



# SOFTWARE LIVRE: TENDÊNCIAS, OPORTUNIDADES E DESAFIOS



TECNOLOGIA DA INFORMAÇÃO BRASILEIRA



# **SOFTWARE LIVRE: TENDÊNCIAS, OPORTUNIDADES E DESAFIOS**

## **ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX**

Presidente

**Rubén Delgado**

Vice-Presidente Executivo

**Ney Leal**

Diretoria de Operações

**Gláucia Chiliatto**

*Controller*

**Fabian AppelPetrat**

Responsável técnica

**Virgínia Duarte**

Equipe técnica

**Virgínia Duarte, Cláudio Ballande Romanelli e Austregésilo Gonçalves**

Revisão e Copidesque

**Área de Inteligência Softex**

Promoção e difusão

**Comunicação Softex**

**MPL Assessoria e Consultoria Técnica de Imprensa**

Projeto gráfico e diagramação

**Fuse Design**

### **RECONHECIMENTO ESPECIAL**

**Suporte financeiro** - Financiadora de Estudos e Projetos (FINEP)

**Apoio** – Centro de Tecnologia da Informação Renato Archer (CTI)/Fundação de Apoio à Capacitação em TI (FACTI); Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Campus Salto; **Comunidade de software livre/código aberto** responsável pela Pesquisa FLOSS 2013, cedida sob licença Creative Commons.

**Integrantes da comunidade brasileira de software livre/código aberto** que, gentilmente, participaram de debates e entrevistas.

As opiniões emitidas nesta publicação são de exclusiva e inteira responsabilidade dos autores não exprimindo, necessariamente, o ponto de vista da SOFTEX ou de seus parceiros e entrevistados.

A duplicação ou reprodução desta obra, sob qualquer meio, só é permitida mediante autorização da Softex. As ideias expressas nesta publicação poderão ser reproduzidas desde que citada a fonte.

Todos os direitos reservados à Associação para Promoção da Excelência do Software Brasileiro – SOFTEX

Copyright@2014 para SOFTEX

### **Associação para Promoção da Excelência do Software Brasileiro – SOFTEX**

Rua Irmã Serafina, 863 - 6o andar  
Edifício Sada Jorge  
Centro - Campinas, SP  
Brasil  
CEP: 13015-914  
[www.softex.br](http://www.softex.br)

## CONHEÇA A SOFTEX

A Associação para Promoção da Excelência do Software Brasileiro – SOFTEX - é uma entidade de direito privado, sem fins lucrativos, que desenvolve ações para promover a melhoria da competitividade da indústria brasileira de software e serviços de TI. É gestora do Programa para Promoção da Exportação do Software Brasileiro – Programa SOFTEX, considerado programa prioritário pelo Ministério da Ciência, Tecnologia e Inovação (MCTI).

Desde a sua criação, em 1996, a SOFTEX vem ampliando a sua área de atuação e contribuindo de maneira significativa para o desenvolvimento socioeconômico brasileiro e para a inserção competitiva do país na economia mundial. Mantém programas e ações nas seguintes áreas: inovação e empreendedorismo; investimento; geração de negócios no exterior; qualidade; capacitação de recursos humanos; e inteligência de negócios.

O Sistema SOFTEX é composto pela SOFTEX e pela rede de Agentes Regionais, distribuídos por 23 cidades em 13 estados brasileiros, que contam com mais de 2 mil empresas associadas.



## CONHEÇA O NISB/OBSERVATÓRIO SOFTEX

O **Núcleo de Inteligência do Software Brasileiro (NISB)/Observatório SOFTEX** é a unidade de estudos, pesquisas e inteligência da SOFTEX. Cabe à unidade coletar, organizar, analisar e difundir dados e informações sobre as atividades de software e serviços de TI realizadas no Brasil. Também faz parte das suas atribuições propor, aplicar e disseminar novos conceitos e novas metodologias para estudos, interagir com universidades e institutos de pesquisa em nível nacional e internacional e incentivar o surgimento de grupos de pesquisa sobre temas de interesse.

A geração de Inteligência Estratégica e Competitiva para o setor de software e serviços de TI é uma ação viabilizada pela manutenção e atualização de Sistema de Informação composto por dados confiáveis oriundos de fontes oficiais diversas e por pesquisas de mercado. As atividades do NISB/Observatório SOFTEX incluem, ainda, a realização de estudos sobre ecossistemas digitais e consultorias sob demanda e a publicação das séries **Software e Serviços de TI: A Indústria Brasileira em Perspectiva e Cadernos Temáticos do Observatório**.



# SUMÁRIO

<b>APRESENTAÇÃO</b>	<b>10</b>
<b>CAPÍTULO 1</b> SOFTWARE LIVRE E SOFTWARE PROPRIETÁRIO: UMA DISPUTA PELA HEGEMONIA NO MERCADO DE SISTEMAS OPERACIONAIS	<b>12</b>
<b>CAPÍTULO 2</b> PERFIL DA COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO: QUEM SÃO E O QUE PENSAM OS SEUS INTEGRANTES	<b>40</b>
<b>CAPÍTULO 3</b> O USO DO SOFTWARE LIVRE NA ADMINISTRAÇÃO	<b>100</b>
<b>CAPÍTULO 4</b> SOFTWARE LIVRE/CÓDIGO ABERTO: DECIFRANDO O PRESENTE E FAZENDO PROGNÓSTICOS PARA O FUTURO	<b>134</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>174</b>
<b>GLOSSÁRIO DE SIGLAS</b>	<b>178</b>
<b>ANEXO</b>	<b>180</b>

# APRESENTAÇÃO

Na virada da década de 2010, o número de usuários de dispositivos móveis, como *smartphones* e *tablets*, ultrapassou a quantidade de usuários de computadores pessoais. A visão de Bill Gates, anunciada nos anos de 1980, de que em cada mesa de cada casa no mundo haveria um computador serviu de guia para a Microsoft e o seu ecossistema baseado na computação pessoal. A premissa sobre a qual este negócio se baseou foi exposta em uma carta aberta do próprio Gates à comunidade de software livre: as empresas precisariam vender o software para poder remunerar os seus programadores.

Como se verificou nas décadas seguintes, a cisão no seio do *Homebrew Computer Club*, onde foram estabelecidas as bases das tecnologias digitais, dividiu o mundo do software basicamente com duas correntes: os que defendiam o software livre/código aberto e os que tinham negócios de alguma maneira baseados em software proprietário.

Apesar do inquestionável sucesso do software proprietário, através do qual a maioria absoluta das pessoas teve acesso às facilidades proporcionadas pelos computadores pessoais, o advento da Internet e da computação ubíqua e o barateamento do acesso aos sistemas de informação criaram um novo paradigma no qual o 'Império Win-Tel' perdeu sua força. Agora é a vez dos *smartphones* com telas *touch* rodando Android em um processador RISC da ARM.

Uma série de novos negócios, alguns deles gigantescos, possui uma estratégia tecnológica baseada, de alguma forma, em software livre ou software de código aberto. Poucos sabem que o núcleo do software no qual os dispositivos Android rodam é totalmente baseado em software livre, o *kernel* Linux. Outra grande força neste mercado, a Apple, com seu iPhone rodando sistema iOS, tem sua plataforma baseada em software aberto, o BSD. Seguindo o modelo do software livre/código aberto, empresas como Google e Facebook têm conseguido êxito comercial.

No contexto de mobilidade, avanço da Internet e surgimento de negócios disruptivos, as discussões travadas há quase quarenta anos voltam à tona, trazendo novos questionamentos: qual seria, agora, o papel do software livre e das comunidades livres no desenvolvimento das tecnologias digitais e de um uso apropriado delas? Faz sentido, ainda, a discussão software livre/código aberto versus software proprietário? O que é possível aprender sobre o passado para inspirar e construir uma nova visão de futuro com respeito à adoção e apropriação das tecnologias digitais?

## Estrutura da publicação

No Capítulo 1, Software Livre e Software Proprietário: uma disputa pela hegemonia no mercado de sistemas operacionais, analisa-se a disputa que ocorreu entre diferentes plataformas tecnológicas, livres e proprietárias, pela hegemonia no mercado de sistemas operacionais para estações de trabalho/servidores. Busca-se, com esta reflexão sobre o passado, produzir um conjunto de lições úteis para compreender o presente e prognosticar o futuro da indústria de software e as perspectivas do software livre e do software de código aberto no novo cenário.

Apresentam-se as duas características básicas da estrutura e dinâmica competitiva da indústria de software: a existência de plataformas tecnológicas concorrentes e o processo de desenvolvimento baseado em módulos. Mostra-se que a hegemonia de uma dada plataforma tecnológica sobre as demais permite o surgimento de externalidades de redes capazes de gerar retornos crescentes para a líder e para o conjunto de parceiros que gravitam em torno dela, ofertando módulos complementares. A consolidação de uma dada plataforma tecnológica provoca efeitos de *lock-in* que, por um lado, inibem a saída de atores da área de influência da plataforma hegemônica e, por outro, desestimulam a entrada de novas candidatas à líder no mercado.

No Capítulo 2, são fornecidas informações sobre o perfil da comunidade brasileira de software livre/código aberto. Busca-se definir quem são e o que pensam os seus integrantes. O seu retrato é comparado com o da comunidade internacional e, também, com o da comunidade brasileira de dez anos atrás.

O uso do software livre/código aberto no Brasil é discutido no Capítulo 3. Pretende-se entender como as disputas pela hegemonia do mercado de sistemas operacionais, que tiveram início nos países centrais, repercutiram no Brasil e foram assimiladas à realidade brasileira. Inicialmente, comparam-se dados e expectativas de crescimento do uso do software livre/código aberto no Brasil, no início dos anos 2000 e no início da década de 2010. Apresentam-se os elos que constituem a cadeia de valor do software e o posicionamento de empresas de capital nacional nesta cadeia de valor e o que isto significou em termos de possibilidades de resistência à hegemonia da plataforma tecnológica líder. Os elementos do cálculo de custo e benefício realizado por demandantes e fornecedores de software e serviços de TI são discutidos. Além disso, mostra-se como, no Brasil, o movimento de software livre/código aberto evoluiu e criou identidade própria, tendo como pano de fundo os fatores limitantes e as possibilidades em aberto.

No Capítulo 4, Software livre/código aberto: prognósticos para o futuro, mencionam-se as tecnologias e tendências de negócios que irão transformar a realidade nos próximos anos. São feitas considerações sobre o impacto que vêm provocando e como irão afetar a indústria brasileira de software e serviços de TI, cujas empresas, na sua maioria, encontram-se atreladas às plataformas tecnológicas líderes, comandadas por empresas-redes com sede nos países centrais. Exploram-se as oportunidades para o crescimento de uma indústria brasileira de software livre/código aberto, antenada com o novo paradigma tecnológico em construção e pontos fortes e fracos, oportunidades e ameaças para a indústria brasileira de software livre/código aberto.

Os autores.

# CAPÍTULO 1

## SOFTWARE LIVRE E SOFTWARE PROPRIETÁRIO: UMA DISPUTA PELA HEGEMONIA NO MERCADO DE SISTEMAS OPERACIONAIS

### APRESENTAÇÃO

Neste capítulo, analisa-se a disputa que ocorreu entre diferentes plataformas tecnológicas, livres e proprietárias, pela hegemonia no mercado de sistemas operacionais para estações de trabalho/servidores. Busca-se, com esta reflexão sobre o passado, produzir um conjunto de lições úteis para compreender o presente e prognosticar o futuro da indústria de software e as perspectivas do software livre e do software de código aberto no novo cenário.

Na Seção 1.1, apresentam-se as duas características básicas da estrutura e dinâmica competitiva da indústria de software: a existência de plataformas tecnológicas concorrentes e o processo de desenvolvimento baseado em módulos. Mostra-se que a hegemonia de uma dada plataforma tecnológica sobre as demais permite o surgimento de externalidades de redes capazes de gerar retornos crescentes para a líder e para o conjunto de parceiros que gravitam em torno dela, ofertando módulos complementares. A consolidação de uma dada plataforma tecnológica provoca efeitos de *lock-in* que, por um lado, inibem a saída de atores da área de influência da plataforma hegemônica e, por outro, desestimulam a entrada de novas candidatas à líder no mercado.

O cenário de construção paulatina de uma plataforma hegemônica no mercado de sistemas operacionais para estações de trabalho/servidores, na década de 90, é apresentado na Seção 1.2. Nas seções 1.3 e 1.4, identificam-se as forças que agiram no sentido de facilitar ou dificultar a consolidação de plataformas livres e proprietárias neste mercado. A história que se segue, na década de 2000, após a consolidação da plataforma líder, é narrada na Seção 1.5.

## 1.1 ESTRUTURA E DINÂMICA DA INDÚSTRIA DE SOFTWARE

*A indústria de software possui duas características principais: organização das empresas em torno de plataformas tecnológicas e o desenvolvimento modular de produtos*

A indústria de software apresenta duas características peculiares: o arranjo típico das empresas em torno de plataformas tecnológicas e o processo de desenvolvimento baseado em módulos (Softex, 2014)<sup>1</sup>. Essas características desempenham um papel importante na dinâmica competitiva da indústria e no jeito específico com que as inovações ocorrem no momento em que atores com diferentes plataformas tecnológicas disputam a liderança do mercado e, no instante seguinte, em que ocorre a consolidação de uma dada plataforma.

*Dois momentos distintos. No primeiro momento, durante a construção da hegemonia, aliados e combinação de ativos tecnológicos jogam um papel relevante. Predominam as inovações radicais, no centro da rede em construção*

O momento de busca pela hegemonia é particularmente favorável ao surgimento de inovações tecnológicas radicais. Várias empresas disputam o mercado, que se oferece relativamente aberto para entrantes e para a celebração de parcerias estratégicas que permitem criar vantagens comparativas frente aos concorrentes. Durante a construção da hegemonia, a busca por aliados e a combinação adequada de ativos tecnológicos jogam um papel relevante na definição de quem finalmente assumirá a liderança. Embora os atributos tecnológicos de uma dada firma sejam importantes na disputa, eles precisam estar relativamente amparados por tecnologias e produtos complementares. As parcerias são importantes neste momento porque criam sinergias e fortalecem os esforços rumo à consolidação de um novo paradigma tecnológico ou modelo de negócios inusitado.

Um elemento importante da dinâmica competitiva da indústria de software diz respeito ao processo de modularização. O crescimento de uma dada plataforma ocorre através da agregação de novos módulos complementares ou pelo aperfeiçoamento de módulos já existentes. Apesar de dependerem estruturalmente da plataforma tecnológica a qual se encontram vinculados, os módulos são independentes entre si, o que facilita a orquestração desde o centro e a agregação contínua de novos módulos.

*Após a construção da hegemonia: externalidades de redes, retornos crescentes e efeitos de lock-in. Predominam as inovações incrementais, nos módulos complementares, ou seja, na periferia da rede construída*

Após a consolidação de uma dada plataforma líder, as inovações mudam a sua característica, de radicais para incrementais. Direcionam-se não tanto para o centro mas, sobretudo, para a experimentação realizada na periferia, ou seja, em cada módulo de software complementar aprimorado ou acrescentado à rede. Assim, é justamente na possibilidade de criação de novos módulos não previstos anteriormente que

---

<sup>1</sup> Esta seção baseia-se em publicação da Softex (2014). Pesquisa, desenvolvimento e inovação em software e serviços de TI. Cadernos Temáticos do Observatório, volume 3 e, mais especificamente, nos capítulos dos autores Roselino e Diegues, já com vários trabalhos divulgados sobre o tema.

reside grande parte do potencial inovador de uma estrutura modularizada como a vigente nas atividades de software. A efetivação deste potencial, por sua vez, é condicionada pelo grau de dispersão das capacitações tecnológicas entre as empresas constituintes da rede. Uma vez que os módulos das plataformas de software são concebidos como blocos de funcionamento autônomo, não é necessário que cada agente, ao buscar desenvolver novas soluções tecnológicas, leve em consideração a interdependência do módulo de software recém-criado com os demais módulos da plataforma. Esta característica cria um ambiente de incentivo ao processo de inovação, pois reduz o conjunto mínimo de capacitações necessárias a cada agente, diminui a complexidade do processo e, em consequência, ameniza os riscos e as incertezas associados à atividade. Ou seja, os ofertantes de módulos complementares para a rede mantêm forte interação com o centro (a plataforma tecnológica a qual estão associados) e relações fracas ou inexistentes com os demais fornecedores de complementos.

**Sucesso dos novos módulos depende mais das externalidades de rede associadas à plataforma do que dos seus atributos tecnológicos**

Após a consolidação de uma plataforma, o sucesso da incorporação de novos módulos depende mais da capacidade destes se beneficiarem das externalidades de rede associadas à plataforma que de seus atributos tecnológicos. Essas externalidades podem ser compreendidas como um fenômeno através do qual a percepção do valor associado a determinado bem ou tecnologia aumenta à medida que o número de usuários e/ou produtores deste bem ou tecnologia também aumenta. Vale destacar que este aumento do valor pode ser percebido (e, normalmente, é) por todo o conjunto de agentes que orbita em torno do bem ou da tecnologia (sejam consumidores ou produtores sejam fornecedores de partes, peças e componentes ou distribuidores). Nas palavras de Cusumano e Gawer (2001, página 5) “quanto mais pessoas usam produtos organizados em torno de plataformas, existem mais incentivos para que sejam introduzidos produtos complementares, os quais estimulam mais pessoas a comprar ou utilizar os primeiros, levando assim a mais inovação, *ad infinitum*” (cit. in Softex, 2014).

**Externalidades de rede exercem efeitos indiretos na consolidação da plataforma e no processo de inovação. Benefícios advindos da introdução de módulos inovadores não são capturados apenas pelo agente da inovação, transbordando para toda a rede**

Assim, ao aumentar a possibilidade de que novos módulos se associem a uma plataforma de TI e, deste modo, incentivar a incorporação de novas firmas e usuários a esta plataforma, a modularização incentiva a geração de externalidades de rede. Tais externalidades, por sua vez, exercem efeitos indiretos muito importantes, tanto sobre a consolidação da plataforma como sobre o processo da inovação. Isso porque, uma vez que a percepção do valor e da utilidade de uma plataforma completa é fortemente influenciada pelo grau de disponibilidade de softwares complementares, os benefícios advindos da introdução de módulos inovadores transbordam para toda a rede (Gawer e Henderson, 2005). Assim, como os benefícios não são capturados unicamente pelo agente responsável pela inovação, é engendrado um cenário no qual o estabelecimento de parcerias para inovação torna-se um elemento estratégico não apenas para a criação de assimetrias competitivas individuais entre os agentes, mas também para a competição da plataforma como um todo frente a outras plataformas que desempenham funções similares.

**Mais externalidades de rede, mais efeitos de lock-in em virtude da existência de retornos crescentes**

Como consequência deste cenário, observa-se que quanto maior é a intensidade das externalidades de rede em uma plataforma de software, maiores são os efeitos de *lock-in*. Ou seja, os agentes tornam-se cada vez mais fiéis a um determinado padrão tecnológico. Nas palavras de Arthur (1990, página 93) “padrões ou convenções tecnológicas, assim como tecnologias específicas, tendem a criar efeitos de aprisionamento em virtude da existência de retornos crescentes”.

**Momento de entrada no mercado é fundamental. Superioridade tecnológica pode ser instrumento importante na busca da liderança entre plataformas quando as externalidades de rede ainda não engendram efeitos de lock-in**

Nas situações em que as externalidades de rede ainda não engendraram fortes efeitos de *lock-in*, a superioridade tecnológica pode se configurar como um instrumento para a busca da liderança entre plataformas. Esta vantagem, por sua vez, é mais intensa em cenários nos quais se observa uma ruptura dos paradigmas tecnológicos e um estado de grande incerteza entre os agentes.

A importância dos efeitos de *lock-in* para a consolidação da plataforma de software faz com que o momento de entrada dos *players* no mercado seja uma variável fundamental para o prevalectimento de determinado padrão tecnológico e para a criação de assimetrias competitivas frente aos concorrentes. Ao analisar este fenômeno, Arthur (1990) procura destacar a maneira através da qual as externalidades de rede implicam assimetrias competitivas sólidas para as empresas pioneiras ou *first movers*. É neste ponto que reside a importância de se construir uma plataforma (mesmo que esta não se configure como a tecnologicamente mais avançada disponível no mercado) capaz de gerar externalidades de rede primeiro que a dos concorrentes.

**Consolidação da plataforma tecnológica depende do tempo certo para entrada no mercado**

Cabe destacar, no entanto, que conforme lembra Rosenberg (1976), a definição do *timing* correto é um fenômeno bastante complexo que depende das expectativas acerca da evolução tecnológica futura. Deste modo, em alguns momentos, assim como uma entrada tardia no mercado, uma entrada prematura pode implicar na dificuldade da consolidação da plataforma tecnológica em questão (*cit in* Softex, 2014).

**Parcerias para a liderança tecnológica reduzem as incertezas, permitem retornos financeiros maiores e incentivos mútuos**

Com o intuito de reduzir tais riscos, autores como Gawer e Cusumano (2002) e Borrus e Zysman (1997) destacam a importância da busca de parcerias no sentido de se estabelecer uma liderança tecnológica dividida no progresso inovativo das plataformas de TI. Além da redução da incerteza, destacam também que tais parcerias tendem a fazer com que os retornos financeiros associados à introdução de inovações no sistema sejam mais do que proporcionais ao investimento em atividades inovativas realizadas em separado, por cada agente. Nesse cenário, os mecanismos de incentivo à inovação são potencializados, os riscos são diluídos (em virtude do estabelecimento de parcerias e do fato de as inovações complementares se beneficiarem da existência de um sistema já estabelecido) e cria-se um ambiente de incentivo mútuo ao aprimoramento constante nos diversos módulos e à introdução de novos módulos.

### Além das parcerias, fusões e aquisições são elementos importantes para a inovação das empresas líderes

Além das parcerias, outro elemento importante para a inovação nas atividades de software e serviços de TI é a realização de fusões e aquisições (F&A). No cenário de avanço tecnológico rápido e de espraiamento das capacitações em um número relativamente grande de agentes que caracteriza a indústria de software e serviços de TI, diversas firmas (principalmente as líderes de plataforma) utilizam as fusões e aquisições como instrumento importante de introdução de inovações. Esse mecanismo é facilitado pelo fato de o setor se organizar em torno de arquiteturas modularizadas, o que permite a incorporação dos módulos complementares com rapidez (em virtude dos padrões de interface já estarem estabelecidos).

Por não exigirem a construção de novas capacitações muitas vezes complexas e não dominadas pelas firmas líderes, as F&A permitem-lhes a incorporação de novas funcionalidades às soluções *core* da plataforma de maneira mais ágil, reduzindo o *time to market*. Além disso, dadas as incertezas elevadas e os riscos associados a tecnologias em fase de gestação, a introdução de novos módulos inovadores por meio de F&A contribui para a redução das incertezas.

### Fusões e aquisições: estratégia alternativa aos esforços próprios de P&D para agilizar inovação

Assim, as estratégias de F&A também se configuram como um esforço adicional ao que é tradicionalmente classificado como atividade inovativa, dentre elas a P&D, por exemplo. Casos ilustrativos destes esforços de P&D qualitativamente distinto das atividades convencionais podem ser facilmente observados quando se analisam as atuações dos fundos de *venture capital* de empresas de TI como Microsoft e Google, entre inúmeras outras. Estes, mais do que apenas a compra de empresas com sólidas posições de mercado, buscam fomentar o desenvolvimento de tecnologias em áreas de interesse estratégico para suas controladoras.

Paralelamente à redução das instabilidades técnicas presentes no processo de incorporação de inovações, as F&A também atuam como mecanismo de consolidação de mercado tanto de plataformas tecnológicas como de suas respectivas firmas líderes. Isso porque permitem que estas empresas incorporem uma gama mais ampla de funcionalidades complementares e, ao mesmo tempo, ampliem escalas, fatos estes que exercem efeitos de *lock-in* no mercado.

## **1.2 A BUSCA DE CONSOLIDAÇÃO DE PLATAFORMA TECNOLÓGICA NO MERCADO DE SISTEMAS OPERACIONAIS PARA ESTAÇÕES DE TRABALHO/SERVIDORES**

O conteúdo desta seção baseia-se em artigo de Comerford (1999). O autor busca mostrar como, ao longo da década de 1990, os vários desenvolvedores de Unix e a Microsoft travaram uma disputa acirrada pela hegemonia de suas plataformas tecnológicas. Na época, a Microsoft reinava soberana no segmento de sistemas operacionais para computadores de mesa e buscava escalar o seu sucesso para equipamentos mais potentes: estações de trabalho e servidores. Em sentido inverso, o sistema operacional Unix, em suas várias versões, respondia por fatia significativa do segmento de sistemas operacionais para estações de trabalho e servidores, mas tinha dificuldade para estender a sua hegemonia para o segmento de computadores de mesa. Neste cenário, Linux surgia como uma terceira opção.

*O mercado de servidores e estações de trabalho parece curvar-se ao Windows NT. No entanto, Unix recusa-se a morrer*

Nos anos 90, mesmo com a cada vez maior hegemonia do Windows NT, a família Unix recusava-se a desaparecer do mercado das estações de trabalho e servidores. O cenário ainda era incerto e a disputa acirrada. Conforme Comerford (1999), a menos que o Windows NT conseguisse reverter a tendência de avanço dos sistemas operacionais Unix, seria muito difícil para a Microsoft replicar, no mercado de estações de trabalho e servidores, o sucesso que obteve no segmento de computadores de mesa.

Apesar da competição acirrada com os sistemas tradicionais Unix, o que na época aparentemente mais incomodava a Microsoft era o surgimento de um novo modelo para o desenvolvimento e distribuição de software, baseado no comportamento de fornecedores que permitiam ao usuário examinar o software na linguagem de programação em que foi escrito. Essa nova geração de fornecedores, que surgia e crescia com a Internet, acreditava que o compartilhamento do código-fonte, permitindo que muitos programadores pudessem inspecionar e melhorar o software, iria resultar em produtos com melhor qualidade e confiabilidade. O princípio contrastava com o da Microsoft, que mantinha em sigilo o código-fonte dos seus produtos.

*A Microsoft reinava soberana no mercado de sistemas operacionais para computadores de mesa; Unix mantinha participação significativa no mercado de sistemas operacionais para estações de trabalho. Aquele equipamento atendia usuários de aplicações de escritório; este era utilizado sobretudo por técnicos e cientistas*

Desde a década de 1980, a Microsoft reinava soberana no mercado de sistemas operacionais para computadores de mesa. No entanto, até 1993, Unix dominava o segmento de sistemas operacionais para estações de trabalho. Naquela época, os computadores de mesa e as estações de trabalho eram muito diferentes. Os computadores pessoais eram adotados, sobretudo, para rodar aplicações de escritório. As estações de trabalho eram utilizadas, principalmente, em tarefas técnicas e científicas. O desempenho das estações de trabalho superava o dos computadores pessoais, embora fosse inferior à *performance* dos minicomputadores. Da mesma forma que os computadores pessoais, as estações de trabalho podiam servir a um usuário único. No entanto, tal como os minicomputadores, que, em geral, atendiam a diversos terminais, as estações de trabalho também podiam atuar como servidores nos ambientes de rede.

*Computadores de mesa e estações de trabalho também se diferenciavam na arquitetura utilizada de processamento. Enquanto os PCs utilizavam a tecnologia CISC, da Intel, as estações de trabalho empregavam processadores RISC, desenvolvidos pelos próprios fabricantes dos equipamentos e por fornecedores independentes*

Outra diferença entre os computadores de mesa e as estações de trabalho tinha a ver com os processadores em uso. No início dos anos 90, os computadores de mesa baseavam a sua arquitetura de processamento em dispositivos CISC (*Complex-Instruction Set Computing*) da Intel. As estações de trabalho, pelo seu lado, utilizavam processadores RISC (*Reduced-Instruction Set Computing*) feitos ou por fabricantes do próprio equipamento ou por *designers* e fabricantes de processadores independentes, tais como a MIPS Technologies e a Motorola.

*Computadores de mesa e estações de trabalho haviam sido projetados para usuários diferentes, rodando aplicações distintas. Aqueles eram utilizados por profissionais com pouco conhecimento de computação; estas estavam nas mãos de profissionais de informática*

Computadores de mesa e estações de trabalho voltavam-se para usuários finais diferentes, com aplicações distintas. Os sistemas operacionais, portanto, tinham que variar, moldando-se às necessidades específicas dos usuários. Os sistemas operacionais DOS e Windows dos computadores de mesa foram projetados para rodar apenas sobre processadores Intel. Windows contava com uma interface gráfica de usuário que permitia atrair um público que apenas iniciava o contato com um computador. A história de Unix era distinta. Desenvolvido no início dos anos 70, na AT&T, tinha como público-alvo os profissionais de informática.

Para migrar Unix de um tipo de microprocessador para outro, os usuários podiam recompilar o seu código-fonte, pagando um valor modesto pela licença para a AT&T, que havia sido proibida pelos tribunais norte-americanos de entrar no mercado de informática. Em virtude do baixo custo e das possibilidades de manuseio do código, Unix tornou-se muito popular nas universidades, passando a rodar em uma variedade de plataformas. No final dos anos 70, estudantes da Universidade da Califórnia desenvolvem uma versão melhorada de Unix: a *Berkeley Software Distribution (BSD)*.

*Através da Sun Microsystem, Unix chega ao ambiente de negócios. Fabricantes de estações de trabalho desistem de utilizar os seus sistemas operacionais proprietários*

A BSD Unix chamou a atenção de estudantes da Universidade de Stanford que, em 1982, tinham criado uma *start-up*, a Sun Microsystems, para dar prosseguimento a um projeto acadêmico, *Stanford University Networked (SUN)*, envolvendo computadores em rede. Seus sócios contrataram integrantes da equipe do BSD para encarregar-se do software da estação de trabalho, o que trouxe Unix, finalmente, para o ambiente de negócios. O uso bem-sucedido que a Sun fez das tecnologias e dos padrões de software abertos disponíveis, tais como Unix e os padrões de Ethernet do grupo IEEE 802, levaram as empresas pioneiras no mercado de estações de trabalho, tais como HP e Apollo Computer, a abandonarem os seus sistemas operacionais proprietários e adotarem Unix, no final dos anos 80.

*Sun também traz modificações para o hardware, passando a utilizar processadores RISC. Em 1988, a Microsoft inicia o desenvolvimento do seu primeiro sistema operacional capaz de funcionar com processador RISC: o Windows NT*

A Sun Microsystems também foi responsável por modificações no hardware das estações de trabalho, que eram baseados, na época, na arquitetura CISC, muito utilizada na indústria de informática, inclusive pela Sun, através da família de *chips* Motorola 68000. Em 1987, a Sun incorporou a tecnologia RISC em sua estação de trabalho, com o uso do microprocessador Sparc (*Scalable Processor Architecture*).

Processadores do tipo RISC, do final dos anos 1980 e início dos 1990, propiciavam maior desempenho que os *chips* CISC tradicionais da Intel e Motorola. Sua velocidade de *clock* permitia trabalhar em mais de uma instrução ao mesmo tempo. Abalada pelo desempenho superior da arquitetura RISC, em 1988 a Microsoft começa a desenvolver o seu primeiro sistema operacional capaz de funcionar com *chips* RISC ou CISC: o Windows NT.

Com o lançamento do Windows NT, em 1993, a Microsoft encontra-se em condições para disputar o mercado de sistemas operacionais para estações de trabalho e servidores. A Intel colabora com a parceira, fabricando a família Pentium de chips

Em meados de 1993, com o lançamento do Windows NT, a Microsoft passa a contar com um sistema operacional de 32 *bits* capaz de rodar nos processadores RISC utilizados em estações de trabalho. O primeiro *chip* a impulsionar o Windows NT foi o processador Alpha RISC, da Digital Equipment Corp., que também era capaz de rodar Unix e o sistema operacional *Virtual Memory System*, VMS.

A Intel absorve a nova arquitetura de processador, criando a família Pentium de *chips* de 32 *bits*. Assim, a parceria Windows e Intel, já consolidada no segmento dos computadores de mesa, passa, também, a funcionar no segmento das estações de trabalho, até então dominadas por sistemas operacionais Unix.

O surgimento de várias versões distintas do sistema operacional Unix para estações de trabalho, cada uma lançada por um fabricante distinto, levou à fragmentação do mercado de aplicativos e de prestação de serviços de suporte técnico ao Unix

Enquanto a Microsoft tentava, através do Windows NT, se infiltrar no mercado de sistemas operacionais para estações de trabalho, vários fabricantes de computador, incluindo na relação, DEC, HP, IBM e Silicon Graphic, impressionados com o sucesso da Sun, mostravam cada vez maior interesse por Unix, passando a desenvolver variações próprias, mais ou menos semelhantes, deste mesmo sistema operacional. A fragmentação do Unix em propostas específicas provocou um efeito inesperado e indesejável: a segmentação do mercado, com os fornecedores parceiros precisando customizar as soluções para cada variedade de Unix que suportavam. Isso dificultou o surgimento de um mercado capaz de propiciar ganhos de escala e incentivar o interesse dos fornecedores. Os profissionais de suporte técnico ao Unix tinham que se dividir entre as versões múltiplas e revisões diversas do sistema, o que aumentava os custos e reduzia a oferta disponível de prestadores de serviço.

Gravitando em torno da plataforma Windows, ao contrário, os fornecedores tinham maior garantia de contar com um mercado amplo, o que permitia diluir os custos com o desenvolvimento de módulos complementares pela quantidade expressiva de unidades vendidas. Além disso, o modelo de negócios adotado pela Microsoft tendia a baixar as despesas com suporte, já que colocava à disposição dos clientes uma gama maior de opções de fornecedores.

A dificuldade de os fornecedores de sistemas operacionais Unix, em geral, os próprios fabricantes de equipamentos, definirem rapidamente um padrão comum para o sistema também prejudicou o surgimento de um mercado pujante

Os fornecedores de Unix perceberam que a fragmentação prejudicava o mercado de produtos complementares (aplicativos diversos) e serviços e, por conta disto, trataram de unificar as várias versões existentes. É com este intuito que, em 1988, surge a *Open Software Foundation*. No entanto, foram necessários sete anos até que uma primeira versão padrão, a *Single Unix Unification*, fosse apresentada ao mercado. A partir daí, os produtos dentro do padrão passaram a utilizar a marca Unix 95.

Os fornecedores de módulos complementares para as estações de trabalho Unix provavelmente viam os ambientes não fragmentados do Windows NT e das versões Microsoft para computadores de mesa como altamente sedutores. No entanto, pelo seu lado, os fornecedores de sistemas operacionais Unix, mostraram ter muita dificuldade para se comprometer e adotar um padrão comum. Para eles, devia ser difícil vencer a tentação de modificar o sistema. Com isso, mantinham várias plataformas tecnológicas distintas, nenhuma delas com força suficiente para garantir uma posição hegemônica.

**A morte de Unix estaria decretada caso a Microsoft não tivesse negligenciado o surgimento, no cenário dos anos 90, de um fator novo, poderoso e muito impactante: a Internet. Unix pode renascer como servidor de rede**

No momento do lançamento do Windows NT, em 1993, muitos observadores pensaram que a morte de Unix seria uma questão de tempo. Segundo Comerford (1999), poderia, de fato, ter sido assim, caso a Microsoft não tivesse negligenciado o surgimento e crescimento rápido da Internet. Durante o período de desenvolvimento do Windows NT, a empresa não se deu conta da importância que a rede teria no futuro e como poderia afetar os requisitos do seu sistema. Apenas em 1994, parece ter finalmente compreendido o efeito que teria nos seus produtos e no futuro das tecnologias de informação e comunicação.

Embora a Microsoft tenha percebido o valor de interconectar os computadores de mesa entre si, a conexão era tratada como um arquipélago formado por um conjunto de ilhas independentes. Cada *desktop* era como se fosse uma ilha, com linhas de comunicações para outros elementos similares, permitindo, por assim dizer, a construção de redes de área local. O equipamento nas mãos dos usuários estava no centro do sistema planejado e cada centro continha o conjunto de aplicativos que poderiam ser rodadas no equipamento. Isso foi pensado deste jeito justamente porque a bandeira do computador pessoal tinha sido levantada contra a ideia de terminais aprisionados pelo *mainframe* ou colocados à mercê dos caprichos do departamento de TI. O Windows NT, portanto, foi concebido como uma espécie de sistema operacional para computador de mesa, embora mais poderoso.

O sistema operacional Unix para estações de trabalho também servia para usuários únicos que necessitavam rodar um tipo particular de aplicativo, em geral, de alta demanda. Mas a sua concepção trazia um DNA diferente. Unix era capaz de acomodar simultaneamente muitos usuários em um único sistema, rodando aplicativos diversos e trabalhando com muitos *inputs* e *outputs* diferentes. No ambiente acadêmico, em virtude da disponibilidade de seu código-fonte, tinha uso disseminado dentro e fora da sala de aula. Quando as universidades norte-americanas começaram a se envolver no desenvolvimento da Arpanet, rede que daria origem à Internet, Unix foi adotado como servidor de rede.

**As simplicidade do Windows NT e a incorporação de uma interface gráfica do usuário conquistaram o público corporativo, constituído por pessoas com baixo conhecimento técnico e pouca habilidade em informática**

Apesar de Unix ter conseguido sucesso entre tecnólogos e acadêmicos, enfrentava competição acirrada no ambiente de negócios, onde a Microsoft reinava soberana. Os fornecedores dos vários sistemas Unix disputavam o mercado entre si e também com a Microsoft, oferecendo várias opções de configuração. Para os tecnólogos, o leque amplo de possibilidades era uma vantagem. No entanto, para as pessoas de negócios, com baixo conhecimento técnico e pouca habilidade em informática, a enorme gama de possibilidades transformava as decisões em um assunto complexo.

Em 1993, com Windows NT, a Microsoft mostrava possuir um produto mais simples e moderno. A empresa se posicionava no mercado oferecendo basicamente duas opções: NT para estações de trabalho e NT para servidores. As opções Windows NT também contavam com um nível tecnológico razoável.

Mesmo assim, o crescimento da Microsoft no segmento de estações de trabalho e servidores deu-se de modo lento. No ano de lançamento do Windows NT, a Microsoft tinha estimado vendas na casa de um milhão de cópias. No entanto, até meados de 1994, conseguiu comercializar apenas 300 mil. A razão tinha a ver com questões técnicas: as aplicações preexistentes em 16 bits rodavam mais vagarosamente sobre NT do que sobre Windows 3.1 e requeriam, também, mais capacidade de memória. A primeira revisão importante do sistema, liberada em 1994, recuperou o desempenho do NT para as aplicações legadas. A atualização permitiu que as vendas saltassem para 1 milhão de cópias, no final do ano.

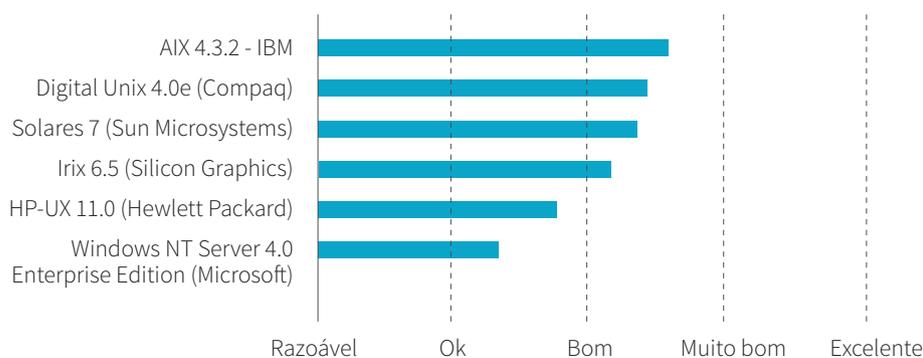
Mas o que realmente ampliou as vendas do Windows NT foi a incorporação de uma interface gráfica de usuário. A solução que veio emprestada do sucessor do Windows 3.1, o Windows 95, tornou o NT mais fácil de usar. A melhoria foi incorporada na versão 4.0 da estação de trabalho Windows NT, lançada em julho de 1996. Em novembro de 1997, de acordo com a IDC, foram licenciadas mais de 11 milhões de cópias. No ano seguinte, 25 milhões.

**De acordo com analistas independentes, apesar do sucesso, o servidor Windows NT fica atrás de seus vários concorrentes Unix no que se refere a desempenho, custo e outros atributos**

Em 1995, considerando métricas de desempenho coletadas pelo *Open Systems Group* do SPEC, os sistemas com mais alto *throughput* e velocidade para operações críticas (que tipificam trabalhos de engenharia) adotavam processadores RISC que rodavam software Unix. O resultado não surpreendia, considerando que várias opções do sistema Unix já estavam disponíveis em versões de 64 bits, enquanto o Windows NT era um sistema de 32 bits.

O desempenho não era a única fragilidade do Windows NT. Características como escalabilidade, confiabilidade, acessibilidade e manutenção poderiam, de alguma forma, contrabalançar a falta de capacidade de processamento. Essas características, no entanto, eram menos evidentes no Windows NT que no Unix comercial, de acordo com avaliações realizadas, na época, por empresas de consultoria isentas. Comparando sistemas diversos, uma delas concluiu que o Servidor 4.0 *Enterprise Edition* do Windows NT estava atrás de versões do Unix em todos os atributos avaliados, exceto suporte ao computador pessoal cliente (Figura 1.1).

**FIGURA 1.1 – AVALIAÇÃO DE SISTEMAS OPERACIONAIS PARA SERVIDORES – 1995**



Fonte: Brown Associates, *apud in* Comerford, 1999.

Em artigo na *web* que comparava as versões de servidores Windows e Unix, uma empresa de consultoria vai além na análise, avaliando, também, os custos associados com a adoção destes sistemas. A sua conclusão é que Unix custava menos que o NT.

Na época, reagindo às avaliações, a Microsoft admitiu as suas deficiências em escalabilidade e confiabilidade, afirmando que pretendia superá-las com a família Windows 2000 de sistemas operacionais para estações de trabalho e servidores. O software estava em teste beta desde 1997, com previsão de lançamento em 1999. Integrantes da nova família iriam suportar até 16 processadores e oferecer uma série de outras vantagens: nível mais alto de segurança, atributos no estado da arte para usuários móveis, melhor confiabilidade e desempenho a um custo total mais baixo.

*Além dos concorrentes tradicionais, a Microsoft precisaria enfrentar, também, movimento que vinha ganhando força, a favor do software livre*

A Microsoft, no entanto, não tinha apenas que competir com fornecedores tradicionais de software. Durante os anos 90, uma nova oposição estava se afirmando na forma de um movimento crescente em defesa do software livre. Entre os defensores deste movimento, encontrava-se a *Free Software Foundation* (FSF), criada para apoiar o projeto GNU, em 1984, por Richard Stallman.

Baseado em Unix, o projeto GNU compunha-se por uma variedade de módulos, incluindo interface gráfica de usuário, editor, sistema de gestão de arquivos e outras utilidades que poderiam ser acrescentadas ao *kernel* do sistema operacional, quando necessário, cabendo ao usuário decidir os módulos a serem utilizados.

Outra contribuição significativa da FSF foi a Licença Pública Geral GNU, um documento legal que permite que o código-fonte seja distribuído livremente através de uma licença de *copyleft*. Esse tipo de licença faz com que qualquer um que redistribua o software, com ou sem mudanças, deva passar adiante a liberdade que lhe foi concedida para melhorar e modificar a cópia.

O Projeto GNU não desenvolveu um *kernel* de sistema operacional próprio. No entanto, em 1991, um estudante finlandês colocou sob licença GNU GPL um *kernel* de sistema operacional que ficou conhecido como Linux. O surgimento do Linux se deveu muito a ausência de um *kernel* GNU que permitisse o uso completo das ferramentas que clonavam as demais estruturas de outros sistemas operacionais Unix. O *kernel* desenvolvido pela FSF, chamado de HURD, até hoje não é uma opção do ponto de vista técnico. O *kernel* do Linux tem o seu projeto baseado no Minix, do Prof. Tannenbaum. Usar o Minix permitiria a Linus Torvalds fechar o código do Linux, mas este preferiu a licença GPL, o que permitiu a rápida adesão dos desenvolvedores em software livre da época ao Linux. O uso do *kernel* Linux com as demais ferramentas GNU desenvolvidas com o apoio da FSF leva à reivindicação do nome GNU/Linux por parte desta.

Linux seduziu os *webmasters* e administradores de redes que começaram a empregá-lo no ambiente de trabalho, de modo velado, no início. Foi possível mantê-lo nesta condição de sigilo por algum tempo, em virtude da sua alta confiabilidade. Como o sistema caía muito raramente, não havia problemas a serem investigados, o que contribuía para que pudesse passar despercebido. O apoio na solução de problemas era fornecido por grupos *online* que também se incumbiam de criar módulos adicionais e realizar correções no sistema. No caso de falhas, não havia a quem responsabilizar.

Linux não era mantido formalmente por nenhuma entidade comercial. Uma dificuldade inerente a esse modelo alternativo de suporte tinha a ver com as dificuldades em caso de falhas no sistema: a quem responsabilizar, nesta situação? No âmbito da FSF, a opção eleita de desenvolvimento, suporte e distribuição do software combinava-se com questões relacionadas com a defesa de direitos humanos e argumentos ideológicos contra injustiças, discriminações, etc., um ponto a mais capaz de provocar objeções para a sua adoção nas empresas.

### Fatores diversos contribuíram para enfraquecer as objeções à adoção do Linux no ambiente corporativo

Alguns fatores contribuíram para enfraquecer as objeções à adoção do Linux nas firmas. Entre eles, destacam-se o desempenho satisfatório do sistema em projetos acadêmicos e militares tidos como críticos e o surgimento de um grupo de distribuidores de software que resolveu utilizar o sistema, apostando em novo modelo de negócios. Foi o caso, por exemplo, das empresas Caldera Systems, Pacific HiTech, Red Hat Software, Suse Holding e VA Research. Essas empresas preferiram não cobrar pela licença de uso do software, buscando ganhos econômicos através de atividades relacionadas com serviços de implantação, documentação e suporte técnico.

### O software de código aberto surge como mais uma alternativa ao software proprietário

Além disso, em 1998, Eric Raymond publica o artigo *Good by, free software; hello open source*, expondo argumentos técnicos a favor da adoção do software de código aberto e deixando de lado aspectos ideológicos presentes no discurso dos defensores do software livre vinculados à *Free Software Foundation*. No mesmo ano, com Bruce Perens, funda a *Open Source Initiative* (OSI).

A seguir, são fornecidas informações sobre os princípios que movem as comunidades de software livre e código aberto e os elementos que as diferenciam.

## **1.3 SOFTWARE LIVRE E CÓDIGO ABERTO: CARACTERÍSTICAS, DIFERENÇAS E SIMILITUDES**

### **CÓDIGO ABERTO: CARACTERÍSTICAS E ELEMENTOS DIFERENCIADORES**

Evangelista, R. (2008) ressalta cinco elementos que, a seu ver, caracterizam o discurso dos defensores do código aberto: elogio ao método de desenvolvimento de software do tipo bazar em contraponto ao método catedral; melhoria técnica obtida através da seleção natural; arrazoado palatável ao ambiente corporativo; distinção clara entre produtores e consumidores; e modelo alternativo à licença tradicional de uso. A seguir, apresenta-se cada um destes elementos do discurso.

*Elogio ao método bazar* - O método de desenvolvimento adotado por Linus e utilizado por Eric Raymond em projetos próprios é descrito no livro *A Catedral e o Bazar*, escrito por Raymond, em 1997. A obra contrasta o novo método com a postura centralizadora de desenvolvimento de software realizada até então. A crítica de Raymond é voltada para o modelo de desenvolvimento do software proprietário, mas é válida, também, para os processos empregados, no geral, pela comunidade de software livre. A seu ver, o modelo de desenvolvimento adotado à época funcionava como uma catedral: era sólido e construído centralmente. Ao contrário, o modelo de desenvolvimento adotado por Linus e por ele próprio era como um bazar:

poroso, erguido e coordenado de modo descentralizado.

**Melhoria técnica a partir da seleção natural** - O movimento do código aberto dava relevância a uma nova prática de produção de software que só se fez possível, a partir dos anos 1990, com o surgimento da Internet. Nessa prática, a rede passou a funcionar como uma metáfora do mundo natural, em que os códigos mais competentes (melhor escritos, mais inovadores) encontravam programadores dispostos a aplicá-los e melhorá-los. A adoção da licença livre passou a ser justificada pela melhoria técnica, pela seleção natural estabelecida na Internet. “Liberar cedo, liberar frequentemente”, como afirmou Raymond, tornou-se a regra para garantir a redução de tempo gasta no desenvolvimento de um software e a adoção das melhores soluções de desenvolvimento, proporcionada pela profusão de olhos a ler o código. A rede passou a funcionar como a experiência de vida que melhoraria o código (Evangelista, 2008).

**Arrazoado palatável ao mundo corporativo** - A metáfora com o mundo natural encontrou grande ressonância nos ouvidos das empresas e nos países centrais. Para aqueles que estão em condição privilegiada em termos sociais, entender sua condição como derivada de uma capacitação, inteligência ou habilidade superior parece mais adequado do que atribuir a diferença a injustiças sociais.

Raymond não esconde que, com o termo código aberto, procurou calar as evidências ideológicas do movimento liderado por Stallman. Poucos meses após a fundação da OSI, após um encontro com a Netscape, confia a um repórter: “Venho mantendo contato com um grupo de *hackers* de software livre e realizamos a nossa própria conferência estratégica. A questão posta é como explorar as oportunidades com a Netscape. Avaliamos algumas estratégias e táticas. Primeira conclusão: o nome *free* software tem que ser descartado. O problema é que ninguém sabe o que significa ‘free’ e, dentro do que eles pensam que sabem, o termo está vinculado a um monte de ideologias...” E complementa, referindo-se a Stallman: “Ele fez um serviço valioso para a nossa comunidade, mas na batalha que lutamos agora a ideologia é apenas um obstáculo. Precisamos argumentar tendo como base a economia e os processos de desenvolvimento e o retorno esperado. Não precisamos nos comportar como comunistas, levantando os nossos punhos nas barricadas. Esta é uma estratégia perdedora. Precisamos de um novo rótulo. Pensamos em vários e, no final, o que venceu foi código aberto. Os defensores do termo código aberto entendem que este termo, mais que software livre, permite ao empresariado perceber que o software sob as licenças previstas pode ser comercializável” (*apud in* Evangelista, 2008).

**Distinção entre produtores e consumidores** - Raymond refere-se com frequência à satisfação dos usuários, mantendo a separação entre produtores e consumidores. No discurso do código aberto, a possibilidade de alteração do código tornou-se algo não essencial, embora ainda desejável. Mais importante que a não-distinção entre produtor e usuário seriam os milhares de olhos capazes de inspecionar o código e fazê-lo evoluir para algo melhor (Evangelista, 2008).

**Modelo alternativo à licença tradicional de uso** - No parágrafo que explica o que é código aberto, na primeira página do *website* da *Open Source Initiative*, consta o seguinte: “A ideia básica por trás do código aberto é muito simples: quando os programadores podem ler, redistribuir e modificar o código-fonte de um software, este evolui. As pessoas melhoram, adaptam e corrigem *bugs* do software. E isto acontece a uma velocidade surpreendente, comparada aos passos vagarosos do desenvolvimento do software tradicional. A comunidade de código aberto aprendeu que este processo rápido de evolução produz software de melhor qualidade do que o modelo fechado tradicional, em que apenas alguns poucos programadores

podem conhecer a fonte e todos os demais devem utilizar cegamente um bloco opaco de *bits*. A OSI existe para abrir esta possibilidade para o mundo comercial” (*apud in* Evangelista, 2008).

O movimento de código aberto, portanto, propunha um modelo de comercialização do software contrário ao modelo tradicional baseado na licença de uso. O modelo colaborativo de produção intelectual oferecia um novo paradigma para lidar com questões envolvendo os direitos autorais (ou patentes em software). Dentro desta perspectiva, várias grandes empresas como IBM, HP, Intel e Dell investiram no software de código aberto, juntando esforços para a criação do *Open Source Development Lab* (OSDL), instituição destinada à criação de tecnologias em código aberto.

## SOFTWARE LIVRE: CARACTERÍSTICAS E ELEMENTOS DIFERENCIADORES

No artigo “Por que o código aberto não compartilha dos objetivos do software livre?”, Stallman (2014) discute as diferenças de princípios e objetivos entre software livre e código aberto. No seu discurso, resumido a seguir, ressaltam-se os seguintes elementos de contraponto ao código aberto: motivos éticos - o software como condição essencial para a liberdade; software robusto e confiável pode ser ruim; ambiguidades existem, também, com o termo *open source*; as comunidades de código aberto se expandem, mas trilham apenas parte do caminho para a liberdade; permissões concedidas aos usuários do software e as condições para a concessão.

**Motivos éticos: software como condição essencial para a liberdade** - Para Stallman, um software é ‘livre’ quando respeita as liberdades essenciais dos usuários: a liberdade de rodá-lo, estudá-lo e modificá-lo e redistribuir cópias com ou sem mudanças. Essas liberdades são essenciais não apenas para os propósitos individuais dos usuários, mas para a sociedade como um todo, pois promovem solidariedade social, isto é, compartilhamento e cooperação. Num mundo de sons, imagens e palavras digitais, o software livre se torna essencial para a liberdade em geral.

Por questões práticas ou crenças pessoais, as razões éticas tendem a ser ignoradas pelos defensores do código aberto. Assim, o termo código aberto se tornou rapidamente associado a ideias e argumentos baseados apenas em valores práticos, tais como criar ou ter software robusto e confiável. O código aberto é uma metodologia de desenvolvimento; o software livre, ao contrário, é um movimento social.

**Software robusto e confiável pode ser ruim** - O desejo de que o software seja robusto e confiável resulta da suposição de que deve servir, da melhor forma possível, aos seus usuários. Se um dado software é robusto e confiável, significa, então, que serve melhor aos usuários. No entanto, essa não deve ser a questão fundamental. Um software só serve, de fato, aos seus usuários quando respeita a sua liberdade. Softwares robustos e confiáveis utilizados para espionagem e outras ações indesejadas e maliciosas, mesmo que mantendo o código aberto, não serão softwares livres.

**Ambiguidades existem, também, no termo open source** -O termo inglês ‘*free software*’ é ambíguo. Pode ser interpretado como software fornecido sem custos e, também, como software que permite certas liberdades ao usuário. É esta segunda interpretação que a comunidade livre quer dar ao termo. A tentativa de solucionar a ambiguidade adotando um novo termo, no caso, *open source*, não resolve o problema, já que este termo também está sujeito a interpretações equivocadas. Para boa parte das pessoas

*open source* parece significar, apenas, o software que permite ao usuário olhar o código-fonte.

**Comunidades de código aberto se expandem, mas trilham apenas parte do caminho para a liberdade** -

Para não deixar as pessoas receosas, os líderes do código aberto decidiram parar de falar de questões éticas, tais como liberdade e responsabilidade. Concluíram que fazendo referência apenas aos benefícios práticos e imediatos de certo software livre poderiam comercializar o software de maneira mais eficaz para certos usuários, especialmente para empresas.

Essa abordagem mostrou-se efetiva. A retórica do código aberto convenceu muitos empresários e indivíduos a usar, e também desenvolver, software livre, ampliando a comunidade — porém, apenas em um nível superficial. A filosofia do código aberto traz muitas pessoas para a comunidade, porém não as ensina a defendê-la. Isso é bom até certo ponto, mas não é o bastante para assegurar a liberdade. Atrair usuários para o código aberto os leva apenas até parte do caminho de se tornar defensores da própria liberdade.

**Permissões concedidas ao usuário final e as condições requeridas para a concessão** -

Parte das distinções entre software livre/código aberto e o software proprietário tem a ver com as permissões concedidas ao usuário do software pelo titular do direito autoral. A distribuição e o uso do software livre/código aberto, tal como acontece com software proprietário, baseiam-se em licenças. Para utilizar os códigos disponíveis, o usuário de um software deve concordar e aceitar a licença associada ao software. A licença possui a força de um contrato de adesão, em que o usuário se compromete a respeitar as regras propostas pelo titular do software e pode ser processado caso não as cumpra. São as regras definidas nessas licenças de uso que determinam se um software pode ser considerado livre, de código aberto ou proprietário.

As licenças de software livre/código aberto autorizam qualquer usuário a utilizar, copiar, modificar e distribuir o software, de acordo com regras determinadas. Pelo seu lado, as licenças de software proprietário permitem que o usuário apenas utilize o software. Em geral, na licença de uso de um software proprietário, reprodução, instalação múltipla, alteração, cessão, revenda ou redistribuição são ações proibidas sem um pagamento adicional.

Apesar do ponto em comum entre software livre e código aberto, existem diferenças no que concerne às condições fornecidas pelo autor para que o usuário usufrua das liberdades concedidas. Também há diferenças no que diz respeito às motivações para a concessão de tais liberdades.

**No software livre, o pensamento predominante é a perpetuação da liberdade. GNU GPL é a licença mais conhecida e adota o copyleft, condição que garante que um software que nasceu livre continue livre**

Os defensores do software livre preocupam-se, sobretudo, com a perpetuação da liberdade. Querem garantir que o produto/conhecimento que nasceu livre continue se desenvolvendo de modo livre, sem ser apropriado por ninguém, em particular. Essa seria a forma de assegurar, de modo contínuo, as quatro liberdades: liberdade para executar o programa para qualquer fim, em qualquer ponto e a qualquer momento; liberdade para redistribuir cópias; liberdade para melhorar o programa; e liberdade para publicar as melhorias.

A licença livre mais conhecida, a GNU GPL, foi proposta pela FSF. Segundo esta licença, o usuário tem liberdade para utilizar o software sem restrições, adaptá-lo às suas necessidades de uso, redistribuir cópias, implementar melhorias e difundi-las. Junto com a licença GNU GPL, a FSF criou o conceito de *copyleft*,

um acrônimo reverso de *copyright*, uma forma de garantir que um software livre e todos os softwares derivados do original continuem livres. Assim, ao utilizar um software sob licença GNU GPL, o usuário se compromete a manter todas as alterações feitas no código-fonte, ou todos os softwares criados utilizando partes do código-fonte, sob o mesmo tipo de licença. Qualquer mudança realizada no software liberado sob licença GPL preservará a mesma licença (o chamado efeito de contaminação), assegurando que o novo software também se torne público, para que a comunidade, que já colaborou com a versão original, também possa desfrutar das melhorias (Taurion, 2004, *apud in* Softex, 2005).

**No geral, as licenças de código aberto atenuam ou descartam o copyleft, permitindo que versões melhoradas de software livre tornem-se software proprietário**

Defensores do software de código aberto tendem a enfatizar mais as questões práticas ligadas ao processo produtivo, tais como o modelo bazar de desenvolvimento de software, do que questões relacionadas com a perpetuação das liberdades defendidas pelo software livre. Sob a égide de código aberto, também foram propostas licenças. Algumas, permitindo que versões alteradas de softwares livres venham a ser comercializadas como software proprietário. Ou seja, na prática, muitas dessas licenças atenuam (ou descartam) a ideia de *copyleft*<sup>2</sup>.

No Quadro 1.1 relacionam-se algumas licenças em software livre/código aberto. A decisão sobre a licença a ser empregada é do autor, que pode eleger entre as disponíveis ou criar uma própria. No entanto, se o desenvolvedor utiliza um código já disponível, poderá ter de se adaptar às condições estabelecidas pelo licenciamento do código usado.

#### QUADRO 1.1 – LICENÇAS EM SOFTWARE LIVRE E CÓDIGO ABERTO

Após o surgimento da licença GNU GPL, várias outras licenças livres e versões destas licenças foram publicadas. Todas permitem uso, modificação e redistribuição do software baixo certas condições. Algumas propõem uma espécie de copyleft atenuado. Outras abandonam este conceito. Destacam-se as seguintes:

- LGPL (Lesser General Public License): versão da GPL tradicional com copyleft atenuado. Permite associar código LGPL a outro código que não esteja sob a mesma licença, desde que respeitadas certas condições.
- BSD (Berkeley System Distribution): não adere ao conceito de copyleft, permitindo que se dê qualquer finalidade ao software, incluindo a associação do código livre original a código não aberto, permitindo que seja posteriormente fechado para criar software proprietário.
- MPL (Mozilla Public License): impõe copyleft apenas para os trechos originais do código, diferenciando o código já existente licenciado por MPL e o código novo, que não necessariamente necessita seguir a mesma licença (e que, inclusive, pode ser proprietário).
- GPL 2.0 (General Public License 2.0): todos os softwares distribuídos ou publicados que contenham código licenciado sob GPL 2.0 devem ser licenciados também sob esta mesma licença. A GPL 2.0 requer aviso de direitos autorais e cópia da licença; que o código-fonte esteja disponível; e que as modificações realizadas estejam documentadas.

<sup>2</sup> Vale ressaltar a possibilidade de licenciamento dual. Isso acontece, por exemplo, com o sistema de banco de dados MySQL, que apresenta uma licença livre (compatível com GNU GPL) para utilizar em projetos de outros softwares livres e uma licença comercial. Neste caso, a forma de licenciamento funciona como em um software proprietário, com cobrança pela cópia do produto, sendo adequada para empresas que desejam desenvolver software utilizando MySQL, mas não querem liberar o seu próprio código com uma licença GPL ou compatível. Outro exemplo de licenciamento dual foi o adotado, no passado, pela empresa Sun para o software OpenOffice.org, disponibilizado sob diferentes licenças livres: LGPL e SISSL (*Sun Industry Standards Source Licence*) (Softex, 2005).

- GPL 3.0 (General Public License 3.0): todos os softwares distribuídos ou publicados que contenham código licenciado sob GPL 3.0 devem ser licenciados também sob esta mesma licença. A versão traz uma série de mudanças, mas as condições se assemelham às da versão anterior. Além de adicionar cláusulas que combatem restrições técnicas que impedem que softwares modificados sejam executados, a GPL 3.0 adquiriu uma linguagem mais robusta, alinhando-se com várias legislações, inclusive a brasileira. Algumas ambiguidades no texto da versão anterior foram removidas e adicionada uma proteção explícita contra patentes. Apesar de incompatível com a versão GPL 2.0, a GPL 3.0 tem compatibilidade com um maior número de licenças, possibilitando integração com softwares sob licença Apache 2.0, por exemplo.
- LGPL 2.1 (Lesser General Public License 2.1) - permite o uso, modificação e redistribuição de bibliotecas de software. Versões modificadas da biblioteca original devem ser distribuídas sob LGPL 2.1. É possível, no entanto, que haja combinação com softwares sob outras licenças. É possível também que cópias da biblioteca sejam licenciadas sob a licença GPL. Seu objetivo principal é permitir que bibliotecas sejam ligadas de forma dinâmica a aplicações sob licença não-GPL.
- MPL 2.0 – (Mozilla Public License 2.0) - versões modificadas do software original devem ser distribuídas também sob MPL 2.0. É possível, no entanto, que haja combinação com softwares sob outras licenças. As principais condições desta licença são: o código-fonte deve ser distribuído sob os termos da MPL 2.0; o código-fonte deve estar disponível; e as modificações devem ser documentadas. Inclui cláusulas de proteção contra patentes.
- BSD Simplificada – versão simplificada da licença BSD original. Versões modificadas do software podem ser licenciadas sob quaisquer outras licenças (inclusive proprietárias) desde que não descumpram as condições da BSD Simplificada, que são: inclusão de aviso de direitos autorais e proibição de uso não autorizado do nome dos autores e dos detentores dos direitos do software para endossar ou promover versões modificadas do software original.
- Apache 2.0 – versões modificadas do software podem ser licenciadas sob quaisquer outras licenças (inclusive proprietárias) desde que não descumpram as condições da Apache 2.0: inclusão de cópia da licença; inclusão de avisos em todos os arquivos modificados informando sobre as alterações que foram feitas; preservação de todos os avisos de direitos autorais, patentes e marcas registradas em distribuições do código-fonte; e, caso haja um arquivo 'Notice' na distribuição, inclusão dos avisos contidos neste arquivo em todas as versões modificadas. Duas características marcantes: presença de proteção contra patentes e proteção das marcas do licenciante, impedindo a utilização destas, exceto em situações descritas pela licença. Os direitos do licenciado são enumerados de modo explícito no texto da licença.
- MIT - versões modificadas do software podem ser licenciadas sob quaisquer outras licenças (inclusive proprietárias) desde que não descumpram a condição da MIT: inclusão de aviso de direitos autorais em todas as cópias ou porções substanciais do software. Os direitos do licenciado são enumerados de modo explícito no texto da licença.

Extraído de Softex (2005) e Wikipédia (novembro de 2014)

## 1.4 A AÇÃO DAS GRANDES FABRICANTES EM APOIO AO SISTEMA OPERACIONAL LINUX E ÀS COMUNIDADES DE CÓDIGO ABERTO

### *Grandes fabricantes investem em software de código aberto e passam a rodar Linux em seus equipamentos*

Em 1999, a Intel (parceira de hardware no duopólio Wintel) e a Netscape Communications (arquirrival da Microsoft) divulgam a realização de investimentos na Red Hat Software, com o objetivo de contribuir para a empresa ganhar musculatura e reforçar o suporte dado ao Linux. A IBM também anuncia apoio comercial para Linux, declarando a adoção do sistema operacional em seus computadores pessoais e na linha de estações de trabalho e servidores RS/6000. Outras fabricantes de computadores, incluindo Compaq, Dell, HP e Silicon Graphics, também passam a rodar Linux em seus equipamentos.

*Apesar do suporte concedido pelas grandes fabricantes ao Linux, no final dos anos 90, o sistema não se mostrava ainda tecnicamente maduro. Unix, considerando as suas várias versões, detinha parte relevante do mercado de estações de trabalho/servidores*

Com todo o suporte concedido pelas grandes fabricantes ao sistema Linux, não seria o momento adequado para que os fornecedores das várias opções de Unix desistissem das suas versões específicas, passando a estimular a adoção do Linux? Comerford faz esta pergunta e busca respondê-la. Segundo ele, mencionando dados da IDC que trata especificamente de sistemas operacionais para servidores, Linux obteve um crescimento expressivo em 1998 (212,5%, respondendo por mais de 17% das remessas no ano). Com mais de um milhão de cópias, Windows NT vinha em segundo lugar em crescimento (27,2%), enquanto as várias versões de Unix aumentavam, no período, 4,1%. No entanto, considerando as várias opções disponíveis, Unix ainda se mantinha como líder de mercado (*apud in* Comerford, 1999). Por esse motivo, não parecia ser o momento apropriado para que os fornecedores de Unix desistissem das suas versões próprias migrando para o sistema Linux.

Linux, pelo seu lado, ainda necessitava de melhorias. De acordo com estudo divulgado em 1999 pela Brown Associates (mencionado em Comerford, 1999), em termos gerais, era tecnicamente inferior aos sistemas operacionais Unix convencionais líderes e ao próprio Windows NT. Na comparação, foram consideradas as versões de Linux da Red Hat e da Caldera, as versões de Unix para servidores da Compaq, HP, IBM, Silicon Graphics e Sun e o Windows NT, da Microsoft. Mesmo achando que Linux apresentava alguns pontos fortes, o estudo salienta a sua ainda baixa maturidade, lembrando que o sistema só estava começando um processo de maturação que era longo e doloroso, mas necessário para o alcance dos níveis elevados de escalabilidade e confiabilidade requeridos no ambiente corporativo.

*Movimento favorável ao código aberto levou fornecedores de sistemas operacionais a realizarem experimentos em plataformas abertas*

Seja como for, no final dos anos 90, o movimento favorável ao código aberto levou vários fornecedores de sistemas operacionais comerciais a realizarem experimentos em plataformas abertas. A Apple lançou o sistema operacional OS X. Parte relevante do código-fonte do sistema era aberta, encontrando-se disponível para escrutínio público. Seguindo os passos da Netscape, grandes empresas desenvolvedoras de software comercial abriram o código de alguns de seus produtos. Linux tornava-se uma alternativa aos produtos da Microsoft e, também, a outras versões de Unix.

Ainda no final da década, a própria Microsoft veio a público para anunciar que estava considerando a possibilidade de abrir o código fonte do *kernel* do seu novo sistema operacional, o Windows 2000. Hoje se sabe que isto não aconteceu. Mas qual teria sido o impacto de uma decisão da Microsoft favorável ao código aberto na indústria de software da década de 2000? O que iria mudar na forma como esta indústria se estrutura e na sua dinâmica concorrencial? Como esta trajetória alternativa poderia afetar a indústria brasileira de software?

### Mercado de estações de trabalho sofre mudanças: não mais máquinas caras e estranhas à arquitetura Wintel

No final dos anos 90, concorrentes no mercado de estações de trabalho introduziam novos sistemas tão rapidamente que as comparações entre eles em um dado momento já não eram válidas dali a poucos meses.

A Compaq reclamou os direitos de apresentar o melhor desempenho quando lançou a *Professional Workstation XT 1000* em fevereiro de 1999. Para a Compaq, era o primeiro em uma linha de novas estações de trabalho de alto desempenho direcionada para o mercado de *design* auxiliado por computador, engenharia e gráficos tridimensionais. Embarcado na estação de trabalho XT 1000 estava um Alpha 21264 de 500-MHz, versão de *chip* projetado pela DEC que, na ocasião, era fabricado pela Intel. Como as versões anteriores do *chip* Alpha, a 21264 podia rodar tanto no sistema operacional Windows NT como no Unix True64, chamado formalmente de Digital Unix. Com uma longa história de sucesso no mercado Wintel, a Compaq estava firmemente ligada tanto ao mundo dos computadores pessoais como ao ambiente das estações de trabalho.

A Silicon Graphics (SGI) também se posicionava nos dois mercados, incorporando, na época, a linha Intel Pentium nas suas ofertas. Em janeiro de 1999, a SGI lançou as estações de trabalho visuais 320 e 520 direcionadas para análises auxiliadas por computador e para aplicações científicas, tais como visualização, em que o *display* tridimensional mostrava-se de grande importância. Ao contrário da oferta da Compaq, os sistemas da SGI ostentavam configurações de processamento múltiplo (usando *chips* Xeon III Pentium de 550-MHz) para agilizar a manipulação e transferir dados gráficos. Essa era a primeira vez que a SGI criava estações de trabalho que não eram baseadas em seus *designs* de MIPS. Enquanto os *chips* MIPS podiam rodar tanto NT como Irix (a versão de Unix da SGI), os novos sistemas eram apenas para NT.

Estimava-se que tanto o sistema 540 da SGI como o XT 1000 da Compaq iriam baixar os preços das estações de trabalho, chegando na loja a algo em torno de \$ 6,500. Mesmo o sistema menos poderoso da SGI, o 320, iniciava em \$ 3,345, o que era consideravelmente mais elevado que os custos médios de um computador pessoal.

A Sun Microsystems, no entanto, fazia o que outros fornecedores não conseguiam. A fabricante anunciou o sistema Ultra 5 com um preço inicial de \$ 2,495. Por este valor, o sistema vinha com um processador Ultra Sparc III de 270 MHz, 64 MB de RAM, 4.3 GB de disco rígido, 24 *bits* na placa-mãe e sistema operacional Unix, o Solares 7. Como a empresa se posicionava em relação ao Windows NT? Através da oferta ao usuário de uma opção de compra de uma placa de coprocessamento SunPCi por \$ 495, possibilitando que NT e as suas aplicações pudessem rodar em um mundo separado.

Mesmo esta pequena amostra de ofertas de estações de trabalho existentes no final dos anos 90 tornava claro que o campo não mais consistia de máquinas caras, estranhas à arquitetura Wintel. Mas era evidente, também que, por algum tempo, ainda, o mundo das estações de trabalho não seria dominado por qualquer uma das opções de hardware ou de arquitetura de software disponíveis.

Além de incluir Linux na disputa pela liderança no segmento de sistemas operacionais para estações de trabalho e servidores, o apoio das grandes fabricantes também desempenhou um papel relevante nos processos de desenvolvimento do software livre/código aberto. Modelos do tipo bazar

passaram a conviver com formatos menos flexíveis. Organizações e empresas desenvolvedoras nem sempre tinham as mesmas motivações que as comunidades que surgiam de modo espontâneo. No bojo destas diferenças, existe toda uma discussão relacionada com a forma de apropriação do software, se individual ou coletiva, e as possíveis trajetórias que podem se abrir, a partir da adoção do sistema operacional Linux. É o que será discutido a seguir.

## SOFTWARE LIVRE/CÓDIGO ABERTO: MODELO COLABORATIVO DE DESENVOLVIMENTO

### Comunidades colaborativas: um novo modelo para desenvolvimento de software, baseado em motivações sociais e suportado pelo ambiente digital em rede

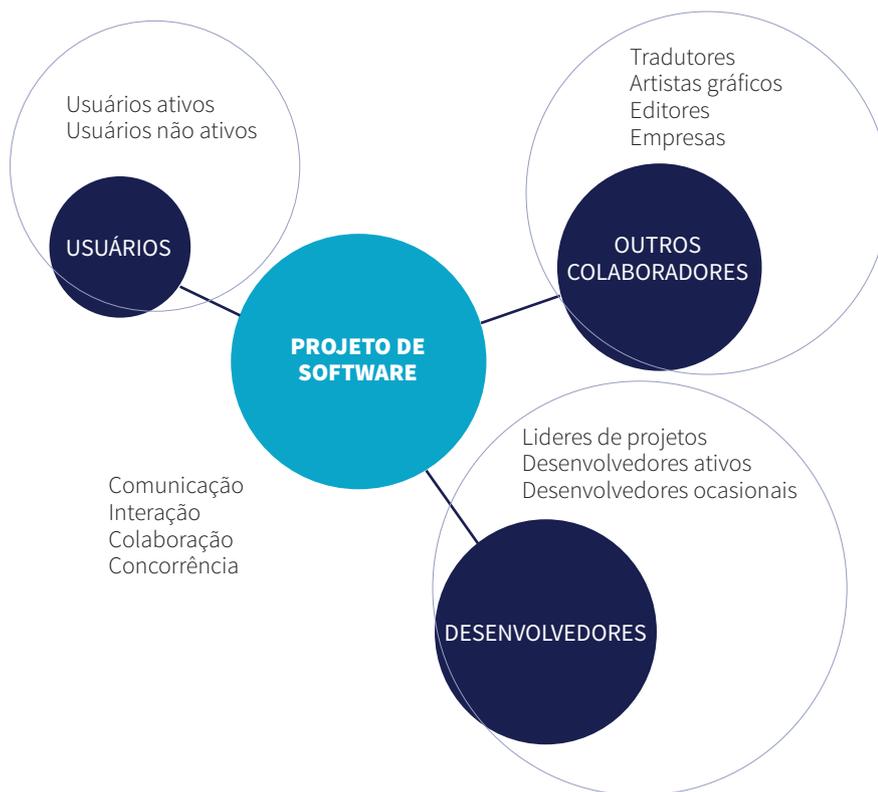
A Internet permitiu o desenvolvimento coletivo de software e se tornou o ambiente de excelência para o exercício da prática colaborativa, suportando a interação entre pessoas ligadas a uma ou mais etapas do ciclo de desenvolvimento do software (programação, indicação de falhas, sugestão de melhorias, tradução, documentação, divulgação e até financiamento). São coletivos heterogêneos e relacionados de modo frouxo, com o objetivo de desenvolvimento de um software específico, em geral, sem contratos formais e, muitas vezes, sem vínculos a empresas ou organizações (Softex, 2005). Os coletivos passam a funcionar após a concepção/especificação de um projeto de software por pessoa, grupo ou empresa que assume a função de líder, detém a autoria do software e determina o tipo de licença a ser adotado.

Para Yochai Benkler (*apud in* Softex, 2005), o modelo que passou a ser amplamente adotado pela comunidade livre e de código aberto seria exemplo de um fenômeno socioeconômico que define um terceiro modo de produção, muito adequado ao ambiente digital em rede. Diferencia-se dos modelos tradicionais baseados em propriedade (empresas) e em contratos (mercados), pois sua característica central é a de grupos de indivíduos que colaboram em grandes projetos por motivações sociais, em vez de preços de mercado e de comando hierárquico-gerencial, motivações típicas das outras formas de produção.

O grupo é comumente chamado de comunidade, formando-se ao redor de um software ou de um projeto de desenvolvimento de software (e de um líder, ou líderes, responsáveis pelo projeto). Inclui desenvolvedores, usuários e interessados para debater e aperfeiçoar uma ferramenta em particular. Podem se subdividir em grupos específicos de desenvolvedores, pessoal de suporte, etc.

Quanto mais visíveis os projetos, maiores as comunidades que se formam ao seu redor. Ainda que o papel central seja o de desenvolvimento, aos responsáveis pela programação se somam outros colaboradores, como tradutores, investidores, artistas gráficos (contribuem com ícones, *layouts*, estudos de usabilidade), editores de livros e, também, usuários. Existem ainda pessoas não ativas, que raramente participam dos debates sobre um software, mantendo-se em posição isolada da comunidade, embora, mesmo assim, sejam considerados seus integrantes (Figura 1.2).

FIGURA 1.2 – REPRESENTAÇÃO ESQUEMÁTICA DA COMUNIDADE E SUBCONJUNTOS RELACIONADOS COM SOFTWARE LIVRE/CÓDIGO ABERTO



Fonte: Softex (2005).

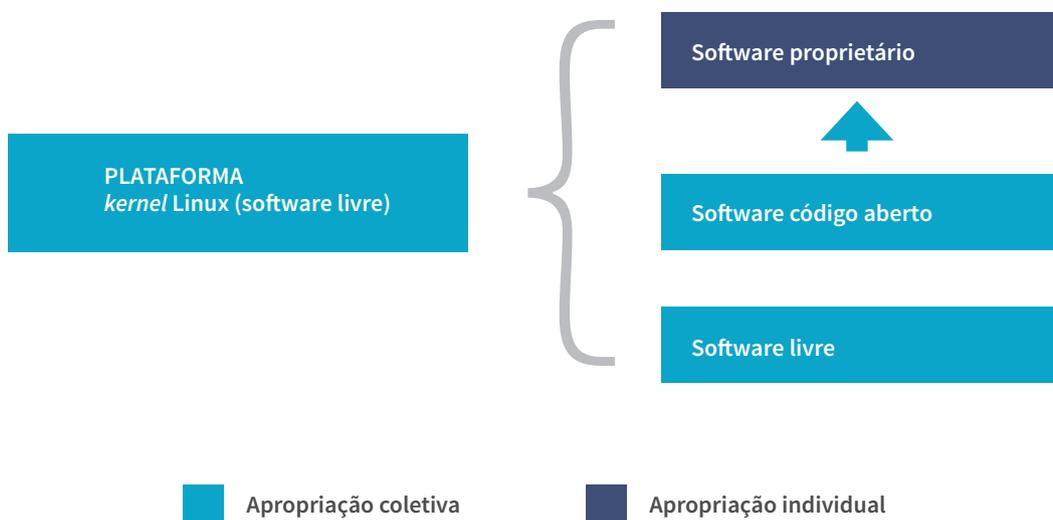
*Apropriação coletiva e apropriação individual: trajetórias alternativas possíveis através do software livre e de código aberto*

Apontam-se, a seguir, algumas características do modelo de desenvolvimento colaborativo (Softex, 2005).

- Participação de muitas pessoas (ou empresas) na criação de um software que ninguém seria capaz (ou gostaria) de desenvolver individualmente, dados tempo, custo e complexidade.
- Correção rápida de falhas e aumento da segurança, já que o código-fonte pode ser inspecionado publicamente, sendo exposto a colaboradores que podem apontar melhorias e detectar falhas.
- Possibilidade de realização de alterações específicas, a partir das necessidades individuais de cada usuário, gerando versões diversas e personalizadas de um mesmo software original.
- Permissão para bifurcação do projeto inicial, denominado FORK no jargão das comunidades que desenvolvem software livre ou aberto, no caso de não haver acordo em relação aos rumos definidos pelo líder, com reaproveitamento das linhas de código geradas.
- Reuso de parte do código de um projeto para desenvolvimento de projetos com outros objetivos.

Observa-se que a produção no formato colaborativo tende a reduzir as incertezas, pois permite, pelo menos em teoria, reunir um número maior e diversificado de competências em torno de um projeto. Também tende a diminuir os custos individuais, já que estes podem ser distribuídos por um número maior de participantes. Em teoria, melhora a qualidade do produto final, acelera o processo de desenvolvimento do software e de aprendizagem e permite extrair vantagens do reuso. No entanto, por se tratar de um trabalho coletivo, cria obstáculos para a apropriação individual dos resultados. Dificuldades de apropriação parecem ter sido solucionadas através de licenças que permitem transformar software livre em software proprietário. Assim, é provável que o software desenvolvido através de processo colaborativo possa se constituir em uma forma muito adequada de empresas proprietárias diluírem custos de P&D&I sem perderem possibilidades de, no futuro, buscarem formas individualizadas de apropriação. Essa é uma trajetória possível, mas não a única. É mais aderente ao modelo de software de código aberto que aos princípios defendidos pela comunidade de software livre, baseados na perpetuação da liberdade (Figura 1.3).

FIGURA 1.3 – TRAJETÓRIAS POSSÍVEIS A PARTIR DO KERNEL LINUX



Fonte: Observatório Softex.

**Equilíbrio adequado no que se refere ao perfil dos participantes de uma dada comunidade é um desafio inerente ao modelo de desenvolvimento colaborativo constituído de modo espontâneo**

Existem algumas dificuldades inerentes a um modelo de desenvolvimento de software baseado na colaboração, em comunidades abertas. Uma primeira questão refere-se às possibilidades, a partir das ações voluntárias, de se chegar a um equilíbrio adequado entre as diferentes competências e os vários perfis de profissionais necessários para levar a cabo, fim-a-fim, o ciclo de vida de um projeto de software. Se o interesse do desenvolvedor é aprender ou compartilhar o seu conhecimento em programação, pode, por exemplo, faltar pessoas para as atividades que se seguem, envolvendo suporte, treinamento e capacitação de pessoal.

Organizações/empresas desenvolvedoras e comunidades espontâneas: nem sempre as mesmas motivações e não necessariamente o mesmo modelo de desenvolvimento

Os profissionais envolvidos no desenvolvimento de projetos corporativos, com objetivos bem definidos e prazos dados, podem acabar se fechando em si mesmos ou controlando as partes críticas do projeto, afastando-se em certa medida das comunidades abertas, mais porosas, versáteis, flexíveis, e adotando um modelo de desenvolvimento mais hierarquizado e rígido. Assim, conforme descrito em Softex (2005), citando Morton, Lerner e Tirole (2002) e Jackson (2004), foi justamente isto que aconteceu após o surgimento do consórcio OSDL, criado por grandes empresas, com o objetivo de melhorar o Linux e disseminar a sua adoção:

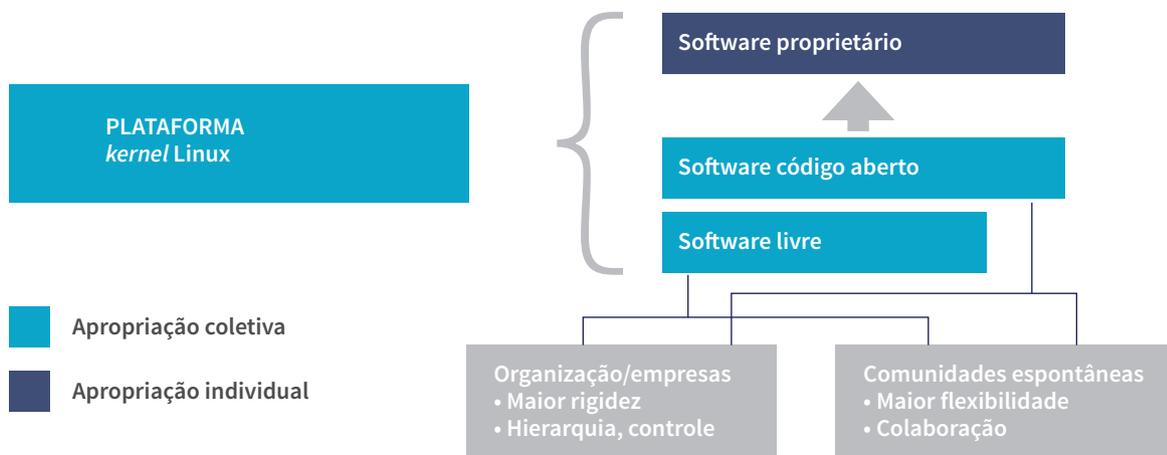
“o perfil dos desenvolvedores, tanto de indivíduos como de empresas, mudou muito após a entrada de grandes atores, como mostra o estabelecimento da *Open Source Developer Labs* (OSDL), que é uma organização sem fins de lucro financiada por corporações como IBM, Intel, Hewlett-Packard e Computer Associates, especificamente para desenvolver Linux para ambientes de grande escala de produção. No início do movimento de software livre, o perfil típico era o do “garoto micreiro apaixonado por computadores” programando em seu sótão “somente por amor à informática”. Mas, após o surgimento da OSDL, a maior parte do código do Linux foi gerado por profissionais programadores empregados em corporações como IBM, Red Hat e SGI, dentro do horário de trabalho.

Nos grandes projetos de software livre/código aberto, o processo de desenvolvimento é bastante hierárquico, pois poucos desenvolvedores (denominados grupo *core* e reconhecidos pelos seus pares) são responsáveis pela maior parte do código, enquanto que um número muito maior de desenvolvedores fazem poucas contribuições, em que a maioria é de informes de erros e problemas (*bugs*). A concentração também se dá em termos geográficos: o grupo *core* do Linux está composto principalmente por desenvolvedores nos Estados Unidos, Europa e Austrália. Há um crescimento representativo dos países do Este Europeu, mas a participação da Ásia e da América Latina permanece pequena.

Esse quadro está relacionado com a emergência do movimento de software de código aberto em complementação ao de software livre. A difusão do Linux atraiu um grande percentual de investimento comercial nos projetos de código aberto e se estabeleceram empresas dedicadas a outorgar suporte ao Linux. Entre as pioneiras estão a VA Linux, estabelecida em 1993, e a Red Hat, em 1995. Red Hat se tornou uma grande empresa que pode ser considerada a principal distribuidora mundial de Linux. De forma semelhante, surgiram outras empresas menores com atuação em software livre/código aberto, tanto em desenvolvimento como em serviços correlatos.”

Na Figura 1.4, apresentam-se, novamente, as trajetórias possíveis dentro de modelo baseado no *kernel* Linux, acrescentando-se novas alternativas amparadas na trajetória livre: o desenvolvimento tendo como base a liderança de organizações e empresas e o desenvolvimento realizado pelas comunidades criadas e mantidas de modo espontâneo.

FIGURA 1.4 – TRAJETÓRIAS POSSÍVEIS A PARTIR DO KERNEL LINUX, CONSIDERANDO ORGANIZAÇÃO PARA A PRODUÇÃO DE SOFTWARE LIVRE/CÓDIGO ABERTO



Fonte: Observatório Softex.

## 1.5 MERCADO DE SISTEMAS OPERACIONAIS PARA ESTAÇÕES DE TRABALHO/SERVIDORES NO FINAL DA DÉCADA DE 2000

Em 2011, no segmento de sistemas operacionais para estações de trabalho e servidores, a plataforma Windows respondia por 49% do mercado, com Unix em segundo lugar, com 22% do total. Linux ocupava a terceira posição (16%) e outros sistemas respondiam pelos 13% restantes (Tabela 1.1).

A hegemonia do Windows continuava incontestável no segmento de sistemas operacionais para computadores de mesa, com participação de 90%. O sistema Mac aparecia na segunda posição, responsabilizando-se por 5% do mercado e Linux por apenas 1%. Outros sistemas respondiam pelos 4% restantes.

**TABELA 1.1 – EVOLUÇÃO DA PARTICIPAÇÃO DAS DIFERENTES MARCAS NO MERCADO DE SISTEMAS OPERACIONAIS PARA SERVIDORES E COMPUTADORES DE MESA - ANOS 2008 E 2011**

### Sistemas operacionais para servidores

Produto	2008	2011
Windows	37%	49%
Unix	33%	22%
Linux	13%	16%
Outros	17%	13%
Total	100,0%	100,0%

### Sistemas operacionais para computadores de mesa

Produto	2008	2011
Windows	88%	90%
Mac	10%	5%
Linux	1%	1%
Outros	1%	4%
Total	100,0%	100,0%

Fonte: HitsLink.

Embora tenha conseguido ganhar algum espaço no mercado de sistemas operacionais para servidores/estações de trabalho, Linux não conseguiu decolar no ambiente de sistemas operacionais para computadores de mesa, apesar dos esforços realizados por consórcios e alianças (por exemplo, a OSDL) no sentido de fortalecer a sua adoção neste segmento de mercado, no início dos anos 2000 (Quadro 1.2). Na ocasião, a Microsoft já havia consolidado a sua hegemonia no segmento dos computadores de mesa, criando efeitos fortes de *lock-in* que dificultaram sobremaneira a inserção do Linux. Além disso, a Microsoft soube

utilizar a sua liderança como trampolim para acesso ao segmento de estações de trabalho/servidores, em que existia, nos anos 90, uma disputa ainda acirrada pela hegemonia, ganhando espaço no segmento.

Seja como for, os fabricantes que se uniram para apoiar o sistema operacional Linux perceberam os benefícios diversos que poderiam ser propiciados com o seu uso embarcado em equipamentos e dispositivos, incluindo, na categoria, eletroeletrônicos e produtos para telecomunicações e mídia. Era uma brecha inexplorada de atuação, em segmento no qual não existia ainda a presença de atores relevantes.

#### QUADRO 1.2 – FATOS MARCANTES NA HISTÓRIA DA APROXIMAÇÃO ENTRE SOFTWARE LIVRE/CÓDIGO ABERTO E AS EMPRESAS

Período	Fatos marcantes
Anos 70	<ul style="list-style-type: none"> <li>• Unix é desenvolvido na AT&amp;T, proibida pela corte norte-americana de entrar no mercado de informática.</li> <li>• O sistema operacional difunde-se nas universidades, passando a rodar em uma variedade de plataformas.</li> <li>• Estudantes da Universidade da Califórnia criam uma versão melhorada de Unix: a BSD.</li> </ul>
1982	<ul style="list-style-type: none"> <li>• Surge a Sun Microsystems. Sun utiliza a equipe desenvolvedora do BSD para desenvolvimento do sistema operacional da sua estação de trabalho.</li> </ul>
1984	<ul style="list-style-type: none"> <li>• Início do desenvolvimento do sistema operacional livre GNU.</li> </ul>
1985	<ul style="list-style-type: none"> <li>• Richard Stallman escreve o Manifesto GNU, um convite para que outros programadores se unam ao esforço da <i>Free Software Foundation</i> para desenvolver software livre.</li> <li>• É criada a Licença Pública Geral (GPL).</li> </ul>
1987	<ul style="list-style-type: none"> <li>• Sun Microsystems incorpora processadores RISC em suas estações de trabalho.</li> </ul>
1988	<ul style="list-style-type: none"> <li>• Microsoft começa a desenvolver o Windows NT: sistema operacional compatível com processadores RISC.</li> <li>• Fabricantes de hardware criam a <i>Open Software Foundation</i> com o objetivo de discutir um padrão único para Unix.</li> </ul>
1991	<ul style="list-style-type: none"> <li>• Linus Torvalds lança a primeira versão do <i>kernel</i> do Linux, licenciado sob licença GNU GPL.</li> </ul>
1993	<ul style="list-style-type: none"> <li>• Surge a empresa VA Linux para fornecer suporte ao Linux.</li> <li>• Lançamento do Window NT marca a entrada da Microsoft no segmento de sistemas operacionais para estações de trabalho.</li> </ul>
1995	<ul style="list-style-type: none"> <li>• Surge a empresa Red Hat para fornecer suporte ao Linux.</li> <li>• Versão padrão para Unix é finalmente apresentada ao mercado. Produtos aderentes ao padrão passam a contar com a marca Unix95.</li> </ul>
1997	<ul style="list-style-type: none"> <li>• Eric Raymond lança o livro <i>A cathedral e o bazar</i>.</li> <li>• Miguel de Icaza inicia o Projeto GNOME.</li> </ul>
1998	<ul style="list-style-type: none"> <li>• Eric Raymond publica o artigo <i>Good by, free software; hello open source</i>. Junto com Bruce Perens, funda a <i>Open Source Initiative</i>.</li> <li>• Intel investe na Red Hat.</li> </ul>

1999	<ul style="list-style-type: none"> <li>• A Red Hat abre o seu capital.</li> <li>• IBM anuncia estratégia de adoção do GNU/Linux.</li> <li>• Miguel de Icaza recebe o prêmio <i>Free Software Award</i> concedido pela FSF. Junto com Nat Friedman funda a Helix Code.</li> <li>• Microsoft lança a família Windows 2000. A empresa anuncia que está considerando a possibilidade de abrir o código-fonte do seu sistema.</li> </ul>
2000	<ul style="list-style-type: none"> <li>• Nasce o OSDL, consórcio envolvendo grandes empresas multinacionais (IBM, HP, NEC, Computer Associates e Intel), com o objetivo de acelerar o desenvolvimento e o uso de Linux nas empresas.</li> <li>• IBM anuncia investimento de US\$ 1 bilhão para compatibilizar software e hardware com o sistema Linux e desloca 250 engenheiros para atuar junto às comunidades de software livre/código aberto.</li> <li>• Sun Microsystems divulga a liberação do código-fonte do Star Office sob licença dual LGPL e SSISSL visando à construção de uma comunidade aberta de desenvolvimento em torno do software. O projeto foi chamado OpenOffice.org</li> <li>• Servidor de <i>web</i> Apache 34 domina amplamente o mercado.</li> <li>• GNU/Linux passa a ser utilizado em eletrônica de consumo: <i>Playstation</i> da Sony e TIVO.</li> </ul>
2001	<ul style="list-style-type: none"> <li>• A Helix Code, agora renomeada Ximian, anuncia o projeto Mono, para suportar o desenvolvimento de software com tecnologia <i>dotnet</i>, da Microsoft, nas plataformas Unix. Icaza é chamado de traidor.</li> </ul>
2002	<ul style="list-style-type: none"> <li>• Hendrik Breikreuz (o Mercur) inicia projeto eMule (aplicativo de compartilhamento de arquivos lançado sob licença GNU-GPL, possuindo versões para Windows Microsoft).</li> <li>• Alcatel, MontaVista Software, Nokia e Toshiba se juntam à iniciativa do OSDL, com o objetivo de introduzir Linux nos mercados de telecomunicações e data-centros. Com a entrada deste novo conjunto, 22 empresas passam a apoiar o consórcio.</li> </ul>
2003	<ul style="list-style-type: none"> <li>• Pela primeira vez, uma empresa não vinculada à indústria de informática integra o consórcio OSDL. A nova entrante, Unilever, caracteriza-se por ser uma forte usuária de tecnologia.</li> <li>• OSDL realiza conversações com várias empresas, com o intuito de expandir ainda mais o número de consorciados. Da relação constam Oracle, Novell, Sun Microsystems e SAP.</li> <li>• Líderes industriais anunciam a criação do <i>Desktop Linux Consortium</i>. Entre os participantes encontram-se empresas responsáveis pela distribuição do sistema operacional Linux, tais como Suse, MandrakeSoft, Lycoris, Xandros, ArkLinux, CodeWeavers, o projeto OpenOffice e o projeto KDE. O consórcio surgiu com o objetivo de promover o crescimento da plataforma Linux nos computadores pessoais, tendo em conta que o sistema tinha conseguido espaço, até o momento, sobretudo no mercado de servidores. Entre os objetivos da associação estão a busca de adoção do sistema operacional no governo, nas escolas e nas pequenas e médias empresas. O financiamento da iniciativa seria realizado através do pagamento de taxas anuais e a direção executiva ficaria a cargo de Bruce Perens.</li> <li>• Grupo Open Office.org trabalha na versão 2.0 do software Open Office com o objetivo de obter melhor compatibilidade com a suíte de escritório da Microsoft, melhor desempenho (maior velocidade e melhor uso de memória), maior integração e usabilidade. A partir da versão 2.0, o Open Office.org passa a ser licenciado exclusivamente sob licença LGPL.</li> </ul>
2004	<ul style="list-style-type: none"> <li>• Novell adquire Ximian e Suse.</li> <li>• OSDL conta com o apoio de 75 empresas, incluindo Alcatel, Ericsson, Bull, Mitsubishi, Nokia, Novell, Unilever e Fujitsu, impactando de modo decisivo o desenvolvimento do Linux e de outros programas livres.</li> <li>• OSDL prepara a especificação de Linux para computadores de mesa com o objetivo de facilitar a instalação e utilização do sistema de código aberto.</li> </ul>

Fonte: adaptação ampliada de Molina (2003), *apud in* Softex (2005).

## CONSIDERAÇÕES FINAIS

Algumas reflexões surgem a partir do arrazoado teórico sobre a estrutura e dinâmica competitiva da indústria de software e das experiências observadas no que se refere às disputas entre software livre/código aberto e software proprietário no mercado de sistemas operacionais para estações de trabalho/servidores.

Um ponto a destacar refere-se à dificuldade aparentemente maior de plataformas abertas se consolidarem. A abertura facilita a criação de trajetórias alternativas e coloca sérios obstáculos para o surgimento de externalidades de redes e efeitos de *lock-in*. Ou seja, aparentemente, as plataformas abertas teriam como efeito colateral perverso a sua fragmentação descontrolada e, como decorrência, uma tendência à segmentação do mercado de módulos complementares, constituídos por softwares aplicativos.

A ausência de um padrão único pode colocar obstáculos ainda maiores para o surgimento de uma plataforma tecnológica aberta e confiável o suficiente para permitir o crescimento de uma rede de parceiros ao seu redor. A adoção de padrões interoperáveis certamente contribui para compatibilizar as várias propostas de plataforma, facilitando a migração sem trauma dos fornecedores de aplicativos (e também usuários) entre as opções existentes. Permite ganhos de escala, estimulando os fornecedores a desenvolver módulos complementares às várias propostas centrais. No passado, como se mostrou ao longo do capítulo, os fornecedores de Unix falharam em introduzir rapidamente no mercado um padrão único a suportar todas as várias opções disponíveis, o que certamente prejudicou a competitividade do sistema.

A licença GNU-GPL parece ter contribuído para impedir que o sistema Linux se abrisse a muitas trajetórias possíveis, evitando o que vinha acontecendo, nos anos 90, com Unix. Aparentemente garantiu que as inovações no sistema pudessem ser melhor controladas e coordenadas pelos responsáveis pelo software, favorecendo a construção de um ambiente adequado para o surgimento de um ecossistema digital relevante ao seu redor. Assim, Linux conseguiu reunir em torno de si um sem número de parceiros que até então desenvolviam sistemas operacionais próprios, baseados em Unix. O consórcio entre as fabricantes de computadores incentivou o crescimento da adoção de Linux e criou as bases para o incentivo ao surgimento de empresas (por exemplo, a Red Hat) com competências para fornecer suporte ao sistema.

O momento, no entanto, não jogou a favor de Linux. Apesar dos esforços das fabricantes, no final dos anos 90, o sistema ainda não tinha maturidade suficiente para substituir as várias versões disponíveis de Unix e enfrentar o sistema da rival Microsoft. Esta, pelo seu lado, soube aproveitar a sua posição já consolidada no mercado de sistemas para computadores de mesa para garantir cada vez maior presença no mercado das estações de trabalho/servidores, tornando-se, no final da década de 2000, a plataforma líder também neste segmento. Salientam-se dois fatores que contribuíram para o aumento de vendas e a caminhada da Microsoft rumo à liderança: a incorporação de uma interface gráfica de usuário no Windows NT, tornando-o mais fácil de usar e a absorção da tecnologia RISC pela parceira Intel, permitindo a fabricação da família Pentium de *chips* e ampliação da parceria Windows e Intel para o segmento de estações de trabalho e servidores.

No período, o consórcio OSDL tentou expandir o Linux para o segmento de computadores de mesa, em movimento em sentido contrário ao que estava sendo realizado pela Microsoft. A investida, no entanto, não obteve o sucesso esperado. Em virtude da presença já consolidada da Microsoft, este mercado mostrava-se especialmente inóspito à entrada de novos candidatos a líderes de plataforma.

Houve um êxito maior com o Linux embarcado em eletroeletrônicos e em equipamentos e dispositivos de telecomunicações e mídia. Nestes segmentos, as soluções em software ainda eram pulverizadas e desenvolvidas *in house*, pelas proprietárias dos equipamentos, inexistindo uma plataforma líder consolidada, com capacidade para criar efeitos de *lock-in*.

Ressalta-se que a decisão das grandes empresas fabricantes de computador de adotar Linux nos seus equipamentos parece ter sido um fator fundamental para o seu fortalecimento no ambiente corporativo, especialmente no mercado de sistemas operacionais para estações de trabalho/servidores. O apoio a Linux ocorreu através da promoção das iniciativas envolvendo o código aberto, mais a gosto do ambiente corporativo, deixando de lado os princípios ideológicos que estavam na origem do pensamento do software livre, tal como proposto inicialmente por Stallman e defendido pela FSL.

Com a adoção de licenças atenuadas, o software de código aberto abriu possibilidades para que produtos desenvolvidos em colaboração pudessem ser apropriados de modo individual, revendo o conceito dado originalmente pela FSF para liberdade.

A aproximação de empresas multinacionais de tecnologias da informação e comunicação dos movimentos de software livre/código aberto trouxe, também, novas nuances para o modelo de desenvolvimento do tipo bazar. Os processos de colaboração tenderam a ser, cada vez mais, coordenados e liderados por grandes corporações, com sede nos países centrais, com as comunidades (e países satélites) exercendo um papel secundário nas interações estabelecidas.

## CAPÍTULO 2

# PERFIL DA COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO: QUEM SÃO E O QUE PENSAM OS SEUS INTEGRANTES

### APRESENTAÇÃO

Neste capítulo, são fornecidas informações sobre o perfil da comunidade brasileira de software livre/código aberto. Busca-se definir quem são e o que pensam os seus integrantes. O seu retrato é comparado com o da comunidade internacional e, também, com o da comunidade brasileira de dez anos atrás.

O perfil dos participantes da comunidade de software livre/código aberto no Brasil e em nível internacional foi obtido através de análise e comparação de dados da pesquisa Software Livre/Código Aberto (SL/CA) Brasil 2014 e do *Free/Libre Open Source Software Survey (FLOSS) 2013*, respectivamente. A edição brasileira da pesquisa foi feita utilizando versão adaptada do questionário FLOSS 2013, cedido sob licença *Creative Commons*. No Anexo 1, encontra-se o formulário da pesquisa brasileira.

### ESCOPO E METODOLOGIA

A Pesquisa SL/CA Brasil 2014, realizada pelo Observatório Softex durante o período 22/09/2014 a 15/11/2014, recebeu 165 respostas, sendo 125 consideradas válidas. As respostas invalidadas referem-se a casos em que o participante não se encontrava envolvido em nenhuma comunidade livre, no momento da pesquisa.

Os participantes da SL/CA Brasil 2014 foram recrutados através de indicações da própria comunidade. Um conjunto inicial de membros ativos e reconhecidos foi convidado a preencher o formulário eletrônico e indicar outros contatos. O método adotado resultou em um número baixo, mas qualificado, de respondentes.

A pesquisa internacional FLOSS 2013 contou com 2.183 respondentes, localizados em vários países.

A metodologia utilizada foi distinta: um formulário eletrônico foi colocado à disposição dos participantes na Internet. A sua participação na comunidade era comprovada através do endereço de *e-mail* fornecido pelo respondente. O método resultou em um número maior de participantes, sem um ponto de partida definido inicialmente.

As diferenças metodológicas entre as edições brasileira e internacional da pesquisa podem explicar eventuais divergências nos resultados. O número baixo de respondentes e a incerteza quanto ao universo de participantes da comunidade de software livre/código aberto recomendam cautela quanto aos achados das pesquisas. Os resultados são preliminares. Outros estudos e levantamentos precisam ser feitos, com o intuito de se obter mais informação sobre o universo, o perfil das comunidades e produtos e serviços que oferecem.

A pesquisa internacional FLOSS 2013 acontece dez anos após a realização de uma pesquisa semelhante. A FLOSS 2013 utilizou o formulário eletrônico adotado na pesquisa anterior, com pequenas modificações. Assim, além de levantar o perfil da atual comunidade, é possível comparar os resultados das duas edições e verificar se houve mudanças no seu perfil.

No início dos anos 2000, a Softex também coordenou uma pesquisa nacional junto à comunidade livre. Os principais achados da época, mencionados em Softex 2005, serão comparados às descobertas da pesquisa SL/CA Brasil 2014.

Este Capítulo possui a seguinte estrutura. Na Seção 2.1, discutem-se os resultados da pesquisa SL/CA Brasil 2014. Na Seção 2.2, estes resultados são comparados com os dados levantados na FLOSS 2013. Na Seção 2.3, comparam-se os dados atuais da edição brasileira com os achados de 2003.

## **2.1 RESULTADOS DA PESQUISA SL/CA BRASIL 2014**

### **PERFIL DOS PARTICIPANTES**

*Maioria é constituída por desenvolvedores ou por pessoas que desempenham atividade híbrida: desenvolvem software e realizam, também, outras tarefas*

Dos 125 respondentes da edição brasileira da pesquisa, 86 (68,8%) se enquadraram na categoria de desenvolvedor, exercendo ou não, simultaneamente, a função de colaborador. Os 39 restantes identificaram-se como colaboradores, contribuindo com as comunidades livres em tarefas diversas, tais como tradução, teste, documentação, especificação do software, etc. Para efeito desta análise, as respostas dos grupos de desenvolvedores e colaboradores foram consideradas separadamente, sendo que, no grupo formado pelos desenvolvedores, encontra-se o desenvolvedor puro, ou seja, a pessoa que participa de pelo menos uma comunidade livre desenvolvendo software, e o desenvolvedor híbrido, ou seja, aquele que, além de atuar como desenvolvedor, envolve-se, também, em outras tarefas.

TABELA 2.1 – PARTICIPANTES DA PESQUISA SL/CA BRASIL 2014

GRUPO	SL/CA BRASIL 2014	
Desenvolvedores ou ambos	86	68,8%
Colaboradores	39	31,2%
TOTAL	125	100,0%

Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.<sup>1</sup>

Nas tabelas 2.2 a 2.10, apresentam-se os resultados do perfil de participantes da pesquisa SL/CA Brasil 2014.

## Perfil dos desenvolvedores

**Entre os desenvolvedores, predominam pessoas do sexo masculino, com nível de escolaridade elevado. A maior parcela é casada ou mantém relação estável e possui filhos crescidos**

Na edição brasileira da pesquisa software livre/código aberto, o grupo de desenvolvedores é formado em grande parte por pessoas do sexo masculino (97,7%), casadas ou com relação estável (71,3%). A maioria possui filhos (57,5% do total) e a idade do filho (ou do filho caçula, no caso de mais de um) é superior a dez anos (34,0%). O perfil predominante, portanto, não é de jovens estudantes, mas sim de pessoas em uma etapa intermediária do ciclo de vida pessoal e profissional.

Os desenvolvedores possuem um nível elevado de escolaridade. Destaca-se a predominância de pessoas com superior completo (33,3%) sendo relevante, também, o percentual daqueles com especialização (25,3%) e mestrado (18,4%).

**No grupo de desenvolvedores, predominam engenheiros de software e programadores. O grupo é composto, sobretudo, por assalariados. A maioria recebe bons salários e gosta do que faz**

Engenheiro de software foi a profissão mais assinalada (27,7% do total), seguida da ocupação de programador (16,9%). No grupo de desenvolvedores existem, também, donos de empresas de TI (14,5%) e consultores de TI (13,3%). Observa-se participação inferior de professores universitários na área de TI (8,4%) e outros profissionais de TI (também 8,4%). A maioria dos desenvolvedores (53,7%) gosta muito do que faz. Parte significativa (62,7%) é constituída por trabalhadores assalariados. Profissionais por conta própria respondem por 26,5% do total; apenas 1,2% dos respondentes é estudante.

No conjunto constituído por desenvolvedores, as faixas salariais de maior remuneração foram as mais pontuadas: 63,5% ganharam salário mensal superior a R\$ 5 mil, em 2013. Quase 30% receberam rendimento bruto superior a R\$ 10 mil no ano em questão.

<sup>1</sup>Na apuração dos resultados de cada questão, foram consideradas apenas as respostas válidas, motivo por que o número de participantes não necessariamente coincide com o total de respondentes em cada questão.

## Perfil dos colaboradores

*Em comparação com o grupo constituído pelos desenvolvedores, entre os colaboradores ocorre maior presença de pessoas do sexo feminino. Predominam, também, pessoas casadas ou com relação estável, com filhos crescidos e com experiência profissional. O nível de escolaridade é ainda superior ao observado para o conjunto de desenvolvedores*

No grupo constituído por colaboradores das comunidades brasileiras de software livre/código aberto, as mulheres também representam um percentual pequeno do total (10,3%). Embora a sua participação seja baixa, ela é superior à verificada para o conjunto formado por desenvolvedores.

A participação baixa de profissionais do sexo feminino não é uma peculiaridade da comunidade livre. Na realidade, de um modo geral, a indústria brasileira de software é composta, em sua grande totalidade, por pessoas do sexo masculino.

O percentual de colaboradores casados ou com relação estável é próximo ao encontrado para o conjunto de desenvolvedores (69,2% e 71,3%, respectivamente). É elevado o número daqueles que têm filhos (71,8%) e que o filho (ou o filho ççula, no caso de mais de um) possui mais de dez anos de idade (42,9%). Assim, também entre os colaboradores, os respondentes não são jovens estudantes, mas pessoas com certa maturidade e experiência profissional. Na média, o grupo parece ser ainda mais sênior do que aquele formado pelos desenvolvedores.

Entre os colaboradores, a maioria possui especialização (38,5%), mestrado (20,5%) ou nível superior completo (33,3%). Na média, o grupo possui um nível de escolaridade superior ao verificado entre os desenvolvedores.

*Entre os colaboradores, predominam os consultores e outros profissionais de TI. As pessoas gostam do que fazem, mas poderiam se imaginar fazendo outras coisas. Em comparação com o grupo constituído por desenvolvedores, é ainda proporcionalmente maior a quantidade de pessoas que receberam salários superiores a R\$ 5 mil, em 2013*

A parcela maior de colaboradores é composta por ‘outros profissionais de TI’ (28,2% do total). No grupo, também é elevada a participação de consultores de TI (20,5%). Tanto entre desenvolvedores como neste grupo, é baixa a presença de executivos de TI, professores universitários e estudantes. Em ambos os casos, predominam pessoas exercendo profissões de TI. A comunidade, de fato, parece estar ainda muito restrita aos profissionais de TI. Os colaboradores gostam do que fazem, mas também se imaginam fazendo outros trabalhos.

Da mesma forma que no grupo composto por desenvolvedores, entre os participantes da pesquisa que se apresentaram como colaboradores, a maioria (71,8%) é de assalariados. Neste grupo, em relação ao de desenvolvedores, é ainda maior o percentual de assalariados e de pessoas que receberam salário mensal acima de R\$ 5 mil em 2013 e, também, daqueles com salários acima de R\$ 10 mil.

TABELA 2.2 - QUAL O SEU GÊNERO?

GÊNERO	DESENVOLVEDORES	COLABORADORES
Masculino	97,7%	89,7%
Feminino	2,3%	10,3%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 87; colaboradores: 39.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.3 - SITUAÇÃO CONJUGAL?

SITUAÇÃO CONJUGAL	DESENVOLVEDORES	COLABORADORES
Solteiro	21,8%	20,5%
Casado ou união estável	71,3%	69,2%
Outro	6,9%	10,3%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 87; colaboradores: 39.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.4 - POSSUI FILHOS?

FILHOS?	DESENVOLVEDORES	COLABORADORES
Sim	57,5%	71,8%
Não	42,5%	28,2%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 87; colaboradores: 39.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.5 - FAIXA DE IDADE DO FILHO (SE MAIS DE UM, INFORMAR A FAIXA DE IDADE DO FILHO CAÇULA)?

FAIXA ETÁRIA	DESENVOLVEDORES	COLABORADORES
Inferior a 2 anos	16,0%	10,7%
Entre 2 e 5 anos	30,0%	25,0%
Entre 6 anos e 10 anos	20,0%	21,4%
Superior a 10 anos	34,0%	42,9%
Total	100,0%	100,0%

Desenvolvedores = 50; colaboradores: 28. Percentual calculado sobre o total de participantes que responderam à questão e informaram possuir filhos: Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.6 - NÍVEL DE ESCOLARIDADE (CONSIDERANDO CURSO COMPLETO DE MAIOR NÍVEL)?

ESCOLARIDADE	DESENVOLVEDORES	COLABORADORES
Fundamental	11,5%	0,0%
Médio	13,8%	5,1%
Superior	33,3%	33,3%
Especialização	25,3%	38,5%
Mestrado	18,4%	20,5%
Doutorado	6,9%	2,6%
Pós-doutorado	1,2%	0,0%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 87; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.7 - PROFISSÃO?

PROFISSÃO/ÁREA DE ATUAÇÃO	DESENVOLVEDORES	COLABORADORES
Engenheiro de software	27,7%	2,6%
Engenheiro (outro)	2,4%	5,1%
Programador	16,9%	2,6%
Consultor de TI	13,3%	20,5%
Consultor (outros setores)	1,2%	2,6%
Executivo (TI)	14,5%	15,4%
Executivo (outros setores)	0,0%	5,1%
Marketing (TI)	0,0%	0,0%
Marketing (outros setores)	0,0%	0,0%
Venda produtos de TI	1,2%	5,1%
Venda outros produtos	0,0%	0,0%
Professor universitário (área de TI)	8,4%	0,0%
Professor universitário (outros setores)	2,4%	5,1%
Estudante de TI	1,2%	0,0%
Estudante outros setores	0,0%	0,0%
Outra (em TI)	8,4%	28,2%
Outra (outros setores)	2,4%	7,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.8 - COMO VOCÊ DESCREVERIA O SEU TRABALHO?

PERCEPÇÃO SOBRE ATIVIDADES EXERCIDAS	DESENVOLVEDORES	COLABORADORES
É o que eu sempre quis fazer. Realmente, gosto muito do meu trabalho.	53,7%	39,5%
Interessante, mas posso me imaginar fazendo outros trabalhos que seriam também interessantes.	42,7%	52,6%
Ganho um bom salário, isto é o que importa para mim.	2,4%	0,0%
Não consigo emprego nas minhas atividades favoritas, mas gosto do que faço.	0,0%	2,6%
Não gosto muito do que faço. Se tivesse oportunidade, mudaria de ocupação.	1,2%	5,3%
Odeio o meu trabalho.	0,0%	0,0
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 82; colaboradores: 38.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.9 - SITUAÇÃO EMPREGATÍCIA?

SITUAÇÃO EMPREGATÍCIA	DESENVOLVEDORES	COLABORADORES
Trabalhador por conta própria	26,5%	18,0%
Empregado	62,7%	71,8%
Desempregado	1,2%	0,0%
Trabalho não remunerado (estudante)	1,2%	2,6%
Trabalho não remunerado (aposentado, etc.)	0,0%	0,0%
Outra	8,4%	7,6%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.10 - FAIXA DE RENDIMENTO BRUTO MENSAL, EM 2013

FAIXA SALARIAL	DESENVOLVEDORES	COLABORADORES
Sem salário	4,9%	0,0%
Menor que R\$ 1 mil	1,2%	2,6%
Maior R\$ 1 mil e até R\$ 2 mil	6,1%	2,6%
Maior R\$ 2 mil e até R\$ 3 mil	9,8%	2,6%
Maior R\$ 3 mil e até R\$ 4 mil	6,1%	15,4%

Maior R\$ 4 mil e até R\$ 5 mil	8,5%	7,7%
Maior R\$ 5 mil e até R\$ 7,5 mil	17,1%	20,5%
Maior R\$ 7,5 mil e até R\$ 10 mil	18,3%	15,4%
Maior R\$ 10 mil	28,1%	33,3%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 82; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## VISÃO SOBRE SOFTWARE LIVRE E SOFTWARE DE CÓDIGO ABERTO

Na pesquisa FLOSS, os termos software livre e código aberto são tratados como sinônimos. No entanto, é sabido que há diferenças entre os termos e que, para alguns integrantes da comunidade, as diferenças têm importância. Nas tabelas 2.11 e 2.12, comparam-se as visões dos desenvolvedores e colaboradores participantes da pesquisa SC/CA Brasil 2014 sobre as diferenças entre software livre e software de código aberto.

### Visão dos desenvolvedores

*Para mais da metade dos desenvolvedores, as diferenças entre software livre e código aberto importam. O grupo identifica-se mais com o software livre que com o software de código aberto. Para 35,4% do total, as diferenças entre as comunidades são relevantes tanto no modo de pensar como no modo de viver*

Para uma quantidade significativa de desenvolvedores participantes da pesquisa brasileira, a diferença entre software livre e código aberto importa (56,6%). Dos que se importam com a diferença, a maior parte se identifica com software livre.

Perguntados se as comunidades de software livre e código aberto são diferentes, uma parte relevante dos desenvolvedores entende que sim (57,8%). Para 34,9% deste conjunto, as diferenças são relevantes tanto no modo de pensar como no de viver. Para 22,9%, as distinções têm a ver apenas com princípios.

### Visão dos colaboradores

*Um percentual ainda maior de colaboradores dá importância à diferença entre software livre e código aberto; com parte relevante do grupo identificando-se com o software livre*

No grupo composto por colaboradores, é muito superior a participação daqueles que acreditam que a diferença entre software livre e código aberto importa: 71,9% deles encontram-se nesta situação. Apesar da enorme preferência pelo software livre observada entre os colaboradores, em comparação com o conjunto de desenvolvedores, é apenas ligeiramente superior a proporção daqueles que acreditam que a diferença entre as comunidades de software livre e código aberto tenha a ver com o modo de pensar e com o jeito de viver.

TABELA 2.11 - VOCÊ SE VÊ COMO PARTE DA COMUNIDADE DE SOFTWARE LIVRE OU COMO PARTE DA COMUNIDADE DE CÓDIGO ABERTO?

SOFTWARE LIVRE OU CÓDIGO ABERTO?	DESENVOLVEDORES	COLABORADORES
A distinção não me importa	43,4%	28,1%
A distinção importa	56,6%	71,9%
<i>Identifico-me como parte da comunidade de software livre</i>	38,6%	53,9%
<i>Identifico-me como parte da comunidade de código aberto</i>	18,1%	18,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.12 - VOCÊ PENSA QUE AS COMUNIDADES DE SOFTWARE LIVRE E CÓDIGO ABERTO SÃO DIFERENTES?

COMUNIDADES SÃO DIFERENTES?	DESENVOLVEDORES	COLABORADORES
Sim, não é apenas um modo diferente de pensar, mas um modo diferente de vida	34,9%	38,5%
Sim, mas somente em seus princípios. O modo como eles trabalham é o mesmo	22,9%	38,5%
Isso não me importa	42,2%	23,1%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## INÍCIO DAS ATIVIDADES EM SOFTWARE E TEMPO DE DEDICAÇÃO

Nas tabelas 2.13 a 2.18, comparam-se dados sobre o início da participação de desenvolvedores e colaboradores na comunidade livre, as relações existentes com software proprietário e o tempo médio de dedicação aos projetos livres e proprietários.

### Início das atividades e tempo de dedicação dos desenvolvedores

#### Parte significativa dos desenvolvedores iniciou o desenvolvimento de software livre/código aberto durante os anos 2000 e 2009

Mais da metade dos desenvolvedores (59,0%) iniciou o desenvolvimento de software livre/código aberto durante os anos 2000 e 2009. Parte também significativa (21,7%) começou antes disto, no período 1990 a 1999.

Na época em que iniciaram na comunidade de software livre/código aberto, 28,9% dos desenvolvedores tinham de 16 a 20 anos de idade; 37,4% de 21 a 25 anos.

Em 2013, os desenvolvedores dedicaram poucas horas semanais aos projetos de software livre/código aberto. Parte gastou um número elevado de horas no desenvolvimento de software proprietário, parte concentrou-se em projetos de software livre/código aberto

Em 2013, os desenvolvedores participantes da pesquisa dedicaram poucas horas semanais aos projetos de software livre/código aberto: 67,5% até 10 horas semanais, em média. No ano em questão, a metade dos desenvolvedores (51,8%) envolveu-se, também, em projetos de software proprietário.

Para os que desenvolveram software proprietário, a maioria (46,5%) também iniciou o desenvolvimento deste tipo de software no período 2000 a 2009; sendo que 23,2% entre 1990 e 1999. Em 2013, 44,2% deste subconjunto de desenvolvedores gastaram, em média, mais de 20 horas semanais com software proprietário.

Observam-se duas situações distintas no que se refere ao envolvimento da comunidade livre com software proprietário. Para um conjunto maior de desenvolvedores, o desenvolvimento de software livre/código aberto é realizado em horários alternativos, por profissionais cujo tempo maior de dedicação é para desenvolvimento de software proprietário. Um conjunto numericamente inferior de desenvolvedores é composto por profissionais dedicados exclusivamente ao desenvolvimento de software livre/código aberto.

### Início das atividades e tempo de dedicação dos colaboradores

Em comparação com os desenvolvedores, parcela superior de colaboradores iniciou na comunidade livre no período 1990 a 1999, um indício a mais de que, na média, este grupo é constituído de pessoas mais velhas. Um número relativamente maior de colaboradores se envolveu com a comunidade livre quando já tinha mais de 25 anos de idade

Entre os colaboradores há um percentual elevado, e muito superior ao que foi observado para o grupo de desenvolvedores, de pessoas que iniciaram na comunidade de software livre/código aberto no período entre 1990 e 1999 (41,0% contra 21,7%, respectivamente). Esse é mais um indício de que, em comparação como o grupo de desenvolvedores, o de colaboradores é formado por pessoas mais velhas.

Comparativamente aos desenvolvedores, os colaboradores envolveram-se com a comunidade de software livre/código aberto com mais idade: 38,5% deles tinham de 26 a 40 anos de idade quando começaram a colaborar com software livre/código aberto (contra 22,8% de desenvolvedores nesta faixa etária).

Em 2013, na média, os colaboradores gastaram ainda menos horas semanais com software livre/código aberto que os desenvolvedores. Uma proporção maior deles envolveu-se com software proprietário, embora com baixa dedicação, em termos de horas

No grupo constituído pelos colaboradores, é ainda menor o percentual daqueles que dedicaram vinte ou mais horas semanais para atividades da comunidade de software livre e código aberto (15,4% do total) e maior a quantidade relativa dos que se envolveram com software proprietário em 2013 (56,4%).

Para os colaboradores que se relacionam com software proprietário, o envolvimento com este tipo de software se deu entre 1980 e 1989 (45,5%) e entre 1990 e 1999 (31,8%). No caso dos colaboradores,

portanto, parece que a interação com a comunidade do software livre/código aberto aconteceu após o envolvimento com o software proprietário. Além disso, é possível que, em um primeiro momento, a participação de brasileiros na comunidade de software livre/código aberto tenha ocorrido através da atuação na condição de colaboradores, com a contribuição em atividades de desenvolvimento passando a ocorrer na década seguinte.

Entre os colaboradores que informaram se envolver com software proprietário em 2013, o envolvimento foi baixo, em número de horas. Para 50,0%, a média foi inferior a 2 horas semanais. Assim, ao baixo tempo de dedicação com software livre e código aberto soma-se, também, um baixo tempo de envolvimento com software proprietário, mostrando que os colaboradores desempenham outras atividades no seu dia-a-dia.

**TABELA 2.13 - QUANDO COMEÇOU A CONTRIBUIR COM PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO?**

PERÍODO	DESENVOLVEDORES	COLABORADORES
Antes de 1960	0,0%	0,0%
Entre 1960 e 1969	0,0%	0,0%
Entre 1970 e 1979	0,0%	0,0%
Entre 1980 e 1989	3,6%	2,6%
Entre 1990 e 1999	21,7%	41,0%
Entre 2000 e 2009	59,0%	48,7%
2010 adiante	15,7%	7,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 83; colaboradores: 39.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.14 - MINHA IDADE NA ÉPOCA ERA...**

FAIXA ETÁRIA	DESENVOLVEDORES	COLABORADORES
10 ou menos	1,2%	0,0%
de 11 a 15	1,2%	0,0%
de 16 a 20	28,9%	20,5%
de 21 a 25	37,4%	30,8%
de 26 a 30	10,8%	18,0%
de 31 a 35	9,6%	7,7%
de 36 a 40	2,4%	12,8%
de 41 a 45	6,0%	7,7%
de 46 a 50	1,2%	2,6%

de 51 a 55	1,2%	0,0%
56 ou mais	0,0%	0,0%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.15 - HORAS POR SEMANA EM PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO, CONSIDERANDO COMO REFERÊNCIA O ANO ANTERIOR À PESQUISA**

HORAS DE DEDICAÇÃO	DESENVOLVEDORES	COLABORADORES
Menos de 2 horas semanais	15,7%	23,1%
De 2 a 5 horas semanais	31,3%	28,2%
De 6 a 10 horas semanais	20,5%	12,8%
De 11 a 20 horas semanais	9,6%	20,5%
De 21 a 40 horas semanais	13,3%	12,8%
Mais de 40 horas semanais	9,6%	2,6%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.16 - TAMBÉM DESENVOLVE/SE ENVOLVE COM SOFTWARE PROPRIETÁRIO?**

DESENVOLVE/SE ENVOLVE?	DESENVOLVEDORES	COLABORADORES
Sim	51,8%	56,4%
Não	48,2%	43,6%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 83; colaboradores: 39.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.17 - QUANDO VOCÊ COMEÇOU A DESENVOLVER/SE ENVOLVER COM SOFTWARE PROPRIETÁRIO?

PERÍODO	DESENVOLVEDORES	COLABORADORES
Antes de 1960	0,0%	0,0%
Entre 1960 e 1969	0,0%	0,0%
Entre 1970 e 1979	0,0%	0,0%
Entre 1980 e 1989	16,3%	45,5%
Entre 1990 e 1999	23,2%	31,8%
Entre 2000 e 2009	46,5%	13,6%
2010 e adiante	14,0%	9,1%
Total	100,0%	100,0%

Para desenvolvedores: 43; para colaboradores: 22. Percentual calculado sobre o total de respondentes que afirmaram desenvolver software proprietário, em cada categoria. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.18 - NÚMERO DE HORAS SEMANAIS TRABALHANDO COM SOFTWARE PROPRIETÁRIO

HORAS DE DEDICAÇÃO	DESENVOLVEDORES	COLABORADORES
Menos de 2 hs. semanais	16,3%	50,0%
De 2 a 5 hs. semanais	16,3%	22,7%
De 6 a 10 hs. semanais	11,6%	18,2%
De 11 a 20 hs. semanais	11,6%	9,1%
De 21 a 40 hs. semanais	32,6%	0,0%
Mais de 40 hs. semanais	11,6%	0,0%
Total	100,0%	100,0%

Percentual calculado sobre o total de respondentes que afirmaram desenvolver software proprietário, em cada categoria. Para desenvolvedores: 43; para colaboradores: 22. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## MOTIVAÇÃO PARA PARTICIPAR DA COMUNIDADE

Nas tabelas 2.19 e 2.20, comparam-se os motivos que levaram desenvolvedores e colaboradores a participarem do movimento software livre/código aberto e, também, as razões para continuarem atuando na comunidade.

### Motivação dos desenvolvedores

*Aprender foi o motivo principal que levou desenvolvedores a participar da comunidade de software livre/código aberto. Ensinar aparece como um motivo forte para continuar participando da comunidade*

Entre os desenvolvedores participantes da edição brasileira da pesquisa software livre/código aberto, motivações envolvendo o aprendizado estão no cerne das razões levantadas para participar da comunidade. Questionados sobre os motivos para iniciar desenvolvimento de software livre/código aberto, o interesse de ‘aprender e desenvolver novas habilidades foi a resposta mais mencionada, seguida das opções ‘compartilhar os meus conhecimentos e habilidades’ e ‘participar no cenário de software livre/código aberto’.

As mesmas razões apontadas para iniciar participação na comunidade de software livre/código aberto foram, também, mencionadas como os principais motivos para continuar participando da comunidade. A opção ‘compartilhar os meus conhecimentos e habilidades’ ganha um número maior de indicações, seguida de ‘aprender e desenvolver novas habilidades’. ‘Melhorar o software livre/código aberto de outros desenvolvedores’ passa a constar entre as três opções mais citadas, indicando que, com o tempo, aquele que entrou na comunidade motivado para aprender começa a se perceber como responsável por ensinar. Há um nítido senso de retribuição.

## Motivação dos colaboradores

*Da mesma forma, entre os colaboradores, predominam os motivos relacionados com o aprendizado. A opção ‘compartilhar os meus conhecimentos’ ganha peso na relação dos motivos para continuar participando da comunidade de software livre/código aberto*

No grupo formado pelos colaboradores, o aprendizado também aparece como a principal motivação para participar da comunidade livre. ‘Compartilhar os meus conhecimentos’ e ‘participar em novas formas de cooperação’ foram as opções que receberam o maior percentual de respostas. A terceira resposta mais pontuada foi ‘aprender e desenvolver novas habilidades’.

As motivações para seguir se envolvendo com a comunidade continuam as mesmas. Mas, tal como ocorre com os desenvolvedores, ganha peso a opção ‘compartilhar os meus conhecimentos’.

Para ambos os grupos, é baixa a indicação fornecida para motivos envolvendo ganhos econômicos. No entanto, em comparação com os desenvolvedores, existe um número relativamente menor de colaboradores que apontam estes motivos para iniciar e se manter na comunidade de software livre/código aberto. Também em termos comparativos, no grupo de colaboradores, há um conjunto maior de pessoas motivadas pelo ambiente propiciado pelas comunidades.

*Tanto para desenvolvedores como para colaboradores, as motivações se alteram ao longo dos anos: os mais velhos e experientes desejam contribuir com o aprendizado dos mais jovens*

As motivações para participar se alteram conforme a idade dos participantes e o tempo de vivência na comunidade. Os mais velhos e experientes parecem se sentir no dever de retribuir os ganhos recebidos no passado, contribuindo para o aprendizado dos mais jovens e novos entrantes. Estes, por seu lado, manifestam interesse em participar e aprender com as experiências de troca propiciadas pelo ambiente comunitário.

TABELA 2.19 - NO PASSADO, QUAL FOI A RAZÃO PARA INICIAR DESENVOLVIMENTO DE SOFTWARE LIVRE/ CÓDIGO ABERTO?

MOTIVOS		DESENVOLVEDORES	COLABORADORES
Aprendizado	aprender e desenvolver novas habilidades	17,7%	15,2%
	compartilhar o meu conhecimento e habilidades	17,4%	17,6%
	melhorar produtos de outros	7,7%	6,4%
	<b>Total</b>	<b>42,8%</b>	<b>39,2%</b>
Ambiente	participar em novas formas de cooperação	10,7%	17,6%
	participar no cenário de software livre/código aberto	13,4%	12,0%
	conseguir reputação no cenário de software livre/ código aberto	3,0%	2,4%
	<b>Total</b>	<b>27,1%</b>	<b>32,0%</b>
Trabalho e dinheiro	melhorar as minhas oportunidades de trabalho	8,0%	4,8%
	distribuir software não comercial	3,0%	2,4%
	obter recursos financeiros	1,7%	0,8%
	<b>Total</b>	<b>12,7%</b>	<b>8,0%</b>
Problema concreto	conseguir ajuda para uma boa ideia para um produto de software	3,0%	2,4%
	resolver problema que não poderia ser feito através de software proprietário	4,7%	8,8%
	<b>Total</b>	<b>7,7%</b>	<b>11,2%</b>
Ideologia	restringir o poder das grandes empresas de software	3,0%	0,8%
	porque eu penso que software não deveria ser um produto proprietário	6,7%	8,8%
	<b>Total</b>	<b>9,7%</b>	<b>9,6%</b>
<b>Não sei dizer</b>		<b>0,0%</b>	<b>0,0%</b>

Permite respostas múltiplas (máximo de 4 opções). Percentual calculado sobre o total de respostas: desenvolvedores = 299; colaboradores = 125. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.20 – E HOJE, QUAL A RAZÃO PARA CONTINUAR DESENVOLVIMENTO DE SOFTWARE LIVRE/ CÓDIGO ABERTO?

MOTIVOS		DESENVOLVEDORES	COLABORADORES
Aprendizado	aprender e desenvolver novas habilidades	16,4%	16,4%
	compartilhar o meu conhecimento e habilidades	18,8%	18,0%
	melhorar produtos de outros	10,6%	8,6%
	<b>Total</b>	<b>45,7%</b>	<b>43,0%</b>
Ambiente	participar em novas formas de cooperação	8,2%	16,4%
	participar no cenário de software livre/código aberto	13,0%	16,4%
	conseguir reputação no cenário de software livre/ código aberto	4,1%	3,1%
	<b>Total</b>	<b>25,3%</b>	<b>35,9%</b>
Trabalho e dinheiro	melhorar as minhas oportunidades de trabalho	7,2%	3,9%
	distribuir software não comercial	2,7%	3,9%
	obter recursos financeiros	3,4%	1,6%
	<b>Total</b>	<b>13,3%</b>	<b>9,4%</b>
Problema concreto	conseguir ajuda para uma boa ideia para um produto de software	2,7%	1,6%
	resolver problema que não poderia ser feito através de software proprietário	4,1%	0,0%
	<b>Total</b>	<b>6,8%</b>	<b>1,6%</b>
Ideologia	restringir o poder das grandes empresas de software	2,0%	3,1%
	porque eu penso que software não deveria ser um produto proprietário	6,8%	7,0%
	<b>Total</b>	<b>8,9%</b>	<b>10,2%</b>
<b>Não sei dizer</b>		<b>0,0%</b>	<b>0,0%</b>

Permite respostas múltiplas (máximo de 4 opções). Percentual calculado sobre o total de respostas: desenvolvedores = 293; colaboradores = 128. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## ENVOLVIMENTO EM PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO

O desenvolvimento de software livre/código aberto requer a existência de líderes de projeto, ou seja, de pessoas responsáveis pela tomada de uma série de decisões envolvendo as prioridades e os rumos do projeto, a incorporação de novas funcionalidades no software e a comunicação com a comunidade que se forma em torno do mesmo. Exemplos claros deste modelo de governança (com variações próprias a cada comunidade) são as comunidades Apache ([www.apache.org](http://www.apache.org)) e de desenvolvimento do *kernel* do Linux ([kernel.org](http://kernel.org)).

À maioria dos projetos de desenvolvimento de software livre/código aberto, aplica-se um princípio baseado em meritocracia. O conceito de mérito pode variar. Mas, em geral, envolve questões como quantidade e qualidade da contribuição em código e das sugestões fornecidas para o projeto, o grau de envolvimento e a coerência nos debates sobre os rumos do projeto.

Nas tabelas 2.21 e 2.22, apresentam-se dados sobre o envolvimento de desenvolvedores e colaboradores em projetos de software livre/código aberto.

### Envolvimento de desenvolvedores em projetos

#### Envolvimento em uma quantidade pequena de projetos, com baixa participação em posição de liderança

A maioria dos desenvolvedores (53,8%) se envolveu em uma quantidade pequena de projetos: no máximo, em cinco. Cerca de um terço (31,3% do total) nunca teve a oportunidade de assumir a liderança. Dos que exerceram o papel de líder, a maior parcela o fez apenas em uma única ocasião.

### Envolvimento de colaboradores em projetos

#### Colaboradores tendem a se envolver em um número maior de projetos. Um percentual maior já assumiu a posição de líder

Entre o grupo dos colaboradores, a maioria (56,8%) também se envolveu, no máximo, em cinco projetos, assumindo a liderança em poucas situações. No entanto, em comparação com o grupo de desenvolvedores, entre os colaboradores, é maior a participação dos que se envolveram em mais de vinte projetos (8,7% do total entre desenvolvedores e 16,2% entre os colaboradores) e assumiram a posição de líder em mais de 10 projetos (1,3% dos desenvolvedores e 10,8% dos colaboradores).

TABELA 2.21 – EM QUANTOS PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO VOCÊ SE ENVOLVEU?

QUANTIDADE DE PROJETOS	DESENVOLVEDORES	COLABORADORES
De 1 a 5	53,8%	56,8%
De 6 a 10	26,2%	18,9%

De 11 a 20	11,3%	8,1%
De 21 a 30	5,0%	5,4%
De 31 a 50	1,3%	2,7%
De 51 a 75	0,0%	2,7%
De 76 a 100	0,0%	0,0%
Mais de 100	1,2%	5,4%
Nenhum	1,2%	0,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 80; colaboradores = 37.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.22 – EM QUANTOS PROJETOS DE DESENVOLVIMENTO DE SOFTWARE LIVRE/CÓDIGO ABERTO VOCÊ JÁ SE ENVOLVEU NA CONDIÇÃO DE LÍDER?**

QUANTIDADE DE PROJETOS	DESENVOLVEDORES	COLABORADORES
Nenhum	31,3%	35,1%
1	22,5%	16,2%
2	16,3%	16,2%
3	12,5%	2,7%
4-5	11,2%	13,5%
6-7	1,2%	2,7%
8-10	3,7%	2,7%
11-15	0,0%	2,7%
16-20	0,0%	2,7%
Mais de 20	1,3%	5,4%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 80; colaboradores = 37.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## RETORNO FINANCEIRO COM SOFTWARE LIVRE/CÓDIGO ABERTO

Na Tabela 2.23, comparam-se dados referentes ao retorno financeiro com software livre/código aberto obtido por desenvolvedores e colaboradores.

## Retorno financeiro obtido pelos desenvolvedores

*Apenas uma pequena parcela dos desenvolvedores é remunerada para desenvolvimento de software livre/código aberto. No entanto, são poucos os que não conseguem algum tipo de ganho monetário, direto ou indireto, com este tipo de software*

Para 86,2% dos desenvolvedores, o software livre/código aberto, de uma ou outra forma, permite retorno financeiro. A sobrevivência financeira da comunidade brasileira de software livre/código aberto, no entanto, é pouco baseada em atividades de desenvolvimento. Apenas 26,3% do total de desenvolvedores são pagos para realizar esta atividade. Muitos obtêm recursos em atuações periféricas, envolvendo o ciclo de vida do software. Uma parte deles (17,5% deles), por exemplo, é remunerada pela prestação de serviços de suporte técnico e administração de software livre/código aberto.

## Retorno financeiro obtido por colaboradores

*Parte significativa dos colaboradores também ganha dinheiro, direta ou indiretamente, com software livre/código aberto*

Entre os colaboradores, também é elevado o percentual daqueles que ganham dinheiro direta ou indiretamente com software livre/código aberto: 81,1%. Embora, em comparação com o grupo de desenvolvedores, há um número relativamente menor de pessoas remuneradas para atividades de desenvolvimento, há uma quantidade maior que é remunerada diretamente para administração de software livre/código aberto: 16,2% do total.

Tanto entre desenvolvedores como entre colaboradores, é maior a proporção daqueles que obtêm ganhos diretos com software livre (58,8% e 54,0%, respectivamente) que ganhos indiretos (27,4% e 27,0% do total, respectivamente). No grupo de desenvolvedores, é maior a quantidade relativa daqueles que obtêm dinheiro desenvolvendo (26,3%) ou fornecendo suporte técnico (11,2%) ao software livre/código aberto. Entre os colaboradores, é maior a quantidade dos que auferem ganhos financeiros com a administração de software livre/código aberto (16,2%) e com outras atividades (21,6%).

Os dados parecem corroborar a hipótese de que, pelo menos em alguns casos, o grupo de colaboradores está na posição de cliente e líder de projetos dos quais os desenvolvedores participam seja em atividades de desenvolvimento seja em atividades de prestação de serviços de TI.

**TABELA 2.23 - GANHA DINHEIRO COM SL/CA DIRETA OU INDIRETAMENTE? (SE MAIS DE UM MOTIVO, ESCOLHA O PRINCIPAL)**

GANHA DINHEIRO COM SL/CA?	DESENVOLVEDORES	COLABORADORES
Não	13,8%	18,9%
Sim, diretamente: sou pago para desenvolver SL/CA	26,3%	5,4%

Sim, diretamente: sou pago para dar suporte técnico para SL/CA	11,2%	10,8%
Sim, diretamente: sou pago para administrar SL/CA	6,3%	16,2%
Sim, diretamente: outras razões.	15,0%	21,6%
Sim, indiretamente: eu consegui o meu emprego em virtude de possuir experiência prévia em SL/CA	6,2%	10,8%
Sim, indiretamente: a descrição da minha função não inclui desenvolvimento de SL/CA, mas também desenvolvo este tipo de software no meu trabalho	10,0%	0,0%
Sim, indiretamente: outras razões	11,2%	16,2%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão, sendo: desenvolvedores: 80; colaboradores: 37.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## SOFTWARE LIVRE, SOFTWARE DE CÓDIGO ABERTO E SOFTWARE PROPRIETÁRIO

Nas tabelas 2.24 e 2.25, apresentam-se os resultados de questões que tratam de investigar como desenvolvedores e colaboradores de projetos de software livre/código aberto posicionam-se sobre os atributos do software. O assunto não é trivial. Como avaliar um software? Que atributos devem ser priorizados: as suas funcionalidades, a garantia que traz de liberdade, o modo como foi desenvolvido? O peso dado a cada fator é uma das diferenças existentes entre os defensores do software livre, do código aberto e do software proprietário. Stallman (2014) coloca esta questão em discussão e resume a sua posição a favor da liberdade como principal atributo de um software, ao comentar as diferenças entre software livre e código aberto e a posição dos defensores destas correntes em relação aos que argumentam a favor do software proprietário:

“Por trás dos que defendem o código aberto está a ideia de que a permissão concedida aos usuários para modificar e distribuir o software iria torná-lo mais robusto e confiável. Porém, isso não é de todo certo. Os desenvolvedores de software proprietário não são necessariamente incompetentes. Às vezes, desenvolvem programas robustos e confiáveis, ainda que desrespeitando a liberdade dos usuários. Os ativistas do software livre e os entusiastas do código aberto irão reagir de modo bem diferente neste caso.

Um entusiasta do código aberto, alguém que não é influenciado pelos ideais do software livre, dirá: “Estou surpreso que você conseguiu fazer um programa rodar bem sem usar nosso modelo de desenvolvimento, mas você conseguiu. Como obtenho uma cópia?”. Essa atitude irá beneficiar modelos que tiram a nossa liberdade.

Um ativista do software livre dirá: “Seu programa é muito atrativo, porém, valorizo mais a minha liberdade. Sendo assim, rejeito o seu programa. Ao invés disso, irei apoiar um projeto para desenvolver um substituto livre. Se nós valorizamos nossa liberdade, podemos agir para mantê-la e defendê-la.”

Nas tabelas 2.26 e 2.27, encontram-se as respostas de desenvolvedores e colaboradores do software livre/código aberto sobre as relações que, na sua percepção, as comunidades livres e pessoas vinculadas com o software proprietário possuem com aspectos econômicos. A discussão sobre ganhos financeiros é um tema que mobiliza muito fortemente a comunidade livre em virtude da ambiguidade inerente à palavra inglesa *free*. Como lembra Stallman (2014):

“O termo *free software* é ambíguo. A ideia, que não é a que queremos passar, de software que pode ser adquirido a custo zero se encaixa tão bem ao termo como a que queremos reforçar de software que concede ao usuário certas liberdades. Resolvemos esse problema de interpretação publicando a definição de software livre e sugerindo: pense em liberdade de expressão, não em cerveja grátis.”

Assim, é de se supor que, para defensores do software livre, não haveria qualquer fator impeditivo à busca por lucro. Ao contrário, até por motivos relacionados com o fortalecimento da imagem/marca do *free* como liberdade, haveria um interesse maior em enfatizar que o cerne das diferenças com o software proprietário não estaria baseado em questões envolvendo dinheiro.

O mesmo raciocínio também cabe para os que defendem o software de código aberto. Se, neste caso, a questão central está no modelo de desenvolvimento e na robustez e qualidade do produto final, então, em tese, não há nenhum impeditivo para a busca por lucro.

## Desenvolvedores no Brasil: software livre, código aberto e software proprietário

*Cerca de um terço dos desenvolvedores afirmam utilizar o melhor software, seja livre ou proprietário. Uma quantidade elevada acredita que as pessoas que trabalham com software proprietário são mais ligadas em dinheiro que as que trabalham com software livre/código aberto*

Parte relevante dos desenvolvedores (68,8%) acredita que o software livre/código aberto satisfaz os requisitos atuais de software melhor que o software proprietário. No que se refere ao uso de software ou nos projetos de desenvolvimento, quase a metade do grupo (48,8%) evita adotar software proprietário. Cerca de um terço (38,7%) afirma utilizar o melhor software, seja livre ou proprietário.

Apesar da tentativa da comunidade de software livre/código aberto de desvincular o movimento livre da ideia de software gratuito, uma quantidade elevada dos desenvolvedores (40,3%) acredita que as pessoas que trabalham com software proprietário são mais ligadas em dinheiro que as que trabalham com software livre/código aberto. O que poderia explicar esta percepção?

Provavelmente tenha a ver com a forma como a acumulação de capital ocorreu na era moderna e com a discussão que desencadeou sobre apropriação individual ou coletiva. O ponto de partida da discussão era a premissa que, em cenário econômico de escassez, o detentor dos meios de produção obtinha vantagens financeiras extraordinárias em prejuízo dos demais. Resultava daí a ideia que o uso compartilhado dos recursos escassos impediria a acumulação de capital nas mãos de poucos, reduzindo a ganância e assegurando justiça, liberdade e igualdade.

No modelo de comercialização de software proprietário, o autor, mediante a licença tradicional, detém a propriedade do software, reservando-se o direito a ganhos financeiros provenientes do seu uso por terceiros. Tendo em vista a discussão acima, é de se pensar que a apropriação individual propiciaria ganhos relevantes para o autor, em prejuízo de outros. No entanto, diferentemente dos meios de produção típicos da era industrial (terra, ferramentas, imóveis), o software é um bem intangível e, como tal, possui propriedades distintas dos bens tradicionais.

Se, por um lado, é correto que os produtos comercializados como software proprietário podem gerar ganhos para o autor, por outro, é certo, também, que os esforços para o desenvolvimento no modelo proprietário tendem a ser superiores. Pelo menos em teoria, na média, o modelo de desenvolvimento baseado na colaboração leva a uma redução sensível no esforço realizado por cada participante e, também em teoria, libera tempo de dedicação para que cada um se envolva em outras atividades lucrativas. Além disso, no modelo colaborativo, existem ganhos intangíveis importantes, relacionados com os processos rápidos de aprendizagem, ou seja, justamente, promovendo e facilitando a geração de conhecimento, uma evidência de que este bem não possui os atributos de escassez que caracterizam outros

meios de produção. O conhecimento não se gasta, como acontece com os bens tangíveis e escassos, mas cresce e se alastra, quanto mais se usa. Enfim, uma compreensão melhor das diferenças entre bens tangíveis e intangíveis poderia, talvez, destruir a percepção comum e difundida de que os adeptos do software proprietário obtenham lucros exorbitantes ou sejam mais ligados em dinheiro.

## Colaboradores no Brasil: software livre, código aberto e software proprietário

### Os colaboradores evitam utilizar software proprietário mais que os desenvolvedores

Entre os colaboradores, é ainda maior o percentual que considera que o software livre/código aberto atende melhor aos requisitos atuais de software. Esta é a opinião de 79,4% dos colaboradores. Eles tentam, mais que os desenvolvedores, evitar utilizar software proprietário. Apenas uma pequena parcela (18,9%) declara utilizar o melhor software, seja livre/código aberto ou proprietário. Um terço (35,3%) dos colaboradores acredita que pessoas que trabalham com software proprietário são mais ligadas em dinheiro que as que trabalham com software livre/código aberto.

Em comparação com os desenvolvedores, no que se refere aos atributos considerados em um software, os colaboradores parecem mais sintonizados com os ideais do software livre, o que, de fato, condiz com posição já declarada por eles, quando questionados a respeito da sua identificação com software livre e código aberto.

**TABELA 2.24 - EM TERMOS GERAIS, VOCÊ CONSIDERA QUE SOFTWARE LIVRE/CÓDIGO ABERTO SATISFAZ OS REQUISITOS ATUAIS DE SOFTWARE MELHOR QUE SOFTWARE PROPRIETÁRIO?**

SATISFAÇÃO MAIOR COM SOFTWARE SL/CA?	DESENVOLVEDORES	COLABORADORES
Sim	68,8%	79,4%
Não	2,6%	5,9%
Um não tem nada a ver com o outro	19,5%	11,8%
Não sei dizer	9,1%	2,9%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.25 – UTILIZA SOFTWARE QUE NÃO É LIVRE/CÓDIGO ABERTO NO SEU DIA-A-DIA OU PARA DESENVOLVIMENTO?**

USO DE SOFTWARE PROPRIETÁRIO	DESENVOLVEDORES	COLABORADORES
Não. Tento evitar utilizar software/serviços que não são livres ou de código aberto	48,8%	59,5%
Escolhi utilizar software livre/serviços que não são livres ou de código aberto para certas tarefas	12,5%	21,6%
Sim, sem problemas. Uso o melhor software/serviços a despeito de ser livre ou de código aberto ou não	38,7%	18,9%

Total	100,0%	100,0%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores: 80; colaboradores: 37.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.26 - PESSOAS QUE TRABALHAM COM SOFTWARE LIVRE/CÓDIGO ABERTO SÃO MAIS LIGADAS EM DINHEIRO DO QUE PESSOAS QUE TRABALHAM COM SOFTWARE PROPRIETÁRIO?**

RELAÇÃO SOFTWARE SL/CA COM DINHEIRO	DESENVOLVEDORES	COLABORADORES
Isso é bom	2,6%	2,9%
Isso é mau	0,0%	0,0%
Isso não é verdade	97,4%	97,1%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.27 - AS PESSOAS QUE TRABALHAM COM SOFTWARE PROPRIETÁRIO SÃO MAIS LIGADAS EM DINHEIRO DO QUE PESSOAS QUE TRABALHAM COM SOFTWARE LIVRE/CÓDIGO ABERTO?**

RELAÇÃO SOFTWARE PROPRIETÁRIO COM DINHEIRO	DESENVOLVEDORES	COLABORADORES
Isso é bom	9,1%	14,7%
Isso é mau	31,2%	20,6%
Isso não é verdade	59,7%	64,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## AMBIENTE DO SOFTWARE LIVRE/CÓDIGO ABERTO

As tabelas 2.28 e 2.29 trazem dados sobre o que desenvolvedores e colaboradores pensam sobre o ambiente do software livre/código aberto.

### Ambiente de software livre/código aberto segundo desenvolvedores

*Desenvolvedores veem a comunidade livre como fórum para troca de conhecimento, liberdade no desenvolvimento e discussão geral sobre software*

Questionados sobre o que é o fórum de software livre/código aberto, os desenvolvedores apontaram o fato de ser um local adequado para troca de conhecimento (67,5%); maior liberdade no desenvolvimento de software (53,1%) e discussão geral sobre software (39,0%).

Quase a totalidade dos desenvolvedores (97,4%) acredita que a vivência na comunidade de software livre/código aberto fornece oportunidades de trabalho.

### Ambiente de software livre/código aberto na percepção dos colaboradores

*Colaboradores preocupam-se mais com as diferenças existentes entre os que fazem parte e os que não fazem parte da comunidade*

Colaboradores pensam o mesmo. Do total de respostas, 67,6% destacam o fórum de software livre/código aberto como um local para troca de conhecimentos e 50,0% como um local com maior liberdade no desenvolvimento de software. No entanto, para os colaboradores, os próximos itens mais indicados foram o fórum se constituir em espaço para pessoas que utilizam novas formas de cooperação (41,1%) e pessoas com os mesmos interesses (35,2%). O senso de pertencimento à comunidade e a ênfase nas diferenças existentes com os que não fazem parte dela, ao invés da percepção do que une a todos, parecem fatores mais acentuados no grupo de colaboradores do que no conjunto de desenvolvedores.

Tal como no grupo de desenvolvedores, também é elevada a quantidade de colaboradores (88,2%) que acredita que a vivência na comunidade de software livre/código aberto fornece oportunidades de trabalho.

TABELA 2.28 - O AMBIENTE DE SOFTWARE LIVRE/CÓDIGO ABERTO É FÓRUM PARA...

AMBIENTE SL/CA	DESENVOLVEDORES	COLABORADORES
Uma discussão geral sobre software	39,0%	23,5%
Uma competição esportiva sobre o melhor código	0,0%	0,0%
Realizar inovações radicais	22,0%	8,9%
Prover imitações de produtos e serviços de software proprietário	7,9%	0,0%
Desenvolvedores de software que necessitam de uma <i>toolbox</i>	1,2%	5,9%
Prover mais variedade de software	28,5%	23,5%
Pessoas com os mesmos interesses	24,7%	35,2%
Pessoas que procuram parceiros de projeto	7,9%	11,7%
Pessoas que utilizam novas formas de cooperação	22,0%	41,1%
Maior liberdade no desenvolvimento de software	53,1%	50,0%

Melhorias na carreira	9,1%	3,1%
Troca de conhecimento	67,5%	67,6%
Pessoas que querem se divertir	10,3%	8,9%

Permite respostas múltiplas (máximo 3 opções). Percentual calculado sobre o número total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.29 - A VIVÊNCIA NA COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO TEM IMPACTO POSITIVO NAS OPORTUNIDADES DE TRABALHO?**

VIVÊNCIA NA COMUNIDADE IMPACTA TRABALHO?	DESENVOLVEDORES	COLABORADORES
Sim	97,4%	88,2%
Não	0,0%	2,9%
Não sei dizer	2,6%	8,8%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores: 34. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## BALANÇO ENTRE FORNECER E RECEBER

Nas tabelas 2.30 a 2.33, compara-se a percepção dos desenvolvedores e colaboradores da comunidade brasileira de software livre/código aberto no que diz respeito à interação entre os membros da comunidade e a percepção sobre dar e receber. Levanta-se a expectativa que os participantes possuem da atuação dos demais como a percepção que têm do que os outros esperam deles.

### Ação conjunta é superior à soma das ações individuais

No geral, entre os participantes da comunidade de software livre/código aberto, existe a crença disseminada de que o resultado da ação conjunta é superior à soma das ações individuais. A situação promove parceria do tipo ganha-ganha, em que todos recebem mais do que ganham.

### Desenvolvedores e o balanço entre dar e receber

Entre os desenvolvedores participantes da edição brasileira da pesquisa, não há uma posição predominante no que se refere ao balanço entre dar e receber. Parte do grupo (31,1%) não se importa com isto. Outra parte (32,5%) acredita que recebe mais do que fornece para o grupo. Os restantes ou se percebem fornecendo tanto como recebem (13,0%) ou fornecendo mais do que recebem (13,0%).

Avaliando o peso da contribuição de outros desenvolvedores para a comunidade de software livre/código aberto, 32,5% acreditam que os outros desenvolvedores também recebem mais que fornecem.

Um número relativamente maior, mas também baixo (18,2%), acha que os outros fornecem mais do que eles próprios. Um número menor (9,1%) também pensa que os demais se importam menos com isto do que eles próprios. Na realidade, eles parecem mais preocupados com o que outros pensam sobre a contribuição deles do que o que eles pensam sobre a contribuição dos outros. Isso pode ser um indicativo da posição que ocupam na hierarquia da comunidade. A sua maior maturidade faz com que se sintam mais responsáveis pelas contribuições.

**Os desenvolvedores acham que os outros esperam mais deles do que eles dos outros no que se refere à cooperação e compartilhamento. No entanto, esperam mais dos outros do que pensam que os outros deles no que se refere à melhoria dos produtos**

Em geral, as expectativas em relação às ações dos demais têm a ver com aspectos sociais e aprendizado. Perguntados sobre o que esperam dos outros desenvolvedores, 90,0% lembram o interesse de que compartilhem os seus conhecimentos e habilidades; 36,4%, que melhorem os produtos de software livre/código aberto dos demais; e, 46,8%, que estejam aptos para cooperar de uma nova maneira.

Ganhos econômicos e questões ideológicas não estão entre as maiores expectativas. Por exemplo, é baixa a quantidade relativa de respondentes que esperam que os outros desenvolvedores ganhem dinheiro com software livre/código aberto (10,4%) e que ajudem a restringir a ação das grandes empresas (7,8%). Perguntados sobre o que pensam que outros desenvolvedores esperam deles, 83,1% acham que há interesse em que compartilhem o seu conhecimento e as suas habilidades; 46,8% que estejam aptos para cooperar de uma nova maneira e 42,9% que sejam capazes de melhorar produtos de software livre/código aberto.

Na contabilidade dos desenvolvedores sobre dar e receber, parece existir a percepção que os outros esperam uma atuação de mais alto nível deles do que eles esperam dos outros, o que reforça a ideia de que os desenvolvedores participantes da edição brasileira da pesquisa possuem uma certa maturidade e percepção de que têm mais a oferecer dada a sua experiência na comunidade. Assim, por exemplo, enquanto 13,0% dos desenvolvedores esperam que os demais tomem parte nas discussões sobre software livre/código aberto, 31,2% acham que os demais esperam que eles tomem parte nas principais comunicações e discussões. Da mesma forma, uma quantidade maior (61,0%) espera que os demais estejam interessados em melhorar os seus produtos e uma quantidade menor (42,9%) julga que os outros desenvolvedores pensam em contar com a sua contribuição na melhoria de produtos.

## Colaboradores e o equilíbrio entre dar e receber

**Em relação aos desenvolvedores, colaboradores parecem se ver menos envolvidos no compartilhamento dos seus conhecimentos e mais voltados para a tomada de decisões relevantes para a comunidade**

No grupo dos colaboradores, quase a metade (44,1%) acredita que recebe mais do que fornece. Comparativamente ao grupo de desenvolvedores, no grupo de colaboradores, existe uma crença maior no ideal de igualdade, com a percepção de que todos contribuem da mesma maneira, independente das atividades que exercem e do empenho e dedicação que fornecem.

As percepções não são divergentes. Mas existem algumas diferenças de grau entre o que desenvolvedores e colaboradores esperam dos seus pares. Assim, por exemplo, entre os colaboradores é proporcionalmente menor o número daqueles com expectativa que os outros colaboradores compartilhem os seus conhecimentos e habilidades (79,4% contra 90,9% do total de desenvolvedores) e que os deixem aprender e desenvolver novas habilidades (14,7% contra 23,4%). No entanto, é consideravelmente maior a quantidade que espera dos demais colaboradores que tomem parte nas principais comunicações e discussões (32,4% em relação a 13,0% dos desenvolvedores) e que lhes forneçam melhores oportunidades de trabalho (11,8% contra 2,6%).

Entre os colaboradores, tal como já observado com os desenvolvedores, são poucos os que esperam que os demais obtenham ganhos econômicos ou contribuam para reduzir o poder das grandes empresas.

*Da mesma forma que entre os desenvolvedores, entre os colaboradores, o número que acredita que os demais desejam que eles tomem parte nas principais ações e discussões é superior à quantidade que espera esta ação dos demais*

No grupo dos colaboradores é proporcionalmente menor a quantidade que acha que outros esperam que eles compartilhem o seu conhecimento (67,6% em relação a 83,1% de desenvolvedores). Quase a metade (44,1%) acredita que os demais desejam que eles tomem parte nas principais ações e discussões, uma proporção mais elevada que a encontrada entre os desenvolvedores (31,2%). Tal como no caso dos desenvolvedores, o número de colaboradores que acredita que esta atividade é requerida deles é superior à quantidade que espera esta ação dos demais.

**TABELA 2.30 – DESENVOLVEDORES E COLABORADORES CONTRIBUEM PARA A COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO DE DIFERENTES MODOS, E RECEBEM MUITOS BENEFÍCIOS EM TROCA. EM GERAL, COMO VOCÊ AVALIA O PESO DA BALANÇA?**

COMO VOCÊ AVALIA O PESO DA BALANÇA?	DESENVOLVEDORES	COLABORADORES
Forneço mais que recebo	13,0%	8,8%
Recebo mais que forneço	32,5%	44,1%
Forneço tanto como recebo	13,0%	17,7%
Não me importo	31,1%	17,7%
Não sei dizer	10,4%	11,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34.  
Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.31 - QUANDO VOCÊ CONSIDERA OUTROS DESENVOLVEDORES OU COLABORADORES DE SOFTWARE LIVRE/CÓDIGO ABERTO, COMO VOCÊ JULGA QUE ELES PENSAM?

COMO OS OUTROS AVALIAM O PESO DA BALANÇA?	DESENVOLVEDORES	COLABORADORES
A maioria fornece mais que recebe	18,2%	17,7%
A maioria recebe mais que fornece	32,5%	47,1%
Fornecem tanto quanto recebem	7,8%	20,6%
Não se importam com isso	9,1%	2,9%
Não me importo com isso	15,6%	2,9%
Não sei dizer	16,8%	8,8%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.32 - O QUE VOCÊ ESPERA DE OUTROS DESENVOLVEDORES/COLABORADORES DE SOFTWARE LIVRE/CÓDIGO ABERTO?

O QUE VOCÊ ESPERA DOS DEMAIS?		DESENVOLVEDORES	COLABORADORES
Ganhos sociais - Ambiente	Estar apto para cooperar de uma nova maneira	46,8%	52,9%
	Tomar parte nas principais comunicações e discussões	13,0%	32,4%
	Respeitar-me e as minhas contribuições para software livre/código aberto	28,6%	23,5%
Aprendizado	Deixar-me aprender e desenvolver novas habilidades	23,4%	14,7%
	Compartilhar os seus conhecimentos e habilidades	90,0%	79,4%
	Melhorar produtos de software livre/código aberto de outros desenvolvedores	36,4%	58,8%
Solução de problemas	Escrever programas adequados desde o ponto de vista estético	3,9%	0,0%
	Contribuir para concretizar ideias de produtos de software	7,8%	2,9%
	Resolver um problema que não poderia ser resolvido por software proprietário	20,8%	20,6%
Ganhos econômicos	Fornecer melhores oportunidades de trabalho	2,6%	11,8%
	Distribuir software não comercial	7,8%	11,8%
	Ganhar dinheiro	10,4%	11,8%

<b>Ideologia</b>	Contribuir para limitar o poder das grandes empresas de software	7,8%	2,9%
	Não sei dizer	0,0%	0,0%

Permite respostas múltiplas (máximo de 4 opções). Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

**TABELA 2.33 - O QUE ACHA QUE OS OUTROS DESENVOLVEDORES/COLABORADORES DE SOFTWARE LIVRE/CÓDIGO ABERTO ESPERAM DE VOCÊ?**

O QUE VOCÊ ACHA QUE OS OUTROS ESPERAM DE VOCÊ?		DESENVOLVEDORES	COLABORADORES
<b>Ganhos sociais - Ambiente</b>	Estar apto para cooperar de uma nova maneira	46,8%	47,1%
	Tomar parte nas principais comunicações e discussões	31,2%	44,1%
	Respeitá-los e as suas contribuições para software livre/código aberto	37,7%	29,4%
<b>Aprendizado</b>	Deixá-los aprender e desenvolver novas habilidades	29,9%	23,5%
	Que eu compartilhe os meus conhecimentos e habilidades	83,1%	67,6%
	Melhorar produtos de software livre/código aberto de outros desenvolvedores	42,9%	29,4%
<b>Solução de problemas</b>	Escrever programas adequados desde o ponto de vista estético	10,8%	0,0%
	Contribuir para concretizar ideias de produtos de software	26,0%	23,5%
	Resolver um problema que não poderia ser resolvido por software proprietário	16,9%	14,7%
<b>Ganhos econômicos</b>	Fornecer melhores oportunidades de trabalho	6,5%	5,9%
	Distribuir software não comercial	7,8%	2,9%
	Ganhar dinheiro	6,5%	17,6%
<b>Ideologia</b>	Contribuir para limitar o poder das grandes empresas de software	3,9%	8,8%
Não sei dizer		2,6%	2,6%

Permite respostas múltiplas (máximo de 4 opções). Percentual calculado sobre o total de participantes que responderam à questão: desenvolvedores = 77; colaboradores = 34. Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## DESAFIOS A SEREM ENFRENTADOS

Nas tabelas 2.34 e 2.35, tendo em vista desafios que devem ser enfrentados durante o processo de desenvolvimento de software livre/código aberto, apresenta-se a distribuição do número de desenvolvedores e colaboradores, considerando as seguintes opções: não é relevante, é importante, é importante e eu podia fazer mais por isto, preocupa-me mas não sei o que fazer, participo deste desafio, este desafio não tem nada a ver comigo.

### Desafios da comunidade de software livre/código aberto

Para os desenvolvedores, de uma lista abrangente de ações, a redução de *bugs* (67,5%) e a comunicação interna do projeto (58,4%) foram consideradas as mais importantes. Eles acreditam que poderiam fazer mais pela documentação dos projetos e pela manutenção de usuários e colaboradores engajados. Suas maiores preocupações têm a ver com a atração de usuários e colaboradores e com a obtenção de recursos financeiros. Alguns se veem envolvidos na redução do tempo entre a liberação de novas versões e, também, nos esforços para adequar o software às novas demandas, incluindo uso em plataformas múltiplas, mobilidade e modelo de negócios *as a service*. No entanto, apenas um terço dos desenvolvedores considera que estas sejam atividades relevantes.

Para os colaboradores, questões legais e a comunicação interna foram apontados como os desafios mais importantes (67,7% e 61,8%, respectivamente). As preocupações dos colaboradores estão voltadas, sobretudo, para a obtenção de recursos financeiros. Eles se veem participando de temas relacionados com documentação, aspectos legais e comunicação interna.

Para alguns desafios, desenvolvedores e colaboradores têm percepções distintas no que diz respeito a sua relevância e a sua capacidade de contribuir. Para outros desafios, parece existir um maior consenso entre desenvolvedores e colaboradores. As diferenças provavelmente têm a ver com as funções destes grupos na comunidade livre. Assim, atrair usuários parece ser um assunto em que os colaboradores estão muito envolvidos; mantê-los é problema dos desenvolvedores. Reduzir *bugs* tem muita importância para os desenvolvedores. No entanto, tratar de questões legais é assunto de maior relevância para os colaboradores. A obtenção de recursos financeiros é tema que preocupa a ambos os grupos, que concordam, também, que poderiam fazer mais pela documentação dos projetos. Tanto desenvolvedores como colaboradores tendem a concordar que alavancar o poder de empresa líder e adaptar o software, tornando-o multiplataforma, mais adequado ao ambiente de mobilidade e baseado em modelo *as a service* não têm nada a ver com eles.

TABELA 2.34 – POSSÍVEIS DESAFIOS QUE TÊM DE SER ENFRENTADOS NOS PRINCIPAIS PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO NO QUAL ESTÁ ENVOLVIDO, AVALIANDO A IMPORTÂNCIA ATRIBUÍDA A CADA UM DELES

Para desenvolvedores

ITEM	NÃO É IMPORTANTE	É IMPORTANTE	É IMPORTANTE E EU PODIA FAZER MAIS POR ISTO	PREOCUPA-ME, MAS NÃO SEI O QUE FAZER	PARTICIPO DISTO	ISTO NÃO TEM NADA A VER COMIGO	NÃO SEI DIZER
Tornar o software estável (reduzir <i>bugs</i> )	0,0%	67,5%	18,2%	0,0%	11,7%	0,0%	2,6%
Definir recursos/características	3,9%	50,7%	22,1%	6,5%	9,1%	2,6%	5,1%
Documentação	1,3%	49,3%	28,6%	5,2%	10,4%	2,6%	2,6%
Tornar o software multiplataforma/móvel/SaaS	10,4%	33,8%	19,5%	5,2%	13,0%	14,3%	3,8%
Reduzir o tempo entre versões	31,2%	26,0%	10,4%	5,2%	13,0%	5,2%	9,0%
Atrair usuários/ colaboradores	2,6%	50,7%	18,2%	13,0%	9,1%	5,2%	1,3%
Manter usuários/ colaboradores engajados	1,3%	48,0%	25,6%	10,4%	7,8%	1,3%	5,2%
Alavancar o poder da empresa líder	33,8%	26,0%	9,1%	3,9%	3,9%	11,7%	11,6%
Obter recursos financeiros	13,0%	44,2%	13,0%	10,3%	6,5%	5,2%	7,8%
Questões legais (licença, <i>copyright</i> , etc.).	5,2%	57,1%	7,8%	5,2%	11,7%	5,2%	7,8%
Comunicação interna do projeto	0,0%	58,4%	20,8%	2,6%	10,4%	3,9%	3,9%
Comunicação ao longo da cadeia de valor	13,0%	33,8%	14,3%	2,6%	13,0%	6,5%	16,8%

Percentual calculado sobre o total de desenvolvedores que responderam a cada um dos itens da pergunta: 77.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

TABELA 2.35 – POSSÍVEIS DESAFIOS QUE TÊM DE SER ENFRENTADOS NOS PRINCIPAIS PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO NO QUAL ESTÁ ENVOLVIDO, AVALIANDO A IMPORTÂNCIA ATRIBUÍDA A CADA UM DELES

Para colaboradores

ITEM	NÃO É IMPORTANTE	É IMPORTANTE	É IMPORTANTE E EU PODIA FAZER MAIS POR ISTO	PREOCUPA-ME, MAS NÃO SEI O QUE FAZER	PARTICIPO DISTO	ISTO NÃO TEM NADA A VER COMIGO	NÃO SEI DIZER
Tornar o software estável (reduzir <i>bugs</i> )	2,9%	47,1%	26,5%	5,9%	2,9%	14,7%	0,0%
Definir recursos/características	2,9%	58,8%	14,7%	2,9%	8,8%	11,8%	0,0%
Documentação	5,9%	32,4%	38,2%	2,9%	14,7%	5,9%	0,0%
Tornar o software multiplataforma/móvel/SaaS	2,9%	52,9%	11,8%	5,9%	2,9%	20,6%	2,9%
Reduzir o tempo entre versões	20,6%	38,2%	14,7%	5,9%	0,0%	17,7%	2,9%
Atrair usuários/ colaboradores	2,9%	52,9%	29,4%	0,0%	11,8%	2,9%	0,0%
Manter usuários/ colaboradores engajados	2,9%	55,9%	26,5%	2,9%	11,8%	0,0%	0,0%
Alavancar o poder da empresa líder	23,5%	38,2%	0,0%	5,9%	5,9%	20,6%	5,9%
Obter recursos financeiros	5,9%	50,0%	5,9%	11,8%	5,9%	17,7%	2,9%
Questões legais (licença, <i>copyright</i> , etc.).	2,9%	67,7%	2,9%	2,9%	14,7%	5,9%	2,9%
Comunicação interna do projeto	2,9%	61,8%	5,9%	0,0%	14,7%	11,8%	2,9%
Comunicação ao longo da cadeia de valor	14,7%	41,2%	5,9%	2,9%	8,8%	17,7%	8,8%

Percentual calculado sobre o total de colaboradores que responderam a cada um dos itens da pergunta: 34.  
 Fonte: Observatório Softex, 2014. Pesquisa SL/CA Brasil 2014.

## 2.2 COMPARAÇÕES ENTRE AS PESQUISAS SL/CA BRASIL 2014 E FLOSS 2013

### PERFIL DOS PARTICIPANTES

Os participantes da edição FLOSS 2013 nasceram e residem em diferentes locais. No entanto, o lugar de moradia de quase setenta por cento deles concentra-se em dez países, com forte presença dos Estados Unidos (27,2% do total de respostas válidas) e da União Europeia, destacando-se Alemanha (8,3%), Reino Unido (6,9%) e Espanha (6,8%).

A edição internacional da pesquisa contou com 2.183 respondentes. A maioria não havia participado da pesquisa de 2002 (93,2%), o que pode indicar uma renovação importante dos integrantes da comunidade ou, também, dificuldades dos organizadores da pesquisa para acessar a antiga comunidade através dos meios utilizados para divulgação. A participação dos desenvolvedores ou de pessoas que desempenham ambas as funções, desenvolvimento e colaboração, é ainda maior que a observada na versão brasileira: 73,2% do total de respondentes.

**Em comparação com a edição brasileira, na edição internacional, observa-se presença maior de desenvolvedores, mulheres e pessoas jovens. Engenheiros de software têm participação elevada no conjunto**

Nas tabelas 2.36 a 2.44, compara-se o perfil de respondentes das edições brasileira e internacional da pesquisa. Observa-se que, no conjunto brasileiro, há relativamente mais pessoas do sexo masculino, casadas ou com união estável e com filhos. Dentro do subgrupo com filhos, no conjunto brasileiro há um percentual maior de pessoas com filhos já crescidos.

Entre os respondentes da pesquisa internacional, há uma quantidade proporcionalmente maior de pessoal com nível de pós-graduação, tanto mestrado como doutorado. Em termos profissionais, a composição dos respondentes da edição brasileira é mais heterogênea. Destaca-se a participação relativamente elevada de executivos de TI (14,8%) e consultores de TI (15,6%) no conjunto Brasil. Chama atenção a participação alta de engenheiros de software na edição internacional (40,3%, contra 19,7%, edição brasileira), embora, em ambas as pesquisas, eles sejam a maioria. A participação de programadores é semelhante naquela e nesta pesquisa.

]Em ambas as pesquisas, predominam as pessoas empregadas. Na versão internacional, existe uma quantidade relativamente maior de estudantes e desempregados e, na brasileira, de trabalhadores por conta própria.

Na edição internacional, é maior a participação de pessoas que recebem salários mensais baixos (valor inferior a R\$2.515) e, também, elevados (superior a R\$ 10.060). Em ambas as pesquisas, a faixa de salários com maior pontuação foi a superior a R\$ 10.060 (29,8% do total dos participantes da edição brasileira e 36,8% dos participantes da edição internacional).

O conjunto constituído pelos respondentes da edição brasileira parece, na média, ser formado por pessoas mais velhas e com maior maturidade pessoal e profissional. Considerando alguns aspectos, por exemplo, situação empregatícia e faixa de remuneração salarial, o perfil dos respondentes da edição interna-

cional mostra-se mais heterogêneo. Levando em consideração outros aspectos, por exemplo, a profissão dos participantes, é mais homogêneo.

**TABELA 2.36 – PARTICIPANTES DA PESQUISA SOFTWARE LIVRE/CÓDIGO ABERTO, VERSÃO BRASILEIRA E VERSÃO INTERNACIONAL**

<b>GRUPO A QUE PERTENCE</b>	<b>SL/CA BRASIL 2014</b>		<b>FLOSS INTERNACIONAL 2013</b>	
Desenvolvedores ou ambos	86	68,8%	1.578	73,2%
Colaboradores	39	31,2%	577	26,8%
TOTAL	125	100,0%	2.155	100,0%

Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.37 - QUAL O SEU GÊNERO?**

<b>GÊNERO</b>	<b>SL/CA BRASIL 2014</b>	<b>FLOSS INTERNACIONAL 2013</b>
Masculino	95,2%	88,7%
Feminino	4,8%	11,3%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 126; para FLOSS Internacional 2013: 2.002. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.38 - SITUAÇÃO CONJUGAL?**

<b>SITUAÇÃO CONJUGAL</b>	<b>SL/CA BRASIL 2014</b>	<b>FLOSS INTERNACIONAL 2013</b>
Solteiro	23,3%	32,8%
Casado ou união estável/outro	76,7%	67,2%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 116; para FLOSS Internacional 2013: 2.021. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.39 - POSSUI FILHOS?

POSSUI FILHOS?	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Sim	61,9%	32,2%
Não	38,1%	67,8%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 126; para FLOSS Internacional 2013: 2.021. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.40 - FAIXA DE IDADE DO FILHO (SE MAIS DE UM, INFORMAR A FAIXA DE IDADE DO FILHOCAÇULA)?

FAIXA ETÁRIA	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Inferior a 2 anos	14,1%	28,3%
Entre 2 e 5 anos	28,2%	21,6%
Entre 5 anos e 10 anos	20,5%	21,3%
Superior a 10 anos	37,2%	28,8%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam que tinham filhos. Para SL/CA Brasil 2014: 78; para FLOSS Internacional 2013: 629. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.41 - NÍVEL DE ESCOLARIDADE (CONSIDERANDO CURSO COMPLETO DE MAIOR NÍVEL)?

ESCOLARIDADE	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Fundamental	1,1%	1,3%
Médio	15,9%	14,4%
Superior	47,7%	43,6%
Mestrado	27,3%	32,1%
Doutorado	8,0%	8,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão e indicaram uma das opções mencionadas na tabela. Para SL/CA Brasil 2014: 88; para FLOSS Internacional 2013: 1.858. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.42 - PROFISSÃO?

PROFISSÃO/ATIVIDADE	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Engenheiro de software	19,7%	40,3%
Engenheiro (outro)	3,3%	3,2%
Programador	12,3%	11,1%
Consultor de TI	15,6%	8,0%
Consultor (outros setores)	1,6%	1,4%
Executivo (TI)	14,8%	2,7%
Executivo (outros setores)	1,6%	0,6%
Marketing (TI)	0,0%	0,3%
Marketing (outros setores)	0,0%	0,2%
Venda produtos de TI	2,5%	0,2%
Venda outros produtos	0,0%	0,2%
Professor universitário (área de TI)	5,7%	4,2%
Professor universitário (outros setores)	3,3%	3,4%
Estudante de TI	0,8%	4,6%
Estudante outros setores	0,0%	3,5%
Outra (em TI)	14,8%	8,2%
Outra (outros setores)	4,1%	7,9%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 122; para FLOSS Internacional 2013: 1.963. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.43 - SITUAÇÃO EMPREGATÍCIA?

SITUAÇÃO EMPREGATÍCIA	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Trabalhador por conta própria	23,8%	19,7%
Empregado	65,6%	66,7%
Desempregado	0,8%	3,3%
Trabalho não remunerado (estudante)	1,6%	6,0%
Trabalho não remunerado (aposentado, etc.)	0,0%	1,6%
Outra	8,2%	2,6%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 122; para FLOSS Internacional 2013: 1.967. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.44 - FAIXA DE RENDIMENTO BRUTO MENSAL

FAIXA SALARIAL	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Sem salário	3,3%	5,9%
Inferior a R\$ 2.515	6,6%	13,0%
Entre R\$ 2.515 a R\$ 5.030	24,8%	18,7%
Entre R\$ 5.030 a R\$ 7.545	18,2%	13,6%
Entre R\$ 7.545 a R\$ 10.060	17,4%	12,0%
Superior a R\$ 10.060	29,8%	36,8%
Total	100,0%	100,0%

Taxa de conversão Euro/RS: em 20/12/2014 = 2,51505. Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 121; para FLOSS Internacional 2013: 1.818. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## VISÃO SOBRE SOFTWARE LIVRE E CÓDIGO ABERTO

*Na edição internacional, tanto entre desenvolvedores como entre colaboradores, é maior a quantidade relativa que se importa com a distinção entre software livre e código aberto e que se identifica com a comunidade de código aberto. Também é maior o percentual que acha que a diferença entre software livre e código aberto é apenas de princípios*

Conforme Tabela 2.45, na comparação entre os respondentes da edição brasileira e da versão internacional, fica nítida a participação bastante relevante, nesta pesquisa, daqueles que se identificam com a comunidade de código aberto (35,6%), quase o mesmo percentual dos que se identificam com o software livre (38,5%). Na edição brasileira, são poucos os que se identificam com código aberto (18,0%).

Na edição brasileira, há uma quantidade relativamente maior de desenvolvedores que pensam que as comunidades de software livre e código aberto são diferentes no modo de pensar e de viver (34,9% do total, contra 23,4% dos participantes da FLOSS internacional) e que não se importam com as diferenças (42,2% e 31,9%, respectivamente). Entre os colaboradores, na edição brasileira, também é maior o percentual que pensa que software livre e código aberto são modos diferentes de pensar e viver (38,5% contra 27,8%). No entanto, diferentemente do caso dos desenvolvedores, há um percentual menor de pessoas que não se importam. Isso ocorre tanto na edição brasileira como na edição internacional. Ou seja, os colaboradores parecem dar maior atenção que os desenvolvedores para a distinção e, em comparação com os respondentes internacionais, os participantes da edição brasileira tendem a valorizar mais as diferenças entre as duas comunidades (Tabela 2.46).

A comunidade internacional parece estar mais dividida entre software livre e código aberto. Para ela, a distinção importa, mas é apenas uma questão de princípios. Para a comunidade brasileira, há um percentual maior para o qual a distinção não importa, mas para aqueles que se importam com a distinção, ela é mais relevante, assim como a identificação com o software livre.

**TABELA 2.45 - VOCÊ SE VÊ COMO PARTE DA COMUNIDADE DE SOFTWARE LIVRE OU COMO PARTE DA COMUNIDADE DE CÓDIGO ABERTO?**

SOFTWARE LIVRE OU CÓDIGO ABERTO?	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Identifico-me como parte da comunidade de software livre	43,4%	38,5%
Identifico-me como parte da comunidade de código aberto	18,0%	35,6%
A distinção não me importa	38,5%	25,9%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 122. Para FLOSS Internacional 2013: 1.935. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.46 – VOCÊ PENSA QUE AS COMUNIDADES DE SOFTWARE LIVRE E CÓDIGO ABERTO SÃO DIFERENTES?**

COMUNIDADES SÃO DIFERENTES?	DESENVOLVEDORES <sup>1</sup>		COLABORADORES <sup>2</sup>	
	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Sim, não é apenas um modo diferente de pensar, mas um modo diferente de vida.	34,9%	23,4%	38,5%	27,8%
Sim, mas somente em seus princípios. O modo como eles trabalham é o mesmo.	22,9%	44,7%	38,5%	49,7%
Isso não me importa. Apenas programo.	42,2%	31,9%	23,1%	22,5%
Total	100,0%	100,0%	100,0%	100,0%

(1) Percentual calculado sobre o total de desenvolvedores que responderam à questão: para SL/CA Brasil 2014 = 83; para FLOSS Internacional 2013 = 1.416. (2) Percentual calculado sobre o total de colaboradores que responderam à questão: para SL/CA Brasil 2014 = 39; para FLOSS Internacional 2013 = 515. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## INÍCIO E TEMPO DE DEDICAÇÃO

Nas tabelas 2.47 a 2.52, apresentam-se dados sobre início e tempo de dedicação dos respondentes das pesquisas brasileira e internacional a projetos de software livre/código aberto e de software proprietário.

Em ambas as edições, a maioria dos respondentes iniciou a sua contribuição a projetos de software livre/código aberto na década de 2000 (55,7% dos respondentes da edição brasileira e 56,5% da edição internacional). Na edição internacional, encontra-se um número relativamente maior de pessoas que iniciaram a

partir de 2010 e, na pesquisa brasileira, que iniciaram na década de 1990, mais um indício da maior participação de pessoas experientes e maduras no conjunto brasileiro de respondentes.

O início de atuação na comunidade livre tende a ocorrer cedo, quando o respondente está na faixa de 16 a 25 anos de idade. Essa é a faixa de idade predominante tanto na pesquisa realizada no Brasil como na edição FLOSS internacional. Mostra que, para atrair novos integrantes, as comunidades precisam conquistar os jovens ainda durante o seu período de estudos de nível médio ou universitário.

**Menos de seis horas semanais: esta é a média de tempo de dedicação a projetos de software livre/código aberto de cerca da metade dos participantes das edições brasileira e internacional da pesquisa. Em ambas as pesquisas, algo em torno de 20% do total de respondentes dedicou mais de 20 horas semanais aos projetos livres**

É pouco o tempo médio de dedicação reservado pelos respondentes para projetos de software livre/código aberto. Para 50,2% da comunidade internacional e 48,3% da brasileira, considerando o ano anterior à pesquisa, o tempo de dedicação foi inferior a seis horas semanais. Entre os participantes internacionais, um número relativamente maior dedicou mais de 20 horas semanais ao software livre/código aberto (23,5% do total contra 20,5% dos participantes brasileiros).

**Em ambas as pesquisas, cerca da metade dos desenvolvedores de software livre/código aberto desenvolvem, também, software proprietário. No grupo internacional, é maior o seu tempo de dedicação ao software proprietário**

No conjunto de desenvolvedores participantes da pesquisa internacional, o percentual que também desenvolve software proprietário é ligeiramente superior ao verificado na pesquisa brasileira. Em ambos os casos, mais da metade dos desenvolvedores que participaram da pesquisa desenvolvem software proprietário.

Em comparação com o resultado encontrado para a edição brasileira, os desenvolvedores participantes da FLOSS internacional 2013 começaram a se envolver mais tarde com software proprietário. No entanto, em ambas as edições, quase a metade iniciou desenvolvimento de software proprietário entre 2000 e 2009.

Os desenvolvedores que participaram da pesquisa internacional dedicaram uma quantidade maior de horas semanais ao desenvolvimento de software proprietário que os participantes da edição brasileira que se envolveram com este tipo de software. No ano anterior à pesquisa, 68,3% daquele grupo declararam gastar, em média, mais de vinte horas semanais com desenvolvimento de software proprietário. No conjunto de respondentes da pesquisa brasileira, 44,2%. No grupo internacional, 21,0% dedicaram até 10 hs. semanais ao desenvolvimento de software proprietário; na edição brasileira, este percentual se eleva para 34,2%.

**Entre os participantes da edição internacional, é maior a proporção de colaboradores que não se envolve com software proprietário. Entre os que se envolvem, parte significativa iniciou o envolvimento em anos recentes e dedicou um tempo maior a este tipo de software, comparativamente aos colaboradores brasileiros**

Diferentemente do que ocorre com os desenvolvedores de software, parte significativa dos participantes da edição internacional da pesquisa que colabora com software livre/código aberto (73,1%) não se envolve com software proprietário. Na pesquisa brasileira, o percentual dos que não se envolve é similar ao verificado para o conjunto constituído pelos desenvolvedores: um pouco menos da metade dos respondentes.

O envolvimento da comunidade brasileira de colaboradores com software proprietário ocorreu cedo (década de 1980 e, também, de 1990). Aparentemente, portanto, este grupo se envolveu antes com software proprietário e, posteriormente, passou a contribuir com software livre/código aberto. A distribuição encontrada para o Brasil é bastante distinta da encontrada para a pesquisa internacional. Neste caso, a minoria de pessoas que contribuem e que se envolvem com software proprietário iniciou o envolvimento mais recentemente (entre 2000 e 2009).

Enquanto, na comunidade internacional, os colaboradores que se envolvem com software proprietário tendem a fazer isto por várias horas semanais, o envolvimento dos colaboradores da edição brasileira com software proprietário acontece por um número menor de horas (para 50,0% deles, menos de duas horas semanais).

**TABELA 2.47 - QUANDO COMEÇOU A CONTRIBUIR COM PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO?**

PERÍODO	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Antes de 1960	0,0%	0,1%
Entre 1960 e 1969	0,0%	0,0%
Entre 1970 e 1979	0,0%	0,7%
Entre 1980 e 1989	3,3%	2,5%
Entre 1990 e 1999	27,9%	19,1%
Entre 2000 e 2009	55,7%	56,5%
2010 e adiante	13,1%	21,1%
Total	100,0%	100,0%

Para FLOSS Internacional 2013: 1.818. Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 122. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional: FLOSS, 2013.

**TABELA 2.48 - MINHA IDADE NA ÉPOCA ERA...**

FAIXA ETÁRIA	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
10 ou menos	0,8%	0,9%
de 11 a 15	0,8%	5,2%
de 16 a 20	26,2%	26,0%
de 21 a 25	35,2%	31,4%

de 26 a 30	13,1%	18,6%
de 31 a 35	9,0%	8,8%
de 36 a 40	5,7%	3,9%
de 41 a 45	6,6%	2,0%
de 46 a 50	1,6%	1,7%
de 51 a 55	0,8%	0,8%
56 ou mais	0,0%	0,7%
Total	100,0%	100,0%

Para FLOSS Internacional 2013: 1.664. Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 122. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.49 – TEMPO MÉDIO DE DEDICAÇÃO A PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO, CONSIDERANDO ANO ANTERIOR À PESQUISA**

### Em horas semanais

MÉDIA DE DEDICAÇÃO SEMANAL	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Menos de 2 horas	18,0%	28,5%
De 2 a 5 horas	30,3%	21,7%
De 6 a 10 horas	18,0%	14,9%
De 11 a 20 horas	13,1%	11,3%
De 21 a 40 horas	13,1%	11,2%
Mais de 40 horas	7,4%	12,3%
Total	100,0%	100,0%

Para FLOSS Internacional 2013: 1.824. Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 122. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.50 – TAMBÉM DESENVOLVE/SE ENVOLVE COM SOFTWARE PROPRIETÁRIO?**

DESENVOLVE/ SE ENVOLVE?	DESENVOLVEDORES <sup>1</sup>		COLABORADORES <sup>2</sup>	
	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Sim	51,8%	54,2%	56,4%	26,9%
Não	48,2%	45,8%	43,6%	73,1%
Total	100,0%	100,0%	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. (1) Para SL/CA Brasil 2014: 83; para FLOSS Internacional 2013: 1.347. (2) Para SL/CA Brasil 2014: 39; para FLOSS Internacional 2013. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.51 – QUANDO COMEÇOU A DESENVOLVER/SE ENVOLVER COM SOFTWARE PROPRIETÁRIO?

PERÍODO	DESENVOLVEDORES <sup>1</sup>		COLABORADORES <sup>2</sup>	
	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Antes de 1960	0,0%	0,1%	0,0%	0,0%
Entre 1960 e 1969	0,0%	0,0%	0,0%	0,8%
Entre 1970 e 1979	0,0%	1,5%	0,0%	3,2%
Entre 1980 e 1989	16,3%	5,6%	45,5%	14,3%
Entre 1990 e 1999	23,3%	24,6%	31,8%	28,6%
Entre 2000 e 2009	46,5%	49,2%	13,6%	37,3%
2010 e adiante	14,0%	19,0%	9,1%	15,8%
Total	100,0%	100,0%	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão e que afirmaram desenvolver software proprietário. (1) Para SL/CA Brasil 2014: 43; Para FLOSS Internacional 2013: 714. (2) Para SL/CA Brasil 2014: 22; Para FLOSS Internacional 2013: 126. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.52 – TEMPO MÉDIO DE DEDICAÇÃO A PROJETOS ENVOLVENDO SOFTWARE PROPRIETÁRIO, CONSIDERANDO ANO ANTERIOR À PESQUISA

HORAS DE DEDICAÇÃO	DESENVOLVEDORES <sup>1</sup>		COLABORADORES <sup>2</sup>	
	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Menos de 2 hs.	16,3%	7,0%	50,0%	21,0%
De 2 a 5 hs.	16,3%	7,0%	22,7%	12,9%
De 6 a 10 hs.	11,6%	7,0%	18,2%	5,6%
De 11 a 20 hs.	11,6%	10,9%	9,1%	8,1%
De 21 a 40 hs.	32,6%	46,7%	0,0%	39,5%
Mais de 40 hs.	11,6%	21,6%	0,0%	12,9%
Total	100,0%	100,0%	100,0%	100,0%
Total	100,0%	100,0%	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão e que afirmaram desenvolver software proprietário. (1) Para SL/CA Brasil 2014: 43; para FLOSS Internacional 2013: 718. (2) Para SL/CA Brasil 2014: 22; para FLOSS Internacional 2013: 124. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## MOTIVAÇÃO PARA PARTICIPAR DA COMUNIDADE

Nas tabelas 2.53 e 2.54, apresentam-se os motivos que levaram os integrantes da comunidade de software livre/código aberto brasileira e internacional a iniciarem participação na comunidade e a permanecerem nela.

*Em ambas as edições, o aprendizado é apontado como motivo principal para começar a participar na comunidade livre. Para a comunidade brasileira, o ambiente jogou um papel mais relevante na motivação de desenvolvedores e colaboradores*

Em ambas as edições, os fatores ideológicos foram pouco ressaltados pelos respondentes como motivo para iniciar a participação na comunidade livre/código aberto. Em contraste, motivos relacionados com o aprendizado receberam um número elevado de indicações.

Em relação aos resultados da edição brasileira, na edição internacional, tanto para desenvolvedores como para colaboradores, é maior o número relativo de respondentes cuja motivação para iniciar na comunidade livre tem a ver com aprendizado e com questões ideológicas. Em contraste, é menor o interesse dos participantes internacionais com o ambiente constituído pela comunidade livre.

O ambiente continua sendo uma justificativa mais utilizada por desenvolvedores e colaboradores brasileiros para continuar participando da comunidade livre. Em contraste, a ideologia é fator proporcionalmente mais ressaltado como motivação para os respondentes da pesquisa internacional permanecerem na comunidade.

Fatores relacionados com ganhos econômicos são mais destacados como motivo para permanecer na comunidade livre do que vir a se juntar a ela. Isso ocorre em ambas as edições. Na versão brasileira, entre os motivos para continuar, cai drasticamente o interesse pela solução de problemas concretos. Esse interesse, no entanto, aumenta entre os respondentes da pesquisa internacional.

**TABELA 2.53- NO PASSADO, QUAL FOI A RAZÃO PARA COMEÇAR A PARTICIPAR DE PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO?**

CATEGORIAS MOTIVACIONAIS	SL/CA BRASIL 2014		FLOSS INTERNACIONAL 2013	
	DESENVOLVEDORES	COLABORADORES	DESENVOLVEDORES	COLABORADORES
Aprendizado	42,8%	39,2%	46,9%	41,6%
Ambiente	27,1%	32,0%	21,8%	22,9%
Ganhos econômicos	12,7%	8,0%	10,7%	9,3%
Solução de problemas concretos	7,7%	11,2%	9,6%	8,2%
Ideologia	9,7%	9,6%	10,4%	16,7%
Total	100,0%	100,0%	100,0%	100,0%

As categorias incluem os seguintes fatores motivacionais: Aprendizado = aprender e desenvolver novas habilidades; compartilhar o meu conhecimento e habilidades; melhorar produtos e outros. Ambiente = participar em novas formas de cooperação; participar no cenário de

software livre/código aberto; conseguir reputação no cenário de software livre/código aberto. Ganhos econômicos = melhorar as minhas oportunidades de trabalho; distribuir software não comercial; obter recursos financeiros. Solução de problemas concretos = conseguir ajuda para uma boa ideia para um produto de software; resolver problema que não poderia ser resolvido através de software proprietário. Ideologia = restringir o poder das grandes empresas de software; porque eu penso que software não deveria ser proprietário. Percentual calculado sobre o total de respostas. Permite respostas múltiplas (máximo de 4 opções). Para SL/CA Brasil 2014: desenvolvedores = 299; colaboradores = 125. Para FLOSS Internacional 2013: desenvolvedores = 4.496; colaboradores = 1.597. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.54 – E HOJE, QUAL A RAZÃO PARA CONTINUAR PARTICIPANDO DE PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO?**

CATEGORIAS MOTIVACIONAIS	SL/CA BRASIL 2014		FLOSS INTERNACIONAL 2013	
	DESENVOLVEDORES	COLABORADORES	DESENVOLVEDORES	COLABORADORES
Aprendizado	45,7%	42,0%	43,8%	41,0%
Ambiente	25,3%	33,9%	19,6%	21,3%
Ganhos econômicos	13,3%	7,4%	14,1%	10,3%
Solução de problemas concretos	6,8%	9,5%	10,2%	8,8%
Ideologia	8,9%	9,2%	11,8%	17,5%
Total	100,0%	100,0%	100,0%	100,0%

Percentual calculado sobre o total de respostas. Permite respostas múltiplas (máximo de 4 opções). Para SL/CA Brasil 2014: desenvolvedores = 293; colaboradores = 128. Para FLOSS Internacional 2013: desenvolvedores = 4.908; colaboradores = 1.690. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## ENVOLVIMENTO EM PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO

A quantidade de projetos de software livre/código aberto com os quais os participantes da pesquisa internacional se envolveram ao longo dos anos é baixa: para 62,5% dos respondentes, de 1 a 5 projetos. Embora parte significativa dos participantes da edição brasileira também estiveram envolvidos em poucos projetos, existe um número relativamente maior que participou em mais de 20, 50 ou 100 projetos (Tabela 2.55).

Além disso, uma proporção maior de respondentes da edição brasileira da pesquisa desempenhou papel de líder, coordenador ou administrador em projetos de software livre/código aberto, embora cerca de um terço dos respondentes, em ambas as edições, nunca tenha assumido esta função (Tabela 2.56).

**TABELA 2.55 – EM QUANTOS PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO VOCÊ SE ENVOLVEU?**

QUANTIDADE DE PROJETOS	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
De 1 a 5	55,4%	62,5%
De 6 a 10	23,9%	22,1%
De 11 a 20	10,3%	8,1%
De 21 a 30	5,1%	2,7%
De 31 a 50	1,7%	1,9%

De 51 a 75	0,9%	0,9%
De 76 a 100	0,0%	0,1%
Mais de 100	2,5%	1,6%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 117; para FLOSS Internacional 2013: 1.795. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.56 –EM QUANTOS PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO VOCÊ JÁ SE ENVOLVEU NA CONDIÇÃO DE LÍDER, COORDENADOR OU ADMINISTRADOR?**

QUANTIDADE DE PROJETOS	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Nenhum	32,5%	35,7%
1	20,5%	23,8%
2	16,3%	14,8%
3	9,4%	8,5%
4 a 5	11,9%	10,1%
6 a 7	1,7%	1,9%
8-10	3,4%	2,2%
11-15	0,9%	0,8%
16-20	0,9%	0,7%
Mais de 20	2,6%	1,3%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 117; para FLOSS Internacional 2013: 1.795. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## RETORNO FINANCEIRO COM SOFTWARE LIVRE/CÓDIGO ABERTO

*Uma proporção maior dos respondentes da pesquisa internacional não ganha dinheiro com software livre/código aberto. No grupo, entretanto, em comparação com a edição brasileira, há uma quantidade superior de pessoas que são remuneradas para desenvolver software. No conjunto de respondentes da edição brasileira, existe uma parcela maior que é remunerada para prestar suporte técnico e administrar software livre/código aberto*

As possibilidades de ganhos econômicos diretos ou indiretos com software livre/código aberto são variadas. Apenas 15,4% da comunidade brasileira que respondeu à pesquisa não obtêm alguma retribuição monetária. Os que ganham dinheiro diretamente com software livre/código aberto representam a maioria (57,3% do total). Em comparação com os resultados observados para o Brasil, uma quantidade maior da comunidade internacional não ganha dinheiro com software livre/código aberto (38,2%). Aqueles que ganham dinheiro diretamente representam 42,3% do total (Tabela 2.57).

No entanto, no conjunto de respondentes da pesquisa internacional, há uma quantidade relativamente maior que é remunerada para desenvolver software livre/código aberto (22,4%). Em contraste, na edição brasileira, é proporcionalmente maior a quantidade que é paga para dar suporte técnico para software livre/código aberto (11,1%) e para administrar software livre/código aberto (9,4%). No conjunto de respondentes da pesquisa internacional, há uma parcela comparativamente maior que declarou obter apenas ganhos financeiros indiretos.

**TABELA 2.57 - GANHA DINHEIRO COM SOFTWARE LIVRE/CÓDIGO ABERTO DIRETA OU INDIRETAMENTE? (SE MAIS DE UM MOTIVO, ESCOLHA O PRINCIPAL)**

GANHA DINHEIRO COM SL/CA?	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Não	15,4%	38,2%
Sim, diretamente: sou pago para desenvolver SL/CA	19,7%	22,4%
Sim, diretamente: sou pago para dar suporte técnico para SL/CA	11,1%	6,7%
Sim, diretamente: sou pago para administrar SL/CA	9,4%	6,9%
Sim, diretamente: outras razões.	17,1%	6,3%
Sim, indiretamente: eu consegui o meu emprego em virtude de possuir experiência prévia em SL/CA	7,7%	8,4%
Sim, indiretamente: a descrição da minha função não inclui desenvolvimento de SL/CA, mas também desenvolvo este tipo de software no meu trabalho	6,8%	11,1%
Sim, indiretamente: outras razões	12,8%	8,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 117. Para FLOSS Internacional 2013: 1.676. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## SOFTWARE LIVRE, CÓDIGO ABERTO E SOFTWARE PROPRIETÁRIO

*Parcela maior dos respondentes da pesquisa brasileira acredita que o software livre/código aberto satisfaz os requisitos atuais de software melhor que o software proprietário*

Mais da metade dos respondentes da pesquisa internacional (59,1%) considera que o software livre/código aberto satisfaz os requisitos atuais de software melhor que o software proprietário. O percentual que pensa desta forma é bastante superior entre os participantes da edição brasileira (72,1%), mostrando que, ao fornecer a sua resposta, esta comunidade talvez esteja mais ligada a atributos não funcionais que aquela (Tabela 2.58).

Na comunidade internacional, é maior a proporção de respondentes que utiliza software proprietário no seu dia-a-dia ou para desenvolvimento (57,2% do total contra 47,9% da edição brasileira). Parcela maior dos respondentes que utiliza software proprietário, decidiu fazê-lo apenas para certas tarefas (Tabela 2.59). Em relação ao conjunto formado pelos participantes da pesquisa internacional, o constituído pelos respondentes da pesquisa brasileira parece mais heterogêneo. Existem dois subconjuntos distintos: um

pequeno, constituído por pessoas pragmáticas, em geral desenvolvedores, e um maior, com forte viés ideológico.

Tal como a comunidade brasileira, a comunidade livre internacional entende que as pessoas que trabalham com o software livre/código aberto têm uma percepção do dinheiro diferente das que trabalham com software proprietário. Para os participantes da edição internacional, não é verdade que as pessoas relacionadas com software livre/código aberto são mais ligadas em dinheiro do que as envolvidas com software proprietário: 92,5% acreditam nisso. No entanto, existe um grupo relativamente grande que acha que as pessoas envolvidas com o software proprietário são mais ligadas em dinheiro e isto, segundo os participantes, é ruim (42,8% contra 27,9%, no Brasil) (Tabela 2.60).

Os participantes da pesquisa brasileira parecem ter menos problemas em ganhar dinheiro com software livre/código aberto que os da comunidade internacional. Estes parecem mais guiados por ideais de colaboração, sem ter em mente ganhos financeiros. Uma parcela deles parece achar que ganhar dinheiro com software não é correto. Não são todos que compartilham desta opinião, mas há uma parcela relevante que parece tender a concordar com isto.

**TABELA 2.58 - EM TERMOS GERAIS, VOCÊ CONSIDERA QUE SOFTWARE LIVRE/CÓDIGO ABERTO SATISFAZ OS REQUISITOS ATUAIS DE SOFTWARE MELHOR QUE SOFTWARE PROPRIETÁRIO?**

SATISFAÇÃO MAIOR COM SOFTWARE SL/CA?	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Sim	72,1%	59,1%
Não	3,6%	13,6%
Um não tem nada a ver com o outro	17,1%	12,8%
Não sei dizer	7,2%	14,5%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 111. Para FLOSS Internacional 2013: 1.651. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.59 – UTILIZA SOFTWARE QUE NÃO É LIVRE/CÓDIGO ABERTO NO SEU DIA-A-DIA OU PARA DESENVOLVIMENTO?**

USO DE SOFTWARE PROPRIETÁRIO	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Não. Tento utilizar software/serviços que não são livres ou de código aberto	52,1%	42,8%
Escolhi utilizar software livre/serviços que não são livres ou de código aberto para certas tarefas	15,4%	30,3%
Sim, sem problemas. Uso o melhor software/serviços a despeito de ser livre ou de código aberto ou não	32,5%	26,9%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 117. Para FLOSS Internacional 2013: 1.802. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.60 - AS PESSOAS QUE TRABALHAM COM SOFTWARE LIVRE/CÓDIGO ABERTO SÃO MAIS LIGADAS EM DINHEIRO DO QUE PESSOAS QUE TRABALHAM COM SOFTWARE PROPRIETÁRIO?

RELAÇÃO SOFTWARE SL/CA COM DINHEIRO	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Isso é bom	2,7%	4,1%
Isso é mau	0,0%	3,4%
Isso não é verdade	97,3%	92,5%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 111. Para FLOSS Internacional 2013: 1.632. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.61 - AS PESSOAS QUE TRABALHAM COM SOFTWARE PROPRIETÁRIO SÃO MAIS LIGADAS EM DINHEIRO DO QUE PESSOAS QUE TRABALHAM COM SOFTWARE LIVRE/CÓDIGO ABERTO?

RELAÇÃO SOFTWARE SL/CA COM DINHEIRO	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Isso é bom	10,8%	18,6%
Isso é mau	27,9%	42,8%
Isso não é verdade	61,3%	38,7%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 111. Para FLOSS Internacional 2013: 1.595. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## AMBIENTE DE SOFTWARE LIVRE/CÓDIGO ABERTO

*Prover inovações radicais foi opção mencionada por uma proporção elevada de respondentes da comunidade internacional. A opção foi bem menos pontuada pelos participantes da edição brasileira da pesquisa. Em contraste, estes indicaram, bem mais que os seus pares internacionais, o ambiente de software livre/código aberto como fórum para discussão geral sobre software*

Questionados sobre o que é o ambiente de software livre/código aberto, os respondentes da versão internacional forneceram um número elevado de pontuações para as seguintes opções: maior liberdade no desenvolvimento de software (43,4%); troca de conhecimento (38,2%) e criação de inovações radicais (30,7%). Em comparação com a versão internacional da pesquisa FLOSS, a versão nacional contém uma proporção menor de pessoas que indicaram a possibilidade de desenvolvimento de inovações radicais e, também, de poder contar com uma *toolbox* para desenvolvimento.

Em contrapartida, os respondentes da edição nacional valorizam mais o ambiente do software livre/código aberto como fórum privilegiado para discutir questões gerais sobre software. A opção recebeu uma proporção bastante inferior de votos da comunidade internacional. Ou seja, em termos gerais, a comunidade brasileira parece estar mais interessada no ambiente propiciado pelas atividades em comunidade. A comunidade internacional, mais ligada a aspectos relacionados com o desenvolvimento de software pro-

priamente dito. É possível que um dos motivos da diferença tenha a ver com a quantidade relativamente maior de desenvolvedores na amostra de respondentes da pesquisa internacional (Tabela 2.62).

Tanto no conjunto formado pelos participantes da edição brasileira da pesquisa como entre os participantes da edição internacional, existe um percentual alto de pessoas que acreditam que a vivência de software livre/código aberto tem impacto positivo nas oportunidades de trabalho (Tabela 2.63).

**TABELA 2.62 - O AMBIENTE DE SOFTWARE LIVRE/CÓDIGO ABERTO É FÓRUM PARA...**

AMBIENTE SL/CA	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Uma discussão geral sobre software	34,1%	9,2%
Uma competição esportiva sobre o melhor código	0,0%	5,0%
Prover inovações radicais	17,9%	30,7%
Prover imitações de produtos e serviços de software proprietário	5,5%	5,3%
Desenvolvedores de software que necessitam de uma toolbox	2,6%	15,3%
Prover mais variedade de software	26,9%	21,7%
Pessoas com os mesmos interesses	28,1%	11,9%
Pessoas que procuram parceiros de projeto	9,0%	1,9%
Pessoas que utilizam novas formas de cooperação	28,1%	14,1%
Maior liberdade no desenvolvimento de software	52,3%	43,4%
Melhorias na carreira	7,2%	5,9%
Troca de conhecimento	67,7%	38,2%
Pessoas que querem se divertir	9,8%	12,9%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 111. Para FLOSS Internacional 2013: 2.183. Permite respostas múltiplas (máximo de três opções). Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

**TABELA 2.63 - A VIVÊNCIA NA COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO TEM IMPACTO POSITIVO NAS OPORTUNIDADES DE TRABALHO?**

VIVÊNCIA NA COMUNIDADE IMPACTA TRABALHO?	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Sim	94,6%	88,0%
Não	0,9%	2,4%
Não sei dizer	4,5%	9,6%
Total	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. Para SL/CA Brasil 2014: 111. Para FLOSS Internacional 2013: 1.650. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## BALANÇO ENTRE FORNECER E RECEBER

No que se refere ao balanço entre fornecer e receber, na pesquisa brasileira, parece existir maior divergência entre as respostas dos desenvolvedores e colaboradores que na pesquisa internacional

Nas tabelas 2.64 e 2.65, comparam-se os achados da edição brasileira e internacional da pesquisa sobre software livre/código aberto no que se refere à percepção de equilíbrio entre fornecer e receber apoio e suporte, considerando desenvolvedores e colaboradores separadamente.

Na edição internacional, não há diferenças significativas entre as respostas encontradas para o conjunto de desenvolvedores e de colaboradores. Em ambos os casos, quase a metade dos respondentes informou que recebe mais que fornece (48,1% e 49,8%, respectivamente). Avaliando o que outros desenvolvedores e colaboradores pensam a respeito, os participantes internacionais acreditam, também, que a maioria julga que recebe mais do que fornece.

Os desenvolvedores brasileiros se julgam, de alguma forma, mais importantes na relação oferta que o conjunto internacional. De modo inverso, o conjunto de colaboradores brasileiros se percebe, de alguma forma, menos decisivo na oferta que o conjunto internacional.

**TABELA 2.64 – VOCÊ CONTRIBUI PARA A COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO DE DIFERENTES MODOS E RECEBE MUITOS BENEFÍCIOS EM TROCA. EM GERAL, COMO VOCÊ AVALIA O PESO DA BALANÇA?**

COMO VOCÊ AVALIA O PESO DA BALANÇA?	DESENVOLVEDORES <sup>1</sup>		COLABORADORES <sup>2</sup>	
	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Forneço mais que recebo	13,0%	12,3%	8,8%	13,3%
Recebo mais que forneço	32,5%	48,1%	44,1%	49,8%
Forneço tanto como recebo	13,0%	15,3%	17,7%	13,7%
Não me importo	31,1%	12,1%	17,7%	11,6%
Não sei dizer	10,4%	12,2%	11,7%	11,6%
Total	100,0%	100,0%	8,8%	13,3%

Percentual calculado sobre o total de participantes que responderam à questão. (1) Para SL/CA Brasil 2014: 77; para FLOSS Internacional 2013: 1.225. (2) Para SL/CA Brasil 2014: 34; para FLOSS Internacional 2013: 430. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.65 - QUANDO VOCÊ CONSIDERA OUTRAS PESSOAS DA COMUNIDADE DE SOFTWARE LIVRE/CÓDIGO ABERTO, COMO VOCÊ JULGA QUE ELES PENSAM?

COMO OS OUTROS AVALIAM O PESO DA BALANÇA?	DESENVOLVEDORES <sup>1</sup>		COLABORADORES <sup>2</sup>	
	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013	SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
A maioria fornece mais que recebe	18,2%	16,7%	17,7%	22,6%
A maioria recebe mais que fornece	32,5%	28,3%	47,1%	25,2%
Fornecem tanto quanto recebem	7,8%	8,4%	20,6%	11,4%
Não se importam com isso	9,1%	2,9%	2,9%	3,5%
Não me importo com isso	15,6%	16,3%	2,9%	14,5%
Não sei dizer	16,8%	27,5%	8,8%	22,8%
Total	100,0%	100,0%	100,0%	100,0%

Percentual calculado sobre o total de participantes que responderam à questão. (1) Para SL/CA Brasil 2014: 77; para FLOSS Internacional 2013: 1.224. (2) Para SL/CA Brasil 2014: 34; para FLOSS Internacional 2013: 429. Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

*Em termos comparativos, participantes da pesquisa brasileira têm expectativas maiores de ganhos econômicos e participação em ambiente que permita a interação. Em contraste, uma proporção maior de participantes internacionais deseja participar, ser respeitada, aprender a escrever programas melhores e combater o poder das grandes empresas*

Quando pessoas se organizam para trabalhar conjuntamente, existem várias expectativas que podem ser criadas no que se refere aos resultados da união. Nas tabelas 2.66 a 2.69, incluem-se os desejos de desenvolvedores e colaboradores, respectivamente. Eles podem estar associados a aspectos diversos: participação em um ambiente propício para a troca, aquisição de conhecimento, obtenção de ganhos financeiros, criação de oportunidades para a solução de problemas concretos e respostas a questões ideológicas. Observa-se que, tanto para desenvolvedores como para colaboradores, as expectativas concentram-se muito no ambiente propiciado pela comunidade de software livre/código aberto, e, também, nas oportunidades criadas para aprendizado. As expectativas de desenvolvedores e colaboradores são semelhantes e não há diferenças entre os resultados levantados nas pesquisas brasileira e internacional.

Uma proporção maior de desenvolvedores participantes da pesquisa brasileira tem expectativas referentes a ganhos econômicos. Espera cooperar, compartilhar, melhorar produtos dos demais e contribuir com soluções. Uma proporção maior de desenvolvedores da edição internacional deseja participar das principais discussões, ser respeitada, poder aprender e escrever programas melhores e, também, reduzir o poder das grandes empresas.

TABELA 2.66 – O QUE VOCÊ ESPERA DE OUTROS DESENVOLVEDORES DE SOFTWARE LIVRE/CÓDIGO ABERTO?

Para desenvolvedores

O QUE VOCÊ ESPERA DOS DE MAIS?		SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Ganhos	Estar apto para cooperar de uma nova maneira	46,8%	28,5%
	Tomar parte nas principais comunicações e discussões	13,0%	39,3%
	Respeitar-me e as minhas contribuições para software livre/código aberto	28,6%	42,6%
Aprendizado	Deixar-me aprender e desenvolver novas habilidades	23,4%	30,4%
	Compartilhar os seus conhecimentos e habilidades	90,9%	74,5%
	Melhorar produtos de software livre/código aberto de outros desenvolvedores	61,0%	38,6%
Solução de problemas	Escrever programas adequados desde o ponto de vista estético	3,9%	21,9%
	Contribuir para concretizar ideias de produtos de software	36,4%	22,9%
	Resolver um problema que não poderia ser resolvido por software proprietário	20,8%	18,3%
Ganhos econômicos	Fornecer melhores oportunidades de trabalho	2,6%	1,9%
	Distribuir software não comercial	7,8%	2,9%
	Ganhar dinheiro	10,4%	3,4%
Ideologia	Contribuir para limitar o poder das grandes empresas de software	7,8%	10,0%
Não sei dizer		0,0%	4,5%

Percentual calculado sobre o total de desenvolvedores que responderam à questão. Para SL/CA Brasil 2014: 77. Para FLOSS Internacional: 1.224. Permite respostas múltiplas (máximo 4 opções). Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.67– O QUE VOCÊ ESPERA DE OUTROS COLABORADORES DE SOFTWARE LIVRE/CÓDIGO ABERTO?

## Para colaboradores

O QUE VOCÊ ESPERA DOS DEMAIS?		SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Ganhos sociais - Ambiente	Estar apto para cooperar de uma nova maneira	52,9%	35,0%
	Tomar parte nas principais comunicações e discussões	32,4%	31,2%
	Respeitar-me e as minhas contribuições para software livre/código aberto	23,5%	32,9%
Aprendizado	Deixar-me aprender e desenvolver novas habilidades	14,7%	33,6%
	Que eles compartilhem os seus conhecimentos e habilidades	79,4%	74,8%
	Melhorar produtos de software livre/código aberto de outros	58,8%	38,9%
Solução de problemas	Escrever programas adequados desde o ponto de vista estético	0,0%	10,3%
	Contribuir para concretizar ideias de produtos de software	38,2%	24,0%
	Resolver um problema que não poderia ser resolvido por software proprietário	20,6%	28,2%
Ganhos econômicos	Fornecer melhores oportunidades de trabalho	11,8%	5,3%
	Distribuir software não comercial	11,8%	3,5%
	Ganhar dinheiro	11,8%	2,3%
Ideologia	Contribuir para limitar o poder das grandes empresas de software	2,9%	19,8%
Não sei dizer		0,0%	3,5%

Percentual calculado sobre o total de colaboradores que responderam à questão. Para SL/CA Brasil 2014: 34. Para FLOSS Internacional 2013: 429. Permite respostas múltiplas (máximo 4 opções). Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.68 – O QUE ACHA QUE OS OUTROS DESENVOLVEDORES DE SOFTWARE LIVRE/CÓDIGO ABERTO ESPERAM DE VOCÊ?

Para desenvolvedores

O QUE OS OUTROS ESPERAM DE VOCÊ?		SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Ganhos sociais - Ambiente	Estar apto para cooperar de uma nova maneira	46,8%	25,9%
	Tomar parte nas principais comunicações e discussões	31,2%	45,7%
	Respeitá-los e as suas contribuições para software livre/código aberto	37,7%	46,4%
Aprendizado	Deixá-los aprender e desenvolver novas habilidades	29,9%	23,5%
	Que eu compartilhe os meus conhecimentos e habilidades	83,1%	68,7%
	Melhorar produtos de software livre/código aberto de outros desenvolvedores	42,9%	35,9%
Solução de problemas	Escrever programas adequados desde o ponto de vista estético	10,4%	21,4%
	Contribuir para concretizar ideias de produtos de software	26,0%	22,3%
	Resolver um problema que não poderia ser resolvido por software proprietário	16,9%	8,6%
Ganhos econômicos	Fornecer melhores oportunidades de trabalho	6,5%	1,3%
	Distribuir software não comercial	7,8%	2,5%
	Ganhar dinheiro	6,5%	1,6%
Ideologia	Contribuir para limitar o poder das grandes empresas de software	3,9%	4,7%
Não sei dizer		2,6%	13,1%

Percentual calculado sobre o total de desenvolvedores que responderam à questão. Para SL/CA Brasil 2014: 77. Para FLOSS Internacional 2013: 1.224. Permite respostas múltiplas (máximo 4 opções). Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

TABELA 2.69 - O QUE ACHA QUE OS OUTROS COLABORADORES DE SOFTWARE LIVRE/CÓDIGO ABERTO ESPERAM DE VOCÊ?

## Para colaboradores

O QUE OS OUTROS ESPERAM DE VOCÊ?		SL/CA BRASIL 2014	FLOSS INTERNACIONAL 2013
Ganhos sociais - Ambiente	Estar apto para cooperar de uma nova maneira	47,1%	33,1%
	Tomar parte nas principais comunicações e discussões	44,1%	42,4%
	Respeitá-los e as suas contribuições para software livre/código aberto	29,4%	42,2%
Aprendizado	Deixá-los aprender e desenvolver novas habilidades	23,5%	24,5%
	Que eu compartilhe os meus conhecimentos e habilidades	67,6%	66,9%
	Melhorar produtos de software livre/código aberto de outros	29,4%	32,6%
Solução de problemas	Escrever programas adequados desde o ponto de vista estético	0,0%	11,4%
	Contribuir para concretizar ideias de produtos de software	23,5%	21,4%
	Resolver um problema que não poderia ser resolvido por software proprietário	14,7%	12,1%
Ganhos econômicos	Fornecer melhores oportunidades de trabalho	5,9%	2,6%
	Distribuir software não comercial	2,9%	4,2%
	Ganhar dinheiro	17,6%	1,6%
Ideologia	Contribuir para limitar o poder das grandes empresas de software	8,8%	7,9%
Não sei dizer		2,9%	13,1%

Percentual calculado sobre o total de colaboradores que responderam à questão. Para SL/CA Brasil 2014: 34. Para FLOSS Internacional 2013: 429. Permite respostas múltiplas (máximo 4 opções). Fonte: Edição brasileira: Observatório Softex, 2014; edição internacional, FLOSS, 2013.

## DESAFIOS A SEREM ENFRENTADOS

Na Tabela 2.70, considerando desafios que devem ser enfrentados durante o processo de desenvolvimento de software livre/código aberto, apresenta-se a distribuição do número de participantes da pesquisa internacional, considerando as seguintes opções: não é relevante, é importante, é importante e eu podia fazer mais por isto, preocupa-me mas não sei o que fazer, participo deste desafio, este desafio não tem nada a ver comigo.

A comunidade internacional está dividida entre os que percebem e os que não percebem importância nas atividades de tornar o software multiplataforma/móvel e disponibilizado no modelo SaaS e na redução do tempo entre *releases*. A comunidade parece entender que questões técnicas, tais como tornar o software estável, definir as suas características e cuidar da documentação fazem parte do seu trabalho. Parte significativa não sabe o que fazer quando o assunto diz respeito a atrair e reter usuários e colaboradores. Algumas atividades são importantes, mas poderiam ser tratadas ainda com mais relevância do que são. É o caso da definição dos recursos e características do software, da documentação e da atração e manutenção de usuários/colaboradores.

Alavancar o poder de empresas líderes e obter recursos financeiros foram consideradas questões sem importância, enquanto questões legais e relacionadas com a comunicação interna ao longo da cadeia de valor foram avaliadas como relevantes. A comunidade internacional entende que poderia ser feito mais pela comunicação.

**TABELA 2.70 – POSSÍVEIS DESAFIOS QUE TÊM DE SER ENFRENTADOS NOS PROJETOS DE SOFTWARE LIVRE/CÓDIGO ABERTO NO QUAL ESTÁ ENVOLVIDO, AVALIANDO A IMPORTÂNCIA ATRIBUÍDA A CADA UM DELES – FLOSS INTERNACIONAL 2013**

ITEM	NÃO É IMPORTANTE	É IMPORTANTE	É IMPORTANTE E EU PODIA FAZER MAIS POR ISTO	PREOCUPA-ME, MAS NÃO SEI O QUE FAZER	PARTICIPO DISTO	ISTO NÃO TEM NADA A VER COMIGO	NÃO SEI DIZER
Tornar o software estável (reduzir <i>bugs</i> )	1,5%	23,4%	22,9%	4,5%	39,6%	2,7%	5,5%
Definir recursos/características	1,3%	25,6%	32,0%	3,2%	29,3%	2,2%	6,4%
Documentação	1,8%	15,6%	39,6%	9,7%	24,7%	3,0%	5,5%
Tornar o software multiplataforma/móvel/SaaS	24,3%	27,8%	19,8%	6,3%	9,9%	1,4%	10,6%

Reduzir o tempo entre versões	23,4%	30,9%	20,7%	7,1%	7,1%	1,6%	9,2%
Atrair usuários/ colaboradores	6,0%	14,7%	31,3%	18,2%	18,7%	3,8%	7,3%
Manter usuários/ colaboradores engajados	4,9%	17,0%	30,3%	19,4%	16,6%	4,0%	7,7%
Alavancar o poder da empresa líder	26,4%	23,9%	15,1%	5,9%	7,1%	2,8%	18,7%
Obter recursos financeiros	29,6%	17,8%	16,1%	12,7%	6,5%	1,8%	15,5%
Questões legais (licença, <i>copyright</i> , etc.).	16,3%	43,9%	11,3%	7,5%	6,3%	1,8%	12,9%
Comunicação interna do projeto	3,6%	33,7%	32,9%	5,3%	12,6%	3,3%	8,5%
Comunicação ao longo da cadeia de valor	5,0%	27,5%	32,3%	8,2%	10,7%	3,5%	12,7%

Percentual calculado sobre o total de respondentes de cada item: tornar o software estável: 1.508; estabelecer as características: 1.573; documentação: 1.560; tornar o software multiplataforma/móvel/SaaS: 1.579; reduzir o tempo entre versões: 1.586; atrair usuários/colaboradores: 1.590; manter usuários/colaboradores engajados: 1.586; alavancar o poder da empresa líder: 1.579; obter recursos financeiros: 1.571; questões legais: 1.580; comunicação interna do projeto: 1.583; comunicação ao longo da cadeia de valor: 1.578. Fonte: FLOSS, 2013.

## 2.3 PERFIL DE DESENVOLVEDORES: O QUE MODIFICOU NOS ÚLTIMOS DEZ ANOS?

Em 2004, a Softex, em parceria com a Unicamp, realizou uma pesquisa eletrônica (SL/CA Brasil 2004) respondida por um número surpreendentemente alto de desenvolvedores de software livre/código aberto no Brasil. Foram consideradas válidas 1.953 respostas, o que permitiu traçar um perfil muito representativo do grupo.

A pesquisa foi divulgada amplamente na comunidade brasileira de software livre/código aberto por meio de listas de discussão, *sites* relacionados e em coletiva na imprensa, durante o lançamento do 5º FISL (Fórum Internacional de Software Livre). Seguindo diretrizes adotadas em estudos internacionais, a amostra foi voluntária e a pesquisa ficou aberta para preenchimento de quem se considerava desenvolvedor ou usuário. A construção do questionário foi inspirada na bibliografia internacional, especialmente em estudos coordenados pela organização FLOSS (Gosh *et al.*, 2002). O formulário eletrônico utilizado foi uma adaptação do FLOSS Europe – 2002, que envolveu 2.784 desenvolvedores, abrangendo pessoas de diferentes países (71% da Europa Ocidental/Rússia; 13% dos EUA e 17% outros).

A seguir apresentam-se os principais achados, comparando-se os resultados do passado com os obtidos em 2014 (Quadro 2.1).

- Nível de escolaridade – o grupo que participou da SL/CA 2014 possui um nível de escolaridade muito superior aos pesquisados em 2004. Neste grupo, dada a forma de condução da pesquisa, havia uma quantidade bastante relevante de estudantes.

- Em ambas as edições, há mais técnicos programadores, administradores de rede e pessoal de suporte que engenheiros de software e correlatos.
- Aparentemente, observa-se uma profissionalização da comunidade de SL/CA, com um percentual maior obtendo ganhos monetários diretos com atividades de software livre/código aberto.
- Os respondentes da SL/CA 2014 iniciaram a participação na comunidade nos anos em que foi realizada a pesquisa SL/CA Brasil, período em que havia grandes expectativas sobre o software livre e código aberto. Divisões e posicionamentos não eram, naquele momento, suficientemente claros.
- Naquela época (da mesma forma que nesta) parte preponderante dos respondentes afirmou desenvolver/se envolver com software proprietário.

**QUADRO 2.1 – COMPARAÇÃO DO PERFIL DOS DESENVOLVEDORES (SL/CA BRASIL 2004 E SL/CA BRASIL 2014): CARACTERÍSTICAS SOCIOECONÔMICAS**

ITENS LEVANTADOS	SL/CA BRASIL 2004	SL/CA BRASIL 2014
Sexo masculino	97%	97,3%
Escolaridade superior	42% (20% com pós-graduação)	85,1% (51,8% com pós-graduação)
Ocupação principal	Técnico programador, administrador de redes, suporte (34%) Engenheiros de software, analistas de sistemas (20%)	Técnico programador e consultor de TI (30,2%) Engenheiros de software (27,7%)
Renda com SL/CA	40% não possui; 36% diretamente; 24% indiretamente	13,8% não possui; 58,8% diretamente; 27,4% indiretamente
Dedicação ao desenvolvimento de SL/CA (hs. semanais)	62%: até 10; 37% mais de 11	67,5%: até 10 32,5%: mais de 11
Participação em projetos	71%: 1 a 5 projetos	53,8%: 1 a 5 projetos
Liderança em projetos	45% nunca; 24%: 1 projeto	31,3% nunca; 22,5%: 1 projeto
Pertencimento às comunidades	26% software livre 10% código aberto 53% ambas	38,6% software livre 18,1% código livre 43,4% ambas
Média do ano de início	2002	Entre 2000 e 2009: 59,0%
Dedicação ao desenvolvimento de software proprietário (hs. semanais)	51,3% também desenvolvem software proprietário, sendo: 39% menos de 10; 14% entre 11 e 20; 43% mais de 21	51,8% também desenvolvem software proprietário, sendo: 44,2% menos de 10; 11,6% entre 11 e 20; 44,2% mais de 21

Fonte: Softex (2005) e Softex (2014).

## CONSIDERAÇÕES FINAIS

Resumem-se, a seguir, alguns achados e apontam-se questões que merecem mais reflexão.

### Desenvolvedores x colaboradores na pesquisa SL/CA Brasil 2014

Os colaboradores iniciaram mais cedo na comunidade livre que os desenvolvedores, um indício de que, na média, aquele grupo é constituído de pessoas com maior idade.

No grupo composto por colaboradores, é muito superior a participação daqueles que acreditam que a diferença entre software livre e código aberto importa. Parte relevante do grupo identifica-se com o software livre. Como explicar a relevância que um percentual tão maior de colaboradores atribui à diferença entre software livre e código aberto? A percepção distinta teria a ver com as atribuições específicas de cada um dos grupos na comunidade livre ou com diferenças relacionadas com a faixa média de idade dos participantes dos grupos, mais elevada entre os colaboradores que entre os desenvolvedores? É de se supor que, ao longo dos anos, existe uma tendência para o movimento de software livre ser substituído pelo de código aberto, mais pragmático e alinhado ao *status quo*? Neste caso, após o movimento do software livre nos anos 80 e o advento do código aberto nos anos 90, o que estaria por vir? A comunidade livre irá se reinventar? E em que direção isto acontecerá? Será que o discurso do software livre/código aberto irá se fragmentar em seus vários elementos constitutivos - desenvolvimento colaborativo, auditabilidade, código aberto, etc. - criando tribos e bandeiras diferenciadas?

Tal como no grupo de desenvolvedores, no grupo dos colaboradores, a maioria se envolveu, no máximo, em cinco projetos, assumindo a liderança em poucas situações. No entanto, em comparação com o grupo de desenvolvedores, entre os colaboradores, é maior a participação dos que se envolveram em mais de vinte projetos e assumiram a posição de líder em mais de 10 projetos. Essas diferenças entre desenvolvedores e colaboradores, no que diz respeito à participação e liderança em projetos seria uma característica da comunidade brasileira ou resultado da posição que os colaboradores possuem na cadeia de desenvolvimento e prestação de serviços envolvendo software livre/código aberto no país? Será que os colaboradores na condição de líderes de projetos seriam os donos do projeto e usuários do software?

### Pesquisa brasileira x internacional

O conjunto de respondentes da edição brasileira da pesquisa é mais heterogêneo que o de participantes da edição internacional. Na edição internacional, observa-se presença maior de desenvolvedores, mulheres e pessoas jovens. Os engenheiros de software têm participação elevada no conjunto.

Em comparação com a versão internacional da pesquisa, um número relativamente menor de participantes da edição brasileira que se apresentaram como desenvolvedores de software desempenha esta atividade em sua vida profissional. Em geral, os desenvolvedores trabalham como administradores de TI ou são prestadores de serviços de suporte técnico. A média de idade do grupo brasileiro de respondentes é superior à do grupo internacional.

Na edição brasileira da pesquisa, existe uma diferença mais significativa entre o perfil de desenvolvedores e colaboradores que na edição internacional. As diferenças têm a ver com a faixa etária, as funções desem-

penhadas e a percepção do papel que lhes cabe. No grupo constituído pelos respondentes da pesquisa no Brasil, também há um número proporcionalmente maior de colaboradores.

Na comunidade internacional, existe uma quantidade relativamente maior de respondentes que se identificam com o código aberto. No Brasil, parte significativa dos participantes da pesquisa se percebe como parte da comunidade de software livre.

Colaboradores têm uma participação diferenciada na comunidade brasileira, assumindo a responsabilidade/liderança de projetos. É possível que parte deles seja demandante/cliente dos desenvolvedores da comunidade local. Isso não é tão nítido entre os participantes da pesquisa FLOSS internacional.

É de se supor que, para defensores do software livre, não haveria qualquer fator impeditivo à busca por lucro. Ao contrário, até por motivos relacionados com o fortalecimento da imagem/marca do *free* como liberdade, haveria um interesse maior em enfatizar que o cerne das diferenças com o software proprietário não estaria baseado em questões envolvendo ganhos monetários. O mesmo raciocínio também cabe para os que defendem o software de código aberto. Se, neste caso, a questão central está no modelo de desenvolvimento e na robustez e qualidade do produto final, então, em tese, não há nenhum impeditivo para a busca por lucro. No entanto, embora não haja restrições, os ganhos financeiros não estão no foco da comunidade brasileira e nem da internacional talvez porque estejam relacionados com a ideia de apropriação individual, indevida, dos resultados produtivos e intelectuais de muitos. Seja como for, no grupo participante da pesquisa internacional, há uma quantidade relativamente maior dos que colaboram sem obter qualquer tipo de retorno financeiro.

Inovações radicais é tema relevante para os respondentes da comunidade livre internacional e foi pouco considerado pela comunidade brasileira pesquisada. Isso talvez ocorra em virtude de um número reduzido atuar profissionalmente como desenvolvedor, neste caso.

Tanto para os respondentes da pesquisa internacional como para os participantes da edição brasileira, o software multiplataforma, móvel e na nuvem, comercializado como serviço, não é percebido como um modelo de negócios de interesse. Esse é um dado que merece atenção (e preocupação), considerando que as oportunidades futuras em software parecem ir, justamente, nesta direção.

# CAPÍTULO 3

## O USO DO SOFTWARE LIVRE/CÓDIGO ABERTO NO BRASIL

### APRESENTAÇÃO

No Capítulo 1, tendo como base a literatura especializada na indústria de software, apontaram-se as limitações para a permanência de plataformas tecnológicas rivais no mercado, após a consolidação de uma líder. Na indústria de software, o vencedor adquire vantagens competitivas relevantes, fechando o mercado para a concorrência. Isso explica, em grande medida, as dificuldades que o movimento de software livre/código aberto teve para avançar tanto no segmento de servidores/estações de trabalho como no de *desktops*. Nem o surgimento de soluções com interfaces gráficas aprimoradas nem as melhorias técnicas substantivas nos produtos foram suficientes para reverter a situação de *lock-in*.

A lógica da estrutura e dinâmica competitiva da indústria de software, que concede à líder condições de mercado muito privilegiadas, funciona em nível global, ultrapassando as fronteiras dos países centrais, onde, em geral, encontram-se as sedes das empresas candidatas ao papel de liderança. Apesar disso, pode receber contornos locais distintos, dependendo das especificidades dos sistemas locais de inovação, do posicionamento das empresas de capital nacional na cadeia de valor do software e das ações de políticas públicas.

Neste capítulo, portanto, busca-se entender como as disputas pela hegemonia do mercado de sistemas operacionais, que tiveram início nos países centrais, repercutiram no Brasil e foram assimiladas à realidade brasileira.

Inicialmente, na Seção 3.1, comparam-se dados e expectativas de crescimento do uso do software livre/código aberto no Brasil, no início dos anos 2000 e no início da década de 2010. Na Seção 3.2, são apresentados os elos que constituem a cadeia de valor do software. Discute-se o posicionamento de empresas de capital nacional nesta cadeia de valor e o que isto significou em termos de possibilidades de resistência à hegemonia da plataforma tecnológica líder. Na Seção 3.3, apresentam-se os elementos do cálculo de custo e benefício realizado por demandantes e fornecedores de software e serviços de TI. Além disso, mostra-se como, no Brasil, o movimento de software livre/código aberto evoluiu e criou identidade própria, tendo como pano de fundo os fatores limitantes e as possibilidades em aberto.

No capítulo, serão utilizados dados e informações mencionados no estudo da Softex (2005) e em estudos recentes, além de depoimentos de integrantes e simpatizantes da comunidade brasileira de software livre/código aberto.

### 3.1 USO E EXPECTATIVAS DE USO DO SOFTWARE LIVRE/CÓDIGO ABERTO NO BRASIL

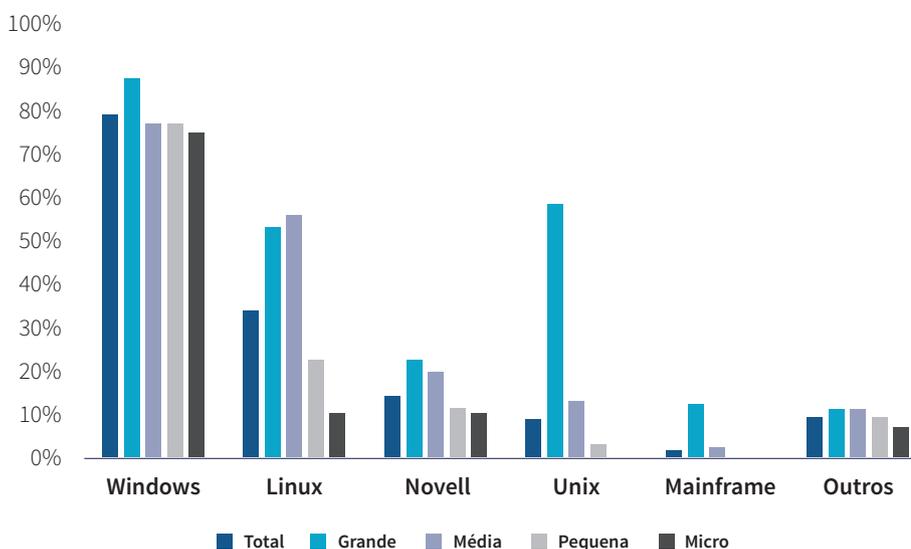
A seguir, apresentam-se dados e informações sobre o uso e as expectativas de uso do software livre/código aberto no Brasil, no início da década de 2000. As situações observadas no período são comparadas com resultados registrados no início da década de 2010. Busca-se responder à seguinte pergunta: o que mudou em uma década e o que explica as mudanças observadas?

#### SOFTWARE LIVRE/CÓDIGO ABERTO NO INÍCIO DA DÉCADA DE 2000

*No Brasil, no início dos anos 2000, o sistema operacional Linux mostrava-se bem posicionado no mercado de servidores para o ambiente corporativo, especialmente em empresas de médio e grande porte.*

No início dos anos 2000, em alinhamento com o que ocorria em nível global, houve um crescimento do uso de software livre no Brasil, no ambiente corporativo. A grande porta de entrada foi o sistema operacional Linux. Dados sobre o uso de software livre em servidores de empresas manufatureiras mostravam que Linux encontrava-se em segundo lugar no *ranking*, rodando em 53% dos servidores das grandes empresas e em 56% dos servidores de empresas de médio porte. Na época, Unix mantinha participação forte nas grandes corporações e Windows, o primeiro colocado no *ranking*, estava presente nos servidores de empresas de todos os portes (FIESP/CIESP e FEA/USP, 2004) (Figura 3.1).

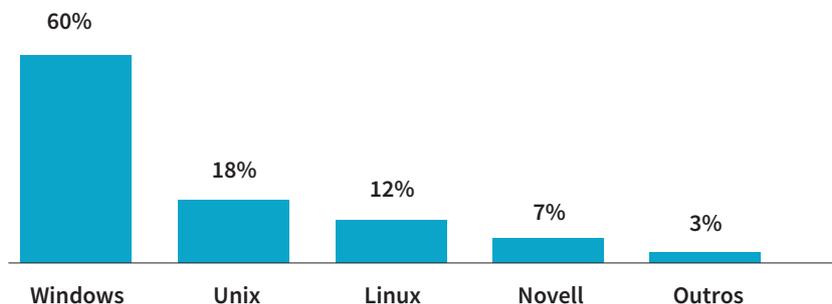
FIGURA 3.1 – PRINCIPAIS SISTEMAS OPERACIONAIS UTILIZADOS NOS SERVIDORES DE EMPRESAS MANUFATUREIRAS, CONSIDERANDO TOTAL E PORTE – BRASIL, 2004



Fonte: FIESP/CIESP e FEA/USP, 2004 *apud in* Softex (2005).

Segundo Meirelles (2004), 12% dos servidores instalados no país operavam em Linux no início dos anos 2000, contra 60% de servidores rodando Windows (Figura 3.2). Os dados estavam em sintonia com as tendências mundiais, pois, de acordo com a revista InfoCorporate (Vieira e Nogueira, 2003), a consultoria Gartner identificou que, em 2002, Linux respondia por 6% do mercado mundial de sistemas operacionais e, em 2003, por 9%. Para 2007, a estimativa era que Linux fosse responsável por 18% do total.

FIGURA 3.2 – DISTRIBUIÇÃO DO NÚMERO DE SERVIDORES INSTALADOS, CONSIDERANDO SISTEMA OPERACIONAL UTILIZADO – BRASIL, 2004



Fonte: Meirelles, 2004, *apud in Softex* (2005).

*A adoção de software livre/código aberto encontrava-se muito concentrada na infraestrutura das empresas. MySQL possuía pouca participação no mercado de banco de dados e Mozilla baixa presença no segmento de navegadores.*

Mais além da infraestrutura, o uso de software livre/código aberto mostrava-se muito restrito. Dados levantados na pesquisa I-Digital registravam que, em bancos de dados transacionais, havia predominância de Oracle para as grandes e de Microsoft SQL Server em empresas manufatureiras de todos os portes. O uso de banco de dados de software livre/código aberto (MySQL) ficava muito atrás de outros de pequeno porte, como Clipper e Access, independentemente do tamanho da empresa.

A penetração também era pequena no segmento de navegadores. O *browser* Mozilla tinha apenas 7% de participação do mercado. Internet Explorer, apesar de haver perdido espaço, seguia dominando o mercado, com 88,9% de participação.

*Software livre/código aberto era muito pouco utilizado nos desktops. A imaturidade das interfaces gráficas era apontada como motivo para a baixa adesão. O mercado de microcomputadores seria o último filão a ser conquistado.*

Segundo estudo da Softex (2005), mencionando dados da ComCiência (2004) e da IDC (2004), o uso de software livre/código aberto no segmento dos computadores de mesa era restrito. Este seria o último filão a ser conquistado. A previsão era que, em 2007, Linux atingiria 6% do mercado global de computadores, um crescimento significativo em relação aos 2,7% de participação que possuía em 2002.

Para muitos, a imaturidade das interfaces gráficas era o fator que justificava o uso restrito do software livre/código aberto nos microcomputadores. Nos PCs, a difusão do Linux ocorria principalmente entre usuários finais sofisticados, como administradores de sistemas e desenvolvedores (Lerner e Tirole, 2002). Estudo de

Reis (2003) mostrava que a usabilidade era atributo pouco considerado no processo de desenvolvimento de software livre/código aberto, o que favorecia a reputação amplamente difundida de que este tipo de software era complexo e difícil de operar.

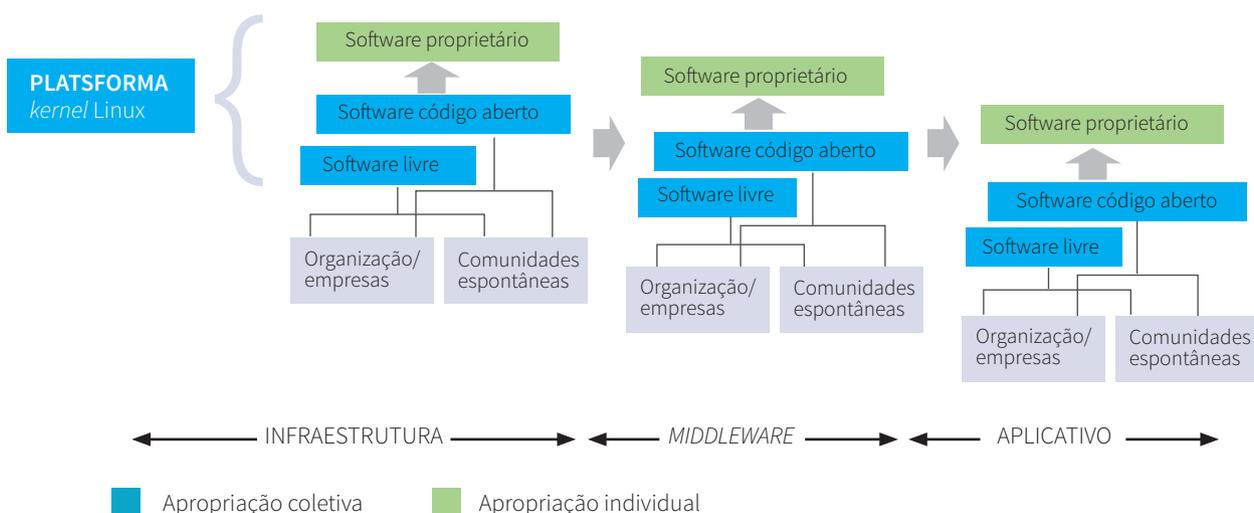
**Projetos Gnome e KDE eram percebidos como boa iniciativa para difundir o software livre/código aberto nos PCs, atingindo, portanto, os usuários finais.**

Na época, entre os desafios a serem enfrentados pela comunidade de software livre/código aberto, estava o desenvolvimento de produtos mais amigáveis para atrair os usuários finais que não possuíam formação em computação, com maior ênfase em aspectos de documentação, suporte, interfaces, entre outros. Os projetos que visavam à criação de interfaces mais adaptadas e próximas do padrão Windows, tais como Gnome e KDE, eram percebidos como uma iniciativa para difundir o software livre/código aberto entre usuários comuns, o território a ser ainda conquistado.

Assim, conforme mostrado na Figura 3.3, no Brasil, da mesma forma que em nível global, o software livre começou a ser utilizado na retaguarda das empresas, como software de infraestrutura. Caminhava, aos poucos, para ampliar a sua participação como software de *middleware*, conquistando os mercados de banco de dados e ferramentas de desenvolvimento. O próximo passo seria chegar aos computadores de mesa, ambiente em que o Windows reinava soberano, e, assim, oferecer aos usuários finais softwares aplicativos para suporte às atividades de rotina, tais como planilhas eletrônicas, soluções para gestão de recursos humanos e relacionamento com os clientes, sistemas de administração hospitalar, etc. A chegada a este último reduto permitiria a construção de uma proposta fim-a-fim em software livre/código aberto.

Licenças do tipo código aberto permitiam que produtos livres dessem origem a produtos proprietários, uma trajetória possível e concorrente da proposta fim-a-fim livre. Nas empresas, no que concerne ao software de infraestrutura e de *middleware*, os softwares com licença livre ou de código aberto passavam a conviver, em certa harmonia, com sistemas proprietários.

**FIGURA 3.3 – TRAJETÓRIAS POSSÍVEIS A PARTIR DA ADOÇÃO DO KERNEL LINUX, CONSIDERANDO ORGANIZAÇÃO PARA A PRODUÇÃO DE SOFTWARE LIVRE/CÓDIGO ABERTO E CADEIA DE VALOR DO SOFTWARE**



Fonte: Observatório Softex.

*Governo brasileiro: importante usuário de software livre/código aberto, no início dos anos 2000.*

Já nos anos 2000, o governo brasileiro dava sinais do seu interesse no uso do software livre/código aberto. Como mencionado em Softex (2005):

“Todas as empresas públicas de informática analisadas possuem algum projeto de migração de plataformas baixas das áreas administrativas para soluções em software livre/código aberto, principalmente com Linux como sistema operacional. Há também iniciativas voltadas aos usuários finais da área administrativa que trata da migração de *desktops*.”

Em um dos casos analisados, uma empresa pública municipal, a utilização do software livre/código aberto iniciou como uma opção técnica em iniciativas isoladas pelo menos oito anos atrás, passando a ser usado pela infraestrutura (como em servidores de arquivos e gestor de arquivos) e depois alcançando o usuário final. Em meados de 2002, foi criado um grupo de trabalho para avaliar o processo de migração. Praticamente toda a equipe interna (operacionais, suporte, *help desk*) recebeu treinamento. A partir de 2003, iniciou firmemente o uso institucionalizado de software livre/código aberto em toda a empresa, incluindo as ferramentas de desenvolvimento e as ferramentas de *desktop*.

Além da migração das plataformas, as empresas públicas municipais estão trabalhando na migração de servidores e estações de trabalho nas áreas de educação e saúde: tanto há experiências de informatização de escolas e montagem de laboratórios e telecentros para as secretarias de educação (estações com Linux ou com *dual boot*), como há experiências em curso de informatização de unidades básicas de atenção à saúde, através da utilização de uma lógica semelhante aos equipamentos dedicados como PDV utilizando software livre/código aberto. Por fim, também há experiências de desenvolvimento, especialmente em certificação digital e segurança.

O grande desafio dessas empresas é tratar com a herança. Muitas empresas têm uma plataforma estável com muitos anos de aprendizagem e interação de equipamentos. Trocar isto significa risco de perda de estabilidade. A grande maioria das grandes organizações não pensa em migrar a herança, somente as plataformas baixas. Trocar a herança exige um investimento elevado e depende de anos de trabalho para converter todos os sistemas de automação governamental.”

*Concluindo: os primeiros passos foram dados, agora é seguir em frente.*

Resumindo o movimento de software livre/código aberto no início dos anos 2000 no país, a Softex (2005) conclui que:

“pode-se dizer que o movimento atual de empresas desenvolvedoras de software livre/código aberto caminha de forma já profissional, mas ainda com muitas incertezas, especialmente no plano regulatório e de perspectivas de desenvolvimento do entorno necessário ao crescimento da importância do software livre/código aberto. A impressão é que todos sabem da importância do investimento em software livre/código aberto (até pelo que se vê no cenário internacional), mas o nível de risco ainda é elevado, especialmente por se tratar de mercado em formação, com um quadro de regras ainda em institucionalização. Portanto, o desenvolvimento profissional de software livre/código aberto no Brasil começa a dar os seus primeiros passos, tanto com empresas nacionais, como com multinacionais e com empresas públicas. A fase é de redução dos ainda elevados custos de transação para a profissionalização de software livre/código aberto no país.”

**SOFTWARE LIVRE/CÓDIGO ABERTO NO INÍCIO DA DÉCADA DE 2010**

Nos últimos anos, houve avanços na adoção das plataformas livres no país e no mundo. No Brasil, o governo desempenhou um papel relevante na adoção destas plataformas. O software livre/código aberto, no entanto, encontrou resistências para chegar às mãos dos usuários finais, apesar das melhorias nas interfaces gráficas e na usabilidade dos aplicativos livres. Também, conforme será visto mais adiante, com o surgimento de um novo paradigma tecnológico, em que a mobilidade ganha importância, os sistemas operacionais livres passaram a ter participação relevante no novo mercado, alcançando os dispositivos eletrônicos acessados por usuários finais, tais como *smartphones*, *tablets*, veículos em geral, relógios de pulso, etc. Entre os aplicativos, no entanto, predominam as versões proprietárias.

## Plataformas livres na administração pública

*No Brasil, quase 70% dos órgãos públicos federais e estaduais contam com sistema operacional livre. A maioria (98%) possui Windows.*

Parte muito significativa (98%) dos órgãos públicos federais e estaduais usa sistema operacional Microsoft/Windows. Software livre aparece como a segunda opção mais mencionada pela administração pública: quase 70% do total de órgãos públicos federais e estaduais que utilizam computador dispõem de sistema operacional livre (Linux/Ubuntu). Existe sobreposição no uso de sistemas operacionais, com os órgãos públicos assumindo estratégia baseada em uso híbrido: software proprietário (em geral, Microsoft/Windows) para algumas tarefas, software livre (Linux/Ubuntu) para outras (Tabela 3.1).

A penetração de sistemas operacionais livres é maior no sistema federal (78%) que no estadual (65%). Maior, também, entre instituições com mais de 250 pessoas ocupadas (PO) (73%) que entre instituições com porte inferior a 250 PO (42%). Observa-se que o uso de Linux/Ubuntu é relativamente maior no legislativo (84%) que no judiciário (79%), executivo (65%) e Ministério Público (69%).

**TABELA 3.1 – PROPORÇÃO DE ÓRGÃOS PÚBLICOS FEDERAIS E ESTADUAIS, CONSIDERANDO SISTEMA OPERACIONAL UTILIZADO NOS ÚLTIMOS 12 MESES – BRASIL, 2013**

Percentual sobre o total de órgãos públicos federais e estaduais que utilizam computador<sup>1</sup>

PERCENTUAL (%)		MICROSOFT/ WINDOWS			LINUX/UBUNTU			MACINTOSH/MAC OS			OUTRO		
		Sim	Não	NR	Sim	Não	NR	Sim	Não	NR	Sim	Não	NR
<b>Total</b>		98	1	0	67	32	1	18	81	1	16	84	1
<b>Poder</b>	Executivo	98	1	1	65	34	1	16	82	1	14	85	1
	Judiciário	97	3	0	79	20	1	25	73	1	35	64	1
	Ministério Público	100	0	0	69	28	3	17	79	3	14	86	0
	Legislativo	100	0	0	84	16	0	36	64	0	19	81	0
<b>Ente federativo</b>	Federal	98	2	0	78	21	1	31	69	1	31	68	1
	Estadual	98	1	1	65	33	1	16	82	1	14	85	1
<b>Porte</b>	Até 249 PO	98	2	0	56	42	1	16	82	2	12	86	1
	De 250 ou mais PO	99	1	0	73	26	1	20	79	1	20	80	0

NR = não sabe ou não respondeu. (1) Base: 1.586 órgãos públicos federais e estaduais que declararam utilizar computador nos últimos 12 meses. Dados coletados entre outubro e dezembro de 2013. Fonte: CETIC.br/CGI.br. Pesquisa TIC Governo Eletrônico 2013 – Órgãos Públicos Federais e Estaduais.

## Plataformas livres nas escolas

*A maioria das escolas brasileiras de ensino fundamental e médio utiliza sistema operacional Windows (87%). Linux aparece em segundo lugar, sendo adotado, sobretudo, em escolas públicas (57% do total). Nas escolas particulares, a adoção do Linux cai para 12%.*

Na Tabela 3.2, apresentam-se dados sobre a adoção de sistema operacional proprietário e livre em escolas brasileiras do ensino fundamental e médio. Observa-se que parte significativa das escolas que usam computadores adota o sistema operacional Microsoft/Windows (87% do total). O uso do Windows encontra-se disseminado nas escolas de todo o país. No entanto, é maior nas regiões Norte (93% do total) e Sudeste (92% do total). A sua presença é inferior nas escolas da região Sul onde, em contrapartida, encontram-se as maiores taxas de uso do sistema operacional Linux.

Nas instituições de ensino particulares, o uso do sistema operacional Microsoft/Windows encontra-se muito disseminado: 99% das que usam computadores adotam este sistema operacional. Entre escolas públicas estaduais e municipais, a adoção do sistema Microsoft também é elevada: 80% e 82%, respectivamente.

Linux é o segundo sistema operacional mais adotado nas escolas. Na média, 43% das que usam computador contam com este sistema operacional, que é empregado, sobretudo, nas escolas públicas (57% do total), com presença pouco significativa nas escolas particulares (apenas 12% do total). Entre as públicas, a presença é maior nas escolas estaduais: 64%. Nas escolas públicas, observa-se sobreposição do uso de Linux e Windows/Microsoft.

**TABELA 3.2 – PROPORÇÃO DE ESCOLAS DE NÍVEL FUNDAMENTAL E MÉDIO, CONSIDERANDO SISTEMAS OPERACIONAIS EM USO POR REGIÃO E DEPENDÊNCIA ADMINISTRATIVA – BRASIL, 2013**

Percentual sobre o total de escolas que possuem computador<sup>1</sup>

Percentual (%)		Microsoft / Windows	Linux	Macintosh / Mac OS	Outro	Não sabe / Não lembra
<b>TOTAL</b>		<b>87</b>	<b>43</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>Região</b>	Norte	93	60	0	0	1
	Centro-Oeste	83	59	0	0	0
	Nordeste	81	41	0	0	0
	Sudeste	92	29	0	0	0
	Sul	82	72	1	0	1

<b>Dependência administrativa</b>	<b>Pública Municipal</b>	82	57	0	0	0
	<b>Pública Estadual</b>	80	64	0	0	0
	<b>Total – Públicas</b>	81	61	0	0	0
	<b>Particular</b>	99	12	0	0	0

(1) Base: 929 escolas de nível fundamental e médio que possuem computador. Respostas múltiplas e estimuladas. Dados coletados entre setembro e dezembro de 2013. Fonte: CETIC/NIC.br. TIC Educação 2013 – Infraestrutura de TIC.

## Plataformas livres nas empresas

*24% das empresas que usam computador adotam sistema operacional de código aberto. Ele está presente em 67% daquelas com 250 ou mais pessoas ocupadas.*

Na Tabela 3.3, apresentam-se dados sobre o uso do sistema operacional de código aberto nas empresas brasileiras. Do total de empresas que possuem computador, 24% adotam sistema operacional de código aberto. Porte é um fator relevante para a adoção de sistema operacional livre. Grandes empresas têm chances maiores de adoção: 67% do total com 250 ou mais pessoas ocupadas contra 41% com faixa de porte entre 50 e 249 PO e 16% entre 10 a 49 PO.

As empresas do Nordeste foram as que apresentaram as maiores taxas de uso de código aberto: 27% do total. Os percentuais foram de 25% na região Centro-Oeste, 24% na Norte, 23% na Sudeste e 23% na Sul.

*A presença de sistema operacional de código aberto é consideravelmente maior em empresas do setor de informática e comunicação (54% do total).*

Mais da metade (54%) das empresas do setor de informática e Comunicação usou sistema operacional de código aberto nos últimos 12 meses. Este é o maior índice encontrado entre os segmentos de atuação cobertos pela pesquisa TIC Empresas e é muito superior ao encontrado em outros segmentos avaliados.

**TABELA 3.3 – PROPORÇÃO DE EMPRESAS QUE UTILIZARAM SISTEMA OPERACIONAL DE CÓDIGO ABERTO NOS ÚLTIMOS 12 MESES, CONSIDERANDO PORTE, REGIÃO E SEGMENTO DE ATUAÇÃO – BRASIL, 2013**

Percentual sobre o total de empresas que utilizam computador<sup>1</sup>

Percentual (%)		Sim	Não	Não Sabe/ Não respondeu
<b>Total</b>		<b>24</b>	<b>73</b>	<b>3</b>
<b>Porte</b>	De 10 a 49 pessoas ocupadas	16	81	3
	De 50 a 249 pessoas ocupadas	41	58	1
	De 250 ou mais pessoas ocupadas	67	33	0

Região	Norte	24	72	4
	Nordeste	27	71	2
	Sudeste	23	74	3
	Sul	23	74	3
	Centro-Oeste	25	72	3
Segmento de atuação - CNAE 2.0	Indústria de transformação	23	75	3
	Construção	18	80	2
	Comércio; reparação de veículos automotores e motocicletas	23	74	3
	Transporte, armazenagem e correio	28	70	2
	Alojamento e alimentação	13	83	4
	Atividades imobiliárias; atividades profissionais, científicas e técnicas; atividades administrativas e serviços complementares	37	62	2
	Informação e comunicação	54	45	1
	Artes, cultura, esporte e recreação; outras atividades de serviços	24	71	5

(1) Base: 6.225 empresas que declararam usar computador, com 10 ou mais pessoas ocupadas, que constituem os seguintes segmentos da CNAE 2.0 (C, F, G, H, I, J, L, M, N, R e S). Dados coletados entre setembro e dezembro de 2013. Fonte: CETIC/NIC.br. TIC Empresas 2013 – Infraestrutura de TIC.

Os dados apresentados mostram que, embora sem conseguir desbancar a posição de líder do Windows/Microsoft, Linux (ou derivados) avançou no país, chegando, no início da década de 2000, a ocupar posição de certo destaque no mercado brasileiro de sistemas operacionais. As plataformas livres foram adotadas, sobretudo, por instituições públicas e por empresas privadas de grande porte. A seguir, nas seções 3.2 e 3.3, busca-se conhecer mais sobre os motivos que limitaram o seu avanço.

## 3.2 A CADEIA DE VALOR DE SOFTWARE E SERVIÇOS DE TI

### CADEIA DE VALOR DO SOFTWARE

Nesta seção, apresentam-se as características dos diferentes elos da cadeia de valor do software (Figura 3.4) para, em seguida, localizar as empresas nacionais nesta cadeia e discutir a sua estratégia de posicionamento.

Upstream da cadeia de valor do software: infraestrutura = proximidade com o hardware, natureza genérica, não requer localização, permite ganhos de escala.

No *upstream* da cadeia de valor do software encontra-se o software de infraestrutura. Este tipo de software está muito próximo do hardware, fornecendo os comandos necessários para a sua operação. Nessa categoria, estão os sistemas operacionais, os softwares de gestão de segurança, armazenagem, etc. Uma das

principais características desse tipo de software tem a ver com o seu alto potencial de ganhos de escala, dada a sua generalidade: serve para diferentes setores econômicos, não requer localização, pode atender usuários finais de diferentes nacionalidades. No entanto, para ter razão de existir, depende de módulos complementares, desenvolvidos por fornecedores a jusante da cadeia de valor. Exemplos de software de infraestrutura são os sistemas operacionais ofertados pela Microsoft (Windows) e pela Red Hat (Linux).

**Middlestream da cadeia de valor do software: middleware = ferramentas para uso do desenvolvedor, natureza genérica, não requer localização, permite ganhos de escala.**

Softwares de *middleware* encontram-se no *middlestream* da cadeia de valor. Nesta categoria, incluem-se as linguagens de programação e os ambientes de desenvolvimento e, também, os softwares para gestão de banco de dados e *business intelligence*. O cliente deste tipo de software é o desenvolvedor de sistemas e soluções. O *middleware* necessita ser compatível com o software de infraestrutura, o que requer, portanto, parcerias entre os fornecedores de ambos os elos da cadeia de valor ou, o que também ocorre, que um mesmo fornecedor atue em ambos os elos. Os softwares de *middleware* também são genéricos, possuindo um alto potencial para ganhos de escala, podendo ser comercializados em diferentes países, prescindindo de localização. Exemplos de software de *middleware* são as linguagens de programação dotnet e Java ou, ainda, os bancos de dados Oracle e MySQL.

**Downstream da cadeia de valor do software: aplicativos = soluções para o cliente final, natureza mais ou menos genérica, requer localização. No geral, inibem ganhos de escala.**

Os aplicativos são softwares para uso do cliente final. Encontram-se no *downstream* da cadeia de valor do software. Podem ser horizontais, tais como as soluções para gestão empresarial (ERPs), gestão de relacionamento (CRMs), de recursos humanos e de projetos e as planilhas eletrônicas. Podem ser verticais, isto é, voltados para atendimento a necessidades específicas de um dado setor econômico, tais como são os softwares específicos para o segmento financeiro, as telecomunicações, o setor de petróleo e gás, saúde, educação, etc.

Empresas bem posicionadas nos demais elos da cadeia de valor costumam, também, complementar o seu portfólio com os aplicativos. Assim, por exemplo, a Oracle possui soluções para gestão empresarial e gestão de relacionamento. A Microsoft comercializa o pacote Office. As grandes empresas que atuam fim-a-fim convivem com empresas dedicadas de diferentes portes.

**Mercados para aplicativos horizontais costumam ser segmentados por território, domínios de aplicação e porte dos clientes, abrindo oportunidades para atuação de pequenas e médias empresas.**

Em comparação com o aplicativo vertical, as soluções horizontais permitem ganhos maiores de escala, pois atingem clientes finais de diferentes setores econômicos e pessoas com perfis heterogêneos. Mas, diferentemente dos softwares de infraestrutura e *middleware*, podem requerer localização, de modo a se adequar ao idioma, ao gosto, à cultura e às necessidades dos clientes de cada lugar específico. No mercado de aplicativos horizontais voltados para o segmento corporativo, costuma ocorrer segmentação baseada na especialização das soluções para um dado conjunto de domínios de aplicação (por exemplo, ERPs voltados para a indústria têxtil mas não apropriados para a indústria siderúrgica, etc.). Além desta segmentação por domínio de aplicação, existe, também, uma tendência para segmentação por porte do cliente, com soluções em gestão indicadas para grandes empresas e outras que se mostram mais adequadas para empresas de pequeno porte, por exemplo.

A segmentação do mercado de aplicativos horizontais e as necessidades de localização dos produtos tendem a diminuir os ganhos de escala, reduzindo barreiras à entrada de empresas de pequeno e médio porte.

Os ganhos menores de escala e a necessidade de localização do software criam barreiras à entrada de grandes empresas no mercado de aplicativos verticais.

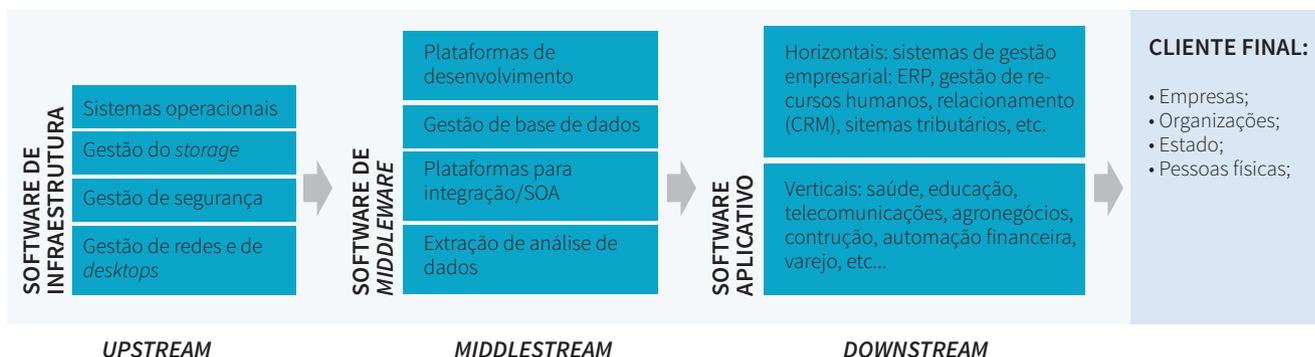
O aplicativo vertical também requer localização para os diferentes mercados. Além disso, propicia ganhos ainda menores de escala, por ser direcionado para um setor (ou segmento) econômico específico. Assim, por criar barreiras para a entrada das grandes, oferece um conjunto ilimitado de oportunidades para a atuação de pequenas empresas. A característica de segmentação das verticais por domínio de aplicação, porte do cliente e local de atuação contribui, também, para permitir o surgimento de um mercado relativamente pulverizado e especializado, constituído de pequenos e médios fornecedores para cada vertical ou segmento de vertical.

Grandes empresas de capital transnacional só entrarão em um dado mercado vertical quando perceberem que ele está suficientemente maduro, sendo capaz de propiciar bons retornos, ou seja, possui um número elevado de clientes requerendo tecnologia e com recursos financeiros disponíveis para arcar com os custos envolvidos na localização/customização e na manutenção de rede de distribuição e suporte. Assim, mercados ainda incipientes e imaturos são um terreno propício e reservado para a atuação de pequenas empresas.

Os aplicativos são desenvolvidos utilizando ferramentas selecionadas (software de *middleware*) que, por sua vez, encontram-se relacionadas aos softwares de infraestrutura. Assim, existe uma relação de parceria entre os atores dos diferentes elos da cadeia de valor do software, baseada em compatibilidade e interoperabilidade de soluções.

Algumas empresas consolidaram estratégia de atuação na cadeia de valor do software fim-a-fim. É o caso, por exemplo, da Microsoft, da SAP e da Oracle. Em alguns casos, a atuação transcende o software, com a empresa também sendo responsável pela fabricação do hardware. Um exemplo, neste caso, é a IBM.

FIGURA 3.4 – CADEIA DE VALOR DO SOFTWARE



Fonte: Observatório Softex.

Existe uma clara segmentação de funções: grandes empresas líderes de plataformas tecnológicas, operando como empresas-rede comandadas pelos países centrais, concentram-se no upstream e middlestream da cadeia de valor do software, podendo, também, complementar o seu portfólio com a oferta de aplicativos. Pequenas e médias empresas distribuídas por diferentes localidades tendem a se concentrar no downstream da cadeia de valor do software.

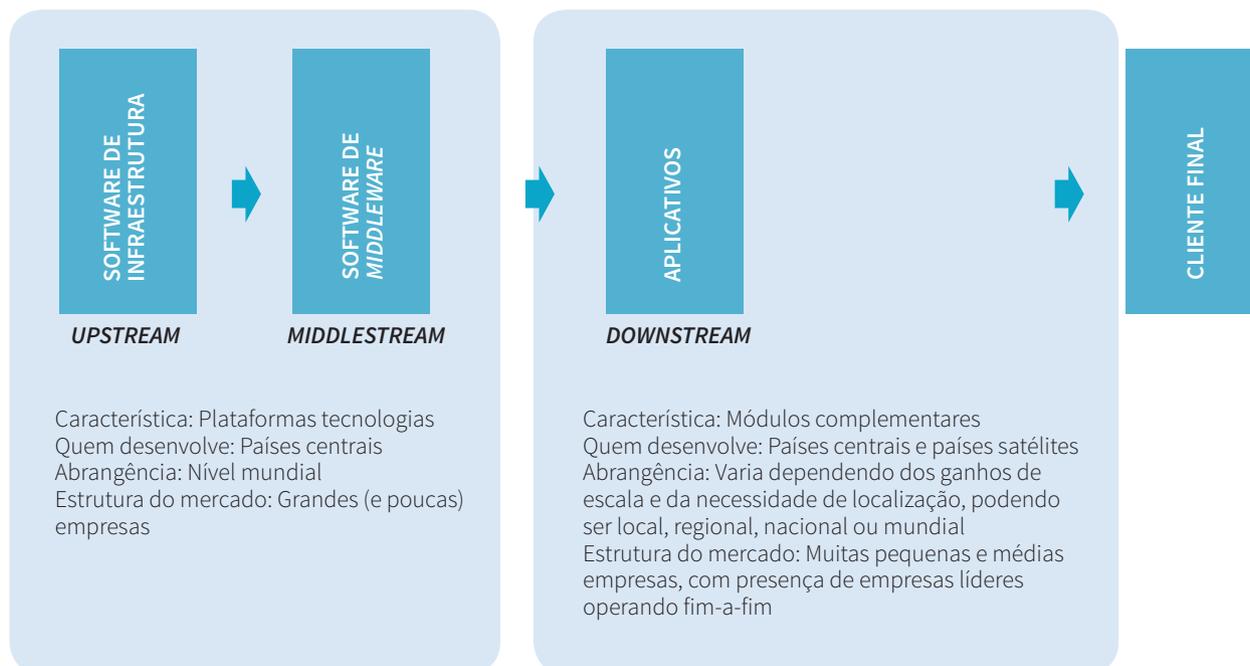
Na cadeia de valor do software, os desenvolvedores de aplicativos gravitam em torno das empresas fornecedoras de infraestrutura e de *middleware*, responsáveis pela consolidação das plataformas tecnológicas. Embora as grandes do setor possam criar estratégias de atuação fim-a-fim, mantendo presença em todos os elos da cadeia de valor do software, sempre haverá espaço para empresas interessadas em desenvolver módulos complementares, que não interfiram com o *core* dos negócios das grandes e que, ao contrário, reforcem os seus ganhos através das externalidades de redes (Figura 3.5).

Existe uma clara segmentação de funções. Grandes empresas líderes de plataformas concentram-se no *upstream* e *middlestream* da cadeia de software, ampliando, a partir daí, a sua atuação para o *downstream*, buscando estratégia fim-a-fim. Empresas que entraram no ambiente de software tardiamente tendem a desenvolver soluções complementares, em geral, aplicativos horizontais e verticais, muito em virtude das barreiras à entrada nos elos a montante da cadeia de valor do software. No mercado de aplicativos, as líderes de plataforma buscam negócios capazes de gerar ganhos de escala, concentrando-se, sobretudo, no atendimento aos clientes de porte. Pequenas empresas desenvolvem soluções altamente customizadas, para mercados de menor escala e clientes de menor porte.

As empresas com atuação no *upstream* e *middlestream* da cadeia de valor do software surgiram nos países desenvolvidos e foram, aos poucos, criando redes globais que se expandiram, sob a sua orquestração, pelos demais países. Trata-se de um grupo selecionado, composto por um número relativamente limitado de grandes empresas, *first movers* que conseguiram se consolidar após o surgimento de um dado paradigma tecnológico. Ao seu redor, gravita uma quantidade elevada de pequenas e médias empresas dedicadas ao desenvolvimento dos aplicativos. Essas empresas surgiram nos diferentes países e tendem a atender clientes locais, dadas as peculiaridades do seu produto: necessidade de localização, suporte ao usuário final, conhecimento profundo dos processos do cliente, etc.

Em alguns casos, pequenas e médias empresas conseguem espaço para desenvolvimento de módulos complementares de software de infraestrutura e *middleware*. No entanto, parte significativa das oportunidades para novas entrantes vincula-se ao desenvolvimento de aplicativos. Assim, no geral, empresas desenvolvedoras de software localizadas nos países periféricos participam das redes globais de software já consolidadas, desenvolvendo aplicativos que rodam em plataformas tecnológicas desenvolvidas por empresas dos países centrais. Dada a forte relação deste tipo de software com as necessidades específicas de um dado porte de cliente e de uma dada maturidade de mercado, a atuação de empresas dedicadas ao desenvolvimento de aplicativos tenderá a ser local.

FIGURA 3.5 – CADEIA DE VALOR DO SOFTWARE: ESPECIFICIDADES ESTRUTURAIS CONSIDERANDO OS DIFERENTES ELOS



Fonte: Observatório Softex.

*O posicionamento das empresas de software na cadeia de valor tem impacto no perfil inovacional. Para empresas no upstream e middlestream, a tecnologia e os aspectos técnicos são muito relevantes. Para empresas no downstream, a tecnologia é um dado; importa mais o conhecimento das necessidades do cliente.*

O posicionamento das empresas de software na cadeia de valor condiciona, ampliando ou restringindo, as possibilidades de ganhos de escala e geração de renda e os riscos envolvidos na empreitada. Além disso, também traz distinções relevantes no que diz respeito ao processo inovativo: atividades de pesquisa e desenvolvimento, parceiros com os quais colaborar e as fontes de informação a serem apreciadas (Quadro 3.1).

O desenvolvimento de software de infraestrutura e de *middleware* requer um conhecimento sobretudo técnico, direcionado para engenharia e ciência da computação. Os fatores críticos de sucesso estão ligados à capacidade técnica de desenvolver uma plataforma tecnológica antes dos demais e conseguir atrair parceiros interessados no seu uso. Ao contrário, desenvolvedores de aplicativos dependem do conhecimento do mercado. Eles se amparam na técnica preexistente (sistemas operacionais disponíveis e ambientes de desenvolvimento, por exemplo) para gerar valor para o cliente. Os fatores críticos de sucesso, neste caso, têm a ver com a compreensão das necessidades do usuário final e com a seleção acertada do parceiro tecnológico, ou seja, a escolha correta da plataforma sobre a qual desenvolver a sua solução de mercado.

Pesquisa e desenvolvimento tecnológico parecem mais relevantes para as empresas posicionadas no *upstream* e no *middlestream* da cadeia de valor do que para empresas atuantes no *downstream*. As

possibilidades de inovar também são diferentes dependendo do posicionamento da empresa na cadeia de valor do software. No primeiro caso, a inovação é amparada pela tecnologia e pela chegada no momento certo ao mercado. No segundo, a inovação tem a ver com o uso eficiente e o domínio apropriado da tecnologia existente para gerar um novo produto para o cliente final.

Essas diferenças explicam por que muitas empresas de software de infraestrutura e de *middleware* surgem das universidades (*startups*) e por que, ao contrário, muitas empresas de aplicativos, especialmente verticais, nascem de *spin-offs* de departamentos de TI de empresas de diferentes setores econômicos (bancos, varejo, manufatura, etc.). Explica a interação maior que empresas desenvolvedoras daquele tipo de software precisam estabelecer com o ambiente universitário e a relevância maior que o ambiente empresarial possui para empresas de aplicativos.

Quanto mais se caminha do *upstream* para o *downstream* da cadeia de valor do software, mais se necessita de conhecimentos do negócio do usuário final (processos e funções), menos se requer conhecimento de tecnologia. O tempo é um fator crítico relevante. Os aplicativos são desenvolvidos após o surgimento das plataformas tecnológicas e, portanto, quando parte significativa do que precisava ser feito em termos tecnológicos já foi realizado, havendo, no entanto, necessidade e brechas para complementações e aprimoramentos contínuos, de natureza incremental, sobretudo no que se refere às funcionalidades a serem entregues para os clientes finais.

Considerando as diferenças na dinâmica, estrutura e história das empresas de software atuantes nos diferentes elos da cadeia de valor do software, seria equivocado, portanto, comparar o resultado dos seus processos inovativos. Mais que isto, seria um erro julgar que empresas de aplicativos poderiam ter comportamento inovativo semelhante às empresas líderes de plataformas tecnológicas.

**QUADRO 3.1 – P&D&I EM SOFTWARE, CONSIDERANDO POSICIONAMENTO NA CADEIA DE VALOR**

ITENS	INFRAESTRUTURA/MIDDLEWARE	APLICATIVO
Fatores críticos de sucesso	Fundamental ser o primeiro. Criar as externalidades de redes e os efeitos de <i>lock-in</i>	Clientes em número suficiente para bancar o negócio. Boa estrutura de marketing e distribuição/revenda. Capacidade de escolher o parceiro ‘certo’ de tecnologia (plataforma tecnológica).
Atividades de P&D	Conhecimentos tecnológicos são relevantes especialmente no momento de construção da plataforma tecnológica.	Conhecimentos das necessidades do usuário final são relevantes.
Inovação	Radical no início, incremental após consolidação da plataforma	Incremental, através de módulos para o cliente final.
Competências	No início, tecnológica. Após consolidação da plataforma, capacidade para articular a rede globalmente, buscando externalidades e efeitos de <i>lock-in</i> . Ampliar o portfólio de produtos, expandir a atuação fim-a-fim.	Conhecer o negócio e criar interação com o cliente, melhorar o produto a partir das avaliações (e do conhecimento) do cliente. Ter domínio da plataforma tecnológica do parceiro e acompanhar a sua evolução.
Relação com universidades e centros de pesquisa	Intensa nos momentos de desenvolvimento do software.	Fraca, sendo mais necessária a relação com o fornecedor de plataforma tecnológica

Fontes de informação	No início, universidades e centros de pesquisa. Após consolidação da plataforma: empresas parceiras.	Cliente final e fornecedores de plataforma.
----------------------	---	---

Fonte: Observatório Softex (2014).

## CADEIA DE VALOR DOS SERVIÇOS DE TI

**Prestadores de serviços de TI (consultorias, reparadores de equipamentos, empresas de suporte técnico, fábricas de software, integradores, revendedores e franqueados, etc.) gravitam em torno das plataformas tecnológicas, cuidando de etapas específicas do ciclo de vida do software.**

Da mesma forma que os desenvolvedores de aplicativos, os prestadores de serviços de TI gravitam, também, em torno das plataformas tecnológicas existentes, fornecendo serviços diversos, incluindo desenvolvimento de software sob demanda, consultoria, suporte técnico, tratamento de dados e reparação e manutenção de computadores e equipamentos periféricos. Os serviços de TI, portanto, abrangem uma quantidade bastante diversificada de atividades econômicas, com potenciais distintos para gerar valor.<sup>1</sup>

Os três tipos de software (infraestrutura, *middleware* e aplicativo) abrem um leque amplo de oportunidades para empresas de serviços de TI. Assim, por exemplo, o surgimento de uma dada linguagem de programação permite que um sem número de empresas forneça capacitação na linguagem. O mesmo ocorre com o surgimento de um novo software de gestão empresarial. Em muitos casos, as próprias empresas globais se incumbem de oferecer cursos que visam à capacitação de profissionais em suas plataformas e de clientes finais no uso de seus aplicativos. Para empresas que desenvolvem aplicativos e para empresas que fornecem consultoria e suporte técnico, a certificação dos seus profissionais nas soluções das empresas globais muitas vezes valem mais que o certificado obtido nas universidades.

A capacitação de recursos humanos não é a única possibilidade de empresas prestadoras de serviços participarem das redes constituídas pelas grandes fornecedoras de plataformas tecnológicas atuantes no *upstream* e *middlestream* da cadeia de valor do software. Existem várias oportunidades para consultores, integradores, revendedores, prestadores de serviços de desenvolvimento, implantação, localização, customização, documentação e testes ao longo da cadeia de valor de software.

**Segmentação do ciclo de vida do desenvolvimento de software abriu oportunidades para empresas prestadoras de serviços realizarem atividades rotineiras, de menor valor agregado.**

Entre os serviços de TI, vale ressaltar o papel dos serviços de desenvolvimento de software (software sob encomenda). Durante muito tempo, os fornecedores de TI foram contratados pelo cliente final para desenvolvimento de todas as etapas do ciclo de vida do software. Mais recentemente, houve uma segmentação

<sup>1</sup> Considerando a versão 2.0 da Classificação Nacional de Atividades Econômicas (CNAE), o Observatório Softex inclui as seguintes classes na categoria de serviços de TI: 6201 – desenvolvimento de software sob encomenda; 6204 – consultoria em TI; 6209 – suporte técnico, manutenção e outros serviços de TI; 6311 – tratamento de dados, provedores de serviços de aplicação e de hospedagem na Internet; 6319 – portais, provedores de conteúdo e outros serviços de informação na Internet; 9511 – reparação e manutenção de computadores e de equipamentos periféricos. Na categoria software, o Observatório Softex inclui duas classes CNAE: 6202 – desenvolvimento e licenciamento de software customizável e 6203 – desenvolvimento e licenciamento de software não customizável.

do ciclo. Etapas de desenvolvimento tidas como estratégicas, tais como a especificação e a arquitetura, são realizadas pela empresa solicitante dos serviços ou por um fornecedor por ela contratado para este fim. As demais etapas, envolvendo atividades rotineiras, menos nobres, tais como codificação, suporte técnico, teste ou localização, foram encomendadas a outros fornecedores. A segmentação deu origem a uma quantidade relativamente grande de empresas especializadas em etapas específicas do ciclo de desenvolvimento, incluindo as chamadas fábricas de software, responsáveis pelas atividades de codificação. Muitas vezes, essas fábricas especializam-se no atendimento de negócios relacionados com uma dada plataforma tecnológica e linguagem de programação.

Empresas globais (tanto as líderes de plataformas tecnológicas como as empresas de outros setores que desenvolvem software *in house*) utilizam serviços de desenvolvimento de software de fornecedores localizados em diferentes países. Os motivos que levaram à segmentação das etapas de desenvolvimento do software entre vários fornecedores têm a ver com questões econômicas (redução dos custos de desenvolvimento), com um melhor aproveitamento das competências existentes em um dado local (por exemplo, abundância de recursos humanos com *expertise* de interesse) e com a carência de recursos humanos no país contratante. Países como a Índia, por exemplo, conseguiram participar das redes globais, assumindo partes menos nobres do processo de desenvolvimento do software. A Irlanda se tornou um *site* importante para a localização de produtos para o mercado europeu.

No Brasil, as fábricas de software encontram-se menos conectadas às redes globais. Em geral, os seus clientes encontram-se no país. São consumidores finais corporativos (bancos, instituições públicas e empresas de serviços de telecomunicações) que especificam o software de interesse internamente (com ou sem apoio de consultores) e terceirizam os serviços de codificação para as fábricas. Parte significativa destes clientes mantém, ainda, software proprietário que roda em sistemas legados.

***Em linhas gerais, com software livre/código aberto não é diferente: pequenas e médias empresas gravitam em torno da plataforma líder.***

Com software livre/código aberto não é diferente. Pequenas e médias empresas, profissionais de desenvolvimento e colaboradores gravitam em torno da plataforma tecnológica Linux e em torno de outras várias plataformas (Python, Mozilla, Apache, etc.), fornecendo apoio para o desenvolvimento e aprimoramento das soluções e oferta de serviços variados de TI (consultorias e capacitação, integração de sistemas e suporte técnico, etc.).

***No universo do software livre/código aberto, por questões relacionadas com as possibilidades maiores de aprendizado técnico, o envolvimento com os softwares de infraestrutura e middleware tenderá a ser privilegiado.***

Quanto mais genérico o software, maiores serão as possibilidades de ganhos de escala. Sendo assim, é possível que empresas e profissionais interessados em garantir retorno financeiro queiram optar pela prestação de serviços relacionados com software de infraestrutura e *middleware*. Além disso, pelo fato de a comunidade de software livre ser altamente motivada pelos ganhos provenientes da aprendizagem colaborativa, é provável que prefiram se juntar às comunidades de software de infraestrutura e *middleware*, onde, em teoria, existem possibilidades maiores de desafios técnicos, que às comunidades de aplicativos, em que os maiores desafios tendem a estar relacionados com os processos de negócios.

Dessa forma, e por razões diversas, para as comunidades de software livre/código aberto, é possível que o mercado de aplicativos tenha se mostrado menos atraente que o de software de infraestrutura e *middleware*. Por um lado, aquele mercado envolvia mais conhecimento do negócio que aprendizado de técnicas. Por outro, permitia retornos menores de escala, trazendo incertezas para o desenvolvedor que visava à obtenção de ganhos provenientes de serviços relacionados ao uso do aplicativo.

**Downstream da cadeia de valor do software: elo mais resistente ao software livre/código aberto.**

Vale lembrar, também, que a chegada pioneira do Windows no mercado de computadores de mesa provocou efeitos de aprisionamento relevantes, que dificultaram a entrada de plataformas concorrentes. Os efeitos de *lock-in* mostraram-se especialmente fortes no *downstream* da cadeia de valor do software. O elo final foi mais resistente ao software livre/código aberto por dois motivos: entre os atores responsáveis pela seleção do software, o usuário final que acessa aplicativos no trabalho ou na residência é o que menos tende a perceber vantagens com a substituição dos programas em uso por outros. Também, em geral, é o que menos sabe ‘mexer’ no computador e o que menos possui condições para comparar o software em uso com o produto de concorrentes eventuais, desde a perspectiva dos seus atributos técnicos ou funcionais. Assim, os aplicativos, por serem endereçados ao usuário final, são o tipo de software mais sujeito aos efeitos de *lock-in*.

**No Brasil, a expectativa criada no ano 2000 de o software livre/código aberto chegar aos microcomputadores, fornecendo aplicativos para o usuário final, não se concretizou. A questão era: como construir um modelo de negócios sustentável?**

No início dos anos 2000, muitos pensaram que o uso de aplicativos livres seria uma questão de tempo. A conquista do mercado dos usuários finais aconteceria com a maturação a caminho das interfaces gráficas. No entanto, esta última fronteira mostrou ser a de mais difícil conquista.

Uma forma para contornar a resistência do usuário final à conveniência dos sistemas proprietários já disseminados seria tornar os aplicativos livres muito fáceis de usar, fáceis a ponto de não requerer custos de aprendizagem e nem a necessidade de suporte técnico para o uso. No entanto, a usabilidade navegava contra uma possível lógica de negócios baseada na oferta de serviços de suporte, implantação, consultoria e treinamento. Como construir um modelo de negócios sustentável, quando a licença de uso é fornecida gratuitamente e não há perspectivas de retorno em serviços?

**Software livre/código aberto e software proprietário: a lógica é semelhante. A plataforma líder exerce um papel preponderante, atraindo empresas ao seu redor. No entanto, no ambiente livre, há maiores desafios para a consolidação de uma plataforma hegemônica.**

O ambiente de software livre/código aberto é estruturado por lógica semelhante à que se verifica no universo do software proprietário: a plataforma líder exerce um papel preponderante, atraindo empresas ao seu redor. Assim, considerando o mundo dos *desktops*, da mesma forma que empresas de capital nacional estruturaram os seus negócios em torno da plataforma Windows, no mundo do software livre/código aberto, as nacionais (ou profissionais vinculados às comunidades brasileiras) orbitaram ao redor do sistema operacional Linux. O fato de que haja liberdade para usar, modificar e distribuir Linux não retira a importância desta plataforma tecnológica como núcleo básico, fundamental, sobre o qual várias empresas e

comunidades gravitam. No entanto, as possibilidades de reuso e recombinações propiciadas pela filosofia de software livre/código aberto estimulam a proliferação de novas plataformas e tendem a agilizar o surgimento de inovações incrementais não controladas pela líder.

A supremacia de uma empresa (ou comunidade) sobre as demais ocorre em virtude das externalidades de redes, ou seja, dos retornos crescentes propiciados pelo fato de uma dada plataforma ser aceita por um número maior de empresas parceiras e usuários. Embora existam efeitos de *lock-in*, eles tendem a ser mais atenuados no ambiente de software livre/código aberto, em virtude das liberdades concedidas. Apesar da maior facilidade de se transitar entre alternativas concorrentes, há custos de migração relacionados com o conhecimento acumulado sobre o manuseio de um dado software e com os ganhos já obtidos com os relacionamentos estabelecidos nas redes criadas, por exemplo. Dada a miríade de opções disponíveis e a maior possibilidade de se transitar entre elas, já que os custos de migração tendem a ser mais baixos (quando comparados com software proprietário), parece existir oportunidades menores de apropriação. Neste cenário, as parcerias surgem como alternativa para reduzir os custos envolvidos nos esforços de desenvolvimento dos novos produtos e há maiores desafios para a consolidação de uma plataforma hegemônica no universo do software livre/código aberto.

### **3.3 SOFTWARE LIVRE/CÓDIGO ABERTO E SOFTWARE PROPRIETÁRIO: CÁLCULO DE CUSTO-BENEFÍCIO**

O surgimento de um mercado de software pujante requer, por um lado, a existência de demanda, ou seja, pessoas físicas, empresas ou organizações que manifestem interesse em utilizar software. Exige, por outro, a presença de fornecedores, ou seja, pessoas, empresas ou organizações que se mostrem interessadas em ofertar software.

Por que um demandante de software iria preferir adotar software livre/código aberto? Em que casos a adoção deste tipo de software traz benefícios e em que situações apresenta obstáculos? Da mesma forma, o que levaria um fornecedor de software a optar pela oferta de software livre/código aberto? A seguir, propõe-se um cálculo de custo-benefício baseado em decisões dos diferentes atores no que se refere à adoção de software. Posteriormente, considerando vários tipos de software, avaliam-se as ameaças e as oportunidades que o software livre/código aberto traz para ofertantes e demandantes.

#### **A PERSPECTIVA DO DEMANDANTE**

*Existem diferentes perfis de usuários. Os cálculos de custo-benefício envolvendo decisões sobre software não são coincidentes.*

Existem diferentes perfis de usuários de software, incluindo consumidores domésticos e corporativos, públicos e privados. No mundo corporativo, os usuários podem ser segmentados em pelo menos três grandes categorias: os responsáveis pela empresa/organização, a quem recai a decisão final sobre as soluções em TI a serem adotadas internamente; a área/departamento de TI da empresa/organização, a quem possivelmente recaia a tarefa de avaliar as soluções em TI disponíveis e guiar as decisões de compra; e os empregados da empresa/organização, que irão empregar produtos e soluções adquiridas pela empresa/

organização no seu dia a dia. No ambiente doméstico, há vários perfis de consumidores, com nível cultural e disponibilidade de recursos financeiros distintos.

As preferências desses usuários por software não são coincidentes, e nem a avaliação que fazem das vantagens e desvantagens de adoção de um tipo ou outro de software, em cenário de mudança. Mesmo dentro de um único subconjunto de usuários, a avaliação do custo-benefício de adoção do software livre/código aberto pode variar. No caso de empresas, as decisões variam, entre outras coisas, conforme porte, aplicação e existência prévia de sistemas legados. No caso das pessoas físicas, as decisões estão ligadas a fatores socioculturais e econômico-financeiros.

### Dirigentes (CEO e CIO) de empresas/organizações demandantes

*Os custos são elementos críticos nas decisões dos dirigentes de empresas e organizações quanto ao software a ser adquirido. Vários custos são considerados, além daqueles envolvidos na aquisição de licenças de uso.*

Segundo levantamento realizado pela Softex (2005), as empresas usuárias decidem adotar software livre/código aberto principalmente para reduzir custos. Quanto maior a empresa, maior o interesse no uso de software livre/código aberto, em virtude da grande economia que pode ser feita em licenças de uso. No entanto, outros fatores afetam a decisão.

Desde a perspectiva da empresa usuária, o custo total de adoção de um dado software inclui, além da licença de uso (L), despesas relacionadas com a instalação do software e a integração entre as soluções em uso e o software a ser adquirido (A). Somam-se, a isto, os custos envolvidos no suporte técnico (S) e, também, as despesas necessárias para capacitação do pessoal (E). Dentro deste contexto, o que, em princípio, poderia parecer uma economia, pode acabar saindo caro.

Além do custo total, a decisão pelo software a ser adquirido deve considerar, também, fatores relacionados com atributos técnicos e funcionais do software (ATF), a disponibilidade de suporte técnico qualificado (DSQ) e as dificuldades de uso e resistências a mudanças da equipe interna, já acostumada a produtos concorrentes (DRE). Questões ideológicas e aspectos subjetivos (IS) podem influenciar, também, na decisão final (Quadro 3.2).

#### QUADRO 3.2 – ELEMENTOS DO CÁLCULO DE DECISÃO POR SOFTWARE LIVRE/CÓDIGO ABERTO, CONSIDERANDO A PERSPECTIVA DOS DIRIGENTES (CEO E CIO) DA EMPRESA/ORGANIZAÇÃO USUÁRIA

$D_{\text{final}} = C_{\text{total}}(L+A+S+E) + ATF + DSQ + DRE + IS$ , sendo:

$D_{\text{final}}$  = decisão final

$C_{\text{total}}$  = L (licença de uso) + A (ambiente preexistente, envolvendo custos com integração, interoperabilidade e softwares complementares) + S (custo do suporte técnico) + E (custo de capacitação da equipe)

ATF = atributos técnicos e funcionais

DSQ = disponibilidade do suporte técnico qualificado

DRE = dificuldade de uso e resistência da equipe (profissionais de TI e outros usuários)

IS = questões ideológicas e subjetividades

Assim, considerando os fatores que determinam a decisão de aquisição de software do responsável pela empresa/organização:

1- Ela tenderia a ser favorável ao software livre/código aberto nos casos em que o diferencial do software baseia-se no preço. Ou seja, nos casos em que produtos rivais possuem os mesmos atributos técnicos e funcionalidades e as mesmas condições de suporte, não havendo vantagens claras no uso de uma ou outra solução.

2- Nas situações em que características técnicas ou funcionais são relevantes, o software livre/código aberto poderia ou não se mostrar a melhor opção, dependendo do atributo de interesse. Assim, por exemplo, a auditabilidade é ponto a favor das soluções livres, o que as credencia especialmente para uso do governo, em aplicações críticas, envolvendo soberania nacional e defesa cibernética. Em situações em que o software cria vantagens competitivas relevantes para a empresa que o possui, o usuário poderia preferir mantê-lo em sigilo, para o seu uso exclusivo, optando, por exemplo, pelo desenvolvimento sob encomenda ou *in house*.

3- Fator que poderia levar empresas/organizações a preferir soluções proprietárias tem a ver com os riscos envolvidos com o futuro das soluções livres, caso a equipe desenvolvedora não dê continuidade ao desenvolvimento. Esse risco pode ser atenuado em empresas de grande porte que mantêm equipe interna de suporte/desenvolvimento, pois estas podem, no caso de abandono da solução pela comunidade desenvolvedora, vir a arcar com os aprimoramentos necessários. Ou seja, a existência de equipe interna de desenvolvimento pode contribuir para a redução de incertezas, pois assegura a continuidade, mesmo na falta de suporte externo. Isso ajuda a explicar por que as grandes empresas parecem mais propícias à migração para software livre/código aberto que empresas de pequeno porte. O software proprietário poderia ser escolhido em casos críticos, dada a suposta maior garantia de continuidade e suporte.

4- Questões ideológicas poderiam fazer com que a escolha recaísse sobre o software livre/código aberto. Isso é o caso, sobretudo, nas instituições governamentais, cuja lógica de ação é guiada pelo bem comum.

5- Insegurança ou desconhecimento poderiam fazer com que as empresas usuárias preferissem as marcas já reconhecidas pelo mercado, seja de software proprietário ou de software livre/código aberto. A posição de seguidor fornece maior segurança e reduz incertezas. O fato tende a privilegiar a plataforma tecnológica líder, seja ela qual for.

6- Em nível local, a inserção de software livre/código aberto seria facilitada em situações de inexistência de plataforma tecnológica proprietária já consolidada. Nos casos em que esta consolidação já existe, a presença de empresas fortes interessadas em competir com a líder e/ou de governo disposto a atenuar ou impedir o poder da líder poderiam levar a um equilíbrio mais favorável à adoção do software livre/código aberto.

## Profissionais do departamento de TI das empresas/organizações demandantes

É provável que profissionais de TI das empresas e organizações tenham interesse em adotar software livre/código aberto por motivo de aprendizado. Atributos técnicos e funcionais do software também tendem a ser considerados no processo decisório.

Desde a perspectiva dos profissionais de TI de empresas e organizações, é provável que haja interesse em adotar o software livre/código aberto por motivos de aprendizagem (AP), já que existem ganhos envolvidos na possibilidade de vasculhar o software, aprender com o uso e interagir com as comunidades livres. A sua motivação seria maior quanto mais o profissional apreciasse o modelo de aprendizado do estilo aprender fazendo e mais aberto estivesse a riscos e incertezas (EE). A existência prévia de legados poderia funcionar como um fator de resistência ou de estímulo à mudança, dependendo do perfil do profissional (A).

A presença de uma comunidade sólida de apoio ou de empresas habilitadas para a prestação de suporte técnico (DSQ) também poderia contribuir favoravelmente para o interesse do profissional de TI pelo software livre/código aberto. Além disso, profissionais de TI são sensíveis aos atributos técnicos e funcionais das soluções, levando-as em conta ao comparar produtos concorrentes (ATF). Questões ideológicas e aspectos subjetivos (IS) poderiam influenciar, também, o seu interesse (IS) na adoção do software livre (Quadro 3.3).

### QUADRO 3.3 – ELEMENTOS DO CÁLCULO DE INTERESSE POR ADOÇÃO DE SOFTWARE LIVRE/CÓDIGO ABERTO, CONSIDERANDO A PERSPECTIVA DOS PROFISSIONAIS DE TI DA EMPRESA/ORGANIZAÇÃO USUÁRIA

$I_{\text{final}} = AP + EE + A + DSQ + ATF + IS$ , sendo:

$I_{\text{final}}$  = interesse final

AP = motivação para aprender fazendo

EE = espírito empreendedor; desejo de assumir riscos e incertezas envolvidos em mudanças

A = ambiente preexistente

DSQ = disponibilidade do suporte técnico qualificado

ATF = atributos técnicos e funcionais

IS = questões ideológicas e subjetividades

## Outros profissionais das empresas/organizações demandantes

Os profissionais que utilizam software aplicativo em suas atividades diárias tendem a resistir bravamente à substituição das soluções em uso. O custo envolvido em novos aprendizados apresenta-se como o motivo principal de resistência.

Os empregados da empresa/organização costumam resistir à migração de um ambiente de soluções em uso para outro ambiente. A resistência é resultado das externalidades de redes e dos efeitos de *lock-in* criados pelas plataformas tecnológicas líderes e afeta especialmente os empregados que trabalham em

seus *desktops* acessando aplicativos diversos. Para esta categoria de usuário final, não parece haver uma grande motivação para a mudança, o que torna especialmente difícil substituir as soluções em uso.

Quanto mais consolidada uma plataforma se encontra, gerando externalidades de redes e efeitos de *lock-in* (A), maiores serão os obstáculos para a substituição dos aplicativos nas mãos destes usuários. Eles têm pouco a ganhar com as mudanças, já que as reduções de custos financeiros, de um modo geral ( $C_{total}$ ) não os afeta. Poderiam, talvez, se beneficiar substituindo um software menos funcional ou muito complexo, por soluções com funcionalidades superiores e mais simples de usar. No entanto, estes usuários são pouco sensíveis a argumentos envolvendo atributos técnicos e funcionais (ATF). Os custos envolvidos em novas capacitações (E) desempenham papel relevante, jogando contra a substituição de soluções. A disponibilidade de suporte técnico qualificado (DSQ) poderia reduzir a sua resistência. Um perfil pessoal mais empreendedor (EE) e aspectos ideológicos e subjetividades (IS) também poderiam contribuir para atenuar as suas resistências à mudança (Quadro 3.4).

**QUADRO 3.4 – ELEMENTOS DO CÁLCULO DE INTERESSE POR ADOÇÃO DE SOFTWARE LIVRE/CÓDIGO ABERTO, CONSIDERANDO A PERSPECTIVA DE OUTROS PROFISSIONAIS DA EMPRESA/ORGANIZAÇÃO USUÁRIA**

**$I_{final} = E + EE + A + DSQ + IS$ , sendo:**

$I_{final}$  = interesse final  
 E = custo envolvido na capacitação  
 EE = espírito empreendedor; desejo de assumir riscos e incertezas envolvidos em mudanças  
 A = ambiente preexistente  
 DSQ = disponibilidade do suporte técnico qualificado  
 IS = questões ideológicas e subjetividades

Em ambiente de existência de plataforma consolidada e considerando o interesse de migração de plataformas tecnológicas, uma questão importante que se coloca para os dirigentes de empresas e organizações diz respeito ao trabalho que deve ser realizado junto aos profissionais da empresa. Como motivá-los à mudança?

No Quadro 3.5, relatam-se as experiências de várias instituições públicas que buscaram migrar de software proprietário para software livre/código aberto. Com exceção do Exército Brasileiro, em que os usuários finais tiveram de acatar decisão que foi tomada pelas instâncias superiores, os entrevistados das demais instituições fazem menção a um trabalho árduo, e necessário, de persuasão dos usuários finais. Em todos os casos, houve resistência por parte dos usuários. A decisão de migração considerou aspectos financeiros, atributos técnicos e questões ideológicas. Mas os entrevistados citam, também, a existência de políticas favoráveis à adoção do software livre/código aberto.

### QUADRO 3.5 – A ADOÇÃO DE SOFTWARE LIVRE/CÓDIGO ABERTO EM INSTITUIÇÕES PÚBLICAS

“A implantação da cultura de software livre no CTI é uma diretriz cuja conquista está sendo obtida paulatinamente, sem imposição. É meta a ser alcançada por meio de um processo pedagógico e de persuasão constante. É necessário cumprir tarefa didática, mostrando as vantagens técnicas e econômicas.

A ênfase tem que ser dada ao software livre uma vez que o usuário, do ponto de vista do governo não pode ser contemplado somente como consumidor. As empresas têm em mente o cliente. O governo pensa no cidadão. Como cidadão, deve ser incluído na sociedade, a qual está se digitalizando a uma velocidade espantosa conforme indicam as últimas pesquisas.

Hoje em dia, o software livre já é bastante aceito no CTI. A comunidade sabe que existe uma legislação brasileira que determina o uso do software livre. O PDTI do CTI determina o alinhamento com a legislação federal. Sempre que houver disponibilidade das duas categorias de software (livre e proprietário), executando as mesmas funções, a escolha deve recair sobre o software livre. A adoção plena não só do Linux como de outras plataformas similares é uma questão de tempo.”

[Extraído de depoimentos de integrantes da diretoria do Centro de Tecnologia da Informação Renato Archer (CTI)]

“A desconfiança [em relação ao software livre] diminuiu, mas ainda tem muito desconhecimento embora seja largamente usado. Na Unicamp, existe uma política de sempre se procurar alternativa em software livre antes de se comprar uma solução comercial. Ao se falar de filosofia de software livre há um bloqueio, um desinteresse. O Open Office é pouco usado embora tenha boa qualidade.”

[Extraído de depoimento de profissional do CCUC/Unicamp]

“Não existe determinação legal para uso de software livre na Prefeitura de Campinas. Existe uma recomendação interna (ordem de serviço). No momento de compra, se houver opção de software livre, ele deve ser usado. Todo o desenvolvimento da IMA é feito em software livre, todo o nosso quadro interno de desenvolvedores conhece software livre, mas as equipes de manutenção trabalham também com software proprietário.”

[Extraído de depoimento de gerente da IMA]

“O Plano de Migração para Software Livre no Exército Brasileiro foi publicado, em sua 1ª edição, em novembro de 2004. A implementação foi iniciada por ordem do Chefe do Departamento de Ciência e Tecnologia do Exército. Atualmente, a adesão é de cerca de 80%/90% no Exército Brasileiro. Software proprietário só é utilizado em caso de extrema necessidade e quando justificado.

No ambiente militar, portanto, a implantação ocorreu através de uma ordem de serviço do comando. De um dia para o outro toda a comunidade foi informada de que dali em diante estava vetado o uso de software proprietário.

No início, houve uma forte resistência. As chamadas ao suporte cresceram de modo significativo e, muitas vezes, tinham a ver com pequenos problemas que, na verdade, não eram de fato problemas e sim falta de familiaridade com o novo ambiente computacional. Isto causou uma demanda de suporte constante. Foi necessária a implementação de um plano de treinamento com palestras explicando as diferenças e as mudanças, mostrando as vantagens do novo ambiente, incentivando a adesão e participação de todos.

Aos poucos, o usuário foi se adaptando e após vários anos de implementação pode-se dizer que hoje a comunidade está gostando e dando preferência ao software livre, inclusive no ambiente domiciliar.

A decisão de mudança foi de caráter nacional, valendo para todas as organizações militares federais do Brasil. Como a hierarquia no setor é muito forte, acredita-se que o nível de implementação seja muito alto nas Forças Armadas.”

[Trecho de entrevista com militares do Exército Brasileiro – Comando de Campinas]

## A PERSPECTIVA DO FORNECEDOR DE TI

### Existem diferentes categorias de pessoas envolvidas na oferta de TI, com lógicas, motivações e expectativas distintas

A categoria ‘desenvolvedor de software livre/código aberto’ abrange pessoas em condições distintas de trabalho, com natureza jurídica variada e motivações diversas para o desenvolvimento. Por um lado, existem pessoas físicas que participam voluntariamente da comunidade livre ou trabalham para uma empresa ou organização, sendo remuneradas pelo serviço de desenvolvimento de software livre/código aberto. Por outro, pessoas jurídicas de natureza privada que surgem com a finalidade de desenvolver software livre/código aberto para oferta no mercado. Existem, ainda, pessoas jurídicas de natureza pública que mantêm desenvolvedores de software livre/código aberto *in house* ou contratam terceiros para o desenvolvimento.

Como visto no Capítulo 2, as motivações dos desenvolvedores para participar das comunidades de software livre/código aberto estão fortemente relacionadas com as oportunidades de aprendizado e de participação no cenário livre. O número de horas dedicadas ao desenvolvimento de software livre/código aberto é reduzido e são poucos os que têm em mente ganhos financeiros. Apesar disso, parte significativa obtém remuneração direta proveniente de software livre/código aberto, embora não, em sua maioria, exercendo atividades de desenvolvimento.

Em virtude das liberdades fornecidas pelas licenças de software livre/código aberto, as linhas de código materializadas em um produto encontram-se disponíveis para acesso dos interessados, reduzindo, portanto, as expectativas de ganhos vinculadas diretamente aos esforços de desenvolvimento do produto. Dentro deste contexto, as atividades de programação realizam-se através dos seguintes mecanismos: 1- ações voluntárias, que acontecem fora do expediente do trabalho, com motivações relacionadas, no geral, com o desejo de aprender e participar; 2- ações patrocinadas por terceiros que, por motivos diversos, remuneram profissionais para que se dediquem ao desenvolvimento em tempo parcial ou integral; 3- ações de desenvolvimento com vistas a uma apropriação futura, na forma de serviços ou através da comercialização de dispositivos com software embarcado.

### Desenvolvimento patrocinado por grandes corporações e instituições públicas

#### Nos países centrais, grandes empresas do setor de TI, em alianças e consórcios, mantiveram organizações voltadas para o desenvolvimento de software livre/código aberto.

Nos países centrais, grandes empresas do setor de TI, em geral através de consórcios tais como o OSDL, fornecem as condições para a existência de um quadro permanente de profissionais para o desenvolvimento de código aberto. O perfil das consorciadas variou ao longo do tempo. No início, parte significativa era constituída por fabricantes de computador e empresas de desenvolvimento de software. Aos poucos, o grupo passou a contar com fabricantes de equipamentos de comunicação, usuários e empresas ofertantes de serviços digitais.

Embora todos esses atores buscassem soluções mais além do monopólio instaurado pela Microsoft/Windows, os interesses variavam a partir deste tema em comum. Os fabricantes de equipamentos queriam

opções de software para embarcar nos seus equipamentos ou dispositivos móveis, sem a necessidade de pagar *royalties* por elas; as empresas de software pretendiam apoio para inovação, usando modelo aberto como mecanismo para reduzir custos em cenário de incertezas e transformações tecnológicas rápidas. As empresas de serviços digitais planejavam utilizar software livre/código aberto como plataforma a partir da qual poderiam gerar serviços de informação para os seus clientes.

Todos esses grupos tentavam alguma forma de apropriação, seja através da comercialização do hardware, seja através da combinação de software de código aberto com software proprietário, seja por meio da prestação de serviços facilitados por software. Assim, as ações consorciadas reduziam as despesas individuais relacionadas com o desenvolvimento de software livre/código aberto ao mesmo tempo que permitiam a criação de propostas individuais de ganhos financeiros.

**No Brasil, as instituições públicas tiveram um papel relevante na manutenção de desenvolvedores de software livre/código aberto.**

Na ausência de um consórcio relevante de empresas, no Brasil, coube especialmente às instituições públicas a manutenção de pessoas voltadas, em tempo integral, para o desenvolvimento de software livre/código aberto. No geral, os órgãos públicos assumiram um modelo híbrido baseado na manutenção de equipes *in house* reforçadas por serviços de terceiros. Comunidades livres e empresas com perfil de fábrica de software passaram a gravitar em torno de projetos em software livre/código aberto propostos pelo governo. Na condição de contratante, coube aos órgãos públicos o papel de conceber e especificar os projetos de interesse. Além de manter quadro próprio de desenvolvedores, as organizações/empresas públicas com atividades em TI, registradas com diferentes naturezas jurídicas, contavam, também, internamente, com vários outros profissionais de TI, incluindo gestores e administradores de projetos de software e profissionais para suporte técnico.

**Nos países centrais: apoiadoras eram grandes empresas de hardware, comunicação e software motivadas e com poder para questionar a hegemonia da líder.**

Existem diferenças na forma como se dá o apoio realizado pelas grandes empresas dos países centrais aos desenvolvedores de software livre/código aberto e o modo com que o governo brasileiro apoiou/vem apoiando estes profissionais. Através de ação consorciada, as grandes empresas de software proprietário colocaram recursos em uma organização que manteve profissionais atuantes em torno de uma plataforma tecnológica em desenvolvimento que surgia como um forte concorrente potencial ao monopólio da Microsoft/Windows. Através do consórcio OSDL, por exemplo, grandes empresas estimularam o surgimento de uma comunidade forte em torno do Linux. Havia uma clara preferência pelo software de código aberto, sem o peso marcante dos ideais de liberdade, tal como divulgados pela FSF. Do trabalho coletivo das comunidades, poderiam sair ideias e estruturas básicas a serem apropriadas de modo individual, futuramente, especialmente através da combinação da infraestrutura e *middleware* livre e/ou de código aberto com software proprietário, nos módulos complementares e nos aplicativos.

**No Brasil: no vácuo deixado pela ausência de grandes empresas, o governo protagonizou as principais iniciativas em software livre/código aberto.**

O Governo brasileiro, pelo seu lado, apoiou serviços de terceiros como parte de uma estratégia em que as

empresas públicas de TI eram os principais sustentáculos do avanço do software livre/código aberto. Parte dos recursos foi canalizada para fortalecimento de um quadro interno de profissionais visando ao desenvolvimento *in house*. Havia um peso marcante na ideia de software como bem público, com um discurso mais próximo dos ideais de software livre que das crenças do código aberto. Como as ações foram protagonizadas pelo governo e centralizadas nele, alterações eventuais de ordem política tinham probabilidade alta de afetar, prejudicando ou beneficiando, as medidas governamentais favoráveis ao software livre.

***Empresas brasileiras de TI mantiveram-se gravitando em torno da plataforma tecnológica Microsoft Windows.***

As empresas nacionais de TI, em geral de pequeno e médio porte, ao contrário das grandes empresas de TI dos países centrais, não viam no monopólio Windows Microsoft uma ameaça para os seus negócios. Direcionadas para o desenvolvimento de aplicativos, gravitavam em torno do Windows como plataforma, de onde surgiam as melhores oportunidades de negócios, especialmente voltadas para clientes privados, já que os clientes públicos, em grande medida, eram atendidos por empresas ou departamentos de TI do próprio governo.

O posicionamento das empresas de software brasileiras no *downstream* da cadeia de valor do software, ou seja, no desenvolvimento de aplicativos horizontais e verticais, possivelmente contribuiu para que estas empresas não desenvolvessem uma visão contrária à da plataforma líder. O perfil característico das brasileiras não era o de um concorrente em potencial da líder, mas o de seguidor. Assim, como percebeu um dos entrevistados, “o empresário brasileiro raramente combate a Microsoft. Além disso, as pessoas que defendem o software livre o fazem por razões ideológicas e não mercadológicas, um comportamento bastante distinto daquele verificado na Europa” (Quadro 3.6).

**QUADRO 3.6 – DIFERENÇAS DE PONTO DE VISTA EUROPEU E BRASILEIRO NO QUE CONCERNE À ADOÇÃO DO SOFTWARE LIVRE**

“Existe um movimento mundial para a padronização em torno de padrões abertos. A OW2 é um consórcio europeu de software livre formado por empresas privadas. Seu objetivo é combater o monopólio ou oligopólio gerado pelo software proprietário e a iniciativa conta com uma forte participação governamental.

No Brasil é o contrário. O empresário brasileiro raramente combate a Microsoft. As pessoas que defendem o software livre o fazem por razões ideológicas e não mercadológicas. Os adeptos do software livre o fazem por ideologia sem considerar a tendência do mercado.”

[Extraído de depoimento de um integrante da comunidade livre/código aberto brasileira – setembro 2014]

***Forte presença do governo, sem um anteparo mais pragmático das empresas, exacerbou o discurso político-ideológico, fornecendo contornos próprios para o movimento brasileiro, mais focado nos ideais do software livre que nos preceitos do código aberto***

No Brasil, portanto, o combate ao software proprietário acabou sendo muito fortemente concentrado e promovido pelo governo. Essa forte presença do governo, sem um anteparo mais pragmático da iniciativa privada, deve ter ajudado a tornar o discurso sobre software livre/código aberto, no Brasil, mais vinculado

a questões ideológicas, envolvendo a percepção do software como bem comum, explicando a força maior que o software livre tem, no país, em relação ao código aberto.

Vários autores (ver, por exemplo, Shaw, 2011; Evangelista, 2014; e Murillo, 2009) chamaram a atenção para a forte orientação política que assumiu o movimento de software livre no país, o que lhe garantiu características únicas, quando comparado a outras experiências internacionais. Como sustenta Evangelista (2014) “o discurso, em especial do grupo *free*, ganhou coloração própria quando reinterpretado por militantes brasileiros. A ideia de cooperação, colaboração, solidariedade e construção de um conjunto de softwares que fosse uma alternativa para o enrijecimento das regras de propriedade intelectual ganhou outra força ao aportar em um país subdesenvolvido de industrialização parcial. Técnicos, muitos deles ligados ao serviço público e com passado nos movimentos de esquerda, entenderam o movimento do software livre também como uma resposta ao domínio das grandes empresas de informática e ao saque de riquezas promovido pelos países desenvolvidos. No horizonte, enxergou-se o software livre até como fator de transformação e superação da economia capitalista” (Quadro 3.7).

No entanto, Evangelista registra a mudança pela qual o movimento vem passando mais recentemente. Segundo ele, “a entrada mais forte das corporações *open* impacta o software livre enquanto movimento, pois traz para ele militantes/colaboradores de perfil diferenciado, mais interessados no software livre pelas perspectivas profissionais que lhes são oferecidas. O perfil da base do movimento está em transformação, agora abarcando jovens interessados em carreiras mais tradicionais nas grandes empresas, carreiras semelhantes às dos executivos tradicionais das empresas transnacionais. Essa transformação pode ser acompanhada, no contexto brasileiro, em eventos como o Fórum Internacional de Software Livre (Fisl)”. Seja como for, no movimento brasileiro favorável ao software livre/código aberto, as motivações de natureza ideológica ainda são fortes.

#### QUADRO 3.7 - O SURGIMENTO DO FISL: ENTRE MOVIMENTOS SOCIAIS E PARTIDOS DE ESQUERDA

“O Fórum Internacional de Software Livre (Fisl) é um evento que reúne, desde o ano 2000, grande parte do que se convencionou chamar de “comunidade software livre brasileira”. Dessa comunidade faz parte uma gama complexa de indivíduos que qualificam a si mesmos principalmente de acordo com suas ocupações: desenvolvedores (que desenvolvem, modificam os softwares), programadores (que oferecem instruções para que os softwares funcionem), usuários, funcionários de governo, políticos, estudantes de computação, jornalistas, ativistas sociais, empresários, etc. Ao longo do tempo, essa comunidade cresceu, superando principalmente o limite do conhecimento técnico, envolvendo cada vez mais usuários de nível intermediário e simpatizantes de algumas das ideias gerais do software livre. Concomitantemente, o movimento software livre também cresceu internamente, conquistando progressivamente a simpatia e/ou interesse de profissionais e estudiosos da computação. Um olhar sobre o Fisl é também um olhar consistente sobre o movimento brasileiro.

Surgido entre funcionários públicos de tecnologia, ligados a sindicatos e movimentos de esquerda, o evento buscou sua base de público entre estudantes e profissionais da computação. Esses profissionais e estudantes convivem, geralmente, com empresas de todos os tamanhos, de onde retiram seu sustento (como empregados ou patrões) e que costumam estar presentes em eventos da área. O que em um primeiro momento pode parecer contraditório (a conjunção entre setores em certa medida anticapitalistas e o espaço para as empresas) faz sentido dado o perfil dos setores mobilizados, refletindo conjuntamente o ambiente de eventos para estudantes, militantes políticos e empresários/trabalhadores.

Aaron Shaw (2011) oferece uma visão interessante sobre alguns dos personagens que construíram as fundações do movimento software livre brasileiro e que participaram ativamente dos primeiros anos da organização do Fisl. Segundo Shaw, parte deles compartilhava uma história nos movimentos de esquerda do país e, quando o governo Lula atingiu o poder, levaram à frente um discurso radical, buscando politizar o papel do Estado desenvolvimentista em uma

economia do conhecimento. Os membros do movimento software livre brasileiro possuíam características únicas, se comparados a seus pares internacionais. A principal delas seria a orientação política, uma mistura de neomarxismo com socialismo.

Um dos indivíduos entrevistados por Shaw e que contribuem para que ele forme essa percepção sobre o movimento brasileiro é Mario Teza, bastante ativo na organização do FISL até hoje. [...] um depoimento de Teza colhido por Shaw (2011, p. 259) é emblemático do significado que parte dos organizadores históricos do Fisl dão ao software livre, mostrando que, pelo menos para alguns deles, o software livre significava uma possível “transcendência do capitalismo” e um meio para superar as limitações naturais das lutas sindicais.

A partir de 1999, quando o PT chega ao governo do estado [do Rio Grande do Sul], os simpatizantes do movimento de software livre fortalecem ligações entre o PT, os sindicatos de Porto Alegre, as empresas estatais, os movimentos sociais e setores interessados em informática, a partir de certas ideias do software livre.

Um exame das páginas publicadas na Internet pelos organizadores do Fisl confirma que a ideia do software livre como fator de mudança social já estava presente nos primeiros anos do evento. Tenta-se combinar o mundo dos negócios com objetivos de transformação da estrutura da economia. Uma das preocupações dos organizadores era impulsionar os negócios das empresas de software livre, vistas como portadoras, em si, de um modelo econômico alternativo.

A importância de se estimular os negócios com as empresas de software livre é uma preocupação que persiste nas diversas edições do Fisl, estando ligada tanto à ideia de que isso levaria mais pessoas a “viverem de software livre” – “libertando-se” do “mundo do software proprietário” como à noção de que não é saudável ao “ecossistema do software livre” estar excessivamente ligado a iniciativas estatais. Desde os primeiros anos, nota-se a importância da estrutura estatal para a promoção das ideias do software livre, exemplificada pela clara interconexão entre o Projeto Software Livre-RS, organizador do Fisl, com o governo do estado do Rio Grande do Sul. As páginas, tanto do Fisl em suas primeiras edições, como do PSL-RS, funcionavam em um domínio “rs.gov.br”, ou seja, estava endereçada em um registro que pertence exclusivamente à administração estadual. Essa forte influência do governo estadual e municipal no evento foi substituída, mais tarde, quando da saída do PT do governo gaúcho e porto-alegrense, em forte influência do governo federal, a partir do governo Lula. Após 2003, o governo federal passou a contribuir mais consistentemente com o evento, oferecendo os patrocínios básicos que garantiram a sua realização em condições mínimas. Além disso, funcionários públicos, ligados neste segundo momento ao governo federal, continuaram colaborando com a organização. Contudo, não se trata necessariamente dos mesmos indivíduos, e estes estão menos ligados à estrutura interna de organização do Fisl (até por não estarem no Rio Grande do Sul) do que os colaboradores iniciais. Em 2003, a organização do Fisl tornou-se autônoma do PSL-RS, fundando uma ONG regularmente formalizada (a ASL.org) para gerir a organização do evento. Ao mesmo tempo, cresceu e diversificou-se o patrocínio oferecido pelas empresas privadas.”

[Extraído, com alterações de Evangelista, R., 2014]

**Ações governamentais envolvendo o desenvolvimento do software livre tenderam a ser realizadas in house, desvinculadas das comunidades. Estas passaram a gravitar em torno de projetos internacionais, descolados da realidade brasileira.**

No Brasil, as oportunidades de programação em software livre/código aberto surgiam no âmbito de ações voluntárias nas comunidades livres, em empregos nas instituições públicas e em empresas que prestavam serviços de desenvolvimento especialmente para o Governo. Abriram-se, também, oportunidades de trabalho em projetos suportados por organizações internacionais. Algumas empresas brasileiras com atividades de serviços de desenvolvimento de software livre, por exemplo, são remuneradas por projetos externos. O sócio de uma dessas empresas lamenta que parte importante dos talentos brasileiros acabe se envolvendo em projetos que não beneficiam, diretamente, o país. Como diz:

“As grandes empresas costumam contribuir com projetos de software livre em andamento, auxiliando para o fortalecimento das comunidades. Elas investem na comunidade livre, remunerando pessoas nos projetos. Essa é uma forma que encontraram para avançar mais rapidamente na obtenção de soluções inovadoras. Elas não buscam inovação internamente, não criam soluções

internas, mas mantêm uma estratégia de inovação aberta: usando os achados livres como referência para, posteriormente, desenvolver atividades internas. A IBM, por exemplo, é uma empresa que adota este modelo. Facebook é um outro exemplo. Toda a infraestrutura deles é livre.

Nós formamos pessoas que acabam sendo utilizadas fora do Brasil, acabamos perdendo os nossos bons profissionais, porque as nossas empresas não entendem o que está acontecendo nas comunidades e não sabem como se beneficiar da interação com as comunidades livres. Os bons profissionais hoje não estão nas nossas corporações: os profissionais acabam trabalhando para empresas estrangeiras.”

**No modelo de software livre/código aberto, os relacionamentos contínuos com a comunidade devem ocorrer. O modelo adotado pelo governo, baseado no desenvolvimento por equipes in house, acabou por isolar a equipe**

O modelo de negócios privilegiado pelo governo, muito baseado na manutenção de equipes internas de desenvolvimento coloca risco para a continuidade do software livre/código aberto, podendo levar as equipes ao isolamento, caso não haja uma interação forte com as comunidades livres. Como lembra um entrevistado:

“Caso a interação com a rede não seja mantida, a empresa ou profissional fica isolado e não consegue evoluir. Ou seja, não se trata apenas de colocar à disposição o código na Internet, em um repositório. A gestão das relações durante o processo de especificação do produto e desenvolvimento é necessária e de alta complexidade. Isso inclui criar metodologia, mobilizar as comunidades e resolver conflitos. É necessário, portanto, trabalhar a gestão do desenvolvimento do software e a gestão da orquestração das equipes. E isso, certamente, é um custo a ser pago pela empresa/instituição interessada em atuar com software livre/código aberto” (Quadro 3.8).”

### QUADRO 3.8 – ESPECIFICIDADES DOS NEGÓCIOS EM SOFTWARE LIVRE/CÓDIGO ABERTO

A grande oportunidade do software livre/código aberto tem a ver com o fato de ser possível pegar uma solução global e trabalhar esta solução para nível local. É o caso, por exemplo, do Word Press. Qualquer empresa hoje pode se tornar uma prestadora de serviços usando o software. Mas, para isso, é necessário interagir com a comunidade internacional. Esse é um ponto relevante a destacar. Não se trata apenas de pegar uma solução estrangeira e começar a utilizá-la. É fundamental manter o relacionamento com a comunidade. Retornar para a comunidade as mudanças e transformações realizadas. Caso esta interação com a rede não seja mantida, a empresa ou profissional fica isolado e não consegue evoluir. Pela mesma razão, as melhorias realizadas em nível local podem ser utilizadas por todos. Mas não se trata apenas de colocar à disposição o código na Internet, em um repositório. A gestão das relações é necessária e de alta complexidade e deve iniciar no começo do processo de concepção do produto.

Isso inclui desenvolver metodologia, mobilizar, resolver conflitos. Toda essa gestão precisa fazer parte do custo da empresa de software livre/código aberto. Há necessidade de desenvolver métricas próprias para precificar: desenvolvimento, gestão, incorporação da história. Isso acaba significando algo em torno de ¼ do custo da empresa que trabalha com software livre. No entanto, é 1/1.000 do custo de uma empresa que inicia o desenvolvimento do zero. Trabalhar em rede fica mais barato, mais viável também. Há reduções do custo de aprendizagem e de desenvolvimento.

O Serpro ainda está buscando *expertise*. No passado, a instituição deixou de interagir com a comunidade. Isso foi um equívoco que levou a consequências ruins. Pegaram uma solução, deixaram de interagir com a comunidade e, agora, precisam retornar à comunidade.

[...] Nas comunidades livres, é mais comumente aceito o *home office* e mais adotado o modelo de desenvolvimento ágil e colaborativo, amparado em redes de desenvolvimento baseadas nas pessoas. Trata-se de um arranjo organizacional aberto em ‘pares’.

A minha empresa utiliza método de desenvolvimento baseado em uma mescla de *scrum* adaptado para ambiente aberto, modelo colaborativo e modelo bazar. No cronograma proposto para o cliente, é necessário considerar o uso dessa metodologia. Existe a necessidade de orquestrar de modo adequado todos os colaboradores, lembrando que eles não estão, necessariamente, no mesmo local físico. Como chegar no Japão? Cria software aberto, busca por equipe que

está sediada na Universidade de Chicago. Eles traduzem e preparam para o ambiente do Japão. Trata-se, com certeza, de uma orquestração diferenciada, embora baseada, também, na gestão conhecida de projetos. A comunidade ajuda a buscar clientes, mapear o mercado. É necessário, portanto, trabalhar a gestão de desenvolvimento e a gestão de orquestração.

[Extraído de depoimento de sócio de empresa que presta serviços de desenvolvimento de software livre/código aberto]

## Profissionais das equipes de desenvolvimento

Desde a perspectiva da equipe interna de desenvolvimento e/ou de suporte, ou seja, desde a perspectiva do profissional assalariado, existem vantagens claras em desenvolver software livre/código aberto. Muitas pessoas inspecionando o software podem, de fato, torná-lo mais adequado, mais rapidamente. Há vantagens, também, quando existem incertezas: nada melhor do que compartilhar os riscos ou os desafios de desenvolvimento com outros.

As razões relacionadas com a formação referem-se às possibilidades de aprendizagem colaborativa (e, por isto, ampliada) e de um desfile de competências frente a um universo relativamente grande de observadores (efeito vitrine). Possibilitam, ainda, ampliar as condições de empregabilidade dos desenvolvedores, exatamente por esta exposição permanente (em rede de competências).

Existem ganhos econômicos relacionados com as economias de rede, ao poder desenvolver um novo produto com a cooperação de um número elevado de desenvolvedores e desta forma, em tempo inferior. No entanto, as empresas desenvolvedoras fornecedoras de software livre/código aberto poderiam se beneficiar dos efeitos positivos de externalidades de redes mais que os empregados de empresas usuárias.

## Oferta de serviços de TI

**Tendência é a oferta de serviços de suporte e/ou consultoria para softwares genéricos, permitindo ganhos maiores de escala, e complexos, necessitando instalação, capacitação e suporte técnico**

Fora da influência da esfera pública, um modo de os profissionais de TI obterem ganhos com software livre/código aberto ocorre através da oferta de serviços de suporte e/ou consultoria. Para gerar negócio sustentável, o software precisa ser relativamente complexo, requerendo customização (incluindo horas de desenvolvimento complementares), serviços de instalação ou capacitação para o uso. Soluções que não exigem customização, comercializadas em pacotes que o próprio cliente instala, não permitem a sobrevivência do prestador de serviços de consultoria e suporte.

Além da complexidade, o software tem que ser genérico o bastante para ser utilizado por muitos, aumentando as chances, pelo menos em teoria, de se conseguir clientes. Provavelmente por estes motivos, a plataforma Linux, as ferramentas para gestão de bancos de dados e as aplicações de uso genérico tornaram-se uma das principais oportunidades de geração de negócios para consultores no mercado de serviços envolvendo sistemas livres. Esta, também, foi a porta de entrada do software livre/código aberto no

mercado de software em geral, com Linux tornando-se forte inicialmente e especialmente no mercado de servidores. Assim, um primeiro conjunto de oportunidades para oferta de serviços de TI envolveu serviços de implantação do Linux e outros módulos genéricos. Isso ocorreu no país, mas está longe de ser uma peculiaridade nacional.

Algumas empresas com modelos livres conseguiram brechas no mercado de aplicativos, explorando usuários finais com um perfil diferenciado: conhecimento maior sobre os atributos técnicos ou funcionais de um software, uso de Linux nas máquinas ou espírito mais motivado a mudanças. Nesta categoria encontram-se, sobretudo, os profissionais vinculados às instituições de ensino e pesquisa, especialmente, instituições públicas. Como diz um professor e empresário, com produto voltado para o apoio a pesquisas acadêmicas:

*“Não existe grande penetração do software livre no mercado brasileiro. O uso é praticamente nulo. As empresas, em geral, usam Microsoft, Cisco, etc. As grandes só confiam nas grandes em virtude dos interesses envolvidos. No caso de problemas com o produto, o cliente quer ter a quem acionar. Com software livre, fica difícil responsabilizar alguém. Para os grandes clientes a aquisição da licença não é um problema porque podem comprar.*

*A nossa escolha por desenvolvimento de software livre se deve à influência do meio acadêmico e, também, a análise do custo/benefício, que se mostrou bastante vantajosa. Na universidade pública, existe maior aderência ao software livre. Nossa empresa consegue atender muito bem os seus clientes da academia desenvolvendo soluções em software livre.”*

### **Alternativa de negócios seria a oferta de serviços do tipo ITO e BPO, que são facilitados por software**

Como alternativa de negócios, surge, também, a possibilidade de oferta de serviços intensivos em TI ou facilitados pela TI, do tipo ITO (*Information Technology Outsourcing*) ou BPO (*Business Process Outsourcing*), em que o software surge como ferramenta para a prestação dos serviços. Para o cliente final, é indiferente se a ferramenta em uso é ou não um software livre. Dentro desta categoria, podem ser mencionados, como exemplos, serviços de gestão do parque computacional de uma empresa, desenvolvimento de páginas na *web*, serviços de auditoria florestal facilitada por software ou recrutamento de recursos humanos. Utilizando software livre/código aberto, o prestador de serviços poupa tempo e recursos no desenvolvimento do software a ser utilizado como suporte às atividades de ITO ou BPO. O modelo pode servir para gerar negócios para consumidores finais e empresas de diferentes portes, pois não requer que o consumidor final possua habilidades para uso de software livre ou disponha de sistema operacional livre instalado.

### **Software livre embarcado em equipamentos e dispositivos: também neste caso uma alternativa possível de negócios**

O software livre embarcado em equipamentos e dispositivos diversos também se torna uma forma muito atraente de uso do software livre, pois resolve, tanto como ITO e BPO, a equação entre a motivação dos desenvolvedores para o uso, por um lado, e o desinteresse do usuário final pela adoção, por outro. Um exemplo está nas empresas que produzem equipamentos eletroeletrônicos e de comunicação, principalmente roteadores e *modems* com software livre/código aberto embarcado. Ainda que este não seja o negócio principal das fabricantes de hardware, estas desenvolvem e adaptam software livre/código aberto internamente. Pesquisa da Softex realizada em 2005 verificou que empresas de hardware estavam customizando versões de suas soluções proprietárias para correr em Linux, especialmente para aplicações de equipamentos para o varejo (PDV), mas, também, em outros equipamentos dedicados.

## CONSIDERAÇÕES FINAIS

No Brasil, no início dos anos 2000, o sistema operacional Linux encontrava-se bem posicionado no mercado de servidores para o ambiente corporativo, especialmente em empresas de médio e grande porte. O uso do software livre/código aberto estava, ainda, muito concentrado na infraestrutura das empresas. Mas se acreditava, na época, que a conquista do mercado de PCs era apenas uma questão de tempo. O governo brasileiro apareceria como um usuário importante de software livre/código aberto.

Passada uma década, o uso significativo de software livre/código aberto na administração pública se confirmava. Em 2013, quase 70% dos órgãos públicos federais e estaduais utilizavam sistema operacional livre. Mas a grande maioria (98%) adotava, também, o sistema Windows.

Windows também era o sistema predominante nas escolas brasileiras de ensino fundamental e médio, rondando em 87% delas. Linux apresentava uma participação inferior, estando presente em 57% das escolas públicas, mas em apenas 12% das escolas particulares.

A utilização de sistemas livres nas empresas era baixa em 2013. O uso restringia-se a 24% delas, embora a presença fosse superior, atingindo 67%, no conjunto de empresas com mais de 250 pessoas ocupadas.

A promessa de chegada dos sistemas livres nos microcomputadores não se confirmou. Também no Brasil, funcionou a dinâmica inerente à indústria de software, que premia a plataforma tecnológica líder com fatias relevantes do mercado.

Ao se observar o perfil de atores distribuídos pelos diferentes elos da cadeia de valor do software, percebe-se que, no Brasil, as empresas de capital nacional, em geral de pequeno porte, tendem a se concentrar no *downstream*, ofertando aplicativos em mercados que permitem ganhos mais baixos de escala. Grandes empresas, comandadas pelos países centrais e operando como empresas-redes, controlam os elos a montante da cadeia de valor do software (softwares de infraestrutura e *middleware*), complementando a sua oferta com aplicativos, especialmente em segmentos que permitem ganhos maiores de escala e necessidades menores de customização/localização do software. Nenhuma empresa de capital nacional conseguiu reunir as forças necessárias para competir minimamente com a plataforma líder de mercado. Na realidade, parte relevante das nacionais tornou-se parceira da líder, assumindo riscos menores.

No Brasil, a expectativa existente no início dos anos 2000 de o software livre/código aberto continuar ganhando espaço e expandir a sua participação no segmento de microcomputadores não se realizou, pelo menos não na dimensão desejada. A dinâmica da indústria de software, baseada na capacidade de a plataforma líder consolidada criar barreiras fortes à entrada de concorrentes, explica a dificuldade, até certo ponto. No entanto, ressalta-se, também, o desafio maior que é a consolidação de uma plataforma tecnológica no ambiente livre, dadas as várias permissões concedidas ao usuário do software.

Existem diferentes perfis de usuários e os seus cálculos de custo-benefício envolvendo decisões sobre aquisição ou migração de software não são coincidentes. Sobre a demanda por software e serviços de TI proveniente de empresas e organizações, pode-se dizer que:

- A adoção de software livre/código aberto nos computadores pessoais foi dificultada pela presença de externalidades de redes e efeitos de *lock-in* provenientes da consolidação da plataforma líder, Windows Microsoft, criando resistências a mudanças por parte dos usuários finais.
- Os ganhos baixos de escala oriundos da oferta de serviços de suporte a softwares do tipo aplicativo devem ter contribuído para prejudicar o uso de software livre/código aberto nos PCs, pois inibiram o surgimento de uma rede sustentável de prestadores de serviços de TI voltada para suporte às aplicações livres.
- Parece mais fácil substituir softwares de infraestrutura e *middleware*, que estão no controle dos profissionais de TI, que softwares aplicativos, em uso por profissionais de perfil não técnico.
- A substituição de software em mãos dos trabalhadores terá de ser acompanhada por atividade relevante e persistente de persuasão, capacitação e evangelização. Essa necessidade ocorre mesmo em organizações em que existe um forte senso de hierarquia.
- Considerando as oportunidades e desafios para adoção do software livre/código aberto, conclui-se que, para a empresa/organização demandante, desde a perspectiva de um cálculo racional, o mais vantajoso seria a migração ou implementação de software livre na retaguarda (ou seja, na infraestrutura de TI), mantendo nas máquinas acessadas pelos trabalhadores aplicativos rodando em software já amplamente difundido e aceito.
- Empresas de grande porte e, especialmente, o Governo teriam mais motivos para optar por software livre/código aberto que pequenas e médias empresas e organizações.
- Grandes empresas e órgãos públicos adotam software livre/código aberto por motivos distintos. No primeiro caso, a motivação diz respeito à redução dos custos elevados de aquisição de licenças de uso. A migração é facilitada pela presença de equipe interna de TI, que pode ficar total ou parcialmente responsável pelas atividades de implantação, capacitação e suporte técnico aos usuários finais. Nos órgãos públicos, as questões econômicas também têm peso, mas se destacam, sobretudo, alguns atributos técnicos críticos do software livre/código aberto, tal como a auditabilidade, e aspectos ideológicos relacionados com o objetivo do bem comum e da inclusão social.
- No ambiente dos computadores pessoais, o modelo de software livre/código aberto ameaça, sobretudo, o software proprietário de infraestrutura e *middleware*, pois são ferramentas genéricas, sem grandes atributos funcionais diferenciadores e estão nas mãos dos desenvolvedores. O modelo tem mais dificuldades no segmento de aplicativos horizontais e verticais, direcionados para os usuários finais, pois as externalidades de redes e os efeitos de *lock-in* tendem a gerar resistências fortes às mudanças e os usuários finais têm menos motivação para a migração de plataforma. Além disso, empresas/desenvolvedores de software livre/código aberto irão preferir se voltar para negócios mais rentáveis e desafiadores, desde o ponto de vista técnico, que o mercado de desenvolvimento e suporte a aplicativos, com ganhos inferiores de escala, ênfase maior no conhecimento do negócio e menor na tecnologia.

Da mesma forma, existem diferentes categorias de pessoas envolvidas na oferta de TI. As suas motivações e expectativas são distintas. Assim:

- Nos países centrais, grandes empresas do setor de TI, em alianças e consórcios, mantiveram organizações voltadas para o desenvolvimento de software livre/código aberto. Os apoiadores eram grandes empresas de hardware, comunicação e software motivados e com poder suficiente para questionar a hegemonia da líder. Em vários países europeus, os governos apoiaram as iniciativas do setor corporativo.
- No Brasil, na ausência de empresas com calibre suficiente e capacidade de articulação adequada, o governo trouxe para si o papel de incentivar o uso do software livre/código aberto. O modelo adotado privilegiou o desenvolvimento *in house*, deixando pouco espaço para o surgimento de um ecossistema pujante constituído por empresas ou sob o comando das comunidades. A equipe interna de TI das instituições e empresas públicas manteve-se isolada das comunidades no país e no mundo, com as interações acontecendo de modo intermitente. Com isso, as comunidades brasileiras livres passaram a gravitar em torno de projetos internacionais, descolando-se da realidade brasileira. Em algumas situações, passaram a atuar como fábricas de software para projetos concebidos e especificados pelo governo.
- O forte protagonismo do governo sem um anteparo mais pragmático das empresas/comunidades exacerbou o discurso político-ideológico, fornecendo contornos próprios para o movimento brasileiro de software livre/código aberto. Também se manteve mais fiel aos ideais do software livre que aos preceitos do código aberto.

# CAPÍTULO 4

## SOFTWARE LIVRE/CÓDIGO ABERTO: DECIFRANDO O PRESENTE E FAZENDO PROGNÓSTICOS PARA O FUTURO

### APRESENTAÇÃO

Estudo da McKinsey (Bughin *et alii*, 2013) e de outras empresas de consultoria apontam as principais tendências envolvendo TIC (tecnologias da informação e comunicação) e negócios para os próximos anos. Entre elas, destacam-se o nexos das forças citado por Gartner - mobilidade, *big data*, redes sociais e computação em nuvem – mas são mencionadas, também, outras várias oportunidades disruptivas: Internet das Coisas (IoT), máquinas inteligentes, impressoras 3D, etc. Todas essas novas tecnologias, especialmente considerando a sua atuação conjunta, têm um enorme potencial para abalar a supremacia das plataformas tecnológicas vigentes, modificando de modo drástico a forma como se desenvolve e se comercializa software e o jeito como as pessoas trabalham e vivem. Elas, de fato, estão permitindo o surgimento de um novo paradigma, que traz muitas oportunidades, mas também gera incertezas para as empresas do setor de TIC.

Na Seção 4.1, apresentam-se as tecnologias e tendências de negócios que estão transformando a realidade muito rapidamente para, na Seção 4.2, tecer considerações sobre o impacto que vêm provocando e como irão afetar a indústria brasileira de software e serviços de TI, cujas empresas, na sua maioria, encontram-se atreladas às plataformas tecnológicas líderes, comandadas por empresas-redes com sede nos países centrais. As oportunidades para o crescimento de uma indústria brasileira de software livre/código aberto, antenada com o novo paradigma tecnológico em construção, são exploradas na Seção 4.3. Finalmente, na Seção 4.4, é realizada análise de pontos fortes e fracos, oportunidades e ameaças para a indústria brasileira de software livre/código aberto.

### 4.1 NOVO PARADIGMA SUPOSTO POR NOVAS TECNOLOGIAS E TENDÊNCIAS DE NEGÓCIOS

*Novo paradigma recoloca a questão da hegemonia. Abrem-se brechas para a entrada de novos atores e o posicionamento das antigas lideranças.*

A chegada de um novo paradigma desestrutura os arranjos empresariais existentes no setor de TIC, levando os líderes a um novo cenário de disputas pela hegemonia, em que nenhuma empresa ou grupo possui ainda a supremacia. Abrem-se brechas para a entrada de novos atores, para um reposicionamento das antigas lideranças e para a celebração de parcerias inusitadas. A seguir, apresentam-se algumas das principais tecnologias e tendências de negócios.

## MOBILIDADE

### *O cenário de mobilidade estimula o surgimento de novas plataformas tecnológicas. Líderes disputam a supremacia.*

O cenário de mobilidade se tornou um novo jeito de vida, trazendo consequências variadas. Entre elas, destaca-se a priorização do uso de dispositivos portáteis, que podem ser carregados de um lado para o outro. O computador de mesa foi substituído pelos *notebooks* e *laptops* que, mais recentemente, passaram a conviver com *tablets* e celulares inteligentes. Além dos dispositivos terem se tornado mais leves, multiplicam-se nas mãos das pessoas, criando a necessidade de os dados e as informações poderem ser acessados nos vários meios possíveis. Soma-se, a isto, a importância cada vez maior da Internet, e um aumento considerável no tempo em que as pessoas permanecem conectadas.

A mobilidade afeta o ambiente de trabalho. Cada vez mais os empregados usam os seus próprios dispositivos (*notebooks*, *tablets*, *smartphones* e outras máquinas inteligentes que estão a caminho) e softwares na empresa. Essa é uma tendência que veio para ficar e que coloca um conjunto todo novo de desafios para o empregador, relacionados com proteção de dados e infraestrutura para suporte a hardwares e softwares diferentes e gestão organizacional baseada em valores que privilegiam a relação de confiança entre patrão e empregado.

As fronteiras da empresa tornam-se menos rígidas. Não só os trabalhadores estão levando os seus dispositivos para o trabalho como o trabalho também está sendo transferido para outros locais, pois cada vez há um número menor de motivos para que seja realizado em um local físico definido. O *home office* ganha força.

A revisão das fronteiras também ocorre em outros ambientes. Assim, por exemplo, nas escolas, as salas de aula e os laboratórios vão aos poucos perdendo a razão de ser. Os alunos agora podem e querem se reunir para troca de experiências e aprendizado em qualquer lugar físico ou virtual, em horário da sua conveniência.

Para o fornecedor de TI, existe o desafio de colocar produtos e serviços ao alcance dos clientes em qualquer lugar, a qualquer hora e, também, nos vários meios possíveis de acesso. Além disso, é necessário integrar dados e informações de forma que a última versão esteja disponível para acesso seja qual for o dispositivo em uso.

Várias empresas disputam a supremacia no ambiente móvel. O conjunto inclui aquelas que surgiram com o deslanchar da Internet, tais como Google, Amazon, Facebook, e empresas líderes nas antigas plataformas tecnológicas, tais como Microsoft e IBM.

## Bring Your Own Device e Bring Your Own App já são realidade

Equipamentos eletrônicos pessoais, tais como celulares e *tablets*, e *apps* baixadas a partir destes dispositivos são cada vez mais utilizados nas empresas, em práticas denominadas BYOD (*Bring Your Own Device*) e BYOA (*Bring Your Own Apps*).

O fenômeno BYOD/OA está crescendo rapidamente, permitindo liberdade, agilidade e produtividade. No entanto, traz uma série de novos questionamentos também. Por exemplo, como trabalhar com plataformas diferentes? Qual o propósito de implantação de políticas de BYOD: redução de custos, satisfação do empregado, expansão das oportunidades de mobilidade? As principais questões têm a ver, sobretudo, com segurança da informação e proteção de dados: como garanti-las em um cenário em que ocorrerá um crescimento significativo na liberdade de escolha dos usuários?

Informações adicionais sobre BYOD são fornecidas no Quadro 4.1.

### QUADRO 4.1 - BYOD

Segundo pesquisa da Associação Brasileira de *E-business* (*ebusiness* Brasil) (*apud in* Talarico, 2013):

- 39% das empresas brasileiras já permitem a prática de BYOD.
- Na avaliação das empresas pesquisadas, o procedimento aumenta a disponibilidade da equipe e reduz custos com aparelhos.
- 49% das empresas afirmam que adotam o BYOD, mas ainda não possuem a área de TI gerenciando o processo; 38% admitem o aumento na produtividade dos funcionários.
- Dentre os benefícios apontados pelos profissionais está, principalmente, a mobilidade, além da liberdade na decisão de escolha do dispositivo (44%) e a redução de custos com aparelhos (42%).

Existem vários benefícios para implementação de política BYOD. Trabalhadores se sentem mais satisfeitos usando dispositivos e aplicativos que gostam de usar e as organizações economizam recursos. O BYOD leva a um aumento do número de usuários de *apps* no local de trabalho, abrindo oportunidades de meios de comunicação além daqueles que vêm sendo tradicionalmente oferecidos pela empresa. O fenômeno irá expandir os acessos e direcionar as inovações. A TI precisará olhar mais além das ferramentas de gestão dos dispositivos móveis para abranger, também, a gestão das *apps* que as pessoas trarão para o ambiente de trabalho, já que muitas são projetadas para uso em dispositivos móveis (Gohring, 2013).

Até 2017, metade dos empregadores irá requerer que os seus empregados utilizem dispositivos móveis pessoais no trabalho. Segundo pesquisa do Gartner, em 2016, 38% dos empregadores esperam não ter mais de fornecer equipamentos para os seus funcionários. Mas quem irá pagar pelos dispositivos móveis e pelos serviços? De acordo com Gartner, metade das empresas com programas de BYOD reembolsa os empregados pelos custos do celular. Mas a tendência é que os subsídios diminuam e que os empregados passem a arcar totalmente com os custos ou que as empresas cubram apenas alguns custos relativos ao uso do aparelho.

Mesmo que o BYOD acarrete corte de gastos com equipamentos, ainda existem custos por parte das empresas referentes à gestão destes equipamentos e *apps*. Segundo dados do Gartner, as organizações gastam, em média, \$600 por empregado com o fornecimento de dispositivos móveis. Com a adoção de programas de BYOD, esse valor médio caiu para \$100. Mas a perspectiva é que suba para \$300 até 2016, principalmente em virtude das licenças com *apps* (*apud in* Gohring, 2013).

Conforme dados de Willis (2013), existe uma grande diferença entre os países no que diz respeito à adoção de práticas de BYOD. Até 2020, 15% das organizações não terão políticas implantadas a respeito; 40% terão implementado uma política mista e 45% trabalharão exclusivamente com BYOD. O *tablet* será o principal dispositivo levado pelo empregado para o local de trabalho.

**BYOD/OA irá afetar a dinâmica de poder entre o departamento de TI e os usuários. Extinguem-se as fronteiras entre personal IT e work IT. Modelo de controle centralizado pela TI é aos poucos substituído por modelo baseado em estruturas em rede.**

A adoção de BYOD/OA nas organizações irá afetar a dinâmica de poder entre o departamento de TI e os usuários. A tecnologia vem se tornando cada vez mais intuitiva e orientada para o usuário. Os usuários passam a controlar o seu uso, ganhando autossuficiência para escolher dispositivos, *apps* e métodos de trabalho e para trazer para dentro das empresas hábitos diários e sociais próprios. Isso pode levar a conflitos com o tipo de gestão cultivado e aperfeiçoado pela área de TI ao longo dos últimos anos.

Como lembra Taurion (2013), nos primórdios da computação, o acesso dos usuários aos sistemas era restrito. No início, realizava-se por meio de cartões perfurados e, posteriormente, por terminais acessados de dentro dos escritórios. Com a chegada da computação pessoal e dos PCs, com suas interfaces de janelas e *mouse*, os usuários passaram a ter mais intimidade com as máquinas, mas o controle do que poderia estar nos computadores das empresas continuava nas mãos do departamento de TI. Hoje, a disponibilidade de computadores de bolso como *smartphones* e *tablets* e de *apps stores* com centenas de milhares de *apps* popularizou a computação. Com estes dispositivos, a fronteira entre *personal IT* e *work IT* é extinta.

O suporte técnico e o *help desk* não são consultados para acesso a serviços como *Twitter*, *Facebook*, *DropBox*, *Waze* e outros mais. As ações envolvendo o uso dos dispositivos móveis e de *apps* acontecem fora do modelo de gestão do departamento de TI. Pelas mídias sociais, a sociedade se mobiliza e se autoajuda na resolução das dificuldades. Essa mudança tem impacto no papel do suporte técnico e do *help desk*. O modelo tradicional de controle centralizado pela TI é aos poucos substituído por um modelo baseado em estruturas em rede.

## **BIG DATA**

**Não é só um crescimento vertiginoso de dados e informações. Trata-se, também, da incorporação de novas fontes de dados e da realização de análises muito mais sofisticadas.**

O conceito de *big data* incorpora várias novidades. Uma delas refere-se à fonte dos dados e informações. Agora, eles virão de todos os lados: dos departamentos da empresa, de fornecedores e clientes, mas, também, das redes sociais e dos objetos. Os dados estruturados da empresa, aqueles que ela produz, mantém sob o seu controle, atualiza e descarta, serão combinados com dados desestruturados, sem controle da fonte e do contexto de produção.

Outra novidade refere-se ao volume de dados. A quantidade a ser armazenada e tratada crescerá de modo muito significativo na próxima década. Além disso, a análise de dados irá se tornar mais complexa. Enquanto os relatórios tradicionais concentram-se muito na descrição e no diagnóstico das situações, as análises do futuro irão valorizar a previsão e prevenção. A sofisticação das análises irá requerer profissionais com um perfil diferenciado, capazes de combinar conhecimentos estatísticos complexos com *insights* sobre relações entre fenômenos que antes não haviam sido colocados juntos. Com a chegada do *big data*, questões relacionadas com gestão, propriedade, proteção e armazenamento de dados deverão, cada vez mais, fazer parte das pautas de discussão, assim como questões relacionadas ao direito de todos de acesso a dados públicos.

*Dados pessoais: o 'novo petróleo da Internet e a nova moeda do mundo digital'.*

O processamento da infinidade de dados digitais que são gerados pelas pessoas durante a realização de atividades diárias, muitos dos quais elas não se dão conta, permitirá identificar padrões de comportamento com precisão e monitorá-los e influenciá-los. Os dados pessoais serão o novo petróleo da Internet e a nova moeda do mundo digital. Uma questão relevante diz respeito a quem terá o domínio desses dados, se eles ficarão sob o controle de empresas privadas, governos ou das pessoas que os geraram.

Ao contrário do que ocorre com pesquisas demográficas e de opinião, a análise de *big data* revela não o que as pessoas pensam ou declaram, mas sim o que fizeram. Isso aumenta exponencialmente a capacidade de prever comportamentos. Essa previsão será útil em muitas áreas, mas duas são especialmente promissoras: saúde pública e planejamento urbano. Na Costa do Marfim, por exemplo, o trânsito diminuiu em 10% apenas com pequenas mudanças no sistema de transporte público, implementadas após coleta de dados de informações do usuário de celulares (Pentland, 2015).

Apesar do imenso avanço que será propiciado pelas análises de *big data*, será necessário evoluir na questão sobre a privacidade dos dados pessoais. Deve existir uma preocupação sobre quem ficará responsável por cuidar desses dados e armazená-los e cuidar para que as pessoas tenham autonomia para decidir o que poderá ser feito com as informações coletadas a seu respeito. Segundo Pentland (2015), seria necessário promover uma reforma legal de grandes proporções destinada a criar um arcabouço legal que permita aos cidadãos ter o direito de propriedade sobre seus dados digitais. Dados como registros de transações financeiras e comunicações telefônicas ou pela Internet precisariam sair do domínio exclusivo de empresas privadas por um motivo simples: o setor privado visa, legitimamente, ao lucro, e não ao bem comum.

Os governos tampouco deveriam ter domínio sobre os dados dos cidadãos. Como ativos individuais, esses dados poderiam ser distribuídos para empresas e governos no caso de o cidadão assim o desejar, mediante condições negociadas. A transformação dos dados pessoais em ativo individual, no entanto, não irá garantir a privacidade. Eles poderão ser identificados, já que deixam rastros digitais. Mas seria possível garantir o sigilo sobre eles. É o que ocorre, por exemplo, no sistema bancário. A maior parte dos gastos de uma pessoa é feita por meio eletrônico. Os bancos sabem exatamente onde e como os correntistas gastam o seu dinheiro, quanto investem e para quem transferem recursos. Mas o sigilo dessas operações encontra-se garantido por contrato.

## REDES SOCIAIS

*Redes sociais: meio rápido e eficiente de comunicação que, com o tempo, irá desbancar o uso de e-mails nas empresas, tornando-as mais ágeis e porosas.*

As redes sociais vieram para ficar. Trata-se de um novo meio de comunicação e troca de informações entre as pessoas e entre as empresas e organizações. As redes irão substituir, pouco a pouco, outros meios de comunicação, desbancando, por exemplo, o uso dos *e-mails* nas empresas. Elas irão reduzir o tempo que as pessoas gastam para se comunicar e trocar informações, facilitando a realização de trabalhos entre equipes distantes, tornando as empresas mais ágeis e porosas.

O *boom* das redes sociais coloca para as empresas a necessidade de incorporar o novo ambiente na sua proposta de *marketing* e de relacionamento com os clientes. Para as fornecedoras de TI, abre oportunidades de negócios relacionadas com o uso eficiente do novo ambiente.

**Nos próximos anos, a Internet será muito diferente do que é hoje. A guerra cibernética irá se intensificar. Os Estados poderão querer criar barreiras para o livre acesso de estrangeiros às suas redes locais.**

A importância que as redes sociais vêm adquirindo é apenas um dos aspectos que confirmam a presença cada vez maior da Internet na vida das pessoas. O ambiente virtual e a conexão em tempo integral caracterizam o novo estilo de vida e trabalho.

Especialistas acreditam que, nos próximos anos, a Internet será muito diferente do que é hoje. Em publicação recente com prognósticos sobre o futuro, executivos do Google (Schmidt e Cohen, 2013) lembram que cada vez mais a rede será utilizada para relacionamentos. Isso irá gerar um acúmulo impressionante de dados pessoais, estimulando o surgimento de empresas especializadas no gerenciamento da reputação *online* e de firmas de seguro, oferecendo proteção contra roubo de identidade virtual e difamação na Internet. A identidade *online* será um bem de alto valor, incentivando o surgimento de mercado negro para a compra de identidades reais ou inventadas.

A guerra digital irá se intensificar. O conflito terá alguma semelhança com a Guerra Fria, baseando-se em provocações que se arrastam no tempo, mas que, desta vez, serão virtuais. Os autores também preveem a possibilidade de ataques terroristas, envolvendo ações *online* e *offline*.

As repartições públicas deverão migrar para a Internet, permitindo aos cidadãos resolverem situações diversas com o Estado através de cliques. Elas se tornarão mais eficientes e protegidas em caso de desastres naturais. Se o prédio de um ministério fosse destruído por um terremoto, por exemplo, a instituição poderia continuar a funcionar *online*, com os funcionários se conectando de qualquer dispositivo com acesso à Internet.

A rede continuará crescendo. Esse crescimento desmesurado irá ameaçar a sua estabilidade e ampliar as possibilidades de ataque cibernético, levando à necessidade de maior vigilância. É muito provável que os Estados tentem exercer controle sobre a Internet, fragmentando-a nacional ou regionalmente. Com o argumento de combater o terrorismo e os crimes *online*, os países poderão criar barreiras à comunicação via rede, fazendo com que pessoas de outros países ou regiões precisem de um passaporte virtual para conseguir acesso. O que, no início, era uma rede ampla, de domínio mundial, começará a se parecer mais com o próprio mundo, com as suas divisões internas e interesses divergentes. Assim, dependendo da forma como o controle estatal seja exercido, os princípios que nortearam o surgimento da Internet e a sua evolução até os dias atuais, baseados na abertura, gratuidade e universalidade, serão substituídos.

Em teoria, a intervenção governamental pode variar desde um controle excessivo até a falta total de controle. Wohnoutka (2013) faz referência a quatro possíveis estratégias de governo no que se refere à Internet. Duas delas consideram uma participação maior do Estado, com ou sem envolvimento forte da sociedade civil, e duas pressupõem uma intervenção estatal baixa, com a busca de controle sendo realizada por grandes corporações ou por organizações da sociedade civil. Cada uma dessas possibilidades traz impactos diferentes para as pessoas e organizações e para o futuro da Internet.

**A rede será cada vez mais inteligente e móvel. Cidadãos, organizações privadas e públicas deverão conjugar esforços para colocar a maior quantidade possível de dados online.**

A rede será mais inteligente e capaz de entender e questionar dados (Cellan-Jones, 2010), o que permitirá a oferta para os internautas de um sistema de buscas refinado. Assim, por exemplo, no futuro, quando alguém buscar ‘próximo ônibus de um dado local para outro’, em vez de aparecerem várias páginas com informações referentes a opções da origem para o destino, a rede trará uma única resposta. A construção dessa nova realidade, no entanto, requer que os dados estejam disponíveis na *web*, o que significa que cidadãos, organizações privadas e públicas devam se esforçar para colocar a maior quantidade possível de dados à disposição *online*, para que possam ser tratados e interpretados por especialistas.

A rede também será onipresente. O seu acesso ocorrerá em qualquer lugar, prescindindo dos dispositivos que são utilizados atualmente. No futuro, por exemplo, a Internet poderá ser acessada por óculos ou por meio de um visor que as pessoas passariam a utilizar. O *browser* irá desaparecer. A interação com a rede ocorrerá por meio de aplicativos, como já é feito atualmente nos *smartphones*.

Os telefones celulares e outros dispositivos móveis inteligentes serão parte essencial da vida das pessoas, o que significa que a sua apropriação (no sentido do acesso e da capacitação para o uso) farão parte dos elementos essenciais que permitem e definem a cidadania.

## COMPUTAÇÃO EM NUVEM

**IaaS, PaaS e SaaS: necessidade de rever a organização do trabalho, reestruturar as equipes e repensar os modelos de negócios.**

No começo, o tema girou em torno do uso da infraestrutura como serviço (IaaS). Havia grandes vantagens na adoção da IaaS. Alugando capacidade de processamento dos data-centros, as empresas podiam reduzir custos de pessoal de suporte e dar novo uso para o espaço antes reservado para os equipamentos. Nos data-centros, a utilização do parque computacional poderia ser otimizada mediante o compartilhamento da capacidade das máquinas. O contratante teria a garantia de recuperação dos dados, caso algo acontecesse no local em que estes se encontrassem, em virtude de redundância para mais de um ambiente físico de guarda.

A IaaS custou a decolar por temor a ataques aos data-centros e roubo de informações. Incertezas quanto ao local em que os dados ficariam armazenados encontram-se entre os motivos que prejudicaram a decolagem. O começo da adoção foi tímido, mas o modelo deverá vingar, já que a lógica se combina com o ambiente de TI cada vez mais focado na mobilidade. A IaaS permite o acesso aos dados onde quer que o

usuário se encontre. Além disso, mesmo acontecendo algo com o dispositivo em mãos do usuário (roubo ou furto, falha do sistema, etc.), os dados encontram-se preservados.

Além das questões envolvendo a infraestrutura como serviço, o interesse também recaiu sobre a plataforma de desenvolvimento como serviço (PaaS). A PaaS permite que equipes localizadas em diferentes países ou em vários estabelecimentos de uma dada unidade empresarial colaborem no desenvolvimento de projetos.

Mais recentemente, o modelo de comercialização de software como serviço (SaaS) veio à tona, trazendo a possibilidade de o usuário final acessar o software disponível na nuvem, mediante o pagamento baseado no uso. Com SaaS, surge mais uma alternativa à licença de uso tradicional. O modelo parece ser vantajoso para o cliente final, pois, pelo menos em tese, reduz o investimento que precisa fazer em um dado produto, o que modifica a relação custo-benefício a seu favor, facilitando a migração de um fornecedor de TI para outro.

Pelo lado do fornecedor de TI, o uso do SaaS traz a necessidade de rever a arquitetura da solução tradicional e decidir o que fazer com a rede física instalada, composta por distribuidores, revendedores e/ou franqueados. Para desenvolvedores de software proprietário, coloca, também, a necessidade de buscar um novo equilíbrio entre receita e despesa, já que os ganhos recorrentes provenientes da licença de uso poderão se transformar em ganhos intermitentes. No futuro, com SaaS, as grandes suítes de software deverão se fragmentar em pequenos módulos funcionais. A redução da abrangência e complexidade, deverá afetar o modelo de negócios tanto de empresas de software proprietário como de empresas/profissionais de software livre/código aberto, reduzindo a necessidade de prestação de serviços diversos (implantação do software, capacitação do usuário, etc.). Esse assunto será retomado mais adiante.

### **De IaaS, PaaS e SaaS para tudo como serviço: uma nova forma de lidar com ativos tangíveis.**

O conceito de infraestrutura, plataforma e software como serviço está avançando, indo além da TI. Daqui para adiante, será tudo como serviço. A otimização do uso dos bens através do compartilhamento abre oportunidades para serviços diversos baseados em um novo modelo de negócios, em que, ao invés da posse de ativos tangíveis se privilegia o uso por tempo determinado, conforme as necessidades.

O modelo de negócios baseado no compartilhamento está mudando o jeito como as pessoas interagem com os objetos. Ao invés de ter um carro próprio, por que não alugar, quando necessário? Por que ter uma bicicleta, se é possível utilizar uma quando necessário? Ao invés de ter um escritório à disposição todo o tempo, por que não compartilhar o espaço com outros profissionais, pagando pelo uso apenas quando for preciso?

A ideia de compartilhamento veio para ficar até porque responde, de modo favorável, ao requerimento de uso inteligente dos recursos disponíveis e cada vez mais escassos, sejam máquinas, prédios, objetos, infraestrutura de comunicação, etc. A mudança terá um impacto importante no comércio de bens e serviços tal como conhecido atualmente.

No futuro, é muito provável, também, que os recursos humanos passem a ser compartilhados. Os trabalhadores prestarão serviços especializados para número cada vez maior de empregadores, que não necessariamente estarão no mesmo local em que se encontra o empregado. As plataformas de desenvolvimen-

to e a infraestrutura em nuvem fornecem o suporte subjacente, necessário para a colaboração à distância.

Com essas alterações, o mercado de trabalho deverá se tornar mais flexível. Os governos terão de repensar os mecanismos legais de proteção ao trabalhador, que favoreçam a contratação de pessoal por horas ou dias, para atividades especializadas pré-definidas. As leis trabalhistas, portanto, precisarão ser revistas. As empresas de software e serviços de TI também terão de se organizar de forma diferente, buscando o talento e as parcerias de negócios onde quer que se encontrem. Atualmente as empresas de grande porte já têm feito este movimento de prospecção e retenção de talentos em nível global. No entanto, no futuro, pequenas e médias empresas também irão se usufruir dos benefícios obtidos com a expansão relativamente sem limites da fronteira do mercado de capital humano.

## INTERNET DAS COISAS (IoT)

*Com a IoT, intensifica-se a convergência entre hardware e software.*

Com a chegada da Internet das coisas, bilhões de novos dispositivos passarão a estar conectados à Internet. Parte dos novos dispositivos contará com capacidade própria de processamento de dados e atuação programada e redefinida por software, parte encaminhará dados coletados para serem processados por um sistema separado fisicamente. Haverá diferentes modelos de interação possíveis, envolvendo homens e objetos, objetos e objetos e objetos e centrais de processamento. A seleção do modelo a ser adotado dependerá, em grande medida, da infraestrutura disponível e das aplicações de interesse.

Existem vários possíveis usos de IoT. As aplicações mais mencionadas incluem a identificação e o rastreamento de pessoas, animais, veículos, cargas e alimentos; o monitoramento de pacientes ou pessoas que requerem cuidados permanentes; a coleta de informações diversas sobre temperatura, uso de energia, capacidade de reservatórios, possibilidade de abalos sísmicos, etc. Inclui, também, a troca de informações entre objetos, tais como veículos automotores, trens e estações.

Como IoT irá afetar a indústria de software e a prestação de serviços de TI? Os objetos, ´animados` por software e conectados à Internet farão com que parte das soluções desenvolvidas em software se tornem obsoletas. O computador deixará de ser o hardware privilegiado. Interfaces e dispositivos irão se multiplicar. Com IoT, haverá uma necessidade maior de recursos humanos com competências em software embarcado e em microeletrônica. Irá crescer o interesse por infraestrutura de comunicações de última geração, incluindo redes sem fio de campo próximo e de longa distância e fibra óptica. Ganham destaque questões relativas ao uso do espectro e a segurança contra ataques cibernéticos que poderiam provocar desastres sem precedentes em aeroportos, rodovias, fábricas, etc. Entra na ordem do dia, também, questões envolvendo a geração e o uso da energia. Vários dispositivos de IoT terão baixa autonomia energética, o que irá requerer soluções de muito baixo consumo.

A chegada da Internet das coisas abre um leque gigantesco de oportunidades. Mas coloca a necessidade imperiosa de capacitar pessoas para lidar com a convergência, que vem a galope, entre hardware e software.

## AUTOMAÇÃO DO TRABALHO DO CONHECIMENTO

### Ocupações em risco: no bojo das novas tecnologias, destruição e criação de empregos.

As máquinas inteligentes estão a caminho e logo farão parte do cotidiano das empresas, domicílios e organizações. Robôs com habilidades diversas irão colaborar com as tarefas diárias dos trabalhadores e, em alguns casos, irão substituí-los. O ambiente de trabalho e as relações no trabalho sofrerão mudanças profundas. Inicialmente, a substituição de humanos por máquinas acontecerá em atividades rotineiras e de menor complexidade. No entanto, a empregabilidade dos trabalhadores do conhecimento, ou seja, de profissionais em ocupações diversas que requerem a tomada de decisão e o processamento de dados e informações complexas, também estará em jogo.

A destruição do trabalho tal como conhecido hoje será acompanhada do surgimento de novas oportunidades, em profissões com contornos ainda pouco definidos. Competências até então inexploradas, porque no cruzamento de saberes que têm sido tradicionalmente tratados por separado (por exemplo, tecnologia, psicologia, sociologia e negócios), passarão a ser fundamentais para garantir a empregabilidade, no futuro. Assim, por um lado, e apesar das incertezas, existe uma certa urgência em repensar o papel das escolas e dos professores e rever o conteúdo a ser ministrado. Por outro, também se faz necessário preparar a empresa do futuro e garantir a empregabilidade dos trabalhadores.

## NOVOS CONSUMIDORES

### Os negócios acontecerão no cruzamento das fronteiras existentes entre os diferentes domínios de aplicação e entre os processos tradicionais de negócios. O entendimento dos contextos de uso deverá guiar a oferta dos novos serviços.

Durante os próximos anos, uma quantidade expressiva de novos consumidores passará a ter acesso às TICs. O contingente será formado por pessoas de baixa renda, localizadas em países emergentes que estiveram, até então, às margens do progresso econômico e social promovido pelo modo de produção capitalista. Para esse contingente de pessoas, o acesso às novas tecnologias poderá se tornar um elemento fundamental de geração de renda e desenvolvimento humano.

Em mercados desenvolvidos e já saturados, os negócios irão surgir no cruzamento das fronteiras existentes entre os diferentes domínios de aplicação e entre os processos tradicionais de negócios. Os novos serviços serão construídos a partir da compreensão das dificuldades diversas dos clientes, considerando os contextos de uso. Assim, por exemplo, o furto de uma casa ou a batida de um carro; a viagem a um outro país ou a doença de um parente; o aniversário de casamento ou o início da colheita de cana-de-açúcar são todos eventos cujos contextos podem gerar uma série interminável de serviços que, atualmente, requerem a ação de diferentes departamentos, instâncias, processos de negócios e profissionais. O desafio será colocar toda esta parafernália junta, surpreendendo o cliente e facilitando a sua vida.

## PRODUTOS SOB MEDIDA, GRATUITOS E FÁCEIS DE USAR

*Um cenário possível: ao invés de plataformas tecnológicas proprietárias, as plataformas serão abertas e o usuário poderá mudar facilmente de plataforma. Os aplicativos complexos e completos serão fragmentados em pequenos módulos funcionais. O usuário poderá combinar módulos de diferentes fornecedores de TI, gerando produtos personalizados.*

As TICs deverão atender um público maior, em mercados de massa. No entanto, mesmo nestes mercados, os consumidores irão querer cada vez mais ser tratados como únicos, vivenciando a sensação de que os produtos ou serviços oferecidos foram feitos exclusivamente para eles. Também irão desejar ter acesso a produtos gratuitos, ou de baixo valor, e fáceis de usar.

Como resultado, observa-se uma mudança radical na entrega de produtos de software e na prestação de serviços de TI. As grandes soluções em software, as suítes completas, perdem espaço. A tendência é que soluções complexas sejam fragmentadas em pequenos módulos funcionais, que possam ser acessados conforme o interesse do cliente. Assim, no modelo SaaS, ao invés de desembolsar recursos para acesso à suíte como um todo, o consumidor irá querer pagar para ter acesso apenas à função específica que tem interesse em utilizar. Essa é a visão que irá se enquadrar com a expectativa de simplicidade do futuro cliente: nem uma funcionalidade a mais do que é necessário.

A fragmentação das suítes em pequenos módulos funcionais exigirá mudanças na forma de conceber e especificar o software. Além disso, irá requerer maior compatibilidade entre os módulos oferecidos pelos diferentes fornecedores de TIC, pois é esperado que os módulos possam ser combinados com outros de uma mesma empresa ou com produtos de outros fornecedores.

Essa nova forma de organização das empresas de software e de seus produtos altera a dinâmica da cadeia de valor tradicional do software. O novo modo de operação pressupõe a adoção de plataformas tecnológicas abertas, como forma de favorecer a combinação de módulos dos vários fornecedores. O comando e a orquestração da rede de fornecedores de TI não serão mais realizados pelos proprietários das plataformas e sim pelo consumidor final. Ou seja, a cadeia de valor do software passará a ser puxada e guiada pelo consumidor final. Diante da complexidade que será colocada nas mãos do consumidor, é muito provável que surjam empresas para desempenhar o papel de articulação dos diferentes módulos e fornecedores de TI, visando a atender, de modo exclusivo, às necessidades específicas de um dado cliente. Essas empresas irão gerar valor, e sobreviver no mercado, com a oferta de serviços fim-a-fim para o consumidor final, ou seja, serviços que irão combinar os pequenos módulos, incluindo produtos de vários fornecedores de TI, para construir solução completa e personalizada, composta de pequenas partes que se encaixam entre si, como num jogo de Lego. Os ofertantes de plataformas abertas, na nuvem, poderão assumir esta tarefa ou deixá-la para empresas dedicadas.

No novo cenário, o dilema passa a ser como criar um modelo de negócios economicamente viável, entregando pequenos módulos funcionais ou soluções personalizadas para consumidores sem disposição para pagar preços vultosos. A sustentabilidade do negócio terá de ser obtida com o aumento dos ganhos de escala e com a contenção de custos, no primeiro caso, ou com a prestação de um serviço diferenciado obtido pela combinação muito adequada de módulos disponíveis (próprios ou de terceiros). Nem todas as empresas atualmente no mercado conseguirão trilhar estas novas opções de geração de negócios.

*Mais poder para o consumidor final. Negócios B2B tendem a se tornar B2C e C2C ou também B2C2T e C2C2T*

Para aumentar as possibilidades de ganho financeiro, reduzir custos e riscos atrelados ao desenvolvimento, o desenvolvedor de software precisará estar cada vez mais próximo ao consumidor final, visando à obtenção de *feedbacks* frequentes e garantindo entregas em mais curto prazo. A aproximação com o cliente permite um conhecimento maior das funcionalidades que o módulo deve conter e agiliza correções de rumo. O desenvolvedor de software também precisará estar atento à concorrência e às possibilidades de parceria, avaliando, de modo sistemático, em que medida os seus produtos se alinham e combinam para gerar serviços fim-a-fim de interesse para o consumidor final.

Os negócios realizados entre as empresas (B2B) tendem a se tornar, cada vez mais, negócios feitos entre as empresas e os usuários finais (B2C) ou diretamente entre usuários finais (C2C) ou, ainda, entre empresas, usuários finais e coisas (B2C2T) ou entre usuários finais e coisas (C2C2T).

*No novo contexto, as externalidades de redes e os efeitos de lock-in irão operar em novas bases.*

O modelo de negócios baseado no pagamento pelo uso de um dado módulo de software (SaaS) e a simplificação de acesso ao software permite a redução do valor do aplicativo, respondendo de modo satisfatório à expectativa do cliente final de contar com produtos gratuitos ou de baixo valor. A redução no preço do aplicativo (e no investimento feito para o seu uso) tende a tornar mais fácil para o cliente final mudar de fornecedor de TI. O uso de plataformas tecnológicas abertas interoperáveis também trabalha no sentido de reduzir os custos dos dispositivos de acesso e de migração de um fornecedor de TI (ou de uma plataforma) para outro. No novo cenário, portanto, as externalidades de redes e os efeitos de *lock-in* passarão a operar em novas bases. As externalidades de redes surgirão da oferta bem alinhada de módulos funcionais capazes de gerar serviços fim-a-fim de alto interesse. A inércia provocada pelo costume e um modelo de negócios capaz de fornecer vantagens e benefícios recorrentes para o cliente final serão mais determinantes para a criação de efeitos de aprisionamento que os pesados custos de aprendizado envolvidos, até então, na saída da órbita de uma dada plataforma tradicional.

## TI BIMODAL

Durante a transição do velho paradigma para o novo, os ofertantes de TI terão de desenvolver um comportamento bimodal: manter o legado e explorar e testar novas possibilidades de produtos, processos e negócios.

Para as mudanças que se fazem necessárias em produtos e processos, os ofertantes de aplicativos irão continuar buscando o suporte de parceiros. No entanto, não necessariamente através da consulta aos velhos fornecedores de plataformas tecnológicas ou livres. E também, não mais contando com a colaboração dos clientes atuais, ainda pesadamente associados ao antigo modelo baseado em licenças de uso, GPL ou atenuações desta. Esses clientes ainda serão de grande utilidade, pois irão permitir a sobrevivência do negócio durante a migração para um novo modelo. Mas as relações de cooperação passarão cada vez mais a ser buscadas em outros clientes ou em parceiros com competências para apoiar a transição. Nesse contexto de mudanças, empresas de consultoria e concorrentes podem ser

mais úteis que os atuais fornecedores e consumidores.

O momento, na realidade, requer a construção de um *modus operandi* bimodal. Este modo se impõe em várias situações. Está presente no processo de desenvolvimento do software: o método antigo, bem estruturado, documentado, mensurado e definido, convive agora com métodos ágeis, rápidos e flexíveis de desenvolvimento.

O caráter bimodal está presente na forma de comercialização do software. Por questões de sobrevivência, ainda por longo tempo, as empresas de software proprietário manterão o modelo tradicional baseado na licença de uso, mas, simultaneamente, começarão a testar o modelo *SaaS*. Para fornecedores de TI de grande porte, é mais fácil assumir este comportamento bimodal. Eles podem manter o modelo de negócios tradicional, baseado em licença de uso, nos grandes clientes e experimentar versões *light* do seu produto, no modelo *SaaS*, em clientes de pequeno e médio porte. Ou seja, para grandes fornecedores, o modelo *SaaS* talvez permita uma expansão dos negócios para clientes de menor porte, sem grandes esforços de reestruturação do produto e sem perda dos ganhos recorrentes.

Desde a perspectiva dos clientes corporativos de pequeno e médio de software proprietário, a implementação de *SaaS* é uma ótima opção, pois baixa o custo de informatização. A adoção de *SaaS* pelos fornecedores de TI possivelmente irá propiciar uma expansão mais rápida da tecnologia neste segmento de empresas.

Para pequenas e médias empresas fornecedoras de TI, a estratégia bimodal das grandes poderá prejudicar os seus negócios que são, em geral, voltados para clientes de pequeno e médio porte, e feitos ainda em modelo baseado em licença tradicional de uso. Para se manter competitivos, os pequenos e médios fornecedores serão obrigados a recompor o seu portfólio de produtos e serviços, buscando se adaptar à nova tendência de negócios. Para eles, isto será mais difícil, pois terão maiores obstáculos para operar em modelo de comercialização bimodal. Provavelmente, o que terão de fazer para sobreviver diante da investida das grandes será substituir o seu modelo tradicional pelo modelo *SaaS*.

A TI bimodal também traz novidades no que se refere às fontes de informação e aos parceiros privilegiados no desenvolvimento e na comercialização de software e serviços de TI. A necessidade de muito rapidamente oferecer novas soluções ao mercado fará com que as empresas busquem apoio de outros fornecedores de TI. Para as grandes empresas, as fusões e aquisições serão uma das formas privilegiadas de incorporação da inovação. *Startups* têm um papel relevante neste cenário, pois tendem a trazer soluções novas, mais sintonizadas com as tendências do mercado. Para pequenas e médias empresas, as fusões e aquisições podem ser mais difíceis de acontecer.

## DE PROCESSOS DE NEGÓCIOS PARA MODELOS DE NEGÓCIOS

*Equipe de desenvolvimento aproxima-se da área de negócios para construção de modelos sustentáveis, inteligentes, humanos, fortemente atrelados às necessidades dos clientes.*

No paradigma tradicional vigente, a TI é chamada a cumprir um papel relevante no mapeamento dos processos de negócios. A sua contribuição consiste em compreender rotinas e fluxos, destrinchar operações

e transferi-las para o mundo digital, automatizando-as. Com isso, várias atividades do dia-a-dia tornam-se mais eficientes e rápidas.

No paradigma em construção, a TI deverá desempenhar um papel diferente. Não se trata mais de transferir o mundo real para o ambiente virtual, e sim de reinventar o jeito de fazer as coisas, desenvolvendo modelos mais inteligentes, humanos e sustentáveis. Para atingir esse novo objetivo, a equipe de desenvolvedores terá de se aproximar da sua área de negócios. O desenho não é mais de processos pré-existentes, mas de serviços ainda não imagináveis.

## EMPRESAS AMBIDESTRAS

*O novo paradigma irá mudar a forma como as empresas dos vários setores econômicos irão trabalhar e os produtos que irão ofertar aos seus clientes. O seu departamento de TI terá de se reinventar.*

Para sobreviver, empresas e organizações de diferentes setores (seguradoras, bancos, varejo, energia, etc.) terão, cada vez mais, de incorporar as novas tecnologias nos seus produtos e processos. E isto irá revolucionar os seus negócios.

Nessas empresas, o departamento de TI passará a desempenhar funções distintas das que vêm realizando até então. Por um lado, irá se liberar de parte das tarefas relacionadas com a manutenção do parque computacional e o suporte ao usuário local, utilizando empresas terceirizadas, em modelo do tipo IaaS. Por outro, irá se aproximar da área de negócios, tornando-se um aliado crítico para o desenvolvimento de produtos exitosos, que incorporem as novas tecnologias. Para sobreviver, as empresas e organizações terão de se tornar uma *startup* de tecnologia. Irão desenvolver um perfil ambidestro: mantendo produtos e serviços tradicionais, para sobreviver no presente, testando novos produtos e serviços, para se perpetuar no futuro.

O departamento de TI das empresas provavelmente irá necessitar rever o seu quadro de pessoal interno e a sua rede de colaboradores externos. O perfil dos seus profissionais e das terceirizadas hoje contratadas pode ser adequado para lidar com as atividades em andamento, mas inapropriado para a geração dos novos negócios.

## 4.2 A INDÚSTRIA BRASILEIRA DE SOFTWARE E SERVIÇOS DE TI (IBSS) NO NOVO PARADIMA

### O SETOR BRASILEIRO DE TIC E A IBSS

*Setor brasileiro de TIC: R\$ 400 bilhões de receita e 93 mil empresas. Parte preponderante no segmento de serviços.*

Segundo estimativas do Observatório SOFTEX, em 2014, o setor brasileiro de TIC gerou algo em torno de R\$ 400 bilhões em receita líquida. O segmento de serviços de TIC (software e serviços de TI e telecomunicações) respondeu por parte significativa desta receita (69,5%), com destaque especial para telecomunicações (45,2%) (Tabela 4.1).

Ao todo, o país contava com algo em torno de 93 mil empresas com atividade principal em TIC. O segmento de serviços também concentrou quantidade relevante delas (94,9%), com atenção para as atividades de software e serviços de TI, que representavam 87,9% do número total de empresas de TIC. A indústria de TIC respondeu por parcela pouco significativa da receita (18,7% do total) e, também, da quantidade de empresas (1,5% do total).

**TABELA 4.1 - NÚMERO DE EMPRESAS E RECEITA LÍQUIDA DE EMPRESAS DO SETOR DE TIC – BRASIL, ESTIMATIVA 2014**

	SEGMENTO DA ECONOMIA DA INFORMAÇÃO	EMPRESAS		RECEITA LÍQUIDA	
		NÚMERO	% SOBRE TOTAL	EM 1.000 R\$	% SOBRE TOTAL
INDÚSTRIA TIC	Fabricação de componentes eletrônicos	613	0,7%	4.276.257	1,1%
	Fabricação de equipamentos de informática e periféricos	320	0,3%	24.971.167	6,3%
	Fabricação de equipamentos de comunicação e aparelhos telefônicos	197	0,2%	21.684.096	5,4%
	Fabricação de aparelhos de recepção, reprodução, gravação e amplificação de áudio e vídeo	233	0,2%	23.470.140	5,9%
	Fabricação de mídias virgens, magnéticas e ópticas	8	0,0%	38.432	0,0%
	<b>TOTAL INDÚSTRIA TIC</b>	<b>1.361</b>	<b>1,5%</b>	<b>74.440.092</b>	<b>18,7%</b>
COMÉRCIO TIC	Comércio por atacado de equipamentos e produtos de tecnologia de informação e comunicação	3.407	3,7%	47.001.578	11,8%
SERVIÇOS TIC	Software e serviços de TI	81.782	87,9%	96.699.963	24,3%
	Serviços de telecomunicações	6.455	6,9%	180.183.992	45,2%
	<b>TOTAL SERVIÇOS TIC</b>	<b>88.237</b>	<b>94,9%</b>	<b>276.883.955</b>	<b>69,5%</b>
<b>TOTAL</b>		<b>93.005</b>	<b>100,0%</b>	<b>398.325.625</b>	<b>100,0%</b>

Dados para indústria TIC incluem apenas empresas com 5 ou mais pessoas ocupadas. Fonte: Observatório Softex, a partir de dados da PAS, PAC e PIA/IBGE, período 2007 a 2012.

**Para a indústria brasileira de TIC, momento é de fragilidade e oportunidade.**

O momento atual da indústria brasileira de TIC pode ser caracterizado como de fragilidade e de oportunidade. As oportunidades combinam fatores diversos. Por um lado, as mudanças em nível global têm um impacto significativo no setor brasileiro das TICs. O país pode se beneficiar com as janelas de oportunidades criadas com a mudança do paradigma dos PCs para os equipamentos e dispositivos móveis.

Por outro lado, em nível nacional, a conjuntura brasileira também abre oportunidades relevantes. Com a economia paralizada e a inflação batendo à porta, as tecnologias podem se tornar um elemento fundamental, e um aliado estratégico, em planos que visam à saída da crise, através da melhoria da produtividade e da competitividade de cadeias produtivas atualmente fragilizadas.

**Destacam-se, em especial, as possibilidades envolvendo as atividades em software e serviços de TI.**

As oportunidades são para empresas posicionadas nos diferentes segmentos da cadeia de valor do setor TIC. Ressaltam-se, em especial, as possibilidades em software e serviços de TI. Essas atividades são destacadas em virtude das suas características intrínsecas: geram empregos qualificados; requerem pouco investimento inicial (em comparação com outras atividades do setor TIC); e possuem baixas barreiras à entrada de novos concorrentes (em particular, no segmento de aplicativos). Por essas características, a Indústria Brasileira de Software e Serviços de TI (IBSS) vem crescendo a taxas significativas, tornando-se uma ótima opção de negócios para pequenos empreendimentos de capital nacional.

Segundo estimativas do Observatório Softex:

- A receita líquida da IBSS chegou à casa dos R\$ 100 bilhões, em 2014. Empresas de software, nas suas diversas modalidades (serviços de desenvolvimento sob encomenda, produto customizável e produto não customizável), foram responsáveis pela metade deste montante;
- em 2014, a IBSS gerou postos de trabalho para 650 mil pessoas ocupadas, entre sócios, cooperados e assalariados.
- a cada ano, aumenta de modo significativo o número de empresas com atividade principal em software e serviços de TI. Em 2007, a IBSS era constituída por 54 mil empresas. O Observatório Softex estima que, em 2014, a quantidade era superior a 80 mil;
- existem oportunidades propiciadas pelo processo de terceirização de atividades de software e serviços de TI realizadas *in house* para a IBSS. A terceirização seria uma medida capaz de fortalecer a indústria local e transferir, para um número maior de consumidores, os benefícios tecnológicos que hoje ainda são apropriados de modo limitado;
- há, também, oportunidades para inserção da IBSS em redes globais e para a atração de investimentos, em virtude da desaceleração de crescimento nos países centrais e a busca por novos mercados.

Por outro lado, existem desafios a serem superados:

- ao longo dos anos, observa-se queda no investimento em P&D e na taxa de inovação da IBSS;
- a margem líquida cai, também, ao longo dos anos, evidenciando dificuldades no equacionamento das relações entre receitas e despesas;
- há queda de produtividade, indicando que, para cada real obtido, são necessárias mais pessoas para obtê-lo;
- a IBSS tende a se reorientar para atividades de menor valor agregado, com diferencial baseado no preço. A concorrência é cada vez mais acirrada;
- a indústria carece de recursos humanos especializados e de nível técnico. Empresários se queixam dos desajustes entre o conteúdo fornecido nas escolas e as competências requeridas no mercado de trabalho;
- observa-se movimento de aquisição de empresas nacionais pelo capital estrangeiro;
- apesar de numerosa, a IBSS é composta por empresas de muito pequeno porte, que enfrentam dificuldades para crescer e ganhar musculatura.

Assim, o país conta com uma indústria de software e serviços de TI que cresce em virtude de um mercado interno que se expande rapidamente. No entanto, essa indústria apresenta cada vez maior dificuldade para sustentar o seu crescimento, manter autonomia tecnológica e inserir-se, de modo privilegiado, nas redes globais. É necessário, portanto, equipar as empresas da IBSS para que possam responder de modo adequado às novas tendências em TIC.

## O futuro da IBSS

No novo cenário em formação, como prever o futuro da indústria brasileira de software e serviços de TI? Haverá espaço para empresas de capital nacional? Onde estão as boas oportunidades e quais são os principais desafios em software e serviços de TI?

### *Presente e futuro em software*

Em sua maioria, a indústria brasileira de software de capital nacional é constituída por pequenas empresas, que desenvolvem software do tipo aplicativo. No país, o mercado de aplicativos horizontais é fragmentado por porte, área de atuação e localização do cliente. As empresas de capital estrangeiro, especialmente SAP e Oracle, dominam o segmento de ERP para grandes clientes. A Totvs, gigante de capital nacional, possui base forte de clientes de médio porte, embora outras empresas de capital nacional também se encontrem bem posicionadas neste porte de cliente, orientando os seus produtos para áreas específicas de atuação neste mercado, criando, por exemplo, ERPs especializados para agronegócios ou, dentro deste, para manejo vegetal, etc. No segmento constituído por clientes de pequeno porte, predominam pequenas empresas de capital nacional que encontram-se bem posicionadas em determinadas áreas de aplicação, com área de influência em mercados locais.

As empresas de software de capital nacional também buscam mercados horizontais específicos (por exemplo, RH, CRM, etc.), especialmente aqueles em que a presença de empresas de capital estrangeiro é dificultada pela necessidade de customização de produtos. É o caso, por exemplo, de soluções para a área fiscal.

Nos mercados de software para verticais (bancos, telecomunicações, saúde, seguros, etc.), existem, ainda, muitas oportunidades a serem exploradas. O desenvolvimento deste tipo de software requer conhecimento especializado da vertical, o que acaba fazendo com que muitas empresas desenvolvedoras de software vertical surjam como resultado de *spin-off* dos departamentos de TI das grandes empresas clientes.

Pelo posicionamento das empresas de capital nacional na cadeia de valor do software, a seleção das plataformas tecnológicas é relevante: da boa escolha depende a sua capacidade de sobrevivência. Por isso, durante muitos anos, elas tenderam a gravitar em torno da plataforma líder, Windows Microsoft. A parceria com Microsoft propiciava externalidades de redes: as empresas de capital nacional se beneficiavam com o uso da marca da empresa líder e, simultaneamente, reforçavam a marca da empresa líder.

O modelo de negócios praticado pelas empresas de capital nacional é ainda muito fortemente baseado na licença de uso tradicional. A oferta de software livre/código aberto é restrita a alguns poucos nichos. Os ativos em relação aos concorrentes são o conhecimento do mercado, das regras locais de negócios e da legislação brasileira, além do bom relacionamento e da proximidade com o cliente. Os mercados de nicho com ganhos baixos de escala e a informatização ainda incipiente de pequenas e médias empresas de vários segmentos geram um leque vasto de oportunidades para as fornecedoras de TI. No entanto, a chegada das novas tecnologias pode afetar de modo profundo a sua realidade.

1- As plataformas móveis ganham cada vez mais peso em nível nacional, reestruturando a oferta de software, os modelos de negócios e as parcerias das empresas. As *apps* oferecidas em lojas de aplicativos na nuvem tornam-se uma alternativa para comercialização de software e oferta de serviços.

2- Em nível nacional, as grandes empresas estrangeiras de ERP começam a oferecer o seu produto em modelo baseado em SaaS para clientes de pequeno porte, disputando este segmento de mercado até então dominado pelas fornecedoras de TI de capital nacional. Para não perder a sua posição, estas empresas possivelmente terão de repensar o seu modelo de negócios e reestruturar rapidamente a sua oferta.

3- As novas tecnologias irão abrir mercado para novos produtos e destruir o mercado para os produtos atualmente oferecidos pelas empresas brasileiras de software. Muitos dos processos de negócios automatizados e materializados em uma solução em software passarão a ser processados por máquinas inteligentes, prescindindo da intervenção humana. O software estará cada vez mais embarcado em dispositivos diversos. Isso, por um lado, acarretará na necessidade de reestruturação dos produtos tal como vêm sendo oferecidos atualmente pelos fornecedores nacionais de TI. E, por outro, irá requerer mudanças no perfil dos profissionais de TI.

4- As empresas brasileiras de capital nacional não possuem os atributos necessários para se organizar como líderes de plataforma, consolidando no mercado um sistema operacional ou uma linguagem de programação próprios, por exemplo. A vocação nacional é para aplicativos.

5- O país tampouco conta com uma indústria de hardware relevante, com a presença de um ecossistema constituído por empresas com potencial para desenvolvimento de famílias de dispositivos móveis nos quais embarcar soluções próprias ou de terceiros. Talvez exista espaço para o desenvolvimento de dispositivos de monitoração e atuação ou de sistemas de identificação, incluindo, neste caso, sensores, atuadores e etiquetas de RFID, especialmente em situações em que a criatividade brasileira se concentre na combinação das partes ou customização de um ou mais integrantes da solução.

### Presente e futuro em serviços de TI

Os serviços de TI criam oportunidades diversas de negócios para as empresas brasileiras de capital nacional. Entre elas, destacam-se as seguintes:

- Oferta de serviços de desenvolvimento de software para empresas que ainda mantêm sistemas legados, em especial, nos setores financeiro, de telecomunicações e da administração pública. A oferta de serviços ocorre, em geral, em modelo do tipo fábrica de software.
- Oferta de serviços de desenvolvimento de software para empresas interessadas em pagar para ter soluções customizadas. O modelo praticado, em geral, é o de desenvolvimento sob encomenda, cobrindo todo o ciclo de vida do produto. O interesse por produto diferenciado pode estar ligado à ausência de soluções padrões disponíveis no mercado ou ao forte ingrediente inovador da solução idealizada.
- Oferta de serviços de consultoria para empresas líderes, tais como SAP, Oracle, Red Hat, etc. As consultorias podem envolver atividades diversas, tais como treinamento, customização e integração de sistemas e distribuição e revenda de produtos das líderes, complementados pela oferta de produtos próprios.
- Prestação de serviços de manutenção e suporte técnico em TI.
- Processamento/tratamento de dados e oferta de serviços de aplicação e de hospedagem na Internet.
- Desenvolvimento de portais e oferta de conteúdo e outros serviços de informação na Internet.
- Reparação e manutenção de computadores e equipamentos periféricos.

O que muda com a chegada das novas tecnologias e como estas transformações afetam os negócios das empresas brasileiras prestadoras de serviços de TI?

1- Com a maior simplicidade das soluções e com a tendência de o software ser embarcado nos dispositivos, parte dos serviços técnicos oferecidos atualmente por empresas com esta atividade-fim (incluindo consultorias e suporte técnico) irá desaparecer.

2- Parte da comercialização deixará de ser realizada através da rede de parceiros e distribuidores e passará a ser feita através de lojas com oferta para *download*, nas nuvens.

3- Surgirão os serviços de orquestração de *apps*, ou seja, empresas dedicadas a montar suítes completas a partir dos vários pequenos módulos funcionais disponíveis a fim de atender a um cliente em particular.

4- As empresas irão se dedicar ao fornecimento de serviços inteligentes, ofertados considerando contextos específicos de uso, muitos dos quais baseados em dados abertos.

5- Os serviços oferecidos deverão estar mais atrelados com os aplicativos e com as necessidades do cliente final. As empresas brasileiras poderão se beneficiar com o conhecimento que possuem do cliente final, o que será o grande negócio do futuro.

6- Haverá espaço para a prestação de serviços de manutenção da infraestrutura em TI e serviços de segurança da informação em data-centros localizados no país.

### 4.3 SOFTWARE LIVRE/CÓDIGO ABERTO NO NOVO PARADIGMA

*Hoje quase tudo é livre. Mas o emprego de software livre/código aberto não acarretou nas liberdades imaginadas.*

No ambiente para *desktops*, as chances de Linux deverão continuar reduzidas. É provável que Linux continue mantendo alguma participação no mercado de servidores/estações de trabalho, em virtude de este segmento ser menos contaminado por escolhas e preferências do usuário final sem perfil técnico.

Linux encontrou menos barreiras no paradigma da mobilidade. Na versão Android, chegou até os usuários finais com pouco conhecimento de informática, incorporado em dispositivos móveis tais como *tablets*, *smartphones*, *iPods*, etc.

O software livre/código aberto está ligado ao sucesso de novas empresas tais como Google e Facebook. No entanto, como lembra um integrante da comunidade livre, as pessoas consultam sistemas de busca na Internet, interagem nas redes sociais ou utilizam dispositivos móveis com sistema operacional livre, mas ignoram que estão utilizando sistemas livres.

O fato de o Android ter se tornado um dos líderes de mercado no segmento de sistemas operacionais móveis não acarretou por si só nas liberdades que, segundo a visão da comunidade livre, estariam associadas ao uso de um sistema livre. Também não levou, no Brasil e em outros países, a um crescimento vertiginoso de empresas desenvolvedoras de software livre, gravitando em torno da nova plataforma. Além disso, conforme a opinião de um integrante da comunidade, tirou a força do movimento pelo uso do software livre, já que, atualmente, quase tudo é livre. Como diz: “Google e Facebook não existiriam sem software livre. O alcance hoje é gigantesco. O Linux está em tudo. O Android é Linux, houve certa restrição no começo mas hoje quase tudo é livre. Não há mais necessidade de movimento para incentivar o uso do software livre. [...] Se cortarem o software livre, o mundo para. Ele está presente em roteadores, televisores, *blue ray*, etc.)”.

A infraestrutura em Linux mostrou-se uma decisão acertada para empresas que utilizam sistemas operacionais e ferramentas livres como meio para gerar serviços facilitados pela TI (*Information Technology Enabled Services, ITES*). Um exemplo, nesta direção, é o ambiente Google disponível na *web*, para pesquisa por dados e informações. Vale recordar, ainda, que o Android se tornou o sistema operacional privilegiado das empresas desenvolvedoras de software que não dispunham de um sistema próprio.

A estratégia Google foi determinante para o sucesso do Android. Um dos seus acertos deve-se à criação de uma aliança de empresas interessadas em gravitar sobre a nova proposta de plataforma tecnológica: a *Open Handset Alliance* (OHA) (Quadro 4.2). Isso permitiu, por um lado, a redução dos custos e das incertezas técnicas relacionadas com o aprimoramento do Android, e, por outro, ampliou as chances de, com apoio de parceiros, lançar rapidamente no mercado equipamentos com o software embarcado e produtos e serviços complementares.

#### QUADRO 4.2 - A HISTÓRIA DO ANDROID

2005 – Google compra a Android Inc, empresa a qual já vinha suportando financeiramente, dando início ao desenvolvimento de um sistema operacional móvel tendo como base o *kernel* Linux.

2007 – Surge a *Open Handset Alliance* (OHA), aliança entre Google e fabricantes de telefones como HTC, Sony e Samsung, operadoras de telefonia como Sprint, Nextel e T-Mobile e fabricantes de *chipsets* como Qualcomm e Texas Instruments. A OHA tinha como objetivo criar uma plataforma de padrão aberto para dispositivos móveis. Na ocasião, o Android foi revelado como o seu primeiro produto: uma plataforma móvel construída no *kernel* Linux de versão 2.6.25.

2008 – Lançamento do primeiro *smartphone* rodando o Android, o HTC Dream.

2010 – Google apresenta uma série de dispositivos Nexus, linha de *smartphones* e *tablets* rodando o sistema Android, fabricado por empresas parceiras.

2013 – *Google Play Store* atinge a marca de 1 milhão de aplicativos disponíveis.

[Extraído, com alterações, de Wikipédia, em novembro de 2014]

Outro aspecto da estratégia foi a decisão de licenciar o Android como código aberto, fornecendo aos parceiros e colaboradores a possibilidade de darem o destino de sua preferência para as soluções criadas a partir de alterações no Android. Como consta da página do Android: “Android tem a ver com liberdade e escolha. O seu objetivo é permitir abertura no mundo móvel e não acreditamos que seja possível predefinir ou determinar todos os usos que as pessoas irão querer dar ao nosso software. Assim, encorajamos a todos a fabricar dispositivos que sejam abertos e modificáveis, mas não acreditamos que esteja sob o nosso controle forçá-los a agir desta forma” [<http://source.android.com/source/licenses.html>. Tradução própria].

A decisão pelo código aberto, ao invés do software livre, atenuou o *copy left* e abriu a possibilidade de o código-fonte vir a sofrer modificações e ser apropriado de modo proprietário por interessados. Além disso, embora Android seja um software de código aberto, as versões embarcadas nos dispositivos móveis não foram desenvolvidas no modelo bazar, tal como defendido por Eric Raymond. Responder de modo proativo às motivações nem sempre convergentes das grandes empresas e das comunidades livres que gravitam em torno do Android deve ter sido um dos desafios a serem enfrentados no processo de consolidação da plataforma. Em teoria, a decisão de licenciamento em código aberto deve ter agradado mais às grandes empresas que às comunidades livres. Para as grandes empresas, a opção pelo código aberto fornecia possibilidades para uma futura apropriação proprietária. As comunidades livres, pelo seu lado, talvez tivessem mais motivos para preferir licença capaz de garantir que o resultado da sua ação voluntária não fosse apropriado por ninguém, de modo particular.

Para Google, ficou o desafio de descobrir como obter ganhos financeiros com o Android, considerando que outros desenvolvedores, sejam empresas sejam integrantes das comunidades, teriam oportunidades para criar sistemas similares melhorados. Google baseou o seu modelo de negócios na obtenção de ganhos econômicos com a plataforma através da oferta complementar de software proprietário. Os fabricantes de dispositivos móveis e as operadoras de telecomunicações que circulavam em torno da plataforma fizeram o mesmo, combinando software proprietário e software de código aberto.

A estratégia traz riscos inerentes. Um deles diz respeito à possibilidade de atores relevantes substituírem a plataforma Android por outra semelhante, que passe a utilizar soluções próprias, ao invés do ambiente proprietário Google. Os usuários poderiam migrar facilmente de uma plataforma para outra, pois os efeitos de *lock-in* são atenuados neste modelo de negócios. Assim, por exemplo, empresas chinesas estão fabricando computadores pessoais com sistema operacional móvel baseado em Android para competir diretamente com o Windows da Microsoft e com o Android do Google. A academia chinesa de engenharia revelou que mais de uma dúzia de companhias estavam customizando o Android, seguindo o banimento do Windows 8 dos computadores pessoais do governo chinês. Um resultado de um ambiente de *lock-in* atenuado é que as escolhas dos usuários passam a estar mais direcionadas para o valor percebido nos aplicativos. Sem a certeza da lucratividade propiciada pelos efeitos de *lock-in*, será decisivo contar com as externalidades de redes, ou seja, criar parcerias que permitam trazer produtos complementares capazes de fornecer valor elevado para a plataforma.

A fragmentação do Android em uma variedade de plataformas similares, cada uma com contornos próprios, pode ser uma ameaça para a possibilidade de consolidação deste sistema operacional ao longo do tempo, podendo prejudicar, em mais longo prazo, a sua capacidade de superar o iOS da Apple ou outras plataformas proprietárias. Como visto no Capítulo 1, no passado, durante o processo de consolidação de plataforma tecnológica para o ambiente de PCs, a dificuldade de os diferentes fabricantes chegarem a um acordo sobre a versão Unix em uso prejudicou a sua concorrência com o Windows NT no mercado de sistemas operacionais para estações de trabalho e servidores. Agora, no novo ambiente móvel, isso poderia acontecer novamente, ou seja, o Android poderia se fragmentar? Google perderia controle em virtude da intensa customização e proliferação de soluções fornecidas por outras empresas? Os fabricantes dos diferentes dispositivos que rodam Android estariam dispostos a usar a plataforma Google e as suas soluções proprietárias, ou iriam preferir criar as suas próprias soluções, afastando-se do Google Play? A Amazon, por exemplo, preferiu esta segunda alternativa (Quadro 4.3).

#### QUADRO 4.3 - A ESCALADA DO ANDROID E AS POSSÍVEIS AMEAÇAS À SUPREMACIA DA GOOGLE

Android é um sistema operacional móvel baseado no núcleo Linux, atualmente desenvolvido pela empresa Google. Projetado para dispositivos móveis (*smartphones e tablets*), possui interface específica para TV, carro e relógio de pulso. Também é utilizado em consoles de videogames, câmeras digitais, computadores e outros dispositivos eletrônicos. Atualmente, é o sistema operacional móvel mais utilizado no mundo. Em 2013, possuía a maior porcentagem de vendas mundiais de sistemas operacionais móveis. Dispositivos com o sistema Android vendem mais que eletrônicos com Windows, iOS e Mac OS X combinados. Em 2013, a loja de aplicativos Google Play possuía mais de 1 milhão de aplicativos disponíveis baixados mais de 500 bilhões de vezes. Em 2013, segundo pesquisa realizada, 71% dos programadores para sistemas móveis desenvolviam para o Android.

O código do Android é disponibilizado como código aberto, apesar de a maior parte dos dispositivos ser lançada com uma combinação de software livre e software privado. Só funciona sem modificações em determinados aparelhos,

como os da série Nexus, da Google. Assim, a cada atualização do Android, o código precisa ser adaptado por cada fabricante e operadora, para cada novo modelo, para que rode nos hardwares específicos. Essa necessidade de adaptação reduz a rapidez com que novos modelos chegam ao mercado e aumenta o custo do produto, tornando-se pontos fracos do Android em comparação com o seu principal rival, o sistema iOS da Apple.

O Android conta com comunidades de desenvolvedores que usam o *Android Open Source Project* (AOSP) para desenvolver e distribuir suas próprias versões modificadas do sistema operacional, incluindo novas características e funcionalidades, que, no geral, tornam-se disponíveis antes das versões dos fabricantes e das operadoras. As comunidades também contribuem fornecendo suporte para dispositivos fora de linha que rodam em Android e levando o sistema para dispositivos que não foram lançados originalmente em Android.

Os fabricantes de *smartphones* e as operadoras de telefonia móvel ficam apreensivos com o desenvolvimento de *firmware* por terceiros, pois temem que os dispositivos rodando softwares desenvolvidos por usuários funcionem mal e aumentem o custo do suporte técnico. Além disso, esses softwares modificados mexem com o modelo de negócios dos fabricantes e das operadoras, oferecendo gratuitamente ferramentas que costumam ser oferecidas mediante pagamento adicional. Por este motivo, inicialmente, os fabricantes e as operadoras criaram obstáculos técnicos para impedir o acesso dos desenvolvedores aos softwares. O cenário mudou mais recentemente, por conta de decisões legais em favor do desbloqueio dos dispositivos. Fabricantes tais como HTC, Motorola, Samsung e Sony passaram a encorajar e até mesmo a oferecer suporte para ações das comunidades livres.

Apesar de o Android circular com licença de código aberto, a maior parte dos dispositivos é colocada à venda com uma grande quantidade de software particular, como as *apps* do *Google Mobile*, que incluem o *Google Play Store*, *Google Search*, *Google Play Services* e outros. Estes dispositivos devem ter licenças expedidas pela Google e só podem ser fabricados se certos requisitos forem alcançados.

Diversos aplicativos nativos do Android que costumavam ser de código aberto, como a Pesquisa, Música e Calendário, passaram a ser de código fechado e começaram a ser distribuídos pela loja de aplicativos da Google com os novos nomes de Pesquisa Google, Google Play e Google Calendário. Richard Stallman e a Fundação para o Software Livre têm sido críticos do Android e têm recomendado a utilização do *Replicant*, pois os *drivers* e *firmwares* vitais para o funcionamento do Android são de propriedade privada.

A mídia especializada elogiou a natureza de código aberto do Android, entendendo que esta característica foi a principal força do sistema, permitindo empresas como Microsoft, Amazon, Barnes & Noble, Ouya, Baidu e outras a criarem ramificações a partir do Android, que se tornou o sistema operacional padrão para empresas interessadas em lançar novos softwares e que não possuem sistema próprio.

Há relatos de que a Google não estaria conseguindo transformar o Android em uma fonte de lucro. A Google estaria perdendo controle do Android em virtude da intensa customização e proliferação de aplicativos de outras empresas. O *tablet* da Amazon, *Kindle Fire OS*, por exemplo, é um ramo do Android extremamente modificado, que não inclui nenhum software proprietário da Google e requer que o usuário compre os aplicativos da *Amazon AppStore*, ao invés da *Play Store*. Em meados de 2014, em um esforço para aumentar a proeminência da marca Android, a Google começou a exigir dos fabricantes que usam os seus aplicativos proprietários a inclusão do logo do Android na tela do *boot* do sistema.

O Android tem sofrido de fragmentação, uma situação em que a variedade dos dispositivos Android, em termos de hardware e software, faz com que seja extremamente difícil desenvolver aplicativos que funcionem em todas as variações, em comparações com plataformas rivais, como a iOS, em que o sistema de hardware e software varia muito menos. Em 2013, haviam 11.868 modelos de Android.

[Extraído, com alterações, de Wikipédia, em novembro de 2014]

No Brasil, predomina o modelo baseado no desenvolvimento *in house*, em instituições governamentais, com posterior inclusão do software em um repositório público

Como já discutido no Capítulo 3, no Brasil, as comunidades locais voltadas para o desenvolvimento de software livre/código aberto se envolvem em projetos liderados pela comunidade internacional e em projetos encomendados por instituições governamentais. Dentro das instituições públicas, existem equipes próprias de desenvolvimento de software livre/código aberto, com projetos *in house* realizados, no geral, sem uma cooperação efetiva das empresas e comunidades livres. Após finalizados, os softwares desenvolvidos por estas equipes costumam ser transferidos para repositórios públicos.

O modelo baseado no desenvolvimento interno e posterior colocação do software em um repositório rompe a lógica do processo preconizado por Raymond, que tem um dos seus alicerces amparado na interação forte de desenvolvedores e colaboradores ao longo de todo o ciclo de vida do produto. O desenvolvimento do software *in house* pode prejudicar ações futuras de *release* de novas versões e suporte técnico ao usuário, se não houver uma comunidade ativamente interessada em dar continuidade à iniciativa. Ou seja, a separação entre desenvolvedor, equipe de suporte, colaborador/usuário não é adequada, pois impede a interação rica necessária para a evolução futura do software. Também retira da equipe de suporte a possibilidade de participar ativamente do processo de especificação e concepção do software. Assim, uma comunidade de interesse, incluindo desenvolvedores, colaboradores e usuários finais (que, no caso de softwares públicos, em última instância, seriam os cidadãos), deveria ser criada desde o momento de concepção do software, gerando aprendizado colaborativo e ambiente propício para a continuidade do produto.

No cenário de mobilidade, as relações entre desenvolvedor, profissional de suporte, colaborador/usuário final ganharão novos contornos. Com a tendência à simplificação dos softwares, os serviços de suporte técnico deixarão de ter a importância que têm em ambiente em que predominam as suítes complexas. Além disso, a mudança no modelo de negócios envolvendo software e serviços de TI, com a tendência de a licença tradicional de uso ser substituída pelo pagamento pelo uso (SaaS) ou pelo fornecimento gratuito do software, reduz uma das motivações dos usuários para o emprego do software livre/código aberto: a redução no custo da licença.

Da mesma forma que a interação rica com a comunidade não ocorre no modelo baseado no desenvolvimento de software livre/código aberto *in house*, a interação também não acontece quando cada novo agente resolve modificar o software para combiná-lo com soluções em hardware e software proprietárias. Essa releitura de uma mesma plataforma cria pequenas ilhas separadas entre si, com graus variados de conexão entre elas. Isso tem ocorrido com a adoção de Linux como plataforma tecnológica predominante no novo paradigma em construção, baseado em cenário de mobilidade e propiciado pelo advento de novas tecnologias.

## NOVAS FRONTEIRAS PARA AÇÃO E DISCUSSÃO

A presença disseminada do Android nos vários dispositivos móveis garantiu, apenas em termos, a liberdade dos interessados em usar, estudar, copiar, modificar e redistribuir o software. Isto por que, por um lado, os grandes fabricantes de dispositivos móveis, as operadoras de telecomunicações e as empresas que dispõem de *app stores* criaram condições para cercear a liberdade, incorporando partes proprietárias à plataforma aberta. Por outro lado, o fato de o software estar cada vez mais imbricado com o hardware, torna difícil, para o usuário final, garantir a sua liberdade para incluir o software de seu interesse no hardware da sua escolha.

*O advento das novas tecnologias trouxe um conjunto todo novo de questões que acabaram retirando o foco do software enquanto tal.*

Além disso, o advento das novas tecnologias trouxe um conjunto todo novo de questões que acabaram retirando o foco do software enquanto tal, recolocando-o no conteúdo propriamente dito. As questões em foco relacionam-se, por exemplo, com o processo de digitalização da vida das pessoas, a criação e distribuição dos conteúdos midiáticos e com o exercício da liberdade através da cidadania digital. Envolve aspectos tais como a gestão dos direitos digitais, a proteção dos dados pessoais, a abertura dos dados e o uso de formatos abertos de dados. É o que será analisado a seguir.

### DRM: gestão dos direitos digitais

*Nova fronteira de discussão: conteúdos digitais e as restrições impostas a sua distribuição*

A gestão de direitos digitais (DRM - *Digital Rights Management*) é uma abordagem sistemática para proteção dos direitos de autor para mídias digitais. O propósito da DRM é impedir a redistribuição não autorizada da mídia digital e restringir as maneiras que os consumidores podem copiar o conteúdo do que eles compraram. Os produtos de DRM foram desenvolvidos em resposta ao crescimento rápido da pirataria *online* de conteúdo comercializado, que proliferou através do uso disseminado dos programas de troca de arquivos *peer-to-peer*. A DRM é tipicamente implementada por código embarcado que previne cópia, especifica um período de tempo no qual o conteúdo pode ser acessado ou limita o número de dispositivos em que a mídia pode ser instalada.

A gestão de direitos digitais aplica-se somente aos meios digitais. Dois motivos explicam a popularidade do conteúdo digital sobre o conteúdo analógico (Wikipedia, 2014): as vantagens técnicas associadas com a sua produção, reprodução e manipulação e a qualidade superior das cópias geradas por meio digital. A DRM surgiu, justamente, como uma medida de controle da duplicação e disseminação deste conteúdo sem limites, protegendo a indústria fonográfica, cinematográfica e de jogos eletrônicos.

Alguns projetos de lei apoiados pela indústria querem proibir a produção e comercialização de qualquer dispositivo que tenha a capacidade de gravar ou reproduzir som, vídeo, texto ou qualquer outra forma de expressão, a menos que esteja equipado com hardware adequado para a implementação de DRM. Mesmo com a intenção de incluir o sistema no hardware, já existem muitos sistemas de DRM baseados em software que já são suficientemente fortes para restringir efetivamente a cópia de conteúdos com direitos autorais. Grande número de *media players* disponíveis hoje incluem formas bastante sofisticadas de DRM sem suporte nativo no hardware (parte física).

Softwares com licenças GPL GNU não permitem distribuição com DRM. A imprensa tem narrado casos em que softwares sob licença GNU foram retirados da loja de aplicativos da Apple, pois esta permite DRM. Foram os casos, por exemplo, do VLC e do jogo GNU Go.

As empresas reclamam que a DRM é necessária para combater as ações *online* contra os direitos do autor e manter os consumidores resguardados de vírus. Mas não há evidências de que a DRM ajude o combate a essas coisas. Ao contrário, a DRM auxilia as grandes empresas a frear a inovação e a concorrência tornando mais fácil evitar usos desautorizados de mídia e tecnologia.

Várias organizações têm se esforçado para explicar as restrições incluídas em novos equipamentos (iPhone, smartphones, DVDs e blue-rays) e lutado pelos direitos do consumidor de fazer cópias do conteúdo digital.

## Backdoors e a proteção dos dados pessoais

### Backdoors põem em risco a proteção de dados pessoais e tiram o controle das máquinas das mãos dos usuários

Existe cada vez maior apreensão sobre a existência de *backdoors* em softwares proprietários que poderiam estar sendo utilizados pelos governos para ter acesso a dados de pessoas e empresas e interceptar comunicações e mensagens. Desconfianças envolvendo as parcerias entre a Microsoft e o governo dos Estados Unidos para desenvolvimento do sistema operacional Windows são antigas. Por várias vezes, a Microsoft negou que tivesse criado *backdoors* em versões do Windows a pedido da *National Security Agency* (NSA) (Quadro 4.4).

#### QUADRO 4.4 – MICROSOFT E O GOVERNO NORTE-AMERICANO TRABALHAM JUNTOS PARA APRIMORAMENTOS NO WINDOWS

Em 2009, a Microsoft negou que tivesse criado uma *backdoor* em seu sistema operacional, suspeita que surgiu após um porta-voz da *National Security Agency* (NSA) ter declarado ao subcomitê do senado para segurança interna e terrorismo que a agência auxiliou no desenvolvimento do Windows 7, com o objetivo de aprimorar a sua segurança e torná-lo compatível com os softwares usados pelo governo. Na ocasião, o diretor executivo do *Electronics Privacy Information Center* (EPIC), questionou o envolvimento da NSA no desenvolvimento do sistema operacional. “O problema chave é que a NSA possui uma missão dupla, trabalhando para zelar pela segurança da informação, mas, também, se envolvendo em espionagens eletrônicas (KEIZER, G., 2009a e 2009b). Para o diretor, a NSA poderia pressionar empresas como a Microsoft para construir *backdoors* em seus códigos de forma a permitir a agência rastrear usuários e interceptar comunicações. Para as empresas, seria muito difícil recusar qualquer sugestão do governo, neste sentido.

Na ocasião, o porta-voz da NSA também informou que a agência estava trabalhando com outros grandes desenvolvedores de software, incluindo Apple, Sun e Red Hat, em padrões de segurança para os seus produtos. Segundo ele, cada vez seria mais necessário pensar na proteção de sistemas de segurança nacional, o que demandaria a realização de alianças entre instituições públicas e privadas, visando ao alcance de padrões maiores de segurança.

Essa não foi a primeira vez que a NSA celebrou parceria com a Microsoft para desenvolvimento do Windows. Em 2007, a agência confirmou envolvimento no Windows Vista como parte de uma iniciativa para assegurar que o sistema operacional fosse seguro a ataques e que fosse compatível com outros softwares do governo. Antes disto, a NSA também

havia fornecido direcionamento para a Microsoft sobre garantias de segurança para o Windows XP e o Windows 2000. Com o Windows Vista foi a primeira vez que a NSA trabalhou com um fabricante antes do lançamento de um sistema operacional. Através do envolvimento cedo no processo, a NSA ajudou a Microsoft a garantir que estava entregando um produto que era seguro e compatível com o software do governo (McMillan, 2007).

*Backdoors* universais permitem que os equipamentos e dispositivos em mãos dos usuários finais possam ser utilizados para fazer absolutamente qualquer coisa, apagando ou substituindo softwares e arquivos e tendo acesso a dados e informações sem o consentimento prévio e o conhecimento do usuário.

Várias organizações têm alertado para o perigo. A *Free Software Foundation*, por exemplo, defende o controle total do computador e dos dispositivos móveis pelo usuário, o que, segundo a organização, é violado atualmente, com o uso de software proprietário, incluindo, entre eles, os sistemas Windows e Mac OS e vários produtos que rodam Android.

As suspeitas também recaem sobre os *chips* da Intel. É possível que possuam *backdoors* que permitam ao governo norte-americano monitorar cidadãos estrangeiros. As mesmas desconfianças surgem em relação aos demais governos. Dispositivos fabricados por empresas de outros países também poderiam conter *backdoors* para espionagem eletrônica do país sede da empresa.

Esse cenário de suspeitas generalizadas acaba colocando para os governos a questão de se deveriam usar sistemas operacionais desenvolvidos em outros países. O conhecimento antecipado de *backdoors* pelos governos (ou a inclusão propositada de *backdoors*) em sistemas operacionais de empresas locais, poderia permitir aos governos atacarem o parque computacional de outros países que rodam aquele sistema operacional.

No cenário de omni-espionagem, as desvantagens recaem sobre os países que não dispõem de empresas líderes fabricantes de dispositivos móveis e de circuitos integrados ou desenvolvedoras de sistemas operacionais, como é o caso, por exemplo, do Brasil. Mas mesmo nos países com maior potencial para o exercício da espionagem, os cidadãos têm motivos para estar preocupados, já que nada impede que os governos direcionem a espionagem para o monitoramento do ambiente doméstico.

A questão da proteção dos dados pessoais não diz respeito apenas ao acesso do governo aos dados e às informações que circulam pelas redes ou são armazenados em computadores pessoais, dispositivos móveis ou na nuvem. Isso porque as empresas líderes fabricantes de hardware e desenvolvedoras de software também poderiam criar *backdoors* para dados e informações úteis para os seus negócios. Várias delas são acusadas de manter *backdoors* (Quadro 4.5).

Já que o software livre/código aberto permite a inspeção do software, por razões de segurança, muitos governos certamente devem cogitar a possibilidade de adotar sistemas livres como regra local. Essa provavelmente será a tendência futura, de mais longo prazo. A questão é como caminhar para obter vantagens com a sua rápida implementação, sem que o direcionamento ocorra por imposição.

#### QUADRO 4.5 - BACKDOORS PROPRIETÁRIOS. VOCÊ SABIA?

- Muitos telefones móveis possuem uma *backdoor* universal.
- A versão chinesa do Android conta com uma *backdoor* universal. Quase todos os modelos de telefones móveis têm uma *backdoor* universal no *chip* do *modem*. Por que Coolpad se preocupa em incluir outra? Por que esta será controlada pela Coolpad.
- O Windows 8 conta com uma *backdoor* para remover *apps* de modo remoto.
- Em virtude das *backdoors* do Windows 8, o governo alemão decidiu que o sistema não era confiável.
- O iPhone tem uma *backdoor* que permite a Apple remover remotamente *apps* que a empresa considera inapropriadas.
- O Android possui uma *backdoor* que permite que Google remova *apps* por meio remoto. A empresa também pode, por conta própria e remotamente, instalar *apps* através do *Google Play*.
- Os dispositivos *Galaxy* da Samsung rodando versões proprietárias do Android vêm com uma *backdoor* que permite acesso remoto para os arquivos armazenados no dispositivo.
- O *Kindle* da Amazon possui uma *backdoor* que tem sido utilizada para apagar livros remotamente. O *Kindle* também conta com uma *backdoor* universal.
- Aplicações de armazenagem da HP que usam o sistema operacional proprietário *Left Hand* têm *backdoors* que fornecem para a HP acesso a elas através de *login* remoto. A HP sustenta que elas não permitem à empresa ter acesso aos dados do usuário, mas se a *backdoor* permite instalar mudanças no software, poderia ser possível instalar mudanças que permitiriam este acesso.
- Muitos modelos de roteadores possuem *backdoors*. É o caso, por exemplo, do TP-Link.
- Roteadores da D-Link têm uma *backdoor* para substituição rápida de configurações.
- Existem suspeitas que o FBI solicita a várias empresas norte-americanas que incluam *backdoors* em programas proprietários. Microcódigos incluídos nos microprocessadores Intel e AMD podem ser um veículo para a NSA invadir os computadores com a ajuda da Microsoft, confirmam especialistas respeitados da área de segurança.

[Extraído, com alterações, de <https://www.gnu.org/philosophy/proprietary-back-doors.html>]

#### A intervenção da NSA no ambiente de nuvem também gera apreensões

Há suspeitas, também, sobre as relações da NSA com os provedores de nuvem norte-americanos (Rousseau, 2013). Em virtude de incertezas sobre se dados armazenados em data-centros no país estariam sendo monitorados pelo governo em nome da segurança nacional, vários países europeus vêm sugerindo às suas empresas a não-utilização de provedores de serviços sediados nos Estados Unidos. O desconhecimento sobre a extensão das relações da NSA com os provedores da nuvem, junto com a certeza de que estas relações tendem a ser cada vez mais frequentes, podem trazer, como efeito colateral, prejuízo para os negócios norte-americanos.

Assim, cada vez mais, os governos deverão pensar em soluções próprias para resguardar os seus dados e o dos seus cidadãos. O que coloca o Brasil em uma situação frágil, já que não dispõe de empresas prontas, renomadas e posicionadas no mercado para o desenvolvimento destas soluções. Uma alternativa seria optar por software livre/código aberto, que permite auditabilidade e avaliação da presença de *backdoors*.

#### Adoção de padrões abertos

Apadronização do formato utilizado para o armazenamento e troca de dados é imprescindível, já que, com a adoção crescente da informática, uma quantidade cada vez maior de dados está sendo armazenada em meio eletrônico

Apesar das diferenças entre os defensores do software livre e de código aberto no que se refere ao copy left, ou seja, ao efeito (ou não) de contaminação da licença livre, ambas as comunidades concordam em vários pontos, incluindo a necessidade de defesa do uso de padrões abertos. Vários formatos adotados atualmente são proprietários, fechados e protegidos por patentes. É o caso, por exemplo, dos padrões Flash e MP3, amplamente utilizados para áudio e vídeo, e dos padrões pdf e doc, difundidos para texto.

Em nível global, várias organizações (por exemplo, *Open Document Foundation*, *Xiph Foundation*, *Open Format*) vêm defendendo a bandeira de uso de padrões abertos. A adoção do formato aberto torna os ambientes e arquivos compatíveis, reduzindo a possibilidade de perda de dados e facilitando a substituição de um determinado fornecedor por outro. Além disso, o formato aberto fornece ao usuário o direito de acesso aos seus arquivos pessoais e ao conteúdo, sem que para isto seja necessário o consentimento prévio da empresa fornecedora de ferramentas para a geração de mídia.

Seja qual for o modelo adotado de distribuição de softwares, o uso de padrões abertos é uma tendência natural, irreversível, no cenário da tecnologia da informação. Padrões abertos são padrões disponíveis para livre acesso e implementação, que independem de *royalties* e outras taxas e sem discriminação de uso.

Pelo Artigo 2 do Protocolo Brasília, entende-se por formatos abertos de arquivos aqueles que: possibilitam a interoperabilidade entre diversos aplicativos e plataformas, internas e externas; permitem aplicação sem quaisquer restrições ou pagamentos de *royalties*; podem ser implementados plena e independentemente por múltiplos fornecedores de programas de computador, em múltiplas plataformas, sem quaisquer ônus relativos à propriedade intelectual para a necessária tecnologia.

Em relação aos padrões fechados, os padrões abertos trazem uma série de vantagens para o usuário. Garantem, por exemplo, a continuidade e longevidade dos dados (sejam textos, vídeos ou áudios), permitindo a abertura dos arquivos em que os conteúdos se encontram armazenados mesmo após muitos anos. Além disso, assegura a independência em relação aos fornecedores de aplicativos, fazendo com que a concorrência entre eles passe a se basear em qualidade técnica e funcionalidade. E, também, reduz os riscos inerentes aos fatores de mercado, que podem levar ao fechamento da empresa fornecedora de um dado aplicativo em uso ou a sua substituição por outro.

O uso de padrões abertos é especialmente indicado para as instituições públicas, pois permite a sua independência de um dado fornecedor e fornece aos cidadãos a possibilidade de interagir com o governo através de um formato eletrônico que aceita as suas escolhas pessoais. Para as organizações privadas, a adoção de padrões abertos permite o estabelecimento de uma base comum de comunicação e interoperabilidade. Para que se conquiste a interoperabilidade, as pessoas precisam se engajar em um esforço contínuo que faça com que sistemas, processos e culturas sejam gerenciados e direcionados para maximizar oportunidades de troca e reuso de informações.

As instituições públicas, em especial, devem cogitar a possibilidade de adotar padrões abertos como regra, considerando as vantagens que possuem em relação aos formatos fechados: redução dos custos envolvidos com o pagamento de *royalties* por uso de algoritmos proprietários e controle maior concedido ao usuário final. A decisão, no entanto, certamente conta com a resistência das grandes empresas do setor de TIC que atualmente usufruem dos benefícios com a manutenção dos formatos fechados, já que os padrões abertos afetam o seu modelo tradicional de negócios, baseado em patentes e *royalties*.

Embora a tendência mostre-se cada vez mais favorável ao emprego de formatos abertos, e este provavelmente será o resultado irreversível a ser alcançado em mais longo prazo, a questão é como, em mais curto prazo, reverter o cenário atual em que os formatos proprietários e fechados ainda são predominantes. E como fazer esta passagem, de formatos fechados para padrões abertos, sem que esta seja uma decisão imposta pelo governo, à revelia dos cidadãos.

O uso de padrão aberto não quer dizer, necessariamente, que o software tenha que estar sob licença livre ou de código aberto. É possível adotar padrão aberto mesmo utilizando sistemas operacionais proprietários. O fato de o formato ser aberto não quer dizer que os dados em si estarão amplamente disponíveis para consulta. O sigilo ocorre no formato aberto. O que significa, fundamentalmente, é que os usuários terão maior controle sobre os seus dados e liberdade para acessá-los utilizando qualquer sistema operacional. No Quadro 4.6 resumem-se as várias possibilidades de combinação entre o tipo de sistema operacional (livre/código aberto ou proprietário) e as possibilidades no que se refere ao padrão (aberto ou fechado). Na nova discussão, portanto, o foco deixa de ser o software enquanto tal e passa a ser o padrão a ser adotado para armazenamento de dados (textos, vídeos ou áudio).

O Governo poderia utilizar produtos apenas em formato de dados abertos, seja assumindo a posição do quadrante 1 ou do quadrante 3 (Quadro 4.6). Também deveria visitar o modelo utilizado em compras públicas, insistindo na adoção dos padrões abertos.

**QUADRO 4.6 – POSSIBILIDADES DE COMBINAÇÃO ENTRE TIPO DE SOFTWARE (SISTEMAS OPERACIONAIS) E PADRÃO DE DADOS**

	Software livre/código aberto	Software proprietário
Padrão aberto (html, LaTeX)	1. Software livre + padrão aberto	3. Software proprietário + padrão aberto
Padrão fechado (doc, mp3, xls, flash, etc.)	2. Software livre + padrão fechado	4. Software proprietário + padrão fechado

As comunidades brasileiras de software livre/código aberto têm assumido um papel relevante no sentido de apoio ao uso de padrões abertos. Mas esta é uma bandeira que poderia ser carregada por outras entidades de apoio ao setor brasileiro de TIC.

**Dados abertos: mais liberdade para todos**

*Dados abertos: Quem irá deter a informação?*

No novo cenário, não apenas as pessoas estarão *online*. Mais e mais objetos - como carros, monitores cardíacos e sensores - estarão conectados à Internet, contribuindo para um fluxo crescente de dados armazenados na nuvem, em bancos de dados gigantescos que, no momento, estão sendo construídos por *players* poderosos da *web*. Quem irá ter acesso à informação?

Google e Facebook, por exemplo, já contam atualmente com uma quantidade relevante de dados sobre as buscas realizadas pelas pessoas, os seus amigos, sua agenda de compromissos, endereços, telefones, senhas e mensagens enviadas. No futuro, a tendência é que saibam muito mais, pois as informações que hoje ficam em bancos de dados separados serão unificadas e integradas em um único arquivo. Essas informações, em princípio, só deveriam ser acessadas com autorização judicial. No entanto, sempre haverá a possibilidade (e o temor) de vazamentos e espionagens.

O manuseio das informações se tornou um bem de valor inestimável para as grandes empresas e para a construção dos negócios do futuro. Muitos deles irão se basear, justamente, na geração de valor através da oferta de informações combinadas sobre pessoas, locais, empresas, etc.

O conceito de dados abertos aplica-se a todo o conjunto de dados que podem ser publicados na *web*, e não apenas aos dados governamentais. Porém, os dados do governo, pelo volume e pela relevância para a sociedade, são os que têm sido o foco principal das iniciativas nessa área.

A mera liberação dos documentos não torna eficaz o processo de publicidade dos dados, pois pode se tornar uma tarefa complexa o manuseio da quantidade significativa de documentos gerados. A democratização e transparência dos dados da administração pública é louvável. No entanto, a escolha de formatos apropriados para coleta, estruturação e exposição dos dados via Internet é fundamental para que esse processo ocorra com a agilidade e a eficiência necessárias.

O objetivo dos dados públicos abertos é a maior participação da sociedade na vida pública, por meio de documentos em formatos de fácil manuseio por humanos e por máquinas inteligentes. O reuso dos dados públicos pode garantir maior transparência, participação e colaboração.

O conceito de dados abertos propõe que todos, sem restrições, tenham acesso aos dados disponíveis. Para facilitar este acesso, eles deveriam ser armazenados em formatos diferentes (ampliando o leque de possibilidades para os usuários) e em padrão aberto, para evitar dependência tecnológica. Ferramentas e especificações técnicas para facilitar o entendimento e a análise dos dados devem ser colocadas à disposição. Os dados precisam estar estruturados e organizados de modo a facilitar o manuseio por humanos ou máquinas inteligentes, permitindo a integração rápida.

A discussão sobre a proteção aos dados pessoais é assunto que interessa a todos, e não apenas às comunidades de software livre/código aberto. A ideia por trás do conceito também pressupõe o seguinte:

- Quanto mais abertos os dados, maiores ganhos econômicos e/ou sociais, pois muitos poderão conhecê-los, reutilizá-los, gerar negócios e propor inovações.
- Quanto mais abertos os dados, maior controle e segurança para os usuários, pois ninguém poderá utilizá-los com exclusividade.
- O reuso adequado dos dados requer uma ação consciente por parte do usuário. A sua capacitação é fundamental para o êxito dos objetivos de colaboração, participação e transparência.

## 4.4 COMUNIDADE BRASILEIRA DE SOFTWARE LIVRE/CÓDIGO ABERTO: PONTOS FORTES E FRACOS, OPORTUNIDADES E AMEAÇAS E UM PLANO PARA AÇÃO

### CENÁRIO E ANÁLISE SWOT

A seguir, apresentam-se cenário, pontos fortes e fracos, oportunidades e ameaças para a construção/manutenção de um ecossistema em software livre/código aberto no Brasil. A discussão baseia-se em consulta a especialistas, reunidos em evento realizado em dezembro de 2014. No Quadro 4.7, resume-se a análise SWOT.

#### O cenário

No novo paradigma em construção, baseado em ambiente de mobilidade, o Android, sistema operacional derivado do Linux, está presente em um sem número de novos dispositivos. No entanto, o uso disseminado do Android e de outras plataformas livres não garantiu aos usuários, como era de se supor, um controle maior das tecnologias e tampouco assegurou as liberdades para acesso, reprodução e alteração do software. As tecnologias passam a ser utilizadas, cada vez mais, como motor do processo de acumulação do capital, apesar da implementação de modelos de negócios inovadores e ofertas de muito baixo custo. Parte ainda significativa das empresas de capital nacional continua gravitando em torno de plataformas líderes estrangeiras. Apesar disso, muitas delas, cada vez mais, utilizam sistemas operacionais e ferramentas de software livre e código aberto.

*As mudanças afetam o modelo de negócios das empresas brasileiras de software estejam elas vinculadas ao software proprietário, estejam elas ligadas aos ideais de software livre/código aberto.*

As novas tecnologias afetam de modo significativo o modelo típico de negócios em software livre/código aberto, muito amparado, ainda, em serviços de desenvolvimento de software sob demanda do governo e revenda, consultoria e suporte técnico aos usuários de Linux. Também provocam impactos relevantes no modelo de negócios das empresas brasileiras de software e serviços de TI que historicamente vêm gravitando em torno de plataformas proprietárias líderes em cenário dominado pelos computadores pessoais.

*Surgimento de novos negócios.*

Apesar de destruir uma série de negócios tradicionais, o novo ambiente cria um leque amplo de oportunidades direcionadas para a oferta de serviços diversos habilitados por software, em especial, *web services*. Considerando que as plataformas serão abertas e os usuários terão acesso a serviços, a diferença entre licença livre e proprietária tenderá a perder a relevância. O usuário terá as suas decisões baseadas, sobretudo, na qualidade dos serviços, no atendimento às suas necessidades e no custo final. Os efeitos de aprisionamento a uma dada plataforma serão atenuados.

O uso intensivo de *tablets* e *smartphones* no ambiente de trabalho irá trazer Linux, através da versão Android, para o ambiente corporativo. Mas as *apps* que rodarão sobre Android provavelmente continuarão sendo, em sua maioria, proprietárias, permitindo ganhos para o desenvolvedor a cada *download* realizado.

## Pontos fortes

O Brasil conta com uma comunidade de software livre/código aberto madura, reconhecida em nível internacional, com casos variados de sucesso e uma carteira abrangente de projetos relevantes. A comunidade demonstra ter convicção, criatividade, ânimo e persistência.

O país é pioneiro mundial em software público e governo eletrônico. O Portal do Software Público Brasileiro representa um legado inestimável e pode ser a base sobre a qual reinventar os próximos passos.

## Pontos fracos

O Brasil ainda possui uma cultura favorável ao software proprietário, com resistência forte do usuário à adoção de novos modelos de negócios e tecnologias. Embora numerosa e reconhecida internacionalmente, a comunidade brasileira de software livre/código aberto ainda se encontra muito vinculada ao setor governamental e às suas iniciativas intermitentes de suporte. Essa comunidade envelheceu e, agora, necessita urgentemente de oxigenação. Precisa captar recursos financeiros e atrair novos talentos. Também necessita criar interações fortes, ainda inexistentes, com a sociedade civil e as instituições do setor de TI.

## Oportunidades de negócios

*Discussões em torno do software livre/código aberto e software proprietário cederam lugar para outras questões ainda mais relevantes: a proteção aos dados individuais, o direito de acesso aos conteúdos e aos dados do governo, as garantias de soberania nacional.*

A possibilidade de acesso e reuso de dados abertos, em especial, de dados públicos, irá criar oportunidades para prestação de serviços relevantes para a população, utilizando modelos de negócios inusitados e sustentáveis, e contribuir para que não só as grandes empresas de TI (ou, eventualmente, também os governos dos países a que elas pertencem) detenham informações sobre as pessoas e organizações públicas e privadas.

A comunidade livre/código aberto terá um papel relevante a desempenhar no conjunto todo novo de discussões que agora ganha peso, incluindo temas tais como a colaboração no desenvolvimento e a participação maior dos usuários na concepção dos produtos e serviços; o acesso ao software, cada vez mais encapsulado em equipamentos e dispositivos proprietários; o acesso ao conteúdo e aos dados que circulam nas redes; a utilização de padrões abertos; e o emprego de dados abertos; e a proteção aos dados pessoais. No novo cenário, é muito provável que o usuário tenha mais poder para decidir o que adquirir, se vale a pena pagar ou não para contar com serviços e produtos e se deseja ou não colocar os seus dados à disposição. A realidade necessariamente será multiplataforma: diferentes modelos de negócios, produtos e serviços irão disputar o mercado.

Que modelo poderia ser utilizado por empresas de software livre/código aberto? Entre as opções estão a oferta de *web services* direcionados para as várias cadeias produtivas, o desenvolvimento de *apps* e a prestação de serviços facilitados por software ou soluções envolvendo hardware e software. Esses negócios irão requerer uma reorientação dos desenvolvedores das comunidades livres/código aberto para projetos

de desenvolvimento e serviços de TI menos relacionados com o software de infraestrutura e *middleware* e mais próximos das necessidades do usuário final. Para trilhar este caminho, a comunidade terá de tirar o foco exclusivo dos aspectos técnicos e adquirir *expertise* em negócios e mercados.

### Ameaças

Ao contrário dos computadores, máquinas abrangentes que fornecem aos usuários a possibilidade de realizar várias tarefas, os novos dispositivos restringem as opções das pessoas a um leque de ações predefinido. Assim, por exemplo, ao adquirir uma plataforma para jogos (um *Playstation*, por exemplo), o usuário perde a possibilidade de acessar jogos *online*, ficando aprisionado ao conjunto de opções fornecidas pela fabricante. O usuário controla pouco o equipamento, e tem a liberdade restringida para inserir o que quiser nele. A vinculação entre hardware e software, no futuro, pode se constituir em uma ameaça.

Existe um risco de que as comunicações também se tornem cada vez mais centralizadas. Não será possível realizar uma pesquisa, pois os resultados da pesquisa já serão fornecidos de antemão, pelo sistema. Não será mais possível extrair informações, pois o acesso será predefinido. O controle dos dados e das informações estará nas mãos de alguns poucos. A capacidade de programar os dispositivos poderá ser restringida.

### QUADRO 4.7 – ANÁLISE SWOT – PONTOS FORTES E FRACOS, OPORTUNIDADES E AMEAÇAS PARA SOFTWARE LIVRE/CÓDIGO ABERTO NO BRASIL

PONTOS FORTES	PONTOS FRACOS
<p>Existência de uma comunidade madura, com casos de sucesso e desenvolvimento de projetos de interesse. Legado relevante.</p> <p>Existência do Portal do Software Público Brasileiro que conta atualmente com mais de 200 mil usuários.</p> <p>País é pioneiro mundial em software público e governo eletrônico.</p> <p>Convicção + criatividade + ânimo + persistência</p>	<p>Cultura forte favorável ao software proprietário com resistência à adoção de novos modelos/tecnologias.</p> <p>Diluição e desgaste da comunidade. Necessidade de oxigenação: atrair um número maior de participantes.</p> <p>Interação fraca com a sociedade civil e as instituições do setor.</p> <p>Inexistência de marcos legais. Apoio intermitente do governo.</p> <p>Falta de recursos para financiamento das iniciativas das comunidades (inexistência de apoio das grandes corporações).</p> <p>Ausência de recursos humanos com <i>expertise</i> em software livre/código aberto.</p> <p>Desconhecimento do potencial da comunidade (quantos somos, o que fazemos, etc.) e pouca reflexão sobre as mudanças em curso.</p>
OPORTUNIDADES	AMEAÇAS
<p>Serviços de orquestração de <i>apps</i> na nuvem + desenvolvimento de <i>apps</i> + <i>web services</i> e outros serviços de TI com uso de dados abertos (especialmente dados públicos).</p> <p>Legados da comunidade livre/código aberto + cenário de crescimento do uso das TICs especialmente entre pequenas e microempresas + demandas reprimidas:</p>	<p>Perpetuação de novas formas de aprisionamento: dispositivos que impedem o controle do usuário e a programação pelos desenvolvedores; máquinas zumbis (ameaça para empresas desenvolvedoras de software aplicativo, proprietário ou livre).</p> <p>Perda de controle de máquinas e dispositivos tornando-se algo natural.</p>

<p>oportunidades para o suporte às cadeias produtivas nacionais.</p> <p>Fim do software proprietário: modelos cada vez mais baseados no compartilhamento e na colaboração.</p> <p>Adoção de padrões abertos.</p> <p>Surgimento de uma nova geração de microempreendimento dedicada ao desenvolvimento de software livre/código aberto.</p>	<p>Dados e informações nas mãos de poucos.</p> <p>Estupidificação do consumidor.</p> <p>Espionagem em massa; guerra cibernética, fim da proteção/desconfiança generalizada.</p> <p>Incapacidade para reter talentos.</p>
--	--

## DESENVOLVENDO UM PLANO DE AÇÃO

### Ações das comunidades livres em prol do crescimento e fortalecimento do ecossistema livre/aberto

As comunidades brasileiras de software livre/código aberto precisarão se organizar para se tornar uma opção concreta para a geração de trabalho, renda e riqueza, contribuindo de modo eficaz para o desenvolvimento do país. Nessa linha, valem algumas recomendações:

- Selecionar segmentos para atuação como, por exemplo, saúde, educação, agronegócios, energia, segurança pública, etc.

- Criar empresa âncora ou organização privada capaz de concentrar profissionais e pequenas e médias empresas em torno de desafios específicos levantados por demandantes públicos e privados dos segmentos selecionados, criando um leque de aplicações e serviços relacionados, articulados de modo colaborativo visando à oferta de um portfólio abrangente em nível nacional. As soluções devem ser fortemente direcionadas para as novas tecnologias (mobilidade, *big data*, computação em nuvem, Internet das coisas, etc.) e tendências de negócios. Entre outras atividades, caberá à empresa/organização a ser criada:

- Reinventar e oxigenar a comunidade de software livre/código aberto, direcionando-a para os temas atuais e para a busca de novos modelos de negócios.
- Ampliar o diálogo com grandes comunidades internacionais (Canonical, Mozilla, Suse, CloudStack, Pentaho, PostgreSQL, Linux, KDE, Gnome, etc. ) e associações locais do setor.
- Buscar investidores parceiros, públicos e privados, para a viabilização de uma rede nacional de serviços em software livre.
- Estimular o surgimento de *startups* voltadas para a fabricação de hardware livre (o software estará cada vez mais embarcado, fazendo parte de um sistema eletrônico, rodando em um equipamento específico).

- Criar um núcleo de inteligência em software livre/código aberto no âmbito da empresa âncora, com os seguintes objetivos, entre outros: levantamento e divulgação de dados e informações sobre o segmento de software livre/código aberto; acompanhamento e divulgação de tendências e cenários em software livre/código aberto; prospecção de mercados específicos e identificação de oportunidades de negócios;

catalogação das soluções existentes e construção de portfólio de serviços a serem ofertados no mercado; estabelecimento de articulações empresariais; definição de padrões de qualidade na prestação de serviços; elaboração de mecanismos de certificação ou de melhores práticas; constituição de técnica e de redes de qualificação para o empreendedorismo; divulgação dos casos de sucesso.

- Propor grade curricular para formação de recursos humanos em software livre/código aberto, voltada, inclusive, para a aprendizagem da programação nas escolas de ensino básico.

- Manter, na pauta de discussão, os temas que poderiam ampliar as chances de desenvolvimento de uma nova economia e de uma nova sociedade mais justa e fraterna, incluindo questões voltadas para a propriedade intelectual e os direitos do autor, o uso de padrões abertos, a proteção dos dados pessoais e a colaboração no desenvolvimento.

- Incentivar a bandeira: todos a favor de um plano nacional de aceleração do crescimento com apoio das TICs.

### **Ações do Estado em prol do ecossistema livre/aberto**

O Estado pode contribuir para o crescimento e fortalecimento da comunidade brasileira de software livre/código aberto, permitindo o surgimento de um ecossistema digital pujante, voltado para o uso das novas tecnologias e para o atendimento mais ágil às necessidades da população, incentivando o protagonismo das empresas e comunidades. Seguem sugestões para as ações de suporte:

- Conceber e implantar infraestrutura em telecomunicações necessária para deslançar a Internet do futuro, coordenando discussão ampla sobre protocolos, padrões e formatos abertos e sobre aspectos legais relacionados com o uso de novas tecnologias.

- Posicionar-se como aliado, e não protagonista, de ações envolvendo o desenvolvimento de soluções em TI, estimulando o uso de padrões abertos adotados em nível global e o uso de dados públicos.

- Liberar os dados públicos, permitindo ampliar o portfólio de serviços que as empresas e comunidades podem prestar à população, ao governo e à iniciativa privada.

- Incentivar o surgimento de plataformas sociais, baseadas na adoção de padrões abertos, que estimulem as empresas (proprietárias e de software livre/código aberto) a participar ativamente no desenvolvimento colaborativo e consorciado de soluções para os grandes desafios nacionais.

- Rever o conceito de repositório público, deixando sob a gestão profissional de organizações da sociedade civil, os atuais repositórios públicos de software.

- Incluir empresas e profissionais de TI como parceiros na especificação e no desenvolvimento de software que atualmente é realizado por equipes *in house*, criando oportunidades futuras para que produtos e serviços gerados de modo colaborativo se tornem uma opção de negócios para muitos.

- Utilizar as compras públicas como mecanismo para incentivar a manutenção de um ecossistema pujante, protagonizado pelas empresas e organizações.

- Incentivar a bandeira: todos a favor de um plano nacional de aceleração do crescimento da economia com apoio das TICs.

## CONSIDERAÇÕES FINAIS

Como ocorre em todo processo de mudança de paradigma, o cenário é pleno de incertezas. Existe uma tensão ainda não resolvida entre apropriação individual e colaboração e compartilhamento. É difícil prever como será o futuro e como os diferentes atores irão responder ao conjunto de novas questões inquietantes. A seguir faz-se um prognóstico.

1 – No novo cenário em construção, as plataformas tecnológicas (incluindo software de infraestrutura e *middleware*) terão chances elevadas de serem em software livre/código aberto. Existem vários fatores que corroboram esta tendência: a supremacia do Android no mercado de sistemas operacionais para dispositivos móveis; as alianças reunindo um número cada vez maior de empresas buscando contribuir para o desenvolvimento de plataformas abertas; e o anúncio recente da Microsoft que irá abrir o código do Windows (Quadro 4.8).

### QUADRO 4.8 – MICROSOFT ANUNCIA INICIATIVAS EM DIREÇÃO A UMA CULTURA DE CÓDIGO ABERTO

Há muitos sinais de que a cultura da Microsoft tem mudado nos últimos anos, especialmente depois que Satya Nadella assumiu o posto de CEO. E novas afirmações de um engenheiro veterano mostram que as estruturas da gigante do software foram mesmo abaladas: Mark Russinovich disse, durante uma conferência no Vale do Silício, que a empresa discute a possibilidade de tornar o Windows *open source*.

Quem acompanha a história da empresa nas últimas décadas sabe que esse seria um passo enorme – e que provavelmente levaria tempo para se concretizar. A Microsoft se tornou a gigante de hoje vendendo software proprietário, de código fechado, principalmente seu sistema operacional e o pacote Office. Segundo Russinovich, isso pode mudar. “É definitivamente possível”, disse Russinovich durante a conferência Chef Con, em Santa Clara, na Califórnia, que reúne desenvolvedores e especialistas em TI.

Recentemente, a Microsoft já tomou iniciativas em direção a uma cultura de código aberto. No mês passado, a empresa abriu a *engine* de compilação do software MS Build. Em novembro do ano passado, foi a vez do popular *framework* .NET, em um movimento que pode ser considerado ousado, visto o histórico da companhia.

Claro que o projeto de um Windows *open source* exigiria muito mais planejamento. “Se você abre o código de um produto mas ele vem com um sistema que exige cientistas e três meses para configurar, qual a vantagem?” diz o engenheiro. “Todo tipo de conversa que você possa imaginar sobre o que podemos fazer com nosso software – aberto versus fechado versus serviço – já aconteceu.”

O desempenho financeiro da divisão Windows tem sofrido com a queda constante das vendas globais de PCs, como revelaram os resultados divulgados pela companhia. Essa pressão pode acelerar as discussões por novas estratégias de mercado, como o investimento em infraestrutura de nuvem. Inclusive, o Azure, solução da companhia na área, já suporta máquinas rodando Linux. “Esta é a realidade em que vivemos”, diz Russinovich.

Apesar de a discussão chamar a atenção quando se trata de Microsoft, o código aberto é quase uma norma em outras gigantes do Vale do Silício. Basta pensar na Google e no sistema operacional Android, ou no Facebook e sua coleção de tecnologias, de servidores a inteligência artificial, disponíveis com licenças *open source*.

[Fonte: Wired e Microsoft]

2 – O software livre/código aberto será utilizado em combinação com software proprietário, permitindo às grandes empresas criarem soluções próprias e se diferenciarem umas das outras. Os softwares aplicativos continuarão sendo, em sua maioria, softwares proprietários, embora continuará existindo, também, e em menor proporção, opções de aplicativos em software livre/código aberto. Existem fatores que corroboram esta tendência: a fragmentação do Android em várias versões específicas, mesclando módulos de código aberto com módulos proprietários; os modelos de negócios que valorizam as patentes e os direitos do titular do software; a convergência entre software e hardware, reduzindo as chances de o usuário decidir o que rodar em seu dispositivo.

3 – É muito provável que a nova economia em construção não se torne, de fato, e no sentido literal da expressão, uma nova economia. Uma nova economia iria requerer uma mudança significativa na tríade capital, propriedade e trabalho. A produção, geração e acumulação de riqueza nos moldes atuais não dizem respeito a mudanças econômicas e sim a alterações de natureza tecnológica que afetam os modelos organizacionais, negociais, trabalhistas e interpessoais. Assim, seria mais adequado tratar essas mudanças como novos modelos empresariais ou novas oportunidades de negócios envolvendo inovação tecnológica do que como um novo modelo econômico (Meffe, 2010).

Para o surgimento de uma Nova Economia, quatro condições deveriam ser alcançadas (Meffe, 2010): o software deveria se tornar um bem intangível através do uso sob licença livre; deveria ser criado um novo marco regulatório para a propriedade intelectual; o *locus* produtivo deveria ser alterado, com o ambiente de trabalho sendo estabelecido pelas redes virtuais; o ecossistema de bens intangíveis, ressaltando a colaboração ao invés da competição, precisaria ser estruturado (Quadro 4.9).

Compartilhando a mesma opinião de que o que se tem, atualmente, é mais do mesmo, Alphen (2014) lembra que os novos empreendedores, as tecnologias inovadoras e as ideias revolucionárias atuais de produção e gestão acabam por perpetuar o jeito antigo de fazer as coisas. O que mudou foi só a velocidade: lucro mais rápido, crescimento mais rápido, participação de mercado mais rápida, exacerbação do trabalho mais rápido. Muito raro, ver no empreendedorismo do século XXI, o respeito ao tempo livre, ao meio ambiente e à saúde física, mental e moral de todos, funcionários e clientes, complementa.

**QUADRO 4.9 - A VERDADEIRA NOVA ECONOMIA VAI CRIAR CONDIÇÕES PARA UMA DISTRIBUIÇÃO MAIOR DA RIQUEZA E EQUILÍBRIO NAS OPORTUNIDADES PARA OFERTANTES DE SOLUÇÕES E PRESTADORES DE SERVIÇO**

O futuro dos negócios agora é grátis. Como avaliar, hoje, a ampliação dos negócios gratuitos? Uma justificção para a gratuidade poderia ser apenas através dos ganhos financeiros indiretos e paralelos, como se a intenção sempre fosse ganhar dinheiro, ou seja, a ideia de que não existe almoço grátis, que alguém necessariamente estará pagando a conta e lucrando com este estado das coisas. Entretanto, esta visão deixaria uma sombra sobre o que existe de mais inovador nesta virada de século: o emergir da economia dos bens intangíveis, que potencializa um fluxo não necessariamente comercial de geração de riqueza.

A Nova Economia do final do século XX não trouxe nada de muito novo do ponto de vista econômico, mas sim tecnológico. A Nova Economia foi o termo adotado para definir a fase mais avançada do capitalismo, onde a internet tem um aspecto central. Nesta fase todos os agentes econômicos têm acesso a todas as informações, criando teoricamente um nível de competitividade maior. Muito importante, neste contexto, fazemos uma linha divisória: a Nova Economia tem

uma relação umbilical com as TICs e bem mais distante de um modelo econômico que justifique a sua adjetivação de “nova”.

Mas por que afinal a Nova Economia não tem elementos que justifiquem a sua adjetivação? Inicialmente, os bens e serviços que se relacionam neste ambiente econômico “inovador” são marcadamente tangíveis e intangíveis – algo já existente na economia. O computador, a impressora e a câmara digital são tangíveis; a informação, o dado e o software são intangíveis. Assim, a Nova Economia composta em grande parte pelo mercado da Tecnologia da Informação e Comunicação tem sustentação em um conjunto de bens tanto tangíveis quanto intangíveis. Neste caso não temos nada de diferente às outras fases do capitalismo moderno.

A questão fundamental é que a produção, geração e acumulação de riqueza têm como base de sustentação modelos tradicionais voltados para bens tangíveis e artificialismos relacionados aos bens intangíveis. Um destes artificialismos, por exemplo, é a licença proprietária de software. A licença nada mais é do que tornar rival um bem que não o é por natureza (bens rivais são aqueles que impedem que duas pessoas usem/consumam um determinado bem ao mesmo tempo), tendo como uma das justificativas para adoção de licença proprietária o retorno de investimentos ao criador ou mantenedor do software.

Em função disso, um conjunto de bens intangíveis alimentam mais uma cadeia de produção tradicional do que um modelo que promova a sustentação da Revolução Tecnológica e Informacional. Este conceito, abordado por Castells, compreende o período histórico caracterizado por inovações, dentre elas a possibilidade de construções de redes sociais compartilhadas, constituídas por conexões múltiplas e não sequenciais – aquela que permite, por parte do usuário, a produção simultânea e colaborativa de informação e conhecimento no ciberespaço. Castells alerta que o atual modelo de produção de bens intangíveis passa a alimentar uma herança do processo de industrialização clássica – no que o autor chamou de capitalismo informacional – do que criar novas oportunidades no mundo virtual e no contexto da produção colaborativa em redes.

Se a Nova Economia mantém os antigos modelos de produção, das relações de trabalho e da apropriação e distribuição de riqueza, significa dizer que a economia dos bens intangíveis ainda não alcançou a sua plenitude e, por consequência, o emprego do termo Nova Economia é inadequado.

Para atingirmos uma verdadeira Nova Economia algumas etapas precisam ser perseguidas, sendo quatro delas abaixo elucidadas: tornar o software intangível através da licença livre; novo marco regulatório para a propriedade intelectual; alteração do locus produtivo: ambiente de trabalho estabelecido pelas redes virtuais; e estruturação do ecossistema de bens intangíveis: colaboração ao invés da competição.

#### **Nova economia, novas vantagens competitivas**

Nesta outra Nova Economia, conceitos subjetivos relacionados aos bens e serviços têm um espaço fundamental na sustentação do modelo; certamente serão responsáveis pela nova fórmula de composição final do preço, como a qualidade, a segurança, a confiança, a pontuação do cliente, a transparência, a meritocracia. Estes, parafraseando Porter, serão “as vantagens competitivas” da economia futura.

[Extraído, com alterações, de Meffe, 2010].

4 – A interoperabilidade será cada vez mais necessária, o que irá estimular o uso de padrões abertos, permitindo o cruzamento de dados e a combinação de serviços de diferentes fornecedores. Os fatores que corroboram esta tendência dizem respeito ao interesse demonstrado pelos usuários de tomar decisões sobre os produtos e serviços que querem consumir, o que conflua com os interesses das empresas de trabalharem com padrões fechados. Apesar da existência de soluções proprietárias, será mais fácil para o usuário transitar entre as várias alternativas disponíveis, já que os padrões serão abertos. Como resultado da facilidade maior que o usuário terá de trocar de fornecedor de TI, os atributos técnicos e funcionais do software (ou do conjunto constituído por uma combinação de softwares e hardwares) passarão a se constituir em fatores cada vez mais decisivos na escolha por um dado fornecedor.

5 – Já que a tomada de decisão irá recair sobre os usuários, a sua capacitação será cada vez mais crucial para que escolhas conscientes possam ser realizadas. Assim, se coloca na ordem do dia, a educação para o uso das tecnologias e a capacitação de recursos humanos para o desenvolvimento de soluções com atributos relevantes.

6 – A participação de empresas brasileiras na indústria global de TI deverá continuar muito localizada no *dowstream* da cadeia de valor de software, ou seja, no desenvolvimento de software do tipo aplicativo e na oferta de serviços de TI (no futuro, em especial, *web services*). Os seguintes fatores corroboram esta tendência: inexistência de empresas locais com vocação para desenvolvimento de hardware e participação modesta das empresas brasileiras no desenvolvimento de software de infraestrutura e *middleware*.

7 – O ecossistema de software livre/código aberto conta atualmente com uma quantidade significativa de software de infraestrutura e *middleware*. As comunidades livres também poderiam, agora, se voltar para o desenvolvimento de aplicativos e para a oferta de serviços capazes de desempenhar um papel relevante nas várias cadeias produtivas locais, contribuindo para a melhoria das condições de vida da população e propiciando ganhos de produtividade e competitividade em setores diversos, incluindo saúde, educação, agronegócios, energia, segurança e administração pública.

Já existem soluções livres ou em código aberto para a maior parte das demandas usuais, que podem servir de base para um conjunto maior de respostas eficientes em tecnologia para a sociedade, em todas as áreas de atuação humana. Além disso, pela possibilidade de modificação do código de programas, aplicações e sistemas livres, é possível qualificá-los ainda mais, corrigindo e complementando suas funcionalidades, de maneira coletiva para o compartilhamento de todos.

# REFERÊNCIAS BIBLIOGRÁFICAS

- ALPHEN, F. “Startups perpetuam o jeito antigo”, 2014. In: <http://webinsider.com.br/2014/09/02/startups-perpetuam-o-jeito-antigo/>, acessado em 02 de setembro de 2015.
- APGAUA, R. “Linux e a perspectiva da dádiva”. *Horizontes Antropológicos*, ano 10, n. 21, 2004.
- ARTHUR, W. “Competing technologies, increasing returns, and lock-in by historical events”. *Economic Journal*, 97, pp. 642-65, 1990.
- AUGUSTO, P. “Um estudo sobre as motivações e orientações de usuários e programadores brasileiros de software livre”. Dissertação de mestrado. Rio de Janeiro, UFRJ/COPPEAD, 2003.
- BACON, J. *The art of community: building the new age of participation*. O` Reilly, 2009.
- BENKLER, Y. “Coase’s Penguin or Linux and the nature of the firm”. *The Yale Law Journal*, 369, 2002.
- BORRUS, M. e Zysman, J. “Wintelism and the changing terms of global competition: prototype of the future?”. *Brie Working Paper*, 96B, pp. 1-23. Fevereiro, 1997.
- BUGHIN, J.; Chui, M.; Manyika, J. “Ten IT-enabled business trends for the decade ahead”. *Mc Kinsey Quartely*, May, 2013.
- CASTELLS. M. *Sociedade em rede: a era da informação: economia, sociedade e cultura*, V. 1. São Paulo: Editora Paz e Terra, 1999.
- CELLAN-JONES. “Qual é o Futuro da Internet?” *BBC Brasil - Super potência*, março, 2010.
- CETIC.br/CGI.br - Pesquisa TIC Governo Eletrônico, 2013.
- \_\_\_\_\_ - Pesquisa TIC Educação, 2013.
- \_\_\_\_\_ - Pesquisa TIC Empresas, 2013.
- COLEMAN, E. “The political agnosticism of free and open source software and the inadvertent politics of contrast”. *Anthropological Quartely*, 2005.
- COMCIÊNCIA. Liberdade para o software? Junho, 2004.
- COMERFORD, R. “The path to open-source systems”. *IEEE Spectrum*, maio, 1999.
- CUSUMANO, M. e Gawer, A. “Driving high-tech innovation: the four levers of platform leadership”. Paper 152, Center for Business@MIT, october, 2001.

DIBONA, C.; Ockman, S.; Sonte, M. (orgs.). *Open sources: voices from the open source revolution*. Sebastopol: O`Reilly, 1999.

EVANGELISTA, R. “O movimento software livre do Brasil: política, trabalho e *hacking*”. *Horizontes Antropológicos*, v. 20, n. 41, 2014.

\_\_\_\_\_. Evangelista, R. (2008). “Política e Linguagem nos Debates sobre o Software Livre”. Rev. 12. Acessado em [http://wiki.softwarelivre.org/CoberturaWiki/ Pol%EDticaELinguagemNosDebatesSobre OSoftwareLivre](http://wiki.softwarelivre.org/CoberturaWiki/Pol%EDticaELinguagemNosDebatesSobreOSoftwareLivre), novembro de 2014]

\_\_\_\_\_. “Política e Linguagem nos Debates sobre o Software Livre”. Dissertação de Mestrado. Unicamp, 2005.

FIESP/CIESP e FEA/USP - “I-Digital: perfil da empresa digital 2002/2003”, 2004.

FORTES, D. “O fenômeno do software livre”. *Info*, fevereiro, pp. 57-64, 2004.

GAWER, A. e Cusumano, M. “Industry platforms and ecosystem innovation”. *Journal of Product Innovation Management*. Volume 31, Issue 3, pp. 417-433, May, 2014.

\_\_\_\_\_. *Platform leadership: how Intel, Microsoft and Cisco drive industry innovation*. Harvard Business School Press, 2002.

GAWER, A. e Henderson, R. “Platform owner entry and innovation in complementary markets: evidence from Intel”. NBER Working Paper, n. 11.852, 2005.

GHOSH et al. Deliverable D18: Final Report. Part 4: Survey of developers. Free/Libre and Open Source Software: Survey and Study. Maastricht: International Institute of Infonomics/University of Maastricht and Berlecon Research GmbH, 2002.

GOHRING, N. “BYOD is a gateway for mobile app use”, 2013.

HEINZ, F. “Los desafíos de la comunidad de software libre en Latinoamérica” in *Un mundo patentado? La privatización de la vida y de conocimiento*. Ed. Heinrich Boell, n. 19, 2004.

IDC. “The Linux marketplace - moving from niche to mainstream”. Preparado para o OSDL, dec. 2004.

JACKSON, J. Linux now a corporate beast, 2004.

KEIZER, G. “Microsoft nega a construção de *backdoor* no Windows 7”. *Computerworld*, 19 de novembro, 2009a.

\_\_\_\_\_. “NSA helped with Windows 7 development: privacy expert voices ‘backdoor’ concerns, security researchers dismiss idea”. *Computerworld*, 18 de novembro. In: <http://www.computerworld.com/article/2521809/government-it/nsa-helped-with-windows-7-development>.

MEIRELLES, S. *Tecnologia da Informação*. São Paulo: EAESP/FGV, 2004.

MOLINA, A. *Long-term business revolution in the global software sector? The emergence of free/libre and open source software for e-government*, 2003.

MURILLO, L. “Tecnologia, política e cultura na comunidade brasileira de software livre e código aberto”. Dissertação de mestrado. Instituto de Filosofia e Ciências Humanas. Universidade Federal do Rio Grande do Sul, 2009.

\_\_\_\_\_. “O virtual e o político: a análise de um confronto discursivo da comunidade brasileira de software livre”. Observatório OberCom, Volume 1, Número 2, 2007. In: <http://obs.obercom.pt/index.php/obs/article/view/67>.

PENTLAND, A. Entrevista à Revista Veja, abril, 2015.

\_\_\_\_\_. *Social Physics: How Good Ideas Spread – The Lessons from a New Science*. MIT, 2015

RAYMOND, E. *The cathedral & the bazaar: musings on Linux and open source by an accidental revolutionary*. Cambridge: O’Reilly, 2001.

REIS, C. “Caracterização de um processo de software para projetos de software livre”. Dissertação de mestrado. São Carlos: ICMC/USP, 2003.

RODRIGUES, A. “O Futuro da Internet (e do mundo) segundo Google”. *Revista Super Interessante*, junho, 2013.

ROSENBERG, N. “On technological expectations”. *Economic Journal*, Volume 86, Issue 343. September, pp. 523-535, 1976.

ROUSSEAU. “Will concerns over NSA monitoring harm US cloud providers?”. *IDG News Service*, jul., 07, 2013.

SALEH, A. M. “Adoção de tecnologia: um estudo sobre o uso de software livre nas empresas”. Dissertação de mestrado. São Paulo: FEA/USP, 2004.

SCHMIDT, E. e Cohen, J. *The new digital age: transforming nations, businesses, and our lives*. New York Times, 2013.

SOFTEX. “Pesquisa, Desenvolvimento e Inovação em software e serviços de TI”. *Cadernos Temáticos do Observatório*, volume 3, 2014.

\_\_\_\_\_. e Unicamp. “Impacto do Software Livre e de Código Aberto na Indústria de Software Brasileira”, 2005.

STALLMAN, R. “Por que o código aberto não compartilha dos objetivos do software livre?”, 2014. Acessado em <http://www.gnu.org/philosophy/open-source-misses-the-point.pt-br.html>, novembro de 2014.

\_\_\_\_\_. “What’s GNU? GNU’s not Unix!”, 1985.

TALARICO, S. “BYOD e a consumerização: o que são e como utilizar” – 9 de agosto, 2013.

TAURION, C. “Consumerização: a mudança na dinâmica entre o setor de TI e os usuários” – 12 agosto, 2013.

\_\_\_\_\_. *Software livre: potencialidades e modelos de negócio*. Rio de Janeiro, Brasport, 2004.

TORRES, A. “A tecnoutopia do software livre: uma história do projeto técnico e político do GNU”. Dissertação de mestrado. Faculdade de Filosofia, Letras e Ciências Humanas. Universidade de São Paulo, 2013.

VIEIRA, E. e Nogueira, J. “Pronto para encarar o Linux?”. Info Corporate, Maio, 2003.

VON HIPPEL, G. e von Krogh, E. “Special issue on open source software development”. *Research Policy*, 32: 1.149-1.157, 2003.

W3C. “Manual dos dados abertos: governo. Série Dados abertos governamentais”. São Paulo: Comitê Gestor da Internet no Brasil, 2011.

\_\_\_\_\_. “Manual dos dados abertos: desenvolvedores. Série Dados abertos governamentais”. São Paulo: Comitê Gestor da Internet no Brasil, 2011.

\_\_\_\_\_. “Melhorando o acesso ao governo com o melhor uso da web. Série Dados abertos governamentais”. São Paulo: Comitê Gestor da Internet no Brasil, 2009.

WILLIAMS, S. “Free as in Freedom: Richard Stallman and the free software revolution”, 2002 e 2010. In: <https://static.fsf.org/nosvn/faif-2.0.pdf>

WILLIS, D. “The Economics of BYOD”. Apresentação realizada no Gartner Symposium/ITxpo – Orlando, Fl. - 10 de outubro, 2013.

WOHNOUTKA, B. “ISPS6 - *Inevitable Cloud Migration: Actionable Strategies for Healthcare Providers*”. Apresentação realizada no Gartner Symposium/ITxpo – Orlando, Fl. – 6 de outubro, 2013.

html, 2009b.

LERNER, J. e TIROLE, J. “Some simple economics of open source”. *The Journal of Industrial Economics*, 52, pp. 197-234, 2002.

McMILLAN, R. “NSA ajudou a Microsoft a tornar o Vista seguro: agência contribuiu para fazer o sistema operacional conforme com os requisitos DOD”. *IDG News Service*, jan. 10, 2007. In: <http://www.computerworld.com/article/2549164/security0/nsa-helped-microsoft-make-vista-secure.html>.

MEFFE, C. “O falso milagre da nova economia”. *Websinder*, 31 de março, 2015. In: <http://webinsider.com.br/2015/03/31/o-falso-milagre-da-nova-economia/>

\_\_\_\_\_. “A nova economia será a dos bens intangíveis”. Artigo publicado na Revista Linux Magazine, julho, n. 68, Linux New Media do Brasil Editora Ltda. *Websinder*, 2010. In: <http://webinsider.com.br/2010/11/24/a-nova-economia-sera-a-dos-bens-intangiveis/>

## GLOSSÁRIO DE SIGLAS

ABNT - Associação Brasileira de Normas Técnicas

AOSP - *Android Open Source Project*

B2B – *Business-to-Business*

B2C – *Business-to-Consumer*

BPO - *Business Process Outsourcing*

BSD - *Berkeley System Distribution*

BYOA - *Bring Your Own Apps*

BYOD - *Bring Your Own Device*

C2C – *Consumer-to-Consumer*

CA – Código aberto

CEO – *Chief Executive Officer*

CETIC – Centro de Estudos sobre as Tecnologias da Informação e Comunicação

CGI.br – Comitê Gestor da Internet no Brasil

CIO – *Chief Information Officer*

CISC - *Complex-Instruction Set Computing*

CRM – *Customer Relationship Management*

DRM - *Digital Rights Management*

EPIC - *Electronics Privacy Information Center*

ERP – *Enterprise Resource Planning*

F&A – Fusões e Aquisições

FISL - Fórum Internacional de Software Livre

FSF - *Free Software Foundation*

FLOSS – *Free/Libre Open Source Software*

GPL - *General Public License*

IaaS – *Infrastructure as a Service*

IBGE – Instituto Brasileiro de Geografia e Estatística  
IBSS – Indústria Brasileira de Software e Serviços de TI  
IoT – Internet das Coisas  
ISO - *International Organization for Standardization*  
ITES - *Information Technology Enabled Services*  
ITO - *Information Technology Outsourcing*  
LGPL - *Lesser General Public License*  
MIPS – *For Millions of Instructions Per Second*  
MPL - *Mozilla Public License*  
NIC. Br – Núcleo de Informação e Coordenação do Ponto BR  
NSA - *National Security Agency*  
OASIS - *Organization for the Advancement of Structured Information Standards*  
ODF - *Open Document Format*  
OHA- *Open Handset Alliance*  
OSDL - *Open Source Development Lab*  
OSI – *Open Source Initiative*  
P&D – Pesquisa e Desenvolvimento  
P&D&I – Pesquisa e Desenvolvimento e Inovação  
PaaS – *Platform as a Service*  
PAC – Pesquisa Anual do Comércio  
PAS – Pesquisa Anual de Serviços  
PIA – Pesquisa Anual da Indústria  
PC – *Personal computer/microcomputador* – computador de mesa  
RISC - *Reduced-Instruction Set Computing*  
SaaS – *Software as a Service*  
SISSL - *Sun Industry Standards Source License*  
SL – Software Livre  
SP – Software Proprietário  
Sparc - *Scalable Processor Architecture*  
SUN - *Stanford University Networked*  
SWOT – *Strengths, Weaknesses, Opportunities and Threats*  
TI – Tecnologia da Informação  
TIC – Tecnologias da Informação e Comunicação  
VMS - *Virtual Memory System*

# ANEXOS

## **ANEXO 1 – Formulário da PESQUISA FLOSS (Free, Livre, Open Source Software) BRASIL 2014**

Bem-vindo à pesquisa FLOSS BRASIL 2014!

Essa pesquisa recria, com algumas pequenas alterações, a pesquisa FLOSS desenvolvida em 2002 e 2013 no exterior e em 2003 no Brasil, que tiveram impacto significativo para a comunidade, academia e políticas públicas.

A FLOSS BRASIL 2014 é aberta para todas as pessoas residentes no país que contribuem com o FLOSS, não apenas desenvolvedores. Estará disponível de 1o de setembro de 2014 a 1o de outubro de 2014. Toma 15 minutos para preenchimento.

Os resultados serão publicados. Uma versão não identificada dos dados será fornecida para outros pesquisadores através da licença *Creative Commons*.

O formulário conta com 58 questões.

### Introdução

**1. Você participou da versão FLOSS BRASIL, realizada pela Softex em 2003?**  
Por favor, escolha somente uma das seguintes:

- Sim
- Não

**2. Por favor, selecione a área em que suas contribuições para projetos de FLOSS acontecem:**  
Por favor, escolha apenas uma das seguintes:

- Código, programação
- Outras contribuições (documentação, traduções, testes, etc.)
- Ambas

**3. De que comunidades de software livre/código aberto você participa?**

## Dados pessoais

### 4. Qual o seu gênero:

Por favor, escolha apenas uma opção:

- Masculino
- Feminino

### 5. Situação conjugal

Por favor, escolha apenas uma das opções

- Solteiro
- Casado
- Separado ou divorciado
- Viúvo

### 6. Você tem filhos?

Por favor, escolha apenas uma opção

- Sim
- Não

### 7. Quantos filhos você tem?

Somente responda a esta questão se você respondeu sim à questão 6.

- 1
- 2
- 3
- 4
- 5
- 6
- Mais que 6

### 8. Qual a idade do seu filho?

Somente responda a esta questão se você respondeu sim à questão 6.

No caso de possuir mais de um filho, por favor, considere a idade do caçula.

- Menos de dois anos
- Entre dois anos e 5 anos
- Entre 6 anos e 10 anos
- Mais de 10 anos

### 9. Você é brasileiro?

Por favor, escolha apenas uma opção.

- Sim
- Não

## Dados sobre escolaridade e emprego

### 10. Qual o seu nível de escolaridade?

Por favor, escolha opção de curso completo de maior nível.

- Fundamental
- Médio
- Superior
- Especialização
- Mestrado
- Doutorado
- Pós-doutorado

### 11. Qual o seu nível de inglês?

Por favor, escolha apenas uma opção.

- Básico
- Intermediário
- Avançado
- Fluente
- É minha língua materna

### 12. Qual é a sua profissão?

Por favor, escolha apenas uma opção.

- Engenheiro de software
- Engenheiro (outro)
- Programador
- Consultor de TI
- Consultor (outros setores)
- Executivo (TI)
- Executivo (outros setores)
- Marketing (TI)
- Marketing (outros setores)
- Venda produtos de TI
- Venda outros produtos
- Professor universitário (área de TI)
- Professor universitário (outros setores)
- Estudante de TI
- Estudante outros setores
- Outra (em TI)
- Outra (outros setores)

### 13. Como você descreveria o seu trabalho?

Por favor, escolha apenas uma opção. Não responda a esta questão caso esteja desempregado ou realiza atividade não remunerada.

- É o que eu sempre quis fazer. Eu realmente adoro o meu trabalho.
- É interessante, mas eu posso me imaginar fazendo outros trabalhos que seriam também interessantes.
- Eu ganho um bom salário, isto é o que importa para mim.
- Eu não consigo emprego nas minhas atividades favoritas, mas gosto do que faço.
- Eu não gosto muito do que faço. Se tivesse oportunidade, mudaria de ocupação.
- Eu odeio o meu trabalho.

**14. Em que situação você se enquadra?  
Por favor, escolha apenas uma opção.**

- Trabalhador por conta própria
- Empregado
- Desempregado
- Trabalho não remunerado (estudante)
- Trabalho não remunerado (aposentado, etc.)
- Outro \_\_\_\_\_

**15. Qual o seu rendimento bruto mensal? Por favor, selecione a faixa apropriada.  
Por favor, escolha apenas uma opção.**

- Sem salário
- Menos de R\$ 1 mil
- Mais de R\$ 1 mil a R\$ 2 mil
- Mais de R\$ 2 mil a R\$ 3 mil
- Mais de R\$ 3 mil a R\$ 4 mil
- Mais de R\$ 4 mil a R\$ 5 mil
- Mais de R\$ 5 mil a R\$ 7,5 mil
- Mais de R\$ 7,5 mil a R\$ 10 mil
- Mais de R\$ 10 mil

**15. Qual o seu rendimento bruto mensal? Por favor, selecione a faixa apropriada.  
Por favor, escolha apenas uma opção.**

- Sem salário
- Menos de R\$ 1 mil
- Mais de R\$ 1 mil a R\$ 2 mil
- Mais de R\$ 2 mil a R\$ 3 mil
- Mais de R\$ 3 mil a R\$ 4 mil
- Mais de R\$ 4 mil a R\$ 5 mil
- Mais de R\$ 5 mil a R\$ 7,5 mil
- Mais de R\$ 7,5 mil a R\$ 10 mil
- Mais de R\$ 10 mil

## Questões gerais – Software livre e código aberto

**16. Nas questões seguintes, usaremos sempre o termo software livre e código aberto. Sabemos que muitos desenvolvedores fazem distinção entre estes conceitos. Você se vê como parte da comunidade de software livre ou como parte da comunidade de código aberto? Por favor, escolha apenas uma opção.**

- Eu me identifico como parte da comunidade de software livre.
- Eu me identifico como parte da comunidade de código aberto.
- A distinção não me importa.

**17. Você pensa que as comunidades de software livre e código aberto são diferentes?  
Por favor, somente responda a esta questão se, na resposta à questão 2 você optou pela 1ª ou pela 3ª alternativa. Por favor, escolha apenas uma opção.**

- Sim, não é apenas um modo diferente de pensar, mas um modo diferente de vida.
- Sim, mas somente em seus princípios. O modo como eles trabalham é o mesmo.
- Isso não me importa. Eu apenas programo.

**18. Você pensa que as comunidades de software livre e código aberto são diferentes?**

**Por favor, somente responda a esta questão se, na resposta à questão 2 você optou pela 2ª alternativa. Por favor, escolha apenas uma opção.**

- Sim, não é apenas um modo diferente de pensar, mas um modo diferente de vida.
- Sim, mas somente em seus princípios. O modo como eles trabalham é o mesmo.
- Isso não me importa. Eu apenas trabalho com software.

**19. Quais desses desenvolvedores de software livre/código aberto você conhece (não necessariamente em pessoa)**

**Por favor, escolha todas as opções que se aplicam.**

- Hendrik Breitkreuz
- Miguel de Icaza
- Rémi Denis-Courmont
- Matthias Ettrich
- Paul Gardner
- Luciano Mendietti
- Dru Mortensen
- Murat Ozgobek
- Igor Pavlov
- Bruce Perens
- Eric Raymond
- Noa Resare
- Richard Stallman
- Linus Torvalds
- Jamie Zawinski

## Questões gerais – Desenvolvimento

**20. Eu comecei em...**

**Por favor, selecione o momento em que você começou a contribuir com projetos de software livre/código aberto**  
**Por favor, escolha apenas uma das opções.**

- Antes de 1960
- Entre 1960 e 1969
- Entre 1970 e 1979
- Entre 1980 e 1989
- Entre 1990 e 1999
- Entre 2000 e 2009
- Entre 2010 e 2014

**21. Minha idade então era...**

**Por favor, selecione a faixa de idade em que você se encontrava quando começou a contribuir com projetos de software livre/código aberto. Por favor, escolha apenas uma das opções**

- 10 ou menos
- de 11 a 15
- de 16 a 20
- de 21 a 25
- de 26 a 30
- de 31 a 35
- de 36 a 40
- de 41 a 45
- de 46 a 50
- de 51 a 55
- 56 ou mais

**22. Em média, quantas horas por semana você trabalhou em projetos de software livre/código aberto, durante o ano de 2013?**

**Por favor escolha somente uma das opções**

- Menos de 2 horas semanais
- De 2 a 5 horas semanais
- De 6 a 10 horas semanais
- De 11 a 20 horas semanais
- De 21 a 40 horas semanais
- Mais de 40 horas semanais

**23. Você também desenvolve software proprietário?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2.**

**Por favor, escolha apenas uma opção.**

- Sim
- Não

**24. Você também se envolve, de algum modo, com software proprietário?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2.**

**Por favor, escolha apenas uma opção.**

- Sim
- Não

**25. Quando você começou a desenvolver software proprietário?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2 e pelo sim na questão 23.**

**Por favor, escolha apenas uma das opções.**

- Antes de 1960
- Entre 1960 e 1969
- Entre 1970 e 1979
- Entre 1980 e 1989
- Entre 1990 e 1999
- Entre 2000 e 2009
- Entre 2010 e 2014

**26. Quando você começou o seu envolvimento com software proprietário?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2 e pelo sim na questão 24.**

**Por favor, escolha apenas uma das opções.**

- Antes de 1960
- Entre 1960 e 1969
- Entre 1970 e 1979
- Entre 1980 e 1989
- Entre 1990 e 1999
- Entre 2000 e 2009
- Entre 2010 e 2014

**26. Quando você começou o seu envolvimento com software proprietário?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2 e pelo sim na questão 24.  
Por favor, escolha apenas uma das opções.**

- Antes de 1960
- Entre 1960 e 1969
- Entre 1970 e 1979
- Entre 1980 e 1989
- Entre 1990 e 1999
- Entre 2000 e 2009
- Entre 2010 e 2014

**28. Em média, quantas horas por semana você se envolveu em projetos de software proprietário, durante o ano de 2013?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2 e pelo sim na questão 24.  
Por favor escolha somente uma das opções**

- Menos de 2 horas semanais
- De 2 a 5 horas semanais
- De 6 a 10 horas semanais
- De 11 a 20 horas semanais
- De 21 a 40 horas semanais
- Mais de 40 horas semanais

**29. No passado, qual foi a razão que levou você a iniciar o desenvolvimento e/ou a distribuição de software livre/código aberto?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2  
Por favor, selecione no máximo quatro opções.**

- participar em novas formas de cooperação.
- aprender e desenvolver novas habilidades.
- compartilhar o meu conhecimento e habilidades.
- participar no cenário de software livre/código aberto.
- melhorar as minhas oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros desenvolvedores.
- conseguir reputação no cenário de desenvolvedores de software livre/código aberto.
- distribuir software não comercial.
- conseguir ajuda para desenvolvimento de uma boa ideia para um produto de software.
- resolver problema que não poderia ser feito através de software proprietário
- restringir o poder das grandes empresas de software.
- porque eu penso que software não deveria ser um produto proprietário.
- para obter recursos financeiros.
- não sei dizer.

**30.No passado, qual foi a razão que levou você a iniciar o desenvolvimento e/ou a distribuição de software livre/código aberto?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2.  
Por favor, selecione no máximo quatro opções.**

- participar em novas formas de cooperação.
- aprender e desenvolver novas habilidades.
- compartilhar o meu conhecimento e habilidades.
- participar no cenário de software livre/código aberto.
- melhorar as minhas oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros desenvolvedores.
- conseguir reputação no cenário de desenvolvedores de software livre/código aberto.
- distribuir software não comercial.

- conseguir ajuda para desenvolvimento de uma boa ideia para um produto de software.
- resolver problema que não poderia ser feito através de software proprietário
- restringir o poder das grandes empresas de software.
- porque eu penso que software não deveria ser um produto proprietário.
- para obter recursos financeiros.
- não sei dizer.

**31. E hoje, qual é a razão que faz você desenvolver e/ou distribuir software livre/código aberto?**  
**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**  
**Por favor, selecione no máximo quatro opções.**

- participar em novas formas de cooperação.
- aprender e desenvolver novas habilidades.
- compartilhar o meu conhecimento e habilidades.
- participar no cenário de software livre/código aberto.
- melhorar as minhas oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros desenvolvedores.
- conseguir reputação no cenário de desenvolvedores de software livre/código aberto.
- distribuir software não comercial.
- conseguir ajuda para desenvolvimento de uma boa ideia para um produto de software.
- resolver problema que não poderia ser feito através de software proprietário
- restringir o poder das grandes empresas de software.
- porque eu penso que software não deveria ser um produto proprietário.
- para obter recursos financeiros.
- não sei dizer.

**32. E hoje, qual é a razão que faz você desenvolver e/ou distribuir software livre/código aberto?**  
**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2**  
**Por favor, selecione no máximo quatro opções.**

- participar em novas formas de cooperação.
- aprender e desenvolver novas habilidades.
- compartilhar o meu conhecimento e habilidades.
- participar no cenário de software livre/código aberto.
- melhorar as minhas oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros desenvolvedores.
- conseguir reputação no cenário de desenvolvedores de software livre/código aberto.
- distribuir software não comercial.
- conseguir ajuda para desenvolvimento de uma boa ideia para um produto de software.
- restringir o poder das grandes empresas de software.
- porque eu penso que software não deveria ser um produto proprietário.
- para obter recursos financeiros.
- não sei dizer.

**33. Você usa software livre/código aberto no trabalho ou, se você está estudando, na escola/colégio/universidade?**  
**Por favor, escolha apenas uma opção.**

- Sim
- Não

**34. Você ganha dinheiro com software livre/código aberto direta ou indiretamente?**

**Se você ganha dinheiro direta ou indiretamente por motivos diversos, por favor escolha o principal deles. Por favor, escolha apenas uma opção.**

- Não
- Sim, diretamente: eu sou pago para desenvolver software livre/código aberto.
- Sim, diretamente: eu sou pago para dar suporte técnico para software livre/código aberto.
- Sim, diretamente: eu sou pago para administrar software livre/código aberto.
- Sim, diretamente: outras razões.
- Sim, indiretamente: eu consegui o meu emprego em virtude de possuir experiência prévia em software livre/código aberto.
- Sim, indiretamente: a descrição da minha função não inclui desenvolvimento de software livre/código aberto, mas eu também desenvolvo software livre/código aberto no meu trabalho.
- Sim, indiretamente: outras razões.

**35. Dos editores mencionados abaixo, qual é o seu favorito?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**  
**Por favor, escolha apenas uma opção.**

- Eclipse.
- Emacs/XEmacs.
- Netbeans.
- Textmate.
- Vi/Vim
- Outro \_\_\_\_\_

**36. Em que ferramentas e linguagens de programação você possui experiência?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**  
**Por favor, escolha todas as opções que se aplicam.**

- C
- C++
- Objective C
- Java
- Python
- Perl
- PHP
- Unix Shell
- HTML
- LISP
- (La)Tex
- Pascal
- Fortran
- Basic
- Visual Basic
- JavaScript
- SQL
- ADA
- Modula
- Eiffel
- Prolog
- XML/SGML
- Ruby
- Smaltalk
- Tcl
- Make
- Scheme
- Versioning systems

**37. Em que ferramentas e linguagens de programação você possui experiência?****Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2****Por favor, escolha todas as opções que se aplicam.**

- C
- C++
- Objective C
- Java
- Python
- Perl
- PHP
- Unix Shell
- HTML
- LISP
- (La)Tex
- Pascal
- Fortran
- Basic
- Visual Basic
- JavaScript
- SQL
- ADA
- Modula
- Eiffel
- Prolog
- XML/SGML
- Ruby
- Smaltalk
- Tcl
- Make
- Scheme
- Versioning systems
- Nenhuma delas

**Questões gerais – Contribuições****38. Você utiliza software/serviços que não são livres ou de código aberto no seu dia a dia ou para desenvolvimento?****Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2****Por favor, escolha apenas uma das opções.**

- Não, eu tento evitar utilizar software/serviços que não são livres ou de código aberto.
- Eu escolhi utilizar software/serviços que não são livres ou de código aberto para certas tarefas.
- Sim, sem problema. Eu uso o melhor software/serviços a despeito de ser livre ou de código aberto ou não.

**39. Você utiliza software/serviços que não são livres ou de código aberto no seu dia a dia ou para desenvolvimento?****Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2****Por favor, escolha apenas uma das opções.**

- Não, eu tento evitar utilizar software/serviços que não são livres ou de código aberto.
- Eu escolhi utilizar software/serviços que não são livres ou de código aberto para certas tarefas.
- Sim, sem problema. Eu uso o melhor software/serviços a despeito de ser livre ou de código aberto ou não.

**40. Em quantos projetos de desenvolvimento de software livre/código livre você já esteve envolvido?  
Por favor, escolha apenas uma das opções.**

- 1 – 5
- 6 – 10
- 11 – 20
- 21 – 30
- 31 – 50
- 51 – 75
- 76 – 100
- Mais de 100

**41. Em quantos projetos de desenvolvimento de software livre/código aberto você já se envolveu na condição de líder, coordenador ou administrador?  
Por favor, escolha apenas uma das opções.**

- Nenhum
- 1
- 2
- 3
- 4 – 5
- 6 – 7
- 8 – 10
- 11 – 15
- 16 – 20
- Mais de 20

## Questões gerais – Ganhos econômicos e sociais

**42. Em termos gerais, você considera que software livre/código aberto satisfaz os requisitos atuais de software melhor que software proprietário?  
Por favor, escolha apenas uma opção.**

- Sim
- Não
- Um não tem nada a ver com o outro
- Não sei dizer

**43. As pessoas que trabalham com software livre/código aberto são mais ligadas a dinheiro do que pessoas que trabalham com software proprietário?  
Por favor, escolha apenas uma opção.**

- Isso é bom.
- Isso é mau.
- Isso não é verdade.

**44. As pessoas que trabalham com software proprietário são mais ligadas a dinheiro do que pessoas que trabalham com software livre/código aberto?**

**Por favor, escolha apenas uma opção.**

- Isso é bom.
- Isso é mau.
- Isso não é verdade.

**45. Se você tivesse que descrever o ambiente do software livre/código aberto em geral, qual das declarações abaixo reflete mais a sua opinião? O ambiente do software livre/código aberto é um fórum para...**

**Por favor, escolha no máximo três opções.**

- uma discussão geral sobre software.
- uma competição esportiva sobre o melhor código.
- prover inovações radicais
- prover imitações de produtos e serviços de software proprietário.
- desenvolvedores de software que necessitam de uma toolbox.
- prover mais variedade de software.
- pessoas com os mesmos interesses.
- pessoas que procuram parceiros de projeto.
- pessoas que utilizam novas formas de cooperação.
- maior liberdade no desenvolvimento de software.
- melhorias na carreira.
- troca de conhecimento.
- pessoas que querem se divertir.

**46. Você acredita que a vivência na comunidade de software livre/código aberto tem impacto positivo nas oportunidades de trabalho?**

**Por favor, escolha apenas uma opção.**

- Sim
- Não
- Não sei dizer

**47. Os desenvolvedores contribuem para a comunidade de software livre/código aberto de diferentes modos, e recebem muitos benefícios em troca. Em geral, como você avalia o peso da sua balança?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**

**Por favor, escolha apenas uma opção.**

- Forneço mais do que recebo.
- Recebo mais do que forneço.
- Forneço tanto quanto recebo.
- Não me importo.
- Não sei dizer.

**48. Os desenvolvedores contribuem para a comunidade de software livre/código aberto de diferentes modos, e recebem muitos benefícios em troca. Em geral, como você avalia o peso da sua balança?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2**

**Por favor, escolha apenas uma opção.**

- Forneço mais do que recebo.
- Recebo mais do que forneço.
- Forneço tanto quanto recebo.
- Não me importo.
- Não sei dizer.

**49. Quando você considera outros desenvolvedores de software livre/código aberto, como você avalia o peso da balança deles?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**

**Por favor, escolha apenas uma opção.**

- A maioria fornece mais do que recebe.
- A maioria recebe mais do que fornece.
- Eles fornecem tanto quando recebem.
- Eles não se importam.
- Eu não me importo.
- Não sei dizer.

**50. Quando você considera outros desenvolvedores de software livre/código aberto, como você avalia o peso da balança deles?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2**

**Por favor, escolha apenas uma opção.**

- A maioria fornece mais do que recebe.
- A maioria recebe mais do que fornece.
- Eles fornecem tanto quando recebem.
- Eles não se importam.
- Eu não me importo.
- Não sei dizer.

**51. Desenvolver software livre/código aberto é uma forma de autoexploração, pois você fornece um conjunto de boas ideias e trabalho e não recebe benefícios em troca! Até que ponto você concorda ou discorda desta afirmação?**

**Por favor, escolha somente uma opção.**

- Concordo totalmente.
- Concordo.
- Tendo a discordar.
- Discordo completamente.
- Não sei dizer.

**52. O que você espera de outros desenvolvedores de software livre/código aberto?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**

**Por favor, escolha no máximo quatro opções.**

- estar apto para cooperar de uma nova maneira.
- deixar-me aprender e desenvolver novas habilidades.
- compartilhar o seu conhecimento e habilidades.
- tomar parte nas principais comunicações e discussões.
- escrever programas bonitos e estéticos.
- fornecer melhores oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros desenvolvedores.
- respeitar-me e as minhas contribuições para software livre/código aberto.
- distribuir software não comercial.
- ajudar na criação de ideias para produtos de software.
- resolver um problema que não poderia ser resolvido por software proprietário.
- contribuir para limitar o poder das grandes empresas de software.
- ganhar dinheiro.
- não sei dizer.

**53. O que você espera de outros apoiadores de software livre/código aberto?**

**Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2**

**Por favor, escolha no máximo quatro opções.**

- estar apto para cooperar de uma nova maneira.
- deixar-me aprender e desenvolver novas habilidades.
- compartilhar o seu conhecimento e habilidades.
- tomar parte nas principais comunicações e discussões.
- escrever programas bonitos e estéticos.
- fornecer melhores oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros.
- respeitar-me e as minhas contribuições para software livre/código aberto.
- distribuir software não comercial.
- ajudar na criação de ideias para produtos de software.
- resolver um problema que não poderia ser resolvido por software proprietário.
- contribuir para limitar o poder das grandes empresas de software.
- ganhar dinheiro.
- não sei dizer.

**54. O que acha que os outros desenvolvedores de software livre/código aberto esperam de você?**

**Questão a ser respondida apenas por aqueles que optaram pela 1ª ou a 3ª alternativa da questão 2**

**Por favor, escolha no máximo quatro opções.**

- estar apto para cooperar de uma nova maneira.
- deixá-los aprender e desenvolver novas habilidades.
- compartilhar o meu conhecimento e habilidades.
- tomar parte nas principais comunicações e discussões.
- escrever programas bonitos e estéticos.
- fornecer melhores oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros desenvolvedores.
- respeitá-los e as suas contribuições para software livre/código aberto.
- distribuir software não comercial.
- ajudar na criação de ideias para produtos de software.
- resolver um problema que não poderia ser resolvido por software proprietário.
- contribuir para limitar o poder das grandes empresas de software.
- ganhar dinheiro.
- não sei dizer.

**55. O que acha que os outros apoiadores de software livre/código aberto esperam de você? Questão a ser respondida apenas por aqueles que optaram pela 2ª alternativa da questão 2. Por favor, escolha no máximo quatro opções.**

- estar apto para cooperar de uma nova maneira.
- deixá-los aprender e desenvolver novas habilidades.
- compartilhar o meu conhecimento e habilidades.
- tomar parte nas principais comunicações e discussões.
- escrever programas bonitos e estéticos.
- fornecer melhores oportunidades de trabalho.
- melhorar produtos de software livre/código aberto de outros.
- respeitá-los e as suas contribuições para software livre/código aberto.
- distribuir software não comercial.
- ajudar na criação de ideias para produtos de software.
- resolver um problema que não poderia ser resolvido por software proprietário.
- contribuir para limitar o poder das grandes empresas de software.
- ganhar dinheiro.
- não sei dizer.

**56. Abaixo você pode encontrar alguns possíveis desafios a enfrentar nos principais projetos de software livre/código aberto no qual está envolvido. Por favor, avalie a importância que você dá para cada um deles. Por favor, selecione uma opção para cada item.**

Item	Não é importante	OK	OK, mas podia ser melhor	Preocupa-me, mas não sei o que fazer	Eu trato dessa questão	Isso me coloca, e a outros, fora do projeto	Não sei dizer
Tornar o software estável (reduzir bugs)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Conjunto de recursos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Documentação	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Tornar o software multiplataforma/móvel/SaaS	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Reduzir o tempo entre versões	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Atrair usuários/colaboradores	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manter usuários/colaboradores engajados	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**57. Abaixo você pode encontrar alguns possíveis desafios a enfrentar nos principais projetos de software livre/código aberto no qual está envolvido. Por favor, avalie a importância que você dá para cada um deles. Por favor, selecione uma opção para cada item.**

Item	Não é importante	OK	OK, mas podia ser melhor	Preocupa-me, mas não sei o que fazer	Eu trato dessa questão	Isso me coloca, e a outros, fora do projeto	Não sei dizer
Alavancar o poder da empresa líder	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Obter fundos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Questões legais (licença, copyright, etc.).	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comunicação interna do projeto	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comunicação de cima para baixo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**58. Por favor, indique outras cinco pessoas do seu conhecimento, que se envolvem, também, com software livre/código aberto para preenchimento do questionário, informando nome completo, e-mail e telefone de contato.**

	NOME	E-MAIL	TELEFONE
1			
2			
3			
4			
5			

Ao entrar em contato com as pessoas indicadas, CONCORDO que o meu nome seja mencionado como referência.

Ao entrar em contato com as pessoas indicadas, NÃO CONCORDO não gostaria que o meu nome fosse mencionado como referência.

**Obrigado por preencher este questionário!  
Acompanhe os resultados pelo site [www.softex.br](http://www.softex.br).**







# SOFTWARE LIVRE: TENDÊNCIAS, OPORTUNIDADES E DESAFIOS



TECNOLOGIA DA INFORMAÇÃO BRASILEIRA