

O Congresso Internacional de Software Livre e Governo Eletrônico (Consegi) tem como premissa apresentar, ao governo e à sociedade, soluções desenvolvidas para Governo Eletrônico, baseadas em Software Livre. São soluções que podem ser copiadas, alteradas e distribuídas gratuitamente – princípios básicos de Software Livre. Assim como a troca de experiências sobre tecnologia da informação e comunicação (TIC) para o governo e sociedade.

O Consegi chega a sua sexta edição em 2013 e será realizado em Brasília, na Escola de Administração Fazendária (Esaf), no período de 13 a 15 de agosto. Palestras, oficinas, painéis temáticos e espaços abertos para debates farão parte da programação do evento.

Nesta edição, serão realizados diversos encontros paralelos, entre eles:

- O primeiro encontro do Libre Office fora da Alemanha;
- Festival de instalação de Software livre (Flisol);
- Encontro de empresas de TI com o Sebrae;
- Encontro da comunidade Expresso;
- Encontro com a comunidade Demoiselle.

Durante todo o evento, sempre estará em ação pelo menos um tutorial de Linux para Principiantes, oferecido por membros das comunidades das principais distribuições do sistema operacional.



Desde 2008, o Consegi é um evento estratégico no estímulo ao debate de temas da política e gestão de tecnologias em Software Livre, permeado pelos conceitos de promoção da cidadania, interoperabilidade dos sistemas de governo, compartilhamento do conhecimento e adoção de normas e padrões.

Em 2009, expôs o imenso acervo de soluções e de códigos desenvolvidos e compartilhados pela Esfera Pública. O evento mostrou que, com a lógica do Software Livre, os recursos podem ser mais bem aproveitados.

Além disso, os conceitos de liberdade e direito de acesso ao conhecimento foram discutidos por mais de 4 mil participantes. Representantes de 17 países (África do Sul, Argentina, Chile, Coreia do Sul, Cuba, Egito, Equador, França, Índia, Indonésia, Malásia, Paraguai, São Tomé e Príncipe, Uruguai, Venezuela e Zimbábue) participaram dessa edição do evento.

Em 2010, o Consegi inovou ao trazer para dentro do governo a discussão sobre Computação em Nuvem e a evolução dos serviços de governo eletrônico a partir dessa nova proposta de infraestrutura tecnológica. A cooperação internacional e a defesa do software livre como melhor alternativa técnica para a Administração Pública continuaram presentes no evento.

Em 2011, o tema Dados Abertos trouxe o debate da liberação de informações públicas para o âmago do governo. Instituições públicas e grupos interessados dedicaram-se durante três dias ao compartilhamento de conhecimento em torno da questão, que ainda era embrionária no país.

A edição 2012, realizada pela primeira vez fora de Brasília, aconteceu em Belém e abordou a “Mobilidade Digital”, um advento que vem alterando a forma como os indivíduos usam e se apropriam das tecnologias. O governo, de maneira a inovar nos serviços públicos aos cidadãos, já começou a pensar e disponibilizar aplicativos voltados para os dispositivos móveis. Como ampliar a atuação na área de governo eletrônico, tendo a portabilidade como elemento essencial para melhor atender às pessoas, foi o grande foco de debates da quinta edição.

651

CONSEGI
VI CONGRESSO INTERNACIONAL
DE SOFTWARE LIVRE
E GOVERNO ELETRÔNICO
2013

FUNDAÇÃO
ALEXANDRE
DE GUSMÃO



Ministério da
Fazenda

Ministério das
Relações Exteriores



Brasília-DF  www.consegi.gov.br

Tema Central do Consegi 2013:
Portabilidade, Colaboração e Integração

Eixos temáticos:

- Aplicações de Governo Eletrônico
- Computação em Nuvem e Infraestrutura de TIC
- Governança, Gestão e Estratégia de TIC
- Ecossistema de Software Livre
- Novas Interfaces Homem-Máquina
- Tecnologias Emergentes
- Portabilidade, Padrões e Interoperabilidade
- Segurança e Defesa Cibernética
- Tecnologias geoespaciais
- Informática e Sociedade





CONSEGI VI CONGRESSO INTERNACIONAL
DE SOFTWARE LIVRE
E GOVERNO ELETRÔNICO
2013



Consegi 2013

VI Congresso Internacional
de Software Livre e
Governo Eletrônico

MINISTÉRIO DAS RELAÇÕES EXTERIORES

Ministro de Estado Embaixador Antonio de Aguiar Patriota
Secretário-Geral Embaixador Eduardo dos Santos

FUNDAÇÃO ALEXANDRE DE GUSMÃO



Presidente Embaixador José Vicente de Sá Pimentel

*Instituto de Pesquisa de
Relações Internacionais*

*Centro de História e
Documentação Diplomática*

Diretor Embaixador Maurício E. Cortes Costa

A *Fundação Alexandre de Gusmão*, instituída em 1971, é uma fundação pública vinculada ao Ministério das Relações Exteriores e tem a finalidade de levar à sociedade civil informações sobre a realidade internacional e sobre aspectos da pauta diplomática brasileira. Sua missão é promover a sensibilização da opinião pública nacional para os temas de relações internacionais e para a política externa brasileira.

Ministério das Relações Exteriores
Esplanada dos Ministérios, Bloco H
Anexo II, Térreo, Sala 1
70170-900 Brasília, DF
Telefones: (61) 2030-6033/6034
Fax: (61) 2030-9125
Site: www.funag.gov.br

Consegi 2013

VI Congresso Internacional
de Software Livre e
Governo Eletrônico



Brasília - 2013

Direitos de publicação reservados à
Fundação Alexandre de Gusmão
Ministério das Relações Exteriores
Esplanada dos Ministérios, Bloco H
Anexo II, Térreo
70170-900 Brasília – DF
Telefones: (61) 2030-6033/6034
Fax: (61) 2030-9125
Site: www.funag.gov.br
E-mail: funag@itamaraty.gov.br

Equipe Técnica:

Eliane Miranda Paiva
Fernanda Antunes Siqueira
Guilherme Lucas Rodrigues Monteiro
Jessé Nóbrega Cardoso
Vanusa dos Santos Silva

Projeto Gráfico:

Daniela Barbosa

Programação Visual e Diagramação:

Gráfica e Editora Ideal

Impresso no Brasil 2013

C755

Consegi 2013 : VI Congresso Internacional de Software Livre e Governo Eletrônico
/ Alison Wilker ... [et al.] – Brasília : FUNAG, 2013.

90 p. : il.

ISBN 978-85-7631-443-1

1. Tecnologia da informação - África do Sul. 2. Software livre - África do Sul.
3. Serviço eletrônico - África do Sul. 4. Tecnologia da informação - congressos etc.
I. Consegi 2013 : VI Congresso Internacional de Software Livre e Governo Eletrônico.

CDD 005.1

Bibliotecária responsável: Ledir dos Santos Pereira, CRB-1/776

Depósito Legal na Fundação Biblioteca Nacional conforme Lei nº 10.994, de 14/12/2004.

**CONSELHO EDITORIAL DA FUNDAÇÃO
ALEXANDRE DE GUSMÃO**

Presidente: Embaixador José Vicente de Sá Pimentel

Presidente da Fundação Alexandre de Gusmão

Membros: Embaixador Ronaldo Mota Sardenberg

Embaixador Jorio Dauster Magalhães

Embaixador José Humberto de Brito Cruz

Ministro Luís Felipe Silvério Fortuna

Professor Clodoaldo Bueno

Professor Francisco Fernando Monteoliva Doratioto

Professor José Flávio Sombra Saraiva



APRESENTAÇÃO

Neste ano de 2013 o VI Congresso Internacional de Software Livre e Governo Eletrônico (Consegi) traz como temas a portabilidade, a colaboração e a integração. O nexos dessas três forças mais a estratégia transversal de se utilizar software livre em sistemas de governo eletrônico são capazes de transformar a maneira como é feita a governança eletrônica de sistemas de informação e a preservação da inteligência nacional com segurança.

Particularmente as empresas que prestam serviços de tecnologia da informação e comunicação aos Governos devem ter, em sua essência, a capacidade de estabelecer políticas para retenção do saber público e para a garantia da segurança das informações soberanas de um país. Estas capacidades estão intrinsecamente ligadas à capacidade das empresas inovarem a partir dos insumos que a tecnologia fornece.

O bom uso dos recursos públicos passa pelo reuso e pela colaboração entre os diversos atores, assim a portabilidade que emerge com a criação de uma nova interface humano-computador para sistemas de governo é viabilizada pelo reuso de serviços que, na era da Web, são integrados através dos chamados padrões de Web Services.

Uma das primeiras multiplataformas, surgida na década de 1990, que permitiu a portabilidade de aplicações entre vários sistemas operacionais de computadores diversos foi a Máquina Virtual Java, que se tornou mundialmente adotada por ser um software de licenciamento livre. Atualmente existem mais de 3 bilhões de equipamentos com Java no mundo entre *tablets*, *smartphones*, *notebooks*, computadores, servidores e *gadgets* diversos.

O debate de estratégias para se construir aplicações que possam ser escritas uma vez e ser executadas em várias plataformas diferentes é um dos desafios que está presente neste VI Consegi. Além disso, estarão presentes em palestras e oficinas do Congresso as demonstrações de como se criar e executar aplicações que exijam uma nova experiência do usuário (por exemplo: *tablets*, *tv's* conectadas e *smartphones*) integrando-se com sistemas legados de Governo Eletrônico de forma colaborativa e com uso de padrões abertos.

Boa leitura!

Marcos Vinicius Ferreira Mazoni
Diretor-Presidente do Serpro

SUMÁRIO

Introdução	11
-------------------	-----------

Portabilidade entre dispositivos móveis: aplicações multiplataformas e adaptadas ao contexto	15
---	-----------

*Alisson Wilker Andrade, Marlon Carvalho, Serge Rehem,
Rodrigo Hjort, Ronaldo Agra, Viviane Malheiros*

Colaboração e Software Livre	43
-------------------------------------	-----------

Fabio Kon, Carlos D. Santos

Os Serviços Eletrônicos de Governo e a Interoperabilidade	67
--	-----------

Marcus Vinicius da Costa, Everson Lopes de Aguiar



INTRODUÇÃO

O Congresso Internacional de Software Livre e Governo Eletrônico (CONSEGI) tem mantido a tradição de, a cada ano, homenagear um país, conferindo-lhe especial realce em palestras e debates realizados em torno da temática da edição do evento. O primeiro país homenageado foi a França, no contexto das comemorações do Ano da França no Brasil, em 2009. Já em 2010, foi a vez da Coreia do Sul. Em 2011 e 2012 os países focais foram, respectivamente, Espanha e Uruguai.

Em 2013, tendo por tema “Portabilidade, Comunicação e Integração”, o VI CONSEGI presta homenagem à África do Sul, país que vem se destacando nos campos das tecnologias da informação e das comunicações. Ao longo da edição deste ano do Congresso, portanto, expoentes sul-africanos do software livre discorrerão sobre soluções desenvolvidas naquele país em matéria de sistemas operacionais, políticas públicas de TI e gestão da informação.

O sistema operacional Ubuntu, que foi desenvolvido na África do Sul, será apresentado com destaque no VI CONSEGI. O referido sistema de código aberto foi batizado com uma palavra que traz uma mensagem de compartilhamento, liberdade e colaboração que são típicos, aliás, do software livre. Conta-se que Nelson Mandela

usava essa expressão para incentivar os sul-africanos a trabalhar juntos para alcançar seus ideais de igualdade e liberdade.

Em uma de suas entrevistas, o ex-presidente da África do Sul e Prêmio Nobel da Paz – que é homenageado pelo CONSEGI nesta sua sexta edição – reforçou, nos seguintes termos, o espírito de coletividade ao qual a palavra remete “Ubuntu não significa que uma pessoa não se preocupe com o seu progresso pessoal. A questão é: o meu progresso pessoal está a serviço do progresso da minha comunidade?”. Seria desnecessário discorrer sobre a importância de manter tal perspectiva no momento atual em que as tecnologias da informação e das comunicações alcançam progresso extraordinário, impactando todas as atividades e as relações entre as pessoas.

Os laços e as similaridades entre o Brasil e a África do Sul vão muito além do fato de ambos estarem localizados no Hemisfério Sul e terem seu litoral banhado pelo Oceano Atlântico. São ambos países em desenvolvimento influentes, dotados de grande capacidade de pesquisa e inovação, populações expressivas e de composição multiétnica. Ambos enfrentam, também, grandes desafios nos campos econômico e social. Em relação aos temas focalizados no CONSEGI, Brasil e África do Sul têm procurado explorar formas de tornar cada vez mais relevante o software livre e aberto, especialmente no que se refere à provisão de serviços eletrônicos pelo governo. Nesse sentido, o intercâmbio entre os dois países, durante o CONSEGI 2013, deverá contribuir para estreitar nossas relações nas áreas de software aberto e governo eletrônico, além de reafirmar a parceria e o apoio mútuo em relação aos constantes e significativos desafios para o contínuo desenvolvimento de sociedades da informação cada vez mais inclusivas e transparentes.

Pode-se argumentar que a promoção do código aberto, além de oferecer soluções importantes para questões de segurança dos

dados, tem o condão de favorecer o aparecimento de tecnologias autóctones e de uma maior variedade de soluções de sistemas, bem como a formação de uma comunidade colaborativa e engajada. Do ponto de vista da estratégia governamental de promoção do desenvolvimento econômico e social, o apoio ao software aberto apresenta-se, portanto, particularmente interessante, não apenas para o avanço das ferramentas e serviços públicos disponibilizados por meio do governo eletrônico, mas, também, pelas oportunidades que proporciona para a afirmação da autonomia tecnológica e para a exploração de possibilidades de cooperação Sul-Sul. Por esses motivos, o Itamaraty tem renovada satisfação e orgulho em colaborar com o SERPRO na organização do CONSEGI 2013.

Os debates sobre a portabilidade entre dispositivos móveis, a colaboração no ambiente do software livre e a relação entre os serviços eletrônicos de governo e a interoperabilidade compõem um mosaico, em que o traço comum é o espírito colaborativo. Todos esses temas serão desenvolvidos nas páginas a seguir, em antecipação aos estimulantes debates que certamente terão lugar ao longo dos três dias desta sexta edição do Congresso.

Benedicto Fonseca Filho

Diretor do Departamento de
Temas Científicos e Tecnológicos
Ministério das Relações Exteriores



PORTABILIDADE ENTRE DISPOSITIVOS MÓVEIS: APLICAÇÕES MULTIPLATAFORMAS E ADAPTADAS AO CONTEXTO

- *Alisson Wilker Andrade*¹
- *Marlon Carvalho*²
- *Serge Rehem*³
- *Rodrigo Hjort*⁴
- *Ronaldo Agra*⁵
- *Viviane Malheiros*⁶

1 <alisson-wilker.silva@serpro.gov.br>.

2 <marlon.carvalho@serpro.gov.br>.

3 <serge.rehem@serpro.gov.br>.

4 <rodrigo.hjort@serpro.gov.br>.

5 <jose-ronaldo.souza@serpro.gov.br>.

6 <viviane.malheiros@serpro.gov.br>.



1. Introdução

O mercado de mobilidade tem crescido bastante nos últimos anos. Segundo informações do *Business Insider*⁷, as vendas globais anuais de *tablets* cresceram de zero a mais de 100 milhões de unidades entre 2009 e 2012 e as de *smartphones* ascenderam de zero a mais de 600 milhões de unidades entre 2007 e 2012. Enquanto isso, as vendas de computadores pessoais e corporativos têm se mostrado estáveis, com pouco menos de 200 milhões de unidades anuais cada, desde 2009.

Com previsão de que as vendas de *tablets* e *smartphones* cheguem a 1,25 milhão em 2013⁸, esse crescimento de dispositivos móveis no mercado atraiu a atenção de pequenas, médias e grandes empresas. Em especial, esses números também têm atraído investidores para diversas *start-ups* que apresentam inovações em mobilidade.

A Apple teve um papel fundamental na aceleração do crescimento da indústria de mobilidade com a criação do iPhone, em 2007, e do iPad, em 2009. Esses aparelhos introduziram um novo conceito de dispositivos móveis que trazem para as mãos do usuário o poder de processamento de computadores portáteis

7 "Mobile is eating the world", Business Insider. Disponível em: <<http://www.businessinsider.com/heres-an-excellent-presentation-about-the-rise-of-mobile-and-the-massive-implications-2013-5#-1>>. Acesso em: 18/06/2013.

8 Idem.

junto com vários recursos, administrados pelo sistema operacional, como: bússola, GPS, câmera e acelerômetro. É importante observar que a computação portátil não é recente, pois teve início nos anos 80, com o advento do *palmtop* ou PDA (*Personal Digital Assistant*), tendo participação de *players* como HP, Palm, IBM, Nokia e inclusive Apple⁹. Todavia, a grande adoção dos *smartphones* a partir de 2007 culminou com a obsolescência dos *palmtops*.

Aplicativos para os dispositivos móveis são construídos por diversos desenvolvedores ao redor do mundo. O conjunto de *software* e *hardware* dos *smartphones* e *tablets* é o que se chama de plataforma móvel.

Depois da grande contribuição da Apple, surgiram vários outros aparelhos e outras plataformas móveis concorrentes. Em especial, a Google destacou-se com o lançamento da plataforma Android, que possibilitou a qualquer fabricante criar seus aparelhos sem se preocupar com o desenvolvimento de um sistema operacional próprio. Utilizando essa plataforma em seus dispositivos, a Samsung também cresceu e destaca-se com mais do que o dobro do mercado de dispositivos móveis em relação à Apple no primeiro trimestre de 2013¹⁰.

Outras plataformas móveis importantes que surgiram nesse período são: (i) o Windows Phone e o Windows 8 da Microsoft, que operam principalmente nos dispositivos da fabricante Nokia; e (ii) o BlackBerry OS da BlackBerry, que renovou o seu sistema operacional para adaptar-se a novos recursos de *smartphones* e também lançou aparelhos do tipo *tablet*. Nesse mesmo tempo, outras plataformas tiveram sua fatia reduzida e

9 "Personal digital assistant", Wikipedia. Disponível em: <http://en.wikipedia.org/wiki/Personal_digital_assistant>. Acesso em: 24/06/2013.

10 "Mobile is eating the world", Business Insider. Disponível em: <<http://www.businessinsider.com/heres-an-excellent-presentation-about-the-rise-of-mobile-and-the-massive-implications-2013-5#-1>>. Acesso em: 18/06/2013.

foram simplesmente descontinuadas, tais como Nokia Symbian¹¹ e Samsung Bada OS¹².

O mercado de mobilidade está fragmentado em, pelo menos, quatro plataformas significativas: Android, iOS, Windows Phone/Windows 8 e BlackBerry. Se, por um lado, esta diversidade tem o benefício de trazer opções para os consumidores e aumentar a concorrência entre os fornecedores, por outro, ela gera para as empresas desafios técnicos e operacionais relacionados à portabilidade. No contexto da informática, a portabilidade de um programa de computador refere-se à característica das aplicações serem executáveis (ou facilmente recompiladas) em outras plataformas além daquela de origem¹³. No contexto da mobilidade, além deste aspecto a diversidade de tamanho de dispositivos também precisa ser considerada.

Com o intuito de diminuir o impacto desta diversidade e das questões associadas à portabilidade propriamente dita, surgiram no mercado soluções para desenvolver o aplicativo móvel uma única vez e distribuí-lo para diversas plataformas e para diferentes formatos de dispositivos.

Este artigo discute as alternativas para o desenvolvimento multiplataforma e “multidispositivos” e apresenta os resultados de uma observação empírica do desenvolvimento de uma solução multiplataforma para Android e iOS. O artigo está estruturado da seguinte forma: na Seção 2 são discutidas as características e tecnologias para o desenvolvimento multiplataforma e uma

11 “Nokia Confirms The PureView Was Officially The Last Symbian Phone”, TechCrunch. Disponível em: <<http://techcrunch.com/2013/01/24/nokia-confirms-the-pure-view-was-officially-the-last-symbian-phone/>>. Acesso em: 24/06/2013.

12 “Samsung may discontinue Bada platform by mid-2013”, Tech2. Disponível em: <<http://tech2.in.com/news/smartphones/samsung-may-discontinue-bada-platform-by-mid2013/303262>>. Acesso em: 24/06/2013.

13 Portabilidade (informática), Wikipedia. Disponível em: <http://pt.wikipedia.org/wiki/Portabilidade_%28inform%C3%Altica%29>. Acesso em: 21/06/2013.

comparação ponto a ponto entre um aplicativo desenvolvido de forma nativa e de forma portátil; na Seção 3 são apresentadas técnicas e tecnologias para adaptar o conteúdo da aplicação móvel às características do dispositivo; na Seção 4 são apresentadas as considerações finais.

2. Desenvolvimento multiplataforma para dispositivos móveis

O desenvolvimento de aplicações para dispositivos móveis pode ser realizado de três formas: (I) com construção de aplicações nativas, (II) com construção de conteúdo web adaptável ou (III) com construção de aplicações híbridas. Essas três arquiteturas de aplicações foram detalhadas em “Consegi 2012 – Inovando Serviços com Mobilidade Digital”¹⁴, em que se discutiu vantagens e desvantagens de cada uma delas. No mesmo artigo, foi apresentado também um fluxo básico para escolha da melhor arquitetura, com base nas características da demanda e no uso esperado da aplicação.

Em resumo, a construção de aplicações nativas utiliza ferramentas e linguagens de programação específicas de cada plataforma. Por exemplo, a construção de uma aplicação para Android utiliza a linguagem de programação Java, o kit de desenvolvimento Android SDK e o ambiente de desenvolvimento Android Studio. Já o desenvolvimento de aplicações para iOS requer o uso da linguagem Objective-C, do iOS SDK e do ambiente de desenvolvimento Xcode. Dessa forma, cada desenvolvimento de aplicação nativa é especializado para uma plataforma, resultando em manutenção de códigos e capacitação de equipes específicas. O desenvolvimento nativo é interessante para explorar a experiência de uso numa

14 “Mobilidade digital aplicada ao governo brasileiro”, ANDRADE, Alisson et al. Em anais do Consegi 2012 – Inovando Serviços com Mobilidade Digital. ISBN 978-85-7631-411-0. Fundação Alexandre Gusmão, 2012. Brasília, DF, pp. 13-40.

plataforma e para um público-alvo específico, o que geralmente significa uma melhor integração com os recursos da plataforma e uma maior aceitação dos usuários¹⁵. Por outro lado, manter uma equipe focada em cada plataforma e manter códigos distintos pode elevar consideravelmente o custo e o tempo de produção.

A construção de conteúdo web adaptável para dispositivos móveis é uma arquitetura que requer um único desenvolvimento. Isto porque o conteúdo web construído de acordo com as especificações da W3C¹⁶ para HTML, JavaScript e CSS possui boa chance de executar sem problemas nos principais navegadores. Por conteúdo web adaptável, leia-se um conteúdo *web* que se adapta ao contexto em que está sendo executado: (i) navegador móvel ou de computador de mesa, (ii) tela de baixa, média ou alta resolução, etc. Além de ser mais econômico por requerer apenas uma equipe de desenvolvimento e a manutenção de um único código, essa alternativa é mais versátil, uma vez que o mesmo conteúdo web pode ser utilizado tanto em dispositivos móveis como em computadores de mesa. Isto pode ser obtido pelo uso de técnicas de design responsivo (Seção 3) e também pelo uso de bibliotecas para desenvolvimento web multiplataforma como o jQuery Mobile¹⁷, que abstraem as especificidades de cada plataforma para o desenvolvedor. Porém, o conteúdo *web* multiplataforma não possui integração com os componentes visuais nativos de cada plataforma e também possui limitações de *performance* e de uso dos recursos do dispositivo (como bússola, câmera, GPS, acelerômetro). Isso pode levar a uma experiência de usuário mais limitada e a uma necessidade maior de conexão com a Internet, visto que a interface e as regras de negócio são inicialmente carregadas de um servidor remoto.

15 Idem.

16 "W3C", W3C. Disponível em: <<http://www.w3.org/>>. Acesso em: 13/06/2013.

17 "jQuery Mobile", jQuery Foundation. Disponível em: <<http://jquerymobile.com/>>. Acesso em: 13/06/2013.

Como meio-termo entre o desenvolvimento de aplicações nativas e o conteúdo web adaptável, a terceira arquitetura é a de construção de aplicações híbridas. Nesse caso, temos a construção de conteúdo *web* empacotado em uma aplicação nativa, obtendo, assim, o melhor de cada uma das outras duas arquiteturas. Ou seja, é possível desenvolver e manter um único código escrito com tecnologia web (HTML, JavaScript e CSS) e, ao mesmo tempo, utilizar os recursos nativos do dispositivo. Como o conteúdo da interface é *web*, não há integração visual da interface da aplicação com os componentes visuais nativos da plataforma. Porém, é possível utilizar estilos CSS e bibliotecas de interface como o jQuery Mobile, por exemplo, para aproximar o componente visual web do estilo dos componentes visuais nativos. Além disso, como a aplicação é embarcada como nativa, é mais fácil disponibilizar conteúdo *off-line*, requerendo menos conectividade com a Internet para usar a aplicação. Para esse tipo de arquitetura, existem *frameworks*, como o Apache Cordova¹⁸, que facilitam o empacotamento do conteúdo web para cada plataforma e viabilizam a utilização dos recursos nativos do dispositivo.

Tendo em vista as três possibilidades apresentadas, uma decisão importante a ser tomada no início do desenvolvimento de uma aplicação para dispositivos móveis é a arquitetura a ser adotada. Considerando que a opção *web* normalmente é utilizada apenas para a adaptação de portais e sítios já existentes à mobilidade e não é eficiente para o uso de recursos como câmera, acelerômetro, entre outros; na maioria das vezes, essa decisão passa por escolher entre aplicação nativa ou híbrida. Cada uma destas arquiteturas tem suas vantagens e desvantagens, conforme será discutido na Seção 2.2, que apresenta um comparativo entre

18 "Apache Cordova", Apache Foundation. Disponível em: <<http://cordova.apache.org/>>. Acesso em: 13/06/2013.

aspectos técnicos e não técnicos do desenvolvimento de uma mesma aplicação construída em ambas as arquiteturas (nativa e híbrida). Além disso, como a interface de uma aplicação híbrida, em geral, é construída com HTML e CSS, podem ser utilizadas técnicas de design responsivo para adaptar a interface a diferentes características do dispositivo que a executa, como os diferentes tamanhos de tela, por exemplo. Essa discussão é apresentada na Seção 3.

2.1. Tecnologias para desenvolvimento multiplataforma

O desenvolvimento multiplataforma é realizado usando tecnologias diferentes daquelas empregadas no desenvolvimento para uma plataforma específica. Essas tecnologias abstraem as especificidades de cada plataforma, provendo um conjunto de componentes e APIs genéricos. Esses componentes e APIs executam em cada plataforma com as devidas adaptações, a fim de fornecer uma experiência de uso similar ao que é observado em uma aplicação nativa. A seguir, são apresentadas algumas dessas tecnologias.

O framework jQuery Mobile

O jQuery Mobile é um *framework* para desenvolvimento de conteúdo *web* adaptado à navegação através de telas sensíveis ao toque e de tamanho restrito, o que é o caso de muitos *smartphones* e *tablets*. Ele é construído a partir de outro *framework* de desenvolvimento *web* muito popular, o jQuery¹⁹. Ambos os *frameworks* são desenvolvidos e mantidos pela The jQuery Foundation e estão disponíveis sob licença MIT License.

O jQuery Mobile, atualmente disponível na versão 1.3.1, é compatível com a versão 1.9.1 do jQuery. Nesta versão, o

¹⁹ "jQuery", jQuery Foundation. Disponível em: <<http://jquery.com/>>. Acesso em: 13/06/2013.

jQuery Mobile disponibiliza diversos componentes visuais (*widgets*), como botões, caixas de verificação, listas, caixas de diálogo, caixas de texto e barras de navegação, entre outros, todos adaptados à navegação através do toque. Todos esses elementos são compatíveis com os padrões de tecnologia *web* da W3C, especialmente com HTML5²⁰ e CSS3²¹. Dessa forma, qualquer navegador *web* atual que suporta esses padrões é capaz de renderizar corretamente os componentes visuais do jQuery Mobile, mesmo os navegadores *web* para *desktop*.

O jQuery Mobile atual suporta uma ampla gama de plataformas. Entre elas estão: Android²², iOS²³, Windows Phone²⁴, BlackBerry²⁵, WebOS²⁶, Tizen²⁷, Bada²⁸ e Meego²⁹. Além disso, vários navegadores – *desktop* e móveis – como Firefox, Chrome, Opera, Safari e Internet Explorer também são suportados. Essa grande variedade de plataformas e navegadores suportados significa que, em geral, o desenvolvedor do conteúdo *web* não precisa se preocupar com a plataforma específica em que o seu código será executado, visto que o jQuery Mobile abstrai essa questão, realizando os devidos testes e adaptando o conteúdo à plataforma específica em tempo de execução. Esse fato reduz bastante o tempo de desenvolvimento e manutenção e, por conseguinte, os custos de produção.

20 "HTML 5", W3C. Disponível em: <<http://www.w3.org/html/wg/drafts/html/master/>>. Acesso em: 13/06/2013.

21 "CSS 3", W3C. Disponível em: <<http://www.w3.org/Style/CSS/current-work>>. Acesso em: 13/06/2013.

22 "Android", Google. Disponível em: <<http://www.android.com/>>. Acesso em: 19/06/2013.

23 "iOS", Apple. Disponível em: <<http://www.apple.com/br/ios/>>. Acesso em: 19/06/2013.

24 "Windows Phone", Microsoft. Disponível em: <<http://www.windowsphone.com/pt-BR>>. Acesso em: 19/06/2013.

25 "BlackBerry OS", BlackBerry. Disponível em: <<http://br.blackberry.com/software/smartphones/blackberry-7-os.html#tab-1>>. Acesso em 19/06/2013.

26 "WebOS", Hewlett-Packard. Disponível em: <<http://www.hpwebos.com/us/>>. Acesso em: 19/06/2013.

27 "Tizen", The Linux Foundation. Disponível em: <<https://www.tizen.org/>>. Acesso em: 19/06/2013.

28 "Bada", Samsung. Disponível em: <<http://www.bada.com/>>. Acesso em: 19/06/2013.

29 "Meego", The Linux Foundation. Disponível em: <<https://meego.com/>>. Acesso em 19/06/2013.

Além dos componentes visuais, o jQuery Mobile possui uma API³⁰ de programação bastante extensa. Através dessa API, é possível realizar diversas operações, como tratar eventos de interação do usuário e executar métodos para navegação entre páginas *web*, por exemplo. Alguns exemplos de eventos que podem ser tratados pelo jQuery Mobile incluem transições de página e criação de componentes visuais, além de todos os eventos comuns de uma página *web*, como clique, duplo clique, arrastar e soltar, entre outros.

O jQuery Mobile disponibiliza para o desenvolvedor um conjunto pré-configurado de estilos visuais (temas) que podem ser utilizados no conteúdo *web*. Esses estilos são baseados nas características visuais dos componentes das principais plataformas, como Android e iOS. Porém, se o desenvolvedor precisar criar um estilo próprio, seja por conveniência ou por interesse do cliente/ usuário, o jQuery Mobile também permite a personalização de temas. Isto pode ser feito de forma muito simples através de uma aplicação *web* disponível no site do *framework* chamada Theme Roller³¹. Com essa ferramenta é possível alterar os estilos pré-configurados do jQuery Mobile ou criar um estilo completamente novo.

O jQuery Mobile é composto por três tipos de conteúdo: JavaScript, CSS e imagens. O JavaScript é responsável por tudo o que é relacionado a comportamento, enquanto que o CSS e as imagens são responsáveis pelo estilo visual do conteúdo *web*. Para utilizar o jQuery Mobile, basta colocar esse conteúdo em um local acessível e referenciá-lo nas páginas HTML. Além do jQuery Mobile, é necessário também importar o *framework* base jQuery.

30 *Application Programming Interface*, do acrônimo inglês.

31 "Theme Roller", jQuery Foundation. Disponível em: <<http://jqueryui.com/themeroller/>>. Acesso em: 13/06/2013.

O framework Cordova

O Cordova é um *framework* que permite empacotar conteúdo *web* com características de aplicação nativa. Apenas a casca de aplicação nativa é específica para cada plataforma e, por isso, o ambiente de desenvolvimento de cada plataforma ainda é necessário para desenvolver aplicações híbridas. Porém, a maior parte da aplicação é escrita utilizando tecnologia *web* padrão W3C, o que facilita bastante a construção e a manutenção do código, além de diminuir os custos de produção.

O Cordova é a versão software livre de um projeto da Adobe Systems Inc. chamado PhoneGap³². O projeto Cordova é mantido pela Apache Software Foundation e distribuído sob a licença Apache License, versão 2.0. Atualmente na versão 2.8.0, o Cordova possui compatibilidade com uma ampla gama de plataformas: Android, Bada, BlackBerry, iOS, Mac OS X, QT, Tizen, WebOS, Windows e Windows Phone.

Na prática, a aplicação nativa gerada pelo Cordova é apenas um contêiner que interpreta conteúdo HTML, JavaScript e CSS, de forma similar ao que um navegador *web* comum faz, exceto pelo fato de não possuir barra de endereço ou histórico de navegação, por exemplo. Dessa forma, não existe nenhuma transformação ou geração de código a partir do conteúdo *web*, que é executado de forma interpretada no dispositivo móvel.

O Cordova também disponibiliza uma API para acesso aos diversos recursos nativos do dispositivo. Na versão 2.8.0, o Cordova suporta, entre outros, os recursos de acelerômetro, câmera, bússola, contatos, arquivos, GPS, banco de dados e globalização. Todos esses recursos da API podem ser invocados através do código JavaScript da aplicação de forma independente da plataforma, ou seja,

32 "PhoneGap", Adobe Inc. Disponível em: <<http://phonegap.com/>>. Acesso em: 13/06/2013.

sem necessidade de escrever código em linguagem nativa. Dessa forma, a aplicação mantém sua característica de multiplataforma, possibilitando que o mesmo código possa ser distribuído para qualquer plataforma suportada. Para isso, basta que seja criado um novo empacotamento para a plataforma alvo usando o conteúdo *web* padrão da aplicação.

O Cordova não é um ambiente integrado de desenvolvimento (IDE). Assim, para utilizá-lo, deve-se escolher uma plataforma inicial para o desenvolvimento, o que significa adotar a IDE padrão dessa plataforma. Assim, se a plataforma escolhida como base for o Android, será utilizado o Android Studio como ambiente de desenvolvimento. Os ambientes das outras plataformas serão montados e utilizados sob demanda apenas para testes específicos de funcionalidades nessas outras plataformas. Porém, caso o desenvolvimento seja realizado por uma equipe cujos membros possuem especialidades diversas (Android, iOS, Windows Phone, ...), cada membro pode montar e trabalhar no ambiente da plataforma que se sentir mais confortável, sincronizando o código do conteúdo *web* em um repositório comum, independente de plataforma. Nesse caso, é possível testar em várias plataformas à medida que o código *web* comum evolui.

2.2. Comparação entre aplicações híbridas e nativas

Para subsidiar uma tomada de decisão em relação ao desenvolvimento nativo ou híbrido, uma análise englobando aspectos técnicos e não técnicos foi conduzida e será apresentada na seção seguinte. Para realizar essa análise, foi implementada uma mesma aplicação em plataforma nativa (Android e iOS) e em multiplataforma. O desenvolvimento multiplataforma foi realizado com os *frameworks* Cordova e jQuery Mobile, apresentados na Seção 2.1.

Os aspectos técnicos considerados na avaliação dos dois tipos de desenvolvimento foram: (i) desempenho (*performance*) nas transições de tela e rolagens de lista, (ii) tamanho do arquivo executável da aplicação, (iii) integração visual da aplicação com a plataforma, (iv) acesso a banco de dados e API de persistência, (v) API de chamada a serviços e transferência de arquivos, (vi) facilidade de acesso a recursos do dispositivo e (vii) licença para uso comercial. A Tabela 1 apresenta o comparativo desses aspectos em arquitetura híbrida e nativa para uma mesma aplicação.

Tabela 1: Comparação dos aspectos técnicos entre arquiteturas nativa e híbrida para uma mesma aplicação

(continua)

Aspecto	Desenvolvimento Nativo	Cordova	Considerações
<i>Performance</i> (desempenho)	Na comparação com abordagens híbridas o desenvolvimento nativo sempre apresentou uma melhor <i>performance</i> .	Em um âmbito geral, aplicativos desenvolvidos com Cordova tendem a ter um desempenho menor quando comparados a um nativo. Contudo, este desempenho menor não é sempre perceptível. Aplicativos feitos em Cordova e que são bem estruturados e desenvolvidos podem passar despercebidos para usuários leigos. Verificou-se alguns pontos de lentidão nos aplicativos em Cordova. A abertura do aplicativo, por exemplo, é perceptivelmente mais lenta. A interação do usuário com os elementos de tela também demonstra uma lentidão maior.	A escolha da abordagem deve balancear necessidades de alta <i>performance</i> com o requisito multiplataforma. Pode ser necessário abrir mão de recursos mais avançados (por exemplo, muitas animações) em favor de uma aplicação mais simples que possa ser utilizada por diferentes dispositivos.

Tabela 1: Comparação dos aspectos técnicos entre arquiteturas nativa e híbrida para uma mesma aplicação

(continuação)

Aspecto	Desenvolvimento Nativo	Cordova	Considerações
Tamanho do Executável	Os executáveis gerados usando apenas recursos nativos da plataforma tendem a ter um tamanho menor. Neste caso, o tamanho final do executável é influenciado, principalmente, pelo tamanho e qualidade dos recursos utilizados, como imagens, sons e outros. Um "Hello World" feito em Android possui cerca de 175 KB.	Embora o Cordova também gere um executável nativo da plataforma, deve-se ter em consideração a adição de bibliotecas JavaScript para a realização de determinadas tarefas. Por exemplo, o jQuery Mobile, usado para criar a interface UI, necessita de três arquivos extras. Os arquivos possuem, aproximadamente, os tamanhos de 93 KB, 142 KB e 90 KB, totalizando 325 KB. Um "Hello World" em Cordova possui cerca de 364 KB. Adicionando a biblioteca do jQuery Mobile e estruturando a página inicial para utilizá-lo, o aplicativo fica com cerca de 689 KB.	Não foram encontradas diferenças significativas nos tamanhos dos executáveis de um projeto em Cordova de outro nativo. As bibliotecas JavaScript extras usadas no desenvolvimento com Cordova não apresentaram tamanho demasiado, de tal forma que torne o executável gerado para esta tecnologia significativamente maior ao que usa somente recursos nativos.
Aparência / Efeitos Visuais	No desenvolvimento nativo, todos os recursos de tela, transições e animações estão disponíveis para o desenvolvedor com o melhor desempenho possível para a plataforma.	Por utilizar tecnologias como HTML e CSS para a criação da interface, um aplicativo em Cordova não possui uma Interface do Usuário (UI) com aparência nativa da plataforma. É possível, no entanto, customizar imagens e estilos para obter-se uma aparência o mais próxima possível com a nativa de uma plataforma específica. Este aspecto pode ser encarado como uma vantagem ou desvantagem a depender do contexto. No caso de um projeto no qual se requer a UI com aparência e usabilidade mais próxima possível do dispositivo, o Cordova com jQuery Mobile mostra-se deficiente. Todos os demais recursos, como animação, transição de telas e outros, são fornecidos por ferramentas como o jQuery Mobile. É possível, também, criar ou utilizar plugins para o Cordova, bem como usar os próprios recursos existentes no HTML5, CSS3 e JavaScript para realizar efeitos visuais. O uso inadequado dessas tecnologias, contudo, pode implicar em recursos visuais com baixo desempenho, portanto deve-se ter cuidado ao utilizá-los.	A escolha entre uma opção ou outra depende bastante da necessidade de se ter um projeto com aparência e usabilidade nativas. Aplicativos em Cordova terão uma aparência igual em todas as plataformas, fato que pode ser uma vantagem, caso se queira manter um único padrão visual para o aplicativo. Deve-se buscar um padrão visual que seja bem aceito por usuários de diferentes plataformas.

Tabela 1: Comparação dos aspectos técnicos entre arquiteturas nativa e híbrida para uma mesma aplicação

(continuação)

Aspecto	Desenvolvimento Nativo	Cordova	Considerações
Acesso a banco de dados embarcado	Utiliza-se dos bancos de dados fornecidos pela plataforma nativa. Normalmente, essas plataformas utilizam o SQLite. O SQLite é uma solução bastante utilizada e com muitos recursos. Permite a utilização de transações, bancos de dados grandes, entre outras funcionalidades ³³ .	O Cordova provê uma API baseada na especificação W3C Web SQL Database para manipular um banco de dados embarcado. Na prática, as implementações dessa especificação têm utilizado o SQLite mesmo em navegadores desktop. Porém, a escolha da implementação é uma decisão do navegador e não pode ser garantida ³⁴ . Como a maioria das plataformas móveis utiliza o SQLite como banco de dados padrão, é normal que os navegadores móveis acompanhem essa tendência. O Cordova também disponibiliza uma API baseada na especificação W3C Web Storage que permite armazenar informações menos estruturadas (pares de chave e valor). Entretanto, foi observado que na implementação do Cordova para Android essa API também utiliza o banco de dados SQLite.	Tanto o desenvolvimento nativo como o desenvolvimento híbrido permitem que sejam armazenadas informações locais em banco de dados, inclusive com a possibilidade de criação de índices para melhorar o desempenho de consultas. Um banco de dados SQLite é limitado, teoricamente, ao tamanho de 2 terabytes, embora seja improvável que o sistema de arquivos permita esse tamanho para um único arquivo. Além disso, a capacidade de armazenamento dos dispositivos móveis ainda não chegou a esse volume de dados ³⁵ .
Chamada a serviços / Transferência de arquivos	As plataformas nativas não possuem suporte nativo a SOAP. Para o uso com serviços REST não há problemas, uma vez que trata-se de uma conexão HTTP comum. No caso do Android existe a biblioteca ksoap-android, que facilita o consumo de serviços SOAP. Em iOS, não foram encontradas bibliotecas para facilitar o acesso a serviços SOAP, portanto há que se ter um maior conhecimento do protocolo para desenvolver a solução.	Através do Cordova é possível realizar conexões HTTP para ter acesso a serviços REST e SOAP.	Em ambos os casos, não existem impedimentos para o uso de serviços SOAP, mas pode haver a necessidade de se desenvolver ou utilizar bibliotecas para facilitar o uso.

33 "SQLite", SQLite Consortium. Disponível em: <<http://www.sqlite.org/>>. Acesso em: 13/06/2013.

34 "W3C Web SQL Database", W3C. Disponível em: <<http://dev.w3.org/html5/webdatabase/>>. Acesso em: 13/06/2013.

35 "Appropriate Uses For SQLite", SQLite Consortium. Disponível em: <<http://www.sqlite.org/whentouse.html>>. Acesso em: 13/06/2013.

Tabela 1: Comparação dos aspectos técnicos entre arquiteturas nativa e híbrida para uma mesma aplicação

(conclusão)

Aspecto	Desenvolvimento Nativo	Cordova	Considerações
Acesso a recursos nativos	Cada plataforma nativa possui recursos que são bastante específicos. Há determinadas funcionalidades no Android, por exemplo, que não existem no iOS. Neste caso, uma mesma funcionalidade pode ser desenvolvida usando recursos diferentes em cada plataforma.	Há determinadas funcionalidades disponíveis nativamente nas plataformas que, em muitos casos, não estão disponíveis no Cordova. A lista completa dos recursos disponíveis por plataforma pode ser encontrada na página oficial do Cordova ³⁶ . Em Android, por exemplo, não está disponível o acesso a Content Providers, Serviços e Intents a partir do Cordova. Para ter estas funcionalidades faz-se necessário escrever plugins para o Cordova, o que acopla a aplicação a código nativo.	Em casos onde há a necessidade de um grande uso de recursos nativos, como serviços, no caso do Android, deve-se considerar a escolha da plataforma nativa. Caso a abordagem multiplataforma venha a implicar em muita especificidade no código, verificando o que pode ser utilizado em cada plataforma, o desenvolvimento nativo deve ser considerado. Um código-fonte com muitas expressões condicionais pode tornar-se difícil de manter e evoluir.
Licença	Cada plataforma nativa provê suas próprias licenças, contudo, todas permitem a criação de aplicativos comerciais.	A licença do Cordova não impõe limites quanto à criação de aplicativos comerciais.	Em ambos os casos, a licença não é um problema para a criação de aplicativos comerciais.

A Tabela 2 apresenta a comparação dos aspectos não técnicos relacionados ao desenvolvimento para as arquiteturas híbrida e nativa. Os aspectos não técnicos considerados na comparação foram: (I) custo, esforço e prazo para desenvolvimento em cada arquitetura, (II) restrições para implementação de requisitos de negócio, e (III) necessidades de capacitação nas tecnologias relacionadas.

³⁶ "Apache Cordova", Apache Foundation. Disponível em: <<http://cordova.apache.org/>>. Acesso em 13/6/2013.

Tabela 2: Comparação dos aspectos não técnicos entre arquiteturas nativa e híbrida para uma mesma aplicação

Aspecto	Desenvolvimento Nativo	Cordova	Considerações
Custo /Esforço / Prazo	Os mesmos requisitos funcionais devem ser implementados para cada plataforma nativa, portanto esforço, custo e prazo tendem a ser maiores do que no Cordova. Nesse caso, geralmente haverá uma equipe de desenvolvedores especializada em cada plataforma, e cada desenvolvedor deve ter seu ambiente (eventualmente <i>hardware</i> e <i>software</i> específicos) disponível.	Nesta abordagem, o código-fonte é desenvolvido apenas uma única vez e executará em todas as plataformas escolhidas dentre as disponíveis. Vale salientar, no entanto, que será necessário preparar <i>builds</i> específicos para cada plataforma, o que pressupõe a utilização dos equipamentos e ambientes de desenvolvimento exigidos por cada uma delas. Ou seja, basta disponibilizar um ambiente para cada plataforma, em vez de um ambiente para cada desenvolvedor.	Esforço, custo e prazo usando o desenvolvimento nativo tendem a ser maiores do que no Cordova. Caso o desenvolvimento nativo seja escolhido, a estratégia de colocar equipes distintas trabalhando em cada plataforma pode ser considerada, como forma de diminuir o impacto no prazo (aumentando, porém, o custo).
Implementação de requisitos de negócio	Sem restrições (semelhante ao desenvolvimento de aplicação <i>desktop</i>).	Sem restrições (semelhante ao desenvolvimento de aplicação <i>web</i> para <i>desktop</i>).	
Capacitação	Equipes de desenvolvimento precisam ser capacitadas em cada tecnologia nativa escolhida. No caso do Android, a linguagem Java e o ambiente de desenvolvimento Eclipse já são amplamente utilizados no mercado e na academia. Para o ambiente iOS, normalmente é necessário capacitar a equipe na linguagem Objective-C e no ambiente de desenvolvimento integrado (IDE) Xcode. Isso vale para outras plataformas nativas (ex: Windows Phone, <i>BlackBerry</i> , etc), cujas ferramentas e ou tecnologias associadas não são tão bem difundidas no mercado/academia. A carga horária estimada para um treinamento de Android é de 40h e pressupõe o conhecimento prévio da linguagem Java. Um treinamento de Objective-C/Xcode é estimado entre 36h e 48h.	Existe alto grau de reaproveitamento do conhecimento em desenvolvimento <i>web</i> . A utilização do Cordova pressupõe o uso dessas tecnologias, adicionando-se o conhecimento específico da API do Cordova, a qual permite a execução de recursos nativos do dispositivo. Pressupondo o conhecimento prévio de HTML e CSS, seria necessário apenas um treinamento de jQuery Mobile e Cordova, cuja carga horária normalmente não excederia 16h.	O desenvolvimento nativo implica em um maior investimento em capacitação, ainda mais quando a solução precisar ser desenvolvida em mais de uma plataforma nativa. Se o projeto é direcionado a apenas uma plataforma, a decisão deve observar outros aspectos. Se o projeto visa desenvolver para mais de uma plataforma, o Cordova passa a se tornar atrativo.

3. Adaptando o design ao contexto

Além da portabilidade entre plataformas, uma questão importante a ser tratada é a diversidade de dispositivos móveis e, principalmente, o tamanho de tela de tais dispositivos. Devido às limitações de tamanho de tela em relação ao tradicional *desktop*, recomenda-se que o desenvolvimento de uma aplicação comece pela sua versão móvel³⁷. Assim, é possível priorizar as principais funcionalidades e partir de uma interface pequena e mais limitada (no dispositivo móvel) para uma maior (em um *desktop*). Esta abordagem de pensar primeiro o conteúdo para o dispositivo menor é ilustrada na Figura 4, na qual se percebe o aumento de conteúdo disponibilizado à medida que o tamanho da tela aumenta.

O trabalho de construção da interface de uma aplicação móvel híbrida é similar ao de um *Web Designer* ao construir uma página *web*. E o trabalho deste, por sua vez, não é muito diferente daquele de qualquer outro artista que trabalha com desenhos, ilustrações e tipografia. Em todos os casos, os artistas possuem uma área de trabalho, normalmente conhecida como *canvas*. Para um pintor, o *canvas* é o quadro de pintura, enquanto para um ilustrador que cria uma capa de revista, seu *canvas* é a folha. Da mesma forma, para um *Web Designer*, o navegador *web* é sua área de trabalho (seu *canvas*)³⁸.

Entretanto, diferente de todos os demais, o *Web Designer* deve conviver com uma incerteza: seu *canvas* não define um limite, um tamanho mínimo ou padrão. De fato, o *Web Designer* está à mercê das configurações do navegador utilizado. Durante muito tempo, houve tentativas para vencer esta barreira. A principal consistia em impor algumas restrições. Não era incomum encontrar sítios na internet com as seguintes palavras em seu rodapé: desenvolvido

37 "Mobile First", WROBLEWSKI, Luke. ISBN 978-1-937557-02-7. A Book Apart, 2011. New York, NY.

38 "Responsive Web Design", MARCOTTE, Ethan. Jeffrey Zeldman, 2011. New York, NY.

para 800x600. Depois que esta resolução tornou-se ultrapassada, a frase mudou para “desenvolvido para 1024x768”.

Mas, em um mundo no qual a diversidade de dispositivos e de resoluções crescem sem parar, esta tática não é mais viável. Então, como criar um sítio sem restrições de resolução? Como ter um sítio que pode ser visto em qualquer dispositivo, conforme a área disponível para ele? É imprescindível ter em mente que o mesmo sítio poderá ser visto tanto de um *tablet* como de um *smartphone* ou *desktop*. Para transpor este problema, algumas técnicas isoladas foram sugeridas. A junção de algumas delas originou o nome *Responsive Web Design*, criado por Ethan Marcotte³⁹.

Muito antes deste nome ser criado, entretanto, um expoente do *Design para Web*, chamado John Allsopp, já havia alertado: “o controle que os designers conhecem na mídia impressa, e sempre desejado na web, é simplesmente uma função da limitação da página que será usada para a impressão. Nós devemos abraçar o fato de que a web não tem estas restrições e projetar com base nesta flexibilidade⁴⁰”.

Neste contexto, cabe ao *Web Designer* projetar páginas que se adaptam ao dispositivo que o está exibindo, no lugar de criar sítios específicos e isolados para cada resolução. As ferramentas para tornar isto possível já existem e são conhecidas: *flexible grid*, *flexible typesetting*, *flexible media* e *media queries*⁴¹. Tais ferramentas são comentadas nas seções seguintes.

3.1. Tipografia e tabelas flexíveis

O primeiro passo para projetar uma página flexível é não usar uma unidade de tamanho que manterá o mesmo aspecto

39 “Responsive Web Design”, Ethan Marcotte. A Book Apart, 2011. New York, NY.

40 “A Dao of Web Design”, John Allsopp. Disponível em: <<http://alistapart.com/article/dao>>. Acesso: em 13/06/2013.

41 “Responsive Web Design”, MARCOTTE, Ethan. Jeffrey Zeldman, 2011. New York, NY.

(ou tamanho absoluto) para fontes e para a área que manterá o conteúdo, independentemente de resolução. Faz-se extremamente necessário definir esses tamanhos usando uma unidade que mude as proporções conforme o dispositivo em que será apresentado o sítio. Neste contexto quando o usuário abrir a página em um *tablet* ou *desktop*, esses tamanhos serão ajustados conforme a necessidade. Definir um tamanho fixo para a fonte, por exemplo, significa que o texto pode ser perfeitamente legível em um *smartphone*, mas o usuário encontrará dificuldades para lê-lo em um *desktop*.

Para definir o tamanho de textos é importante usar a unidade conhecida como *em*. Esta unidade não define um tamanho absoluto, mas variável, que determina o tamanho real baseado em uma função que usa a fonte padrão do navegador do usuário como argumento.



Figura 1: Leiaute de uma página com tamanhos definidos em pixel.

É natural que, inicialmente, quando está sendo criado o leiaute da página em um editor gráfico (ver Figura 1), defina-se todos os tamanhos em *pixels*, incluindo o tamanho das fontes. Isto ocorre porque no editor gráfico tem-se um tamanho fixo de área para

definir o leiaute. Contudo, é imprescindível, quando for traduzir este leiaute para HTML, transformar esta unidade do texto em *em* (unidade de medida relativa disponível no CSS: 1em é igual ao tamanho padrão do elemento).

Para alcançar este objetivo, faz-se necessário adotar uma fórmula conhecida dos web designers: *alvo / contexto = resultado*. O *alvo* é o tamanho da fonte definido quando o leiaute foi criado no editor gráfico. Pode-se supor, para fins de exemplificação, que este valor é de 24 *pixels*. O valor a ser utilizado na variável *contexto* será 16 e o motivo de usar este valor é que se trata do tamanho padrão usado pelos navegadores para os textos. Neste exemplo, o resultado será $24/16=1.5em$, e deve ser definido usando código CSS, conforme demonstrado na Figura 2.

```
h1 {  
    font-size: 1.5em; /* 24px / 16px */  
    font-style: italic;  
    font-weight: normal;  
}
```

Figura 2: Definição de tamanho do texto em CSS.

Assim como ocorre com os textos, é necessário definir o próprio leiaute da página usando uma unidade de medida também flexível. Para este caso, entretanto, usamos a unidade de porcentagem. Similar ao que foi adotado no exemplo das fontes, o leiaute criado também estará usando uma medida em *pixel* e precisará ser transformado para porcentagem. A mesma fórmula será usada. Para a área total do leiaute, pode-se fazer uma regra com a área total da tela em uma resolução de 1024x768 *pixels*. Caso o leiaute no editor gráfico tenha a largura de 960 *pixels*, teremos $960/1024 = 93,75\%$.

Para as demais áreas internas que possuem conteúdo, esta mesma regra deve ser adotada, contudo, usando agora a medida de 960 *pixels* no lugar da variável *contexto*.

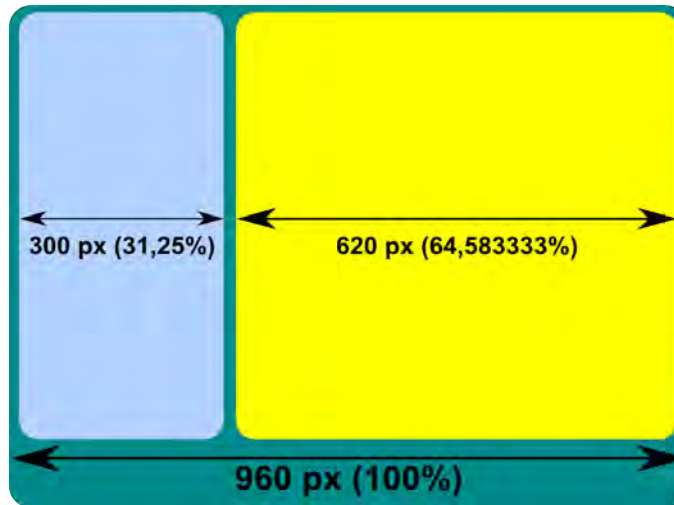


Figura 3: Definição de um grid flexível.

Neste caso, a Figura 3 exemplifica duas áreas internas do leiaute, na qual há uma área de 620px e outra de 300px. A primeira precisa ser expressa com o valor de 64%, devido ao resultado da fórmula $620/960=0,64583333$.

Adotando esta estratégia para definir o leiaute em HTML, pode-se ter um sítio adaptável ao tamanho da tela em que será exibido. O conteúdo será reajustado para caber na tela. Contudo, ainda são necessárias outras técnicas para tornar este sítio totalmente flexível.

3.2. Mídias flexíveis

A portabilidade de imagens e vídeos é outra fonte de preocupação. Uma imagem pode ser de bom tamanho e resolução

para um *desktop*, mas ruim para visualização em um *smartphone*. Neste caso, o problema é agravado também pelo tamanho físico da imagem. Para o *desktop*, ela pode ser grande e ainda assim não causará problemas. Mas, para o *smartphone*, ser grande significa consumir desnecessariamente dados.

Para resolver a questão da resolução da imagem, utiliza-se uma característica do CSS para contornar. Trata-se do atributo *max-width*. Este atributo é suportado pelos principais navegadores tanto de desktop quanto de dispositivos móveis, entre eles: Chrome, Firefox, Internet Explorer, Opera e Safari⁴².

Seu funcionamento é bastante simples e facilita o trabalho dos *Web Designers*. Quando usado em uma *tag *, ele define que a imagem deve ocupar o tamanho disponível para ela, aumentando e diminuindo conforme ocorrem mudanças no tamanho da janela do navegador. Este atributo também pode ser usado em outras *tags* que definem mídias, como *tags* específicas para a exibição de vídeos.

3.3. *Media Queries*

O último ingrediente para criar um sítio totalmente responsivo são as *Media Queries*. Trata-se de uma característica que surgiu a partir da versão 3 do CSS. Contudo, na versão 2 já existia uma característica muito similar, denominada *Media Types*. Embora seja possível criar sítios com um bom grau de flexibilidade usando apenas as duas características já citadas nos tópicos anteriores, ainda há um problema que apenas as *Media Queries* podem tratar: diferentes resoluções e dispositivos impõem organização de leiaute diferente para que a informação esteja bem organizada e fácil de ser acessada.

42 "W3C Schools", W3C. Disponível em: <http://www.w3schools.com/cssref/pr_dim_max-width.asp>. Acesso em: 24/06/2013.

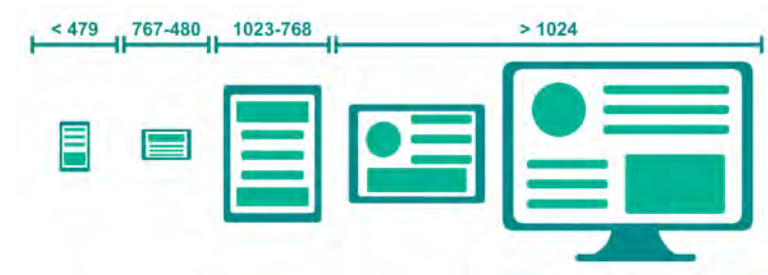


Figura 4: Leiautes diferentes para cada dispositivo.

Fonte: adaptado de *What is Responsive Design*⁴³.

As *Media Queries* resolvem isso separando pedaços de código CSS específicos para determinadas resoluções. Com isso, é possível definir o que está visível ou não, a disposição dos elementos na tela, quais imagens devem ser exibidas, entre outras muitas possibilidades.

A Figura 4 exemplifica um sítio que se adapta, alterando totalmente seu leiaute, para exibir as informações da melhor forma para cada tipo de dispositivo e seu tamanho de tela. Isto é possível adotando o recurso de *Media Queries*, conforme pode ser visto na Figura 5, que demonstra o uso deste recurso em uma página HTML com CSS.

```
@media screen and (max-width: 768px) {
  .main-title {
    font: normal 1.5em Calibri, Candara, Segoe, »
      "Segoe UI", Optima, Arial, Helvetica, »
      sans-serif; /* 24px / 16px */
  }
}
```

Figura 5: Utilização do recurso de *Media Queries*.

43 "What is Responsive Design". Disponível em: <<http://vinaganda.com/what-is-responsive-design/>>. Acesso em: 24/06/2013.

4. Conclusões

Esse artigo apresentou algumas características do mercado de mobilidade, destacando a questão da diversidade de plataformas e os desafios que esta impõe às empresas que buscam desenvolver aplicativos para este mercado. Além de questões relacionadas à portabilidade das aplicações, questões relacionadas a variações de tamanho e forma de dispositivos também precisam ser tratadas.

Como forma de minimizar o impacto dessa diversidade, algumas alternativas tecnológicas para desenvolvimento multiplataforma foram apresentadas. Essas soluções tecnológicas visam à construção de aplicativos móveis uma única vez, mas que podem ser distribuídos para execução em múltiplas plataformas e em diferentes dispositivos.

A primeira solução comentada foi o *framework* jQuery Mobile, que abstrai as especificidades dos navegadores *web* de cada plataforma, possibilitando a construção de um único conteúdo compatível com uma ampla gama de plataformas.

A segunda solução para mobilidade multiplataforma apresentada foi o *framework* Apache Cordova. Esse *framework* possibilita a construção de aplicativos móveis híbridos, ou seja, que são construídos com tecnologia *web* (i.e., HTML, CSS e JavaScript) e são empacotados como aplicativos nativos para cada plataforma. Essa solução amplia as possibilidades de acesso a recursos do dispositivo, como câmera, acelerômetro, bússola, entre outros, e pode ser utilizada em conjunto com *frameworks* *web* como o jQuery Mobile.

Adicionalmente, considerando que para dispositivos móveis, além da portabilidade, a questão da diversidade de telas também é importante, foram apresentadas técnicas para adaptar o conteúdo *web* a vários contextos (tamanho e resolução de tela, principalmente). Essas técnicas em conjunto correspondem ao que se denomina design responsivo.

Por fim, o artigo discutiu também os resultados de uma observação empírica do desenvolvimento de uma solução multiplataforma Android e iOS. Sob diversos critérios estabelecidos, o protótipo multiplataforma foi comparado com o mesmo protótipo desenvolvido como aplicação nativa. A observação indicou que para aplicações mais simples, o desenvolvimento multiplataforma pode ser mais indicado. Esta constatação é muito importante no contexto de desenvolvimento de aplicativos para governo. Isto porque, além da preocupação com a otimização de recursos, é preciso uma preocupação com a diversidade de plataformas que os cidadãos utilizam no país e a importância dos serviços de governo estarem disponíveis para o maior número possível de pessoas.



COLABORAÇÃO E SOFTWARE LIVRE

- *Fabio Kon*⁴⁴
- *Carlos D. Santos*⁴⁵

44 Centro de Competência em Software Livre, IME-USP.

45 Departamento de Administração, UnB.



A célebre frase escrita por Isaac Newton em suas cartas, “Se fui capaz de enxergar mais longe, foi porque estava me apoiando em ombros de gigantes”, deriva de uma tradição de vários séculos da ciência. As descobertas científicas e as inovações tecnológicas não são nunca resultado de um trabalho de uma só pessoa. Qualquer nova ideia resulta da união de ideias mais antigas e, quase sempre, da colaboração entre vários indivíduos refletindo e discutindo sobre um determinado assunto em um certo contexto. São raríssimos os casos de grandes cientistas ou inovações tecnológicas surgidas em um ambiente de isolamento a partir de um indivíduo desconectado de uma rede de relacionamentos e colaborações. Os grandes avanços no conhecimento humano foram produzidos por ambientes de grande colaboração em que pensadores puderam compartilhar ideias, discutir, questionar, discordar, refutar e contrapor seus pontos de vista.

1. Colaboração e Ciência Aberta

Como forma de maximizar a colaboração em ciência em escala planetária, a ideia de que os resultados da pesquisa financiada com dinheiro público devem ser disseminados amplamente de forma aberta está ganhando cada vez mais força na comunidade científica internacional. Algumas das principais universidades do planeta, tais como MIT e Harvard, governos, tais como os dos EUA e da Inglaterra e agências de fomento, tais como FAPESP, European Commission e NSF, têm começado a incentivar

explicitamente a Ciência Aberta (*Open Science*). No entanto, em algumas áreas, ainda existe uma forte resistência de grupos de pesquisa acostumados a esconder os seus dados e procedimentos de trabalho. Em alguns casos, o segredo é buscar uma forma de vantagem comercial – o que, dependendo do contexto, não é necessariamente uma má prática, às vezes é uma imposição dos financiadores da pesquisa. Em outros casos menos nobres, é uma forma de evitar críticas e esconder possíveis fragilidades do trabalho. Finalmente, em muitos casos, as informações deixam de ser disponibilizadas de forma aberta simplesmente por inércia e falta de uma cultura de abertura e compartilhamento.

Os três pilares da Ciência Aberta que permitem aumentar a intensidade e a qualidade da colaboração entre grupos de pesquisa são:

- **Acesso aberto**, i.e., as descobertas científicas devem ser publicadas em artigos de acesso aberto de forma que qualquer interessado, em qualquer país, tenha acesso imediato e irrestrito. Uma forma de viabilizar isso é através de revistas de acesso aberto onde os artigos podem ser obtidos gratuitamente através de portais web acessíveis a todos.
- **Dados abertos**. É fundamental que os grupos que desejem fazer ciência de alta qualidade divulguem de forma aberta os dados que utilizaram para realizar os seus experimentos ou análises. *Open Data* não significa somente disponibilizar centenas de megabytes em um sítio web qualquer, mas também oferecer metadados e documentação de boa qualidade utilizando, sempre que possível, padrões internacionais de codificação de informação, facilitando a reutilização dos dados por terceiros. Dessa forma é possível garantir a reprodutibilidade dos experimentos e das análises científicas e permitir a colaboração efetiva entre grupos espalhados pelo globo.

- **Ferramentas abertas.** As ferramentas de processamento dos dados (abertos), que são utilizadas para filtragem, consolidação, análise e visualização da informação, devem ser disponibilizadas como software livre, ou seja, um software com código-fonte distribuído segundo uma licença aprovada pela *Open Source Initiative* (OSI) ou *Free Software Foundation* (FSF), preferencialmente sob uma das licenças populares listadas em <www.opensource.org/licenses>.

2. Colaboração e o avanço da Computação

Voltando nossas atenções à evolução da Ciência da Computação e da Tecnologia da Informação nos últimos 80 anos, vemos que a quase totalidade dos grandes avanços originaram-se em ambientes de trabalho altamente colaborativos.

Alan Turing, o pai da Ciência da Computação, era um indivíduo introvertido e que formulou parte de suas revolucionárias teorias trabalhando horas a fio em suas notas pessoais. No entanto, ele era uma figura muito querida por seus colegas e que se beneficiou enormemente pelos ambientes de forte colaboração intelectual por quais passou, a partir da década de 1930, incluindo King's College, Cambridge, Princeton University e Bletchley Park. Suas colaborações nestas instituições formaram a base para a ciência e tecnologia da computação que temos hoje.

Também na década de 1930, Bill Hewlett e Dave Packard formaram-se na Universidade de Stanford e criaram uma empresa numa pequena garagem de Palo Alto, Califórnia, sob a orientação e tutoria de Frederick Terman, professor de Stanford. Ao redor dessa empresa, posteriormente conhecida como Hewlett-Packard, ou simplesmente HP, formou-se um aglomerado de pequenas, médias e grande empresas e laboratórios de pesquisa, hoje conhecido como Vale do Silício. A proximidade física de milhares de pessoas

compartilhando informações, trocando ideias, cooperando e competindo, ou seja, trabalhando colaborativamente em torno de inovação tecnológica, levou ao ambiente mais prolífico para inovação em TI que se tem notícia.

Essa colaboração atingiu um nível de inovação sem precedentes em toda a história da Computação quando um grupo de cientistas e técnicos se reuniu no Vale do Silício dentro do XEROX Palo Alto Research Center (PARC). No PARC, entre 1970 e 1996, foram inventados nada menos que o mouse, a interface gráfica, o sistema de janelas, a rede Ethernet, a impressora laser, o editor de textos WYSIWYG, a linguagem orientada a objetos Smalltalk, o padrão MVC de arquitetura de software interativo e a programação orientada a aspectos, entre muitas outras inovações.

3. Colaboração e Software Livre

Com a popularização da Internet a partir da década de 1990, o alto nível de colaboração diário que antes só podia ser encontrado em lugares fisicamente definidos, como Bletchley Park, o Vale do Silício ou o XEROX PARC, passou a ser viável no mundo virtual da teia global (Wasserman, 2011).

Ao identificar as dificuldades de colaboração que o software restrito e fechado começavam a trazer ao mundo da tecnologia, Richard Stallman funda, em 1983, o projeto GNU e, logo em seguida, desenvolve a ideia do Software Livre. Sua proposta era utilizar o trabalho voluntário de centenas de programadores ao redor do globo para colaborativamente construir um sistema computacional completo incluindo todos os componentes, desde o núcleo do sistema operacional até suas bibliotecas, ferramentas e aplicativos.

Um dos primeiros programas de sucesso desenvolvido nesse modelo foi o editor de texto Emacs, um poderoso programa

extensível e configurável desenvolvido pelo próprio Richard Stallman. Utilizando a metáfora proposta por Eric Raymond (1997), dizemos que o Emacs foi desenvolvido por meio de um modelo *Catedral* de colaboração. Assim como as grandes catedrais góticas europeias, que eram projetadas por um renomado arquiteto que gerenciava todo o seu processo de construção ao longo de vários anos, Stallman foi o grande arquiteto do Emacs, controlando todos os detalhes de sua implementação e definindo os próximos passos do seu desenvolvimento.

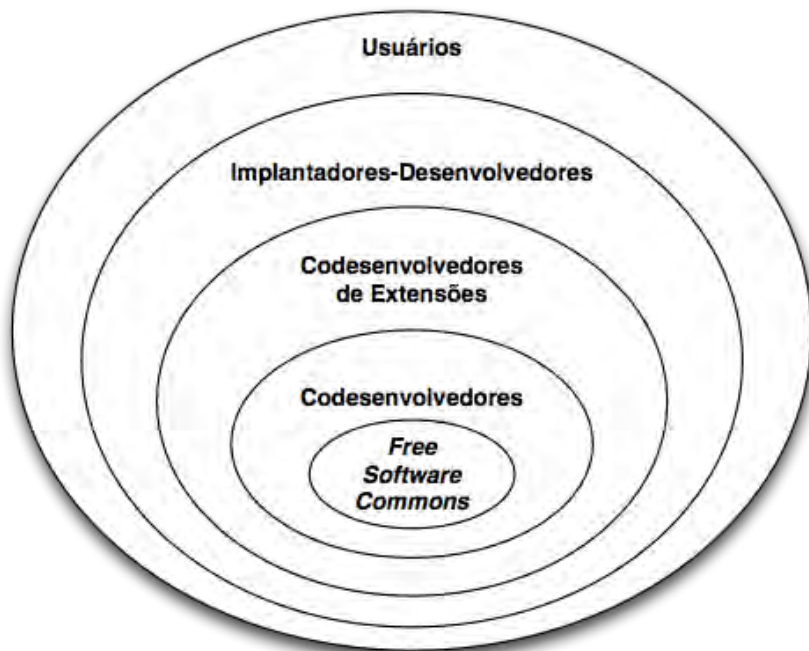
Já o sistema Linux, por sua vez, nos demonstrou que outro modelo de colaboração, mais aberto e caótico, poderia também levar a resultados poderosos e de alta qualidade. O Linux foi desenvolvido no modelo *Bazar* (Eric Raymond, 1997). Linus Torvalds liberou o seu código na Internet em 1991 e convidou programadores de todo o mundo a colaborarem com o seu desenvolvimento. No ano seguinte, Torvalds transformou o Linux em software livre associando a ele a licença GPL. A partir daí, uma grande rede de colaboração internacional composta por milhares de programadores e usuários emergiu e, sob a liderança do “ditador benevolente” Linus Torvalds, auto-organizou-se e formou um ecossistema de colaboração internacional que levou à criação de um dos sistemas operacionais mais robustos e sofisticados da atualidade, o sistema GNU/Linux. No mundo do GNU/Linux, milhares de voluntários e profissionais de empresas trabalham sem um plano global definido e sem uma hierarquia organizacional rígida, a organização do projeto é muito bem estruturada, mas segue um modelo diferente daquele encontrado nas corporações tradicionais. Milhares de melhorias ao sistema são desenvolvidas todos os meses e algumas delas são incorporadas ao núcleo central do sistema após a aprovação de Torvalds ou de alguém designado por ele. Esse modelo de bazar funciona como uma feira livre onde todos têm o direito de questionar os rumos do projeto e

apresentar novas soluções. A partir dessa comunidade vibrante e caótica, pessoas-chaves da comunidade são incumbidas de identificar os focos de inovação mais interessantes e incorporá-las ao núcleo do sistema. A partir de 2000, criou-se uma ONG, a Linux Foundation, com o intuito de promover, proteger e padronizar o desenvolvimento do Linux garantindo que a evolução do sistema não dependa unicamente de Linus, seu criador.

Simon Phipps, presidente da Open Source Initiative (OSI), descreve esse modelo de colaboração como uma estrutura em camadas conforme ilustrado na Figura 1. No núcleo do modelo temos o *Free Software Commons*, ou seja, o conhecimento compartilhado por toda a comunidade, normalmente, sob a forma de software livre e documentação aberta. Em seguida, temos uma camada de codesenvolvedores que atuam diretamente no desenvolvimento do software, possuem permissão para realizar *commits* diretamente em um repositório de código-fonte e têm a autonomia para decidir os rumos de um projeto específico e decidir qual contribuição da comunidade mais externa deve ser aceita ou não; tipicamente o número de codesenvolvedores centrais de um projeto varia entre um e uma dezena. Depois, temos a camada de codesenvolvedores de extensões, que pode variar entre algumas dezenas até centenas de programadores nos projetos maiores. Essa camada é composta por indivíduos capazes de compreender o código-fonte do projeto e implementar extensões, *plug-ins*, melhorias, localização, etc. Em seguida, temos uma camada ainda maior de implantadores-desenvolvedores, que são profissionais ou usuários que instalam e utilizam o software em suas organizações (empresas, universidades, órgãos públicos, etc.) ou para fins pessoais. Normalmente, são pessoas com um bom conhecimento de TI e, portanto, capazes de escrever pequenos trechos de código com melhorias específicas ou consertos de erros e fornecer informações técnicas relevantes para as camadas mais internas

do modelo de forma a aprimorar o software. Finalmente, na camada mais externa, aparecem os usuários finais que, em grandes projetos, podem chegar a milhares ou até milhões de indivíduos que também contribuem ao projeto fornecendo um *feedback* sobre os pontos fortes e fracos do programa, eventuais defeitos e também solicitações de novas funcionalidades.

Figura 1. Camadas de uma comunidade de colaboração em Software Livre



4. Repositórios de Software, Dados e Artigos

O conhecimento, ou a fonte para seu surgimento e aprimoramento, se materializa em diversos artefatos na Internet. Artigos, aplicativos e bases de dados são algumas das principais materializações encontradas. Esses artefatos do conhecimento estão disponíveis em variações de flexibilidade para a evolução colaborativa, ou seja, alguns têm permissão de escrita enquanto outros não têm nem de leitura (gratuita). Quando o primeiro autor opta por liberar o aparecimento e incorporação de coautores – habilitar a escrita no artefato – ele se torna um “common”, ou bem público, disponível em repositórios equipados com ferramentas que dão suporte ao seu aprimoramento compartilhado, assíncrono e regulado por normas técnicas e valores culturais.

A utilização de repositórios pelos criadores de bens públicos oferece a eles uma opção de baixo custo para encontrar coautores e usuários. Ferramentas básicas de comunicação (e de memória dessa comunicação) estão disponíveis e a popularidade do repositório acaba se refletindo na imagem do artefato, que passa a poder ser encontrado por meio de buscas.

Entre os serviços criados para atender essa demanda por suporte ao desenvolvimento público de artefatos, os mais populares atualmente são a Wikipedia, para artigos (administrado pela fundação Wikimedia), o SourceForge.net, para software livre (Dice Holdings, S.A.), e as dezenas de repositórios de bases de dados, que podem ser encontrados nas várias áreas de conhecimento, sendo administrados por organizações como o Banco Mundial⁴⁶ e o governo brasileiro⁴⁷, por exemplo. A praticidade de tal solução e o apelo mercadológico de seu fornecimento revelou a magnitude da intenção de compartilhar da sociedade.

46 <<http://sdwebx.worldbank.org/climateportal/index.cfm>>.

47 <<http://dados.gov.br/>>.

Hoje em dia, a Wikipedia hospeda cerca de 12 milhões de artigos⁴⁸, o SourceForge mais de 300 mil programas⁴⁹ e a Freebase, 39 milhões de tópicos (i.e., entidades)⁵⁰. Apesar de enormes, esses números dos repositórios mais populares não representam bem as iniciativas menos comunitárias e amadoras, que são significativas em quantidade e impacto social. Temos importantes repositórios e organizações envolvidas na empreitada de curadoria pública de artefatos tecnológicos. Por exemplo, a fundação Apache possui um portfólio de mais de 100 projetos em um ecossistema que envolve, além dos voluntários, Yahoo!, Microsoft, Facebook, dentre outros⁵¹. Além disso, temos o software público brasileiro do governo federal (com mais de 60 programas), o Java.net com mais de dois mil projetos e os 593 repositórios de bases de dados que podem ser encontrados no Databib⁵², um metabuscador hospedado na Universidade de Purdue.

Os números impressionam, mas devemos estar atentos ao fato de que a maior parte desses milhões de artefatos não supera os primeiros estágios de desenvolvimento, isto é, não chegam a ser validados e aprimorados ciclicamente em interações sociais de grande escala, podendo assim entrar para o rol de conhecimento “duro”, científico. Parte desse efeito *quase-todos-falham* pode ser explicado através de argumentos competitivos, pois não há oferta de mão de obra suficiente para trabalhar no desenvolvimento de todos os artefatos. Mas essa explicação é incompleta, pois nós sabemos que a distribuição de contribuições entre projetos similares, que atendem uma mesma demanda, não é equitativa, e segue um padrão de redução exponencial, ou seja, muitos recebem

48 <http://www.wikidata.org/wiki/Wikidata:Main_Page>.

49 <<http://sourceforge.net/about>>.

50 <<https://developers.google.com/freebase/faq>>.

51 <<http://www.apache.org/foundation/thanks.html>>.

52 <<http://databib.org/>>.

pouca ou nenhuma contribuição, a quantidade de projetos vai diminuindo a taxas cada vez maiores à medida que a quantidade de contribuições recebidas aumenta e pouquíssimos chegam a recebê-las em quantidade e qualidade significativas.

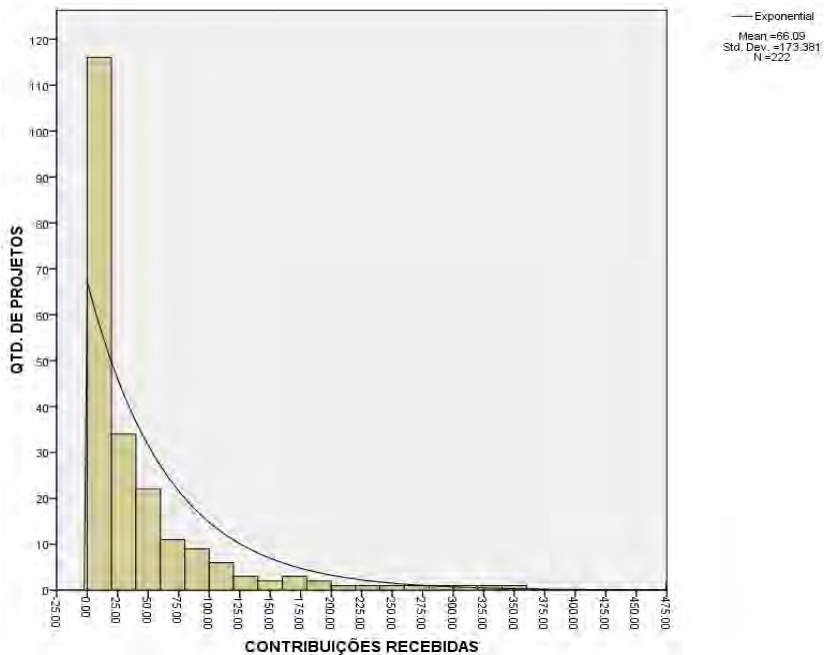
Para ilustrar esse ponto, obtemos e analisamos todos os projetos do SourceForge em 2008 do tipo “banco de dados”. A amostra encontrada foi de 4.105 projetos deste tipo, sendo que apenas 222 destes receberam pelo menos uma contribuição⁵³. Dentre os 222 projetos ativos⁵⁴, quase 120 receberam até 25 contribuições, sendo seguidos por cerca de 35 projetos que receberam aproximadamente 40 contribuições. Na sequência, cerca de 20 projetos receberam até aproximadamente 60 contribuições e o padrão de redução da quantidade de projetos segue de forma contínua, em uma distribuição que se aproxima da exponencial, com a quantidade de projetos diminuindo gradativamente à medida que as contribuições recebidas aumentam (veja Figura 2).

Considerando nossas pesquisas anteriores (Santos, Kuk, Kon, Pearson, 2013; Meirelles et al., 2010) e o padrão observado para os projetos de bancos de dados agora relatados, estimamos que algo como 5% do total de projetos criados se tornam bem-sucedidos nos termos discutidos.

53 Contribuições representam a soma das quantidades de erros reportados, requisições de suporte e novas funcionalidades e *patches* submetidos.

54 Ativos são aqueles projetos que receberam pelo menos uma contribuição ao longo de sua existência no repositório (a média de idade dos projetos da amostra de bancos de dados é de 5,7 anos – o mais novo tem 3,4 e o mais velho, 8,2 anos). O total de projetos encontrados no Sourceforge na época da pesquisa foi de 143.591. Com nossa definição de “ativo”, a quantidade de projetos passou a apenas 4.661.

Figura 2: Distribuição de contribuições recebidas em 222 projetos de software livre ativos do tipo banco de dados



Fonte: Amostra obtida do SourceForge.net via SRDA⁵⁵.

Essa distribuição desigual de contribuições por projetos implica que alguns são mais atrativos do que outros na percepção de quem decide como e onde contribuirá, ainda que dentro de um único domínio de aplicação. Alguns poucos projetos concentram, e atraem, a maior parte das contribuições geradas pela comunidade

⁵⁵ <<http://zerlot.cse.nd.edu>>.

de forma constante, um fenômeno conhecido na teoria de redes como “conexão preferencial”, seguindo um padrão no qual os “ricos” tendem a ficar ainda mais “ricos” (Barabási e Albert, 1999; Santos et al., 2013).

5. Atratividade de Artefatos Digitais

O reconhecimento do fenômeno de que poucos recebem muito em esforços colaborativos e abertos torna o entendimento da atratividade do projeto uma atividade relevante do ponto de vista acadêmico-teórico, justificando estudos dos mecanismos causais envolvidos que possam gerar assim conhecimento gerencial aplicável (Santos Jr., 2010). Esse conhecimento, apesar de ter sido e estar sendo desenvolvido no contexto dos projetos de software, tem seu arcabouço teórico considerado válido analogicamente para contextos similares, como o da produção coletiva de artigos (p.ex., Wikipedia) ou de qualquer outro artefato digital aberto (Santos et al., 2013).

Com isso em mente, podemos avançar à análise dos fatores influenciadores da atratividade de projetos de software livre. No entanto, é importante salientar que, nesta parte, não focaremos em fatores como disponibilidade financeira para campanhas publicitárias e contratação de mão de obra (i.e., patrocínio), ou constatações estatísticas como a de que navegadores de Internet são mais atrativos do que protocolos (por terem maior audiência). Apesar desses fatores afetarem a atratividade dos projetos, nosso foco é em contribuições espontâneas e, principalmente, em sua distribuição desigual entre projetos que competem diretamente por “resolverem um mesmo problema”. Dessa forma, apresentamos a seguir os principais fatores que se encaixam em nossos objetivos, ou seja, que possibilitam a diferenciação competitiva entre projetos

através de ação gerencial. Os fatores podem ser divididos em dois grupos: nos relativos ao código-fonte (Meirelles et al., 2010) e à política de propriedade intelectual, ou licença adotada (Stewart e Maruping, 2006; Santos Jr. et al., 2011).

Sobre o código-fonte, Meirelles et al. (2010) discutiram as influências que diversas métricas, como complexidade estrutural do código-fonte e quantidade de módulos, têm na atratividade do projeto para os desenvolvedores em potencial. Os resultados indicam que quando desenvolvedores voluntários estão em busca de projetos para experimentar e depois contribuir, eles analisam a estrutura do código-fonte e, se ela estiver complexa e contraintuitiva, preferem continuar suas buscas por software interessante e de fácil entendimento, levando suas contribuições para outros locais (ou mesmo iniciando um novo projeto similar).

O zelo e o monitoramento das métricas de código-fonte foram novamente enfatizados por Meirelles (2013), que inclusive desenvolveu o Mezuro⁵⁶, uma ferramenta de software livre que visa a apoiar os gestores de projetos de software livre na atividade de controle das características do código de seu projeto, especialmente em comparação com outros projetos similares. A justificativa de tal esforço está centrada no aumento da propensão de sucesso do projeto em termos de aprimoramento contínuo através de contribuições espontâneas da comunidade, o que indiretamente reduz o desperdício de esforço social por evitar a duplicidade de desenvolvimento de soluções de software com objetivos similares.

Os impactos da política de propriedade intelectual adotada também têm sido estudados no contexto da difusão e evolução dos projetos de software (Colazo e Fang, 2009; Lerner e Tirole, 2005;

56 <www.mezuro.org>.

Sen; Subramaniam e Nelson, 2008). Recentemente, por exemplo, Santos Jr. et al. (2011) focaram nos impactos da mudança da licença escolhida pelo gestor na atratividade, isto é, verificaram se um projeto mudar de licença depois de existir influencia na chance dele ser visitado, encontrar usuários, contribuidores de código-fonte e reportadores de erros. Os resultados indicam que o efeito existe, especialmente um efeito negativo consistente quando o software deixa de ter uma licença especificada.

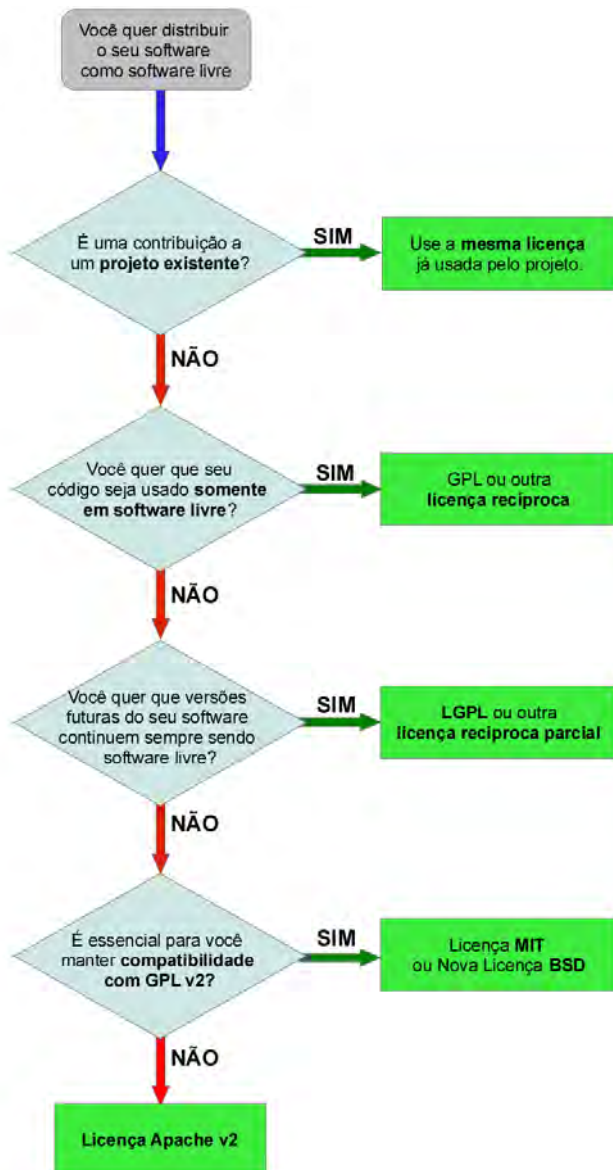
Alternativamente, de forma positiva em atratividade, o efeito da mudança de licença ocorre de forma mais saliente quando o software passa de uma licença mais restritiva (p.ex., GPL) para uma menos (p.ex., LGPL). Segundo a literatura da área (Santos Jr., 2013; Fershtman e Gandal, 2007), os efeitos da escolha da licença ocorrem devido às restrições que elas impõem nas modificações do original e nos trabalhos derivados daquele software licenciado. O entendimento dessas restrições é então crucial para os criadores de artefatos digitais públicos.

Segundo Lerner e Tirole (2005), as licenças podem ser mais bem compreendidas se agrupadas de acordo com as restrições impostas ao software que a utilizam. O nível mais alto de restrição é aquele que contém a mais popular de todas as licenças, a GPL. Esse tipo de licença obriga qualquer modificação do código original ser distribuída como software livre também. Além de restringir as adaptações e aprimoramentos do original, licenças como a GPL obrigam trabalhos derivados, ou seja, código-fonte que utilize software GPL durante a compilação, a serem licenciados como o original. Essas são as características da GPL que a tornaram conhecida como viral, contaminando qualquer aprimoramento do código original e trabalhos derivados como GPL também. Com esses dois mecanismos, os criadores da GPL buscaram garantir o crescimento do modelo livre. Essas licenças altamente restritivas são também conhecidas como de *reciprocidade total*.

No grupo de licenças menos restritivas, temos a LGPL, que relaxa a segunda restrição da GPL apresentada no parágrafo anterior. Ou seja, software LGPL não obriga trabalhos ligados a ele a serem livres também, apenas que alterações no código-fonte original, quando distribuídas pelo agora coautor, sejam disponibilizadas conjuntamente com o novo código-fonte. Essas licenças são conhecidas como de *reciprocidade parcial*. O terceiro tipo de licença relaxa todas as duas restrições apresentadas, possibilitando tanto que alterações do código-fonte original quanto trabalhos derivados não sejam disponibilizados como software livre. Essas licenças são conhecidas como *permissivas* e alguns de seus conhecidos representantes incluem a BSD, MIT e ASL (Fershtman e Gandal, 2007; Lerner e Tirole, 2005; Sabino e Kon, 2009). A facilidade de integração com software proprietário que as licenças permissivas possuem as tornaram populares entre muitas empresas privadas.

A escolha da melhor licença a ser adotada em um novo projeto de software livre não é uma tarefa difícil, mas deve ser efetuada com cuidado e, se possível, com a ajuda de um indivíduo experiente na área. O diagrama que apresentamos na Figura 3 pode servir de guia para tal escolha de forma a maximizar as possibilidades de colaborações e de sucesso do projeto.

Figura 3 - Possível algoritmo para escolha de licença para um novo projeto



6. Conclusões

A metáfora da descoberta repentina, da revelação depois de longas horas de reflexão, gerou o mito da epifania. Esse mito é reforçado pela sensação que muitas vezes experimentamos de ter entendido ou resolvido um problema científico, ou mesmo do cotidiano. Mas se tivermos a oportunidade de seguir pensando criticamente e testando nosso entendimento ou solução em diferentes contextos, a sensação de entendimento se dissipará e rapidamente será substituída por várias perguntas sem resposta imediata. Até as coisas mais simples, quanto mais pensamos sobre elas, mais complexas elas se tornam. É como se a verdade fosse passageira e socialmente (des)construída, a qualidade fosse dinâmica e o prazo de validade estivesse sempre a expirar. A impressão que dá é que nosso entendimento das coisas evolui ao longo do tempo, que nada foi concebido em sua forma “perfeita” originalmente. A compreensão “total”, entre aspas porque isso só quer dizer correta ou aceita para o entendimento da época, parece sempre ocorrer aos poucos, influenciada por muitos.

O desenvolvimento do conhecimento científico e das soluções tecnológicas deve estar pautado no pressuposto de aprimoramento contínuo da forma mais ampla e pública possível. Como disse Popper (1959), a ciência em princípio deve ser uma empreitada (“jogo”) sem fim; se um pesquisador/desenvolvedor decidir que encontrou a verdade ou a solução absoluta, ele deve se retirar do jogo. A aceitação do estado perpétuo de qualquer conhecimento ou solução como incompleta, no mínimo, ou não ótima, no máximo, quando não equivocada, reposiciona o foco da propriedade e controle dos artefatos para o seu aprimoramento e reutilização coletiva sustentável. Isso faz com que tenhamos que nos preocupar mais com a forma que disponibilizamos os artefatos e menos com as possibilidades de apropriação indevida ou não contribuição de alguns usuários do conhecimento disponibilizado (*free-riders*). Diferentes pessoas e comunidades podem sempre redefinir os

objetivos e utilidades do conhecimento, ou até mesmo como devemos entender a sua acurácia.

Ao focarmos no aprimoramento público e colaborativo dos artefatos digitais, passa a ser mais interessante atrair os usuários e colaboradores mais diversos possíveis, com todas as motivações que possam os ter atraído, complementares e conflitantes, pois é dessa configuração que surgem os paradoxos que dão vazão para a criatividade e levam à inovação (Garud et al., 2011). Obviamente, o surgimento de conflitos traz consigo efeitos negativos também, mas de forma mais proeminente para os indivíduos e menos importante para a sociedade. Por exemplo, casos de *forks* agressivos em projetos de software livre, quando conflitos de interesses impedem que uma determinada parceria seja mantida em um mesmo projeto, já foram observados. Mas, mesmo nesses casos, a sociedade em geral, no longo prazo, se beneficia se os artefatos duplicados forem mantidos e competirem para se tornar a melhor solução.

Dessa forma, buscando estimular a atração irrestrita de colaboradores e o recebimento de suas contribuições, os disponibilizadores de software, dados ou artigos científicos devem sempre buscar reconhecer e explicitar as limitações e os problemas atuais da solução apresentada. Em software, essas limitações podem aparecer em formato de *bugs* abertos, ou funcionalidades desejadas. No caso das teorias e bases de dados científicas, elas podem tomar a forma de variáveis relevantes ainda não obtidas, ou hipóteses não testadas em determinados contextos ou utilizando outras técnicas estatísticas. Em todos os casos, as limitações identificadas se parecem com uma *to-do list*, que na verdade são oportunidades de contribuição. Além do cuidado de demonstrar a incompletude, promova e conduza discussões de forma transparente, séria e informal. Comunicação efetiva é um fator-chave em projetos colaborativos, favorecendo inclusive o retorno de um contribuidor espontâneo.

Como foi dito anteriormente, essas oportunidades de contribuição devem vir junto de um “fonte” (escrito em uma linguagem

de programação, português ou matemática) bem estruturado, simples e flexível, além de possuir uma política de propriedade intelectual condizente com as expectativas das comunidades e organizações interessadas em modificar e desenvolver trabalhos derivados daquele. Mobilizações sociais em larga escala são possíveis, é só uma questão de conceber e comunicar os objetivos claramente, fornecer as ferramentas certas e os incentivos apropriados (Scekic et al., 2013).

Por fim, gostaríamos de neutralizar um pouco nossas posições ideológicas e convidar os leitores, eventuais contribuidores do conhecimento, para uma autorreflexão do que significa o conhecimento e a liberdade que deve estar associada a ele. Questione-se: Liberdade inclui a opção de complementar uma aplicação, base de dados ou artigo e não disponibilizar aquela informação/conhecimento original com os autores? E com a sociedade como um todo? É proibido proibir? É só com a resolução dessas perguntas no âmbito individual que uma escolha adequada de política de propriedade intelectual pode ser feita naquilo que você disponibilizar para a sociedade e seus diversos atores com motivações muitas vezes contraditórias (p.ex., retornos pessoais ou disponibilidade ampla irrestrita).

A sociedade já trabalha como codesenvolvedora do conhecimento, da ciência e suas ferramentas, mas ela pode fazer muito mais, aprimorando, por exemplo, a administração pública. Não só através do monitoramento das ações e dos resultados dos gestores e legisladores, mas nas escolhas dessas ações e na intervenção das causas desses resultados observados. Imaginativamente, é possível conceber vias de aprimoramento social dos desenhos processuais internos, por exemplo, ou do gerenciamento das filas dos órgãos públicos. Para isso, obviamente, existe a necessidade de institucionalizar a estrutura de absorção das contribuições sociais na administração pública, o que também precisa ser pensado por todos nós.

Referências bibliográficas

Barabási, A. L. and Albert, R. (1999). *Emergence of scaling in random networks*. Science, 286, 509-512.

Colazo, J.; Fang, Y. (2009) *Impact of license choice on Open Source Software development activity*. Journal of The American Society for Information Science and Technology, v.60, n.5, p.997-1011, jan.

Fershtman, C.; Gandal, N. (2007) *Open Source Software: Motivation and Restrictive Licensing*. International Economics and Economic Policy, v.4, n.2, p.209-225, ago.

Garud, Raghu, Joel Gehman & Arun, Kumaraswamy. (2011) *Complexity Arrangements for*

Sustained Innovation: Lessons from 3M Corporation. Organization Studies. 32: 737-767.

Lerner, J.; Tirole, J. (2005) *The Scope of Open Source Licensing*. Journal of Law, Economics and Organization, v.21, n.1, p.20-56.

Meirelles, Paulo ; Santos Jr., Carlos ; Terceiro, Antonio ; Miranda, Joao ; Chavez, Christina; Kon, F. (2010). *A Study of the Relationships between Source Code Metrics and Attractiveness in Free Software Projects*. In: Simpósio Brasileiro de Engenharia de Software (SBES).

Paulo Meirelles (2013) *Monitoramento de métricas de código-fonte em projetos de software livre*. Tese de doutoramento. Ciência da Computação. IME - Universidade de São Paulo.

Popper, K. R. (1959), *The Logic of Scientific Discovery*, Hutchinson, London.

Eric Raymond. (1997) *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. Disponível em: <<http://www.catb.org/esr/writings/homesteading>>.

Santos Jr, C. (2010) *Atratividade de projetos de software livre: importância teórica e estratégias para administração*. RAE, v. 50, pp. 424-438.

Santos Jr., Carlos ; Bonci, Marcos ; Kon, F. ; Singer, Julio ; Ritter, Victor ; Regina, Damaris ; Tsujimoto, Tamy. (2011) *Intellectual Property Policy and Attractiveness: A Longitudinal Study of Free and Open Source Software Projects*. In: ACM Computer Supported Cooperative Work (CSCW), Hangzhou, China.

Santos, C.; Kuk, G.; Kon, F.; Pearson, J. (2013) *The Attraction of Contributors in Free and Open Source Software Projects*. Journal of Strategic Information Systems (JSIS), vol. 22, issue 1.

Ognjen Scekic, Hong-Linh Truong, Schahram Dustdar. (2013). *Incentives and rewarding in social computing*. Commun. ACM 56, 6, June, 72-82.

Sen, R.; Subramaniam, C.; Nelson, M. L. (2008) *Determinants of the Choice of Open Source Software License*. Journal of Management Information Systems, v.25, n.3, p.207-239.

Stewart, K. Ammeter, A., Maruping, L. (2006) *Impact of License Choice and Organizational Sponsorship on Success in Open Source Software Development Projects*. Information Systems Research, 17 (2), 136-144.

Anthony Wasserman. (2011) *How the Internet transformed the software industry*. *Journal of Internet Services and Applications*, 2(1):11-22, Springer-Verlag.

Para saber mais:

- <<http://ccsl.ime.usp.br/textos>>
- <<http://webmink.com/essays/community-types>>

OS SERVIÇOS ELETRÔNICOS DE GOVERNO E A INTEROPERABILIDADE

- *Marcus Vinicius da Costa*⁵⁷
- *Everson Lopes de Aguiar*⁵⁸

57 SERPRO – Coordenação Estratégica de Tecnologia. Brasília – DF – Brasil, <marcus-vinicius.costa@serpro.gov.br>. Coordenador do Projeto Plataforma de Integração e Interoperabilidade para Serviços Públicos no Serpro, esteve envolvido em iniciativas de gestão e inovação estratégicas também no Ministério da Fazenda e na Companhia Nacional de Abastecimento, é pós-graduado em Análise e Projeto de Sistemas e em Organização, Métodos e Sistemas, possui MBA nas áreas de Estratégia Empresarial pela Fundação Getúlio Vargas, e de Comunicação com o Mercado pela Escola Superior de Propaganda e Marketing. Com artigos publicados a partir de sua experiência em Integração de Dados e Processos, desde 2010 é Coordenador do GT 5 – Áreas para Integração do Governo Eletrônico, da Arquitetura e-PING de Interoperabilidade do Governo Brasileiro.

58 MPOG – Secretaria de Logística e Tecnologia da Informação. Brasília – DF – Brasil, <everson.aguiar@planejamento.gov.br>. Coordenador-Geral de Integração e Interoperabilidade no Departamento de Sistemas de Informação da SLTI/MP – Graduado em Filosofia pela Universidade Católica de Brasília, Especialista em Sistemas de Informação e em Administração Pública pela Fundação Getúlio Vargas – FGV, e especialista em Governabilidade Eletrônica pela União Ibero-americana de Municípios. Realizou diversos cursos de Governo Eletrônico pela Organização dos Estados Americanos – OEA. Finalizou Mestrado em Governo Eletrônico pela Universidade Tecnológica Metropolitana do Chile. Na área de pesquisa, realiza estudos sobre mobilidade, acessibilidade e interoperabilidade em governos. Trabalha como Coordenador Geral de Integração e Interoperabilidade no Departamento de Sistemas de Informação da Secretaria de Logística e Tecnologia da Informação do Ministério do Planejamento, Orçamento e Gestão.



Resumo

Uma maior expectativa para com os resultados possíveis a partir de um governo eletrônico emerge da sociedade brasileira, com demandas crescentes por melhores serviços e informações.

No que interoperabilidade e padrões podem contribuir e qual as dimensões que devem representar foco de atenção em um momento em que a sociedade exige a eficiência e a efetividade das políticas públicas, a meritocracia e o profissionalismo?

Não dá para consertar a máquina administrativa federal de uma vez, sem correr o risco de um colapso. Nem na iniciativa privada isso é possível. No tempo que terei na Presidência vou fazer a minha parte, que é dotar o Estado de processos transparentes em que as melhores práticas sejam identificadas, premiadas e adotadas mais amplamente. Esse será o meu legado. Nosso compromisso é com a eficiência, a meritocracia e o profissionalismo.

(Entrevista da Presidenta Dilma Rousseff à revista Veja, 28/3/2012)

1. Introdução

Nos últimos anos os avanços na aplicação das tecnologias da informação e comunicação (TICs) permitiram disciplinar um uso mais intensivo de padrões, contribuindo de forma bastante

positiva para o aumento da demanda por integração pelos órgãos e entidades no âmbito da Administração Pública.

Governos e sociedade avançam na soma de esforços, fortalecendo novos canais de comunicação e métodos para participação via governo eletrônico, eliminando barreiras interpessoais e geográficas, requerendo das tecnologias novos métodos de produção e uso que aumentem a exatidão, fluidez e o intercâmbio de dados, viabilizando procedimentos tradicionais e melhorando a eficiência e a transversalidade de informações e serviços públicos.

No Estado Brasileiro são várias as iniciativas que vem ao encontro da melhoria da gestão pública, dos serviços prestados pelo governo, da transparência de informações e da prestação de contas à sociedade. Entre estas podemos destacar o Guia de Gestão de Processos de Governo, o Guia “d” Simplificação, a Infraestrutura Nacional de Dados Abertos Governamentais – INDA, o Sistema Federal de Acesso à Informação e o Cadastro Empresa Pró-Ética, suportados pelo Programa Brasileiro de Gestão Pública e Desburocratização – GesPública e pela Arquitetura de Interoperabilidade de Governo Eletrônico e-PING⁵⁹, como bons exemplos de um novo posicionamento de gestão do Estado frente a uma sociedade ainda mais consciente e exigente, que deseja acessar o governo por um único meio de acesso. Isto é, o Estado deve ser percebido como um “governo integrado e convergente”, que disponibiliza e maximiza serviços e interações de maneira transversal e que usa várias plataformas tecnológicas para atingir os diversos contextos em que as pessoas se encontram. Para Capanegra (2002) o Estado precisa ser repensado a partir de uma perspectiva mais integrada e planejada tendo em vista o cenário que combinou a era da informação e a gestão por resultados.

59 O Guia de Gestão de Processos de Governo e os Dados Abertos Governamentais são iniciativas que vêm sendo conduzidas por grupos de trabalho formados por representantes de vários órgãos e entidades do Governo e da Sociedade Civil, dentro do GT 5 “Áreas de Integração para o Governo Eletrônico”.

Este artigo apresenta a evolução dos trabalhos em curso no Governo Federal Brasileiro no sentido da interoperabilidade, a partir da Arquitetura e-PING, com foco na melhoria contínua da prestação de serviços e, em especial, nas iniciativas conduzidas no segmento Áreas de Integração para Governo Eletrônico.

Ele foi escrito com base na experiência dos autores como líder de projeto e superintendente da Unidade de Integração de Dados e Processos no Serviço Federal de Processamento de Dados – Serpro, no período compreendido entre dezembro de 2008 e abril de 2012, como tutor especialista no Curso de Especialização em Interoperabilidade e Processos Públicos Institucionais, realizado pelo Ministério do Planejamento e pela Universidade de Brasília e como Coordenador na Arquitetura de Interoperabilidade de Governo Eletrônico e-PING, no segmento Áreas de Integração para o Governo Eletrônico, cuja gestão iniciou-se em 2010 e segue até os dias de hoje; e de Everson Aguiar, como Especialista em Sistemas de Informação, em Administração Pública e em Governabilidade Eletrônica pela União Ibero-americana, mestre em Governo Eletrônico, nas diversas pesquisas e estudos sobre mobilidade, acessibilidade e interoperabilidade em governos, como Coordenador-Geral de Integração e Interoperabilidade no Departamento de Sistemas de Informação da Secretaria de Logística e Tecnologia da Informação do Ministério do Planejamento, Orçamento e Gestão e Coordenador na Arquitetura de Interoperabilidade de Governo Eletrônico e-PING .

2. A Interoperabilidade

Diversos são os conceitos para interoperabilidade. Entre eles podemos destacar aqueles que são considerados no documento de referência da Arquitetura e-PING, que disciplina os padrões para interoperabilidade no âmbito do Poder Executivo Federal no Brasil:

- “Intercâmbio coerente de informações e serviços entre sistemas. Deve possibilitar a substituição de qualquer componente ou produto usado nos pontos de interligação por outro de especificação similar, sem comprometimento das funcionalidades do sistema” (Governo do Reino Unido);
- “Habilidade de transferir e utilizar informações de maneira uniforme e eficiente entre várias organizações e sistemas de informação” (Governo da Austrália);
- “Habilidade de dois ou mais sistemas (computadores, meios de comunicação, redes, software e outros componentes de tecnologia da informação) de interagir e de intercambiar dados de acordo com um método definido, de forma a obter os resultados esperados” (ISO);
- “Interoperabilidade define se dois componentes de um sistema, desenvolvidos com ferramentas diferentes, de fornecedores diferentes, podem ou não atuar em conjunto” (Lichun Wang, Instituto Europeu de Informática – CORBA Workshops).

Complementam esta conceituação a percepção destacada no Livro Branco de Interoperabilidade de Governo Eletrônico para América Latina e Caribe, no qual interoperabilidade é a “a habilidade dos sistemas de TIC e dos processos de negócios que elas suportam de intercambiar dados e possibilitar o compartilhamento de informação e conhecimento”; e o Manual do Gestor do Guia de Interoperabilidade da e-PING que a define como “uma característica que se refere à capacidade de diversos sistemas e organizações trabalharem em conjunto (interoperar) de modo a garantir que pessoas, organizações e sistemas computacionais interajam para trocar informações de maneira eficaz e eficiente”.

Todos os conceitos explícitos convergem no sentido de direcionar a solução para um dos principais dilemas da sociedade

atual, em que convivem dezenas de sistemas de diversas áreas de negócio de entidades governamentais das três esferas de governo, que suportam grandes volumes de informações e cuja evolução não remete para uma simples reconstrução dos sistemas, visto que o tempo necessário seria maior que o razoável e a dinamicidade dos requisitos e dos gestores envolvidos não permitiriam que os resultados atendessem às necessidades de informação.

Assim, interoperabilidade pode ser entendida como a soma das integrações de sistemas e de redes, a troca de dados entre sistemas e a existência de um legado, as definições de tecnologia e as plataformas de hardware e software instalados, a diversidade de componentes e a utilização de produtos diversos de fornecedores distintos.

Porém um ponto deve ser observado – e reforçado – é que interoperabilidade não se restringe somente às dimensões tecnológica, semântica ou organizacional, mas também à dimensão pessoas. É fato que, para se conquistar a interoperabilidade, os profissionais devem estar engajados num esforço contínuo, de forma a assegurar que sistemas, processos e culturas de uma organização sejam gerenciados e direcionados para maximizar oportunidades de troca e reuso de informações.



3. A Arquitetura e-PING

A Arquitetura e-PING foi implementada em 2005 pela Portaria SLTI nº 05 e preconiza que

a existência de uma Tecnologia de Informação e Comunicação (TIC) que se preste como o alicerce para a criação dos serviços de governo eletrônico é o pré-requisito para o fornecimento de melhores serviços a sociedade, a custos mais baixos. Um governo moderno e integrado exige sistemas igualmente modernos, integrados, interoperáveis e acessíveis, trabalhando de forma íntegra, segura e coerente em todo o setor público.

As discussões da e-PING ocorreram a partir de 2003, envolvendo dirigentes e técnicos e, já naquela ocasião, segmentou-se na estrutura mantida até hoje, com um grupo de coordenação e cinco grupos de trabalho, responsáveis por estudar o uso e a evolução de padrões nos segmentos de interconexão, segurança, meios de acesso, organização de informações e áreas de integração para governo eletrônico. O primeiro grupo foi composto pela Secretaria de Logística e Tecnologia da Informação – SLTI/MP, pelo Instituto Nacional de Tecnologia da Informação – ITI/PR, e pelo Serviço Federal de Processamento de Dados – Serpro. Os demais grupos, por profissionais de vários órgãos de governo.

Além de seu objetivo principal, a melhoria constante da qualidade dos serviços prestados, a Arquitetura e-PING vai ao encontro de demandas decorrentes de diversas outras iniciativas em curso no âmbito da melhoria da prestação de serviços e do acesso à informação e à transparência.

O Programa Nacional de Gestão Pública e Desburocratização, o GesPública, implementado em 2005 pelo Decreto nº 5.378, tem como foco a melhoria da gestão e da governança das organizações públicas visando promover a gestão democrática, participativa,

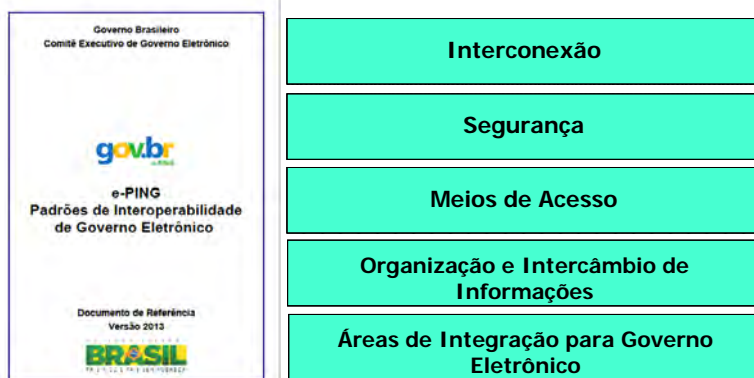
transparente e ética. O “Decreto Cidadão” nº 6.932/2009 tem por objetivo integrar, racionalizar e simplificar o atendimento público prestado ao cidadão, visando uma atuação integrada e sistêmica na expedição de atestados e certidões, com racionalização dos métodos e simplificação de processos e procedimentos para o atendimento e o compartilhamento de informações. Prevê ainda a eliminação de formalidades e exigências que possam gerar um custo econômico ou social desnecessário. Esses dois marcos legais têm, em última instância, o foco na melhoria dos processos e na capacitação dos agentes para prestação de serviços de melhor qualidade à sociedade.

A Lei nº 12.527, de 18 de novembro de 2011, dispõe sobre os procedimentos a serem observados pela União, pelos Estados, pelo Distrito Federal e pelos Municípios para

assegurar o direito de acesso à informação tendo como diretrizes: a publicidade como preceito geral e o sigilo como exceção; a divulgação de informações de interesse público, independentemente de solicitações; a utilização de tecnologia para viabilizar a comunicação; o fomento ao desenvolvimento da cultura de transparência na administração pública; e o desenvolvimento do controle social da administração pública.

A e-PING está consolidada em um documento de referência, hoje em sua 9ª versão, no qual encontram-se disciplinados os processos de governança e dos ciclos de análise de uso, avaliação e atualização de padrões, apresentando tabelas com os componentes que viabilizam as integrações. Estas informações resultam das discussões mantidas nos Grupos de Trabalho que atuam em cada um dos cinco segmentos que compõe sua estrutura, com participação de aproximadamente 400 profissionais, que representam diversas esferas, organizações e poderes.

Segmentação da e-PING



No segmento Interconexão, estão preconizadas as condições para que os órgãos de governo se interconectem e interoperem. São estudados protocolos de transferência de hipertexto, transporte de mensagem eletrônica, segurança de conteúdo de mensagem eletrônica, acesso às caixas postais, serviços de diretório, serviços de nomeação de domínio, endereços de caixa postal eletrônica, protocolo de transferência de arquivos, intercomunicação LAN/WAN, transporte, e troca de informações estruturadas em plataforma descentralizada e/ou distribuída.

No segmento Segurança, são abordados padrões para segurança de IP, segurança de correio eletrônico, criptografia, desenvolvimento de sistemas, serviços de rede, e coleta e arquivamento de evidências.

O segmento Meios de Acesso disciplina questões relativas aos padrões de dispositivos que viabilizam o acesso aos serviços de governo eletrônico, dividindo-se em quatro subgrupos.

No subgrupo Estações de Trabalho são disciplinados os padrões afetos a conjunto de caracteres e alfabetos, formato de intercâmbio de hipertexto, arquivos do tipo documento, arquivos do tipo planilha, arquivos do tipo apresentação, arquivos do tipo banco de dados para estações de trabalho, especificação de intercâmbio de informações gráficas e imagens estáticas, gráficos vetoriais, especificação de padrões de animação, arquivos do tipo áudio e do tipo vídeo, compactação de arquivos de uso geral e arquivos para georreferenciamento.

No subgrupo *Smart-cards*, *Tokens* e Cartões em geral, encontramos definição de dados, aplicações (inclusive multiaplicações), componentes elétricos, protocolos de comunicação, padrões de interface físico, segurança, infraestrutura do terminal.

No subgrupo Mobilidade são tratados protocolo de transmissão, navegador, padrão de hipertexto, programação estendida, mensageria, arquivos de vídeo e som, arquivos de imagem, arquivos de escritório e leitor PDF.

No subgrupo TV Digital são discutidas as Normas da ABNT e as especificações de padrões relacionados.

Retornando aos segmentos, em Organização de Informações são estudados os aspectos relativos ao tratamento e à transferência de informações nos serviços de Governo Eletrônico, incluindo padrão de estrutura de assuntos de governo e de metadados, compreendendo linguagem para intercâmbio de dados, linguagem para transformação de dados, definição dos dados para intercâmbio, o Vocabulário Controlado de Governo Eletrônico, e Padrão de Metadados do Governo (e-PMG).

O quinto e último segmento, Áreas de Integração para Governo Eletrônico, estabelece a utilização ou a construção de especificações técnicas para sustentar o intercâmbio de informações em áreas transversais da atuação governamental, cuja padronização seja relevante para a interoperabilidade de serviços de Governo

Eletrônico, tais como Dados e Processos, Informações Contábeis e Informações Geográficas, entre outras.

4. Áreas de Integração para Governo Eletrônico

Atualmente o segmento é responsável pelos estudos dos padrões para intercâmbio de informações financeiras (XBRL – *eXtensible Business Reporting Language*), intercâmbio de informações estatísticas (SDMX – *Statistical Data and Metadata eXchange*), intercâmbio de informações jurídicas e legislativas (LexML – *Lex Markup Language*), intercâmbio de informações de planejamento e desempenho (StratML – *Strategy Markup Language*), além dos padrões para interação de dados e processos (MGD – Modelo Global de Dados) e integração e gestão de processos (BPM – Business Process Management).

O segmento viabiliza também as Infraestruturas Nacionais de Dados Espaciais (INDE) e de Dados Abertos (INDA), os serviços web – *World Wide Web* e promove a integração entre os Governos Federal, Estadual e Municipal por meio do uso dos padrões, com a participação da Associação Brasileira de Entidades Estaduais de Tecnologia da Informação e Comunicação (ABEP).

Como padrões específicos, o XBRL, SDMX, LexML e StratML têm em comum a necessidade de uma taxonomia que dê suporte a sua aplicação e é este o ponto focal dos trabalhos desenvolvidos pelo SubGrupos de Trabalho (SGTs), em que estão os especialistas e representantes do governo e da sociedade interessados em promover e expandir seu uso. Nestes SGTs o que se busca atualmente é trazer os envolvidos – órgãos de governo, academia e sociedade – para uma discussão conjunta, para que se compartilhe melhores práticas e se gere sinergia na solução de problemas e implementação dos padrões.

O uso destes padrões está diretamente relacionado a uma melhoria da qualidade e da garantia da confiabilidade no intercâmbio,

interligação de bancos de dados distintos e concentração na estrutura da informação e não em sua aparência.

Buscando uma visão mais ampla e estratificada (alto nível) dos dados existentes e necessários para a realização das diversas operações nas quais há envolvimento do governo, foram criados dois SubGrupos de Trabalho. O primeiro tem seus trabalhos focados na evolução do padrão MGD, fazendo revisões constantes de metodologia e notação, para que atendam as necessidades de seus usuários (analistas, gestores, desenvolvedores etc); aprimoramento dos cursos de capacitação, atualmente na modalidade de ensino a distância; manutenção de seu modelo de maturidade e de seu modelo de governança e gestão. O segundo inicia trabalhos semelhantes em relação aos padrões BPM, observando seu alinhamento e sua integração ao padrão MGD.

O uso destes padrões está diretamente relacionado à necessidade de cada parte envolvida ter visibilidade quanto ao processo do outro, promovendo sua integração, evitando o retrabalho e gerando sinergia para prestação de serviços de melhor qualidade ao cidadão; ou ainda sobre como poderão ser reconhecidos os *donos de processos*, responsáveis por promover tal visibilidade e integração também no que se relaciona a soluções de TI, que dão o suporte necessário às operações.

Aspectos como infraestrutura e serviços, situação atual, cenários, maximização e expansão do uso são discutidos pelos SGTs INDE, INDA e WEB. A INDE envolve representantes de diversos órgãos de governo cujas informações geoespaciais são condição fundamental para o desenvolvimento de suas atribuições regimentais, como por exemplo, o Ministério do Planejamento, Orçamento e Gestão, a Casa Civil da Presidência da República e as Forças Armadas. Já a INDA se caracteriza pela forte presença de representantes da sociedade, que demanda uma abertura cada vez maior de dados e informações para participação dos cidadãos

e desenvolvimento do mercado privado. O SGT WEB encontra-se envolvido com o uso e avaliação de padrões que garantam a qualidade das informações nos serviços disponibilizados por meio de *web services*, no sentido de uma arquitetura orientada a serviços.

Não há como se pensar em implementar e monitorar programas sociais e de desenvolvimento ou promover a transparência sem o suporte destas infraestruturas e padrões.

A transversalidade no caso do segmento Áreas de Integração para Governo Eletrônico pode ser entendida como horizontal – integrando ministérios, agências, empresas, etc. – e vertical – integrando o nível federal, estadual e municipal, governo e sociedade.

Hoje ainda é fato que as diversas informações disponibilizadas pelos vários sistemas não são de fácil integração. Tal premissa não envolve somente os sistemas de TICs que apoiam os processos ou macroprocessos de Governo, mas também a forma de trabalho de centenas de pessoas – gestores governamentais, analistas de informações estratégicas, analistas de negócios, gestores de informações, gestores de processos, administradores de dados, desenvolvedores de soluções, administradores de bancos de dados, etc. – localizadas em dezenas de entidades do Governo, pessoas que, em seu cotidiano, atuam em diversos Domínios como Saúde, Educação, Planejamento e Arrecadação, entre outros.

Promovendo a aproximação das iniciativas conduzidas em nível municipal e estadual àquelas conduzidas em nível de governo federal, soma-se o SGT ABEP que, por conta de sua capilaridade, estreita relação com as Casas Civas e Empresas de Tecnologia de Informação municipais e estaduais, torna-se o ambiente propício para utilização de padrões e desenvolvimento de iniciativas conjuntas que promovam a interoperabilidade.

Uma das principais e mais reconhecidas evidências de que o uso dos padrões tende a solucionar mais que uma demanda

pontual pode ser observada a partir do Modelo Global de Dados – MGD, aplicado no Macroprocesso de Planejamento, Orçamento e Finanças.

Aplicado com objetivo de identificar as entidades necessárias à realização do negócio e, conseqüentemente, ao desenvolvimento de soluções que viessem a suportar esses negócios, a metodologia e a notação permitiram trazer ao conhecimento comum as integrações atuais, as integrações necessárias, as redundâncias e as diversas formas de interações existentes e necessárias. A “fotografia” obtida pelo modelo – que, por ser de alto nível, sofre menos alterações – foi disponibilizada para o Ministério do Planejamento, Orçamento e Gestão, que implementa uma área de gestão de dados voltada à evolução do modelo, e a aplicação para o desenvolvimento de soluções afetas aquele ministério, e para o Ministério da Fazenda, que teve recentemente aprovado um projeto de *web service* para provimento de serviços a partir que consideram aquela visão.

É visível que os resultados obtidos não se limitaram ao escopo do macroprocesso em estudo, extrapolando a outros, entre eles o macroprocesso de Gestão Organizacional, tendo sido utilizadas as informações geradas pelo MGD para o desenvolvimento do “Novo SIORG”, fonte oficial de informações sobre estrutura organizacional dos órgãos do Poder Executivo – Administração Direta, Autarquias e Fundações⁶⁰.

Portanto, somente neste exemplo, já se consegue perceber que a modelagem de processos e a abertura de dados de governo tendem a gerar maior qualidade e economicidade nas etapas de geração e disponibilidade de serviços públicos.

60 Do sítio <<http://www.siorg.redegoverno.gov.br/>>.



5. O Governo Eletrônico

Na medida em que, em seu nível mais alto, os dirigentes perceberam o potencial das tecnologias para a melhoria da gestão, estimularam a estruturação de políticas de Estado a partir delas. Segundo Almeida (2002), foi na década de 2000 que se deu a percepção inicial de que as TICs são instrumento de apoio ao processo de Reforma do Estado. Assim, no primeiro ano daquela década o governo federal lançou a “Proposta de Política de Governo Eletrônico para o Poder Executivo”. Esse documento sistematizou e estruturou as várias ações de governo eletrônico entre os órgãos da administração direta, demonstrou a importância das TICs como suporte ao processo de modernização, a melhoria da prestação de serviços aos cidadãos, o acesso a informações, a redução de custos e o controle social sobre as ações do governo. Para Ferrer (2007) o

governo eletrônico é um meio de modernização da gestão pública, mas depende fundamentalmente da mudança de gestão.

Em princípio se identificou uma proliferação de iniciativas isoladas, sem mecanismos de coordenação e integração de diferentes soluções, principalmente nas áreas de arrecadação e tributação especificamente para automação de serviços e investimentos em aquisições de parques computacionais. Em um segundo estágio, o direcionamento de recursos foi para a prestação de serviços que gerassem benefícios à sociedade. Nessa fase o Estado se deu conta da necessidade de aproveitamento e confluência de recursos públicos e que esse processo deveria ter uma estratégia e objetivos de desenvolvimento da democracia, de um bom governo e de direcionamento ao cidadão.

Foi criado, em 2000, o Comitê Executivo de Governo Eletrônico (CEGE) que, vinculado à Presidência da República, pode ser considerado um dos marcos do Conselho de Governo em prol da melhoria da prestação de serviços e informações à sociedade.

Em 2003 as lideranças ampliaram os comitês técnicos do CEGE de quatro para oito, por intermédio do Decreto não numerado de 28 de outubro de 2003, com a finalidade de coordenar e articular o planejamento e a implementação de projetos e ações em suas respectivas áreas de competência. Assim, foram criados os comitês de: Implementação do Software Livre; Inclusão Digital; Integração de Sistemas; Sistemas Legados e Licenças de Software; Gestão de Sítios e Serviços On-line; Infraestrutura de Rede; Governo para Governo e Gestão do Conhecimento e Informação Estratégica.

Neste sentido, o Comitê Executivo destacado apontou um novo norte para o programa governo eletrônico a partir de 2004, fortalecendo algumas temáticas. As atuais diretrizes estão explicitadas no documento “Oficinas de Planejamento Estratégico – relatório consolidado – Comitês Técnicos”, e são:

- Promoção da cidadania como prioridade;
- Indissociabilidade entre inclusão digital e o governo eletrônico;
- Utilização do software livre como recurso estratégico;
- Gestão do Conhecimento como instrumento estratégico de articulação e gestão das políticas públicas;
- Racionalização dos recursos;
- Adoção de políticas, normas e padrões comuns; e,
- Integração com outros níveis de governo e com os demais poderes.

Entre as diretrizes citadas acima se evidencia a preocupação com a otimização no uso das TICs – Tecnologia da Informação e Comunicações, seja por conta da variável economicidade, possível a partir do alto grau de reuso e conseqüente redução dos custos unitários, seja por conta da variável racionalidade, alcançada a partir do compartilhamento de recursos entre órgãos públicos, que poderá se dar tanto em tempo de desenvolvimento quanto de operação de soluções, inclusive por meio do compartilhamento de equipamentos e recursos humanos e do desenvolvimento compartilhado em ambiente colaborativo, envolvendo múltiplas organizações.

Não obstante, o governo eletrônico, em sentido amplo, impõe uma nova visão para processos, pessoas e tecnologia como um todo, inclusive alterando leis, modificando estruturas institucionais, revendo e aprimorando serviços, promovendo maior transparência e desenvolvendo novas capacidades e conhecimentos.

No país, o Programa de Governo Eletrônico Brasileiro visa transformar as relações do governo com os cidadãos, empresas e também entre os órgãos do próprio governo, de forma a aprimorar a qualidade dos serviços prestados, promover a interação com empresas e indústrias e fortalecer a participação cidadã por meio do acesso à informação e a uma administração mais eficiente.



Em face disso, o Governo Eletrônico é hoje um ponto de encontro político em relação a sua importância, visto os impactos das informações disseminadas à sociedade, seja pela mídia ou pelos próprios sistemas e soluções do governo e das entidades voltadas à transparência, que refletem os esforços investidos para uma melhor gestão da coisa pública.

Dentre as capacidades intrínsecas que fortalecem o Governo Eletrônico, podemos destacar:

- a) viabilizar uma troca mais rápida de informações intra e intergovernos;
- b) propiciar a alfabetização e a inclusão digital, bem como uma maior difusão da cultura;
- c) facilitar o relacionamento entre o fisco e contribuintes e governo e fornecedores;

- d) melhorar a qualidade dos serviços prestados aos cidadãos por meio do atendimento de demandas específicas;
- e) fortalecer o processo democrático a partir de uma participação popular mais efetiva na administração pública, gerando o que denomina de e-democracia;
- f) propiciar maior transparência da gestão pública e incentivar a prestação de contas; facilitando o exercício de *accountability* e o acesso ao mercado internacional de crédito;
- g) promover um maior estímulo aos negócios;
- h) contribuir para o fortalecimento da competitividade sistêmica da economia.

6. Considerações Finais

O contexto do governo eletrônico brasileiro nos últimos anos demonstra o quanto as tecnologias vêm melhorando a governabilidade dos processos internos e o relacionamento do Estado com os diversos entes sociais. Todavia, fazem-se necessárias políticas de governo eletrônico mais integradas, sinérgicas e articuladas que compreendam a incorporação das TICs e plasmem mudanças culturais consideradas num contexto de Reforma do Estado. Vislumbra-se que essas políticas possam emergir com uma Agenda de Serviços de Governo Eletrônico e de Modernização ora em desenvolvimento no Ministério do Planejamento, embora se tenha consciência de que ainda são muitos os desafios e barreiras que devem ser transpostas para se implantar o Estado Virtual, como: a universalização do acesso a serviços; a diversidade de produtos e tecnologias ofertados pelo mercado para integração e interoperação de serviços e sistemas; questões de identificação e autenticação eletrônica; o desalinhamento entre a oferta de

serviços públicos eletrônicos; a própria falta de confiança ou de privacidade da sociedade nos serviços digitais de governo; as falhas de comunicação e de divulgação das iniciativas, bem como questões orçamentárias, entre outras que necessitam de atenção e se apresentam como obstáculo para um governo mais matricial e integrado.

Com o amadurecimento das tecnologias e o uso intensivo de padrões, do crescimento das demandas por integração de órgãos e entidades no âmbito da Administração Pública, em que Governo e Sociedade se valem de novos canais de comunicação e métodos para participação, é natural uma maior expectativa para com os resultados possíveis a partir de um governo eletrônico, principalmente no que se caracteriza como melhoria da gestão pública, dos serviços prestados pelo governo, da transparência de informações e do consequente desenvolvimento da sociedade brasileira, no sentido da economia, da igualdade e do desenvolvimento sustentável.

Conclui-se que mesmo com a maturidade no uso de padrões tecnológicos tanto nos governos quanto nas sociedades, ainda há de se empreender esforços na disciplina e no uso desses padrões para que se promovam a interoperabilidade semântica e organizacional, contextos em que o envolvimento das pessoas e a absorção de experiências de sucesso são fatores chaves de sucesso.

Referência Bibliográfica

ALMEIDA, Marcos Osório. *Governo eletrônico no Brasil*. En: VII Congreso Internacional del CLAD sobre la Reforma del Estado y de la Administración Pública. Portugal, Oct. 2002.

BRASIL. MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO. *Arquitetura de Interoperabilidade de Governo Eletrônico e-PING*. Brasília, DF. 2012. 62p. Disponível em: <<http://www.governoeletronico.gov.br/acoes-e-projetos/e-ping-padrees-de-interoperabilidade>>. Acesso: 10/5/2013.

BRASIL. MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO. Decreto nº 5.378 – de 23 de fevereiro de 2005 – Alterado pelo Decreto nº 6.944, de 21 de agosto de 2009, institui o Programa Nacional de Gestão Pública e Desburocratização – GESPÚBLICA e o Comitê Gestor do Programa Nacional de Gestão Pública e Desburocratização, e dá outras providências.

BRASIL. MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO. *Decreto nº 6.932, de 11 de agosto de 2009*, dispõe sobre a simplificação do atendimento público prestado ao cidadão, ratifica a dispensa do reconhecimento de firma em documentos produzidos no Brasil, institui a “Carta de Serviços ao Cidadão” e dá outras providências.

BRASIL. MINISTÉRIO DO PLANEJAMENTO, ORÇAMENTO E GESTÃO. *Instrução Normativa nº 04*. Brasília, DF. 2008. Disponível em: <<http://www.governoeletronico.gov.br/>>. Acesso: 10/5/2013.

CAPANEGRA, Horacio. *El Gobierno electrónico: hacia una verdadera reforma del estado*. En: XVI Concurso de Ensayos y Monografías

del CLAD sobre Reforma del Estado y Modernización de la Administración Pública “Gobierno Electrónico”. CARACAS, 2002. Disponible en: <http://portal.cnti.ve/cnti_docmgr/sharedfiles/gobiernoelectronico2.pdf>. Acesso: 25/6/2013.

COMISIÓN ECONÓMICA PARA AMÉRICA LATINA (CEPAL). *Libro blanco de interoperabilidad de gobierno electrónico para América Latina y el Caribe*. Versión 3.0, Septiembre de 2007.

CONGRESSO INTERNACIONAL DE SOFTWARE LIVRE E GOVERNO ELETRÔNICO, *Amãpytuna: computação em nuvem: serviços livres para a sociedade do conhecimento*, Brasília, FUNAG, 2011.

COSTA, M. V.; et al. *A Integração de Dados no âmbito do Macroprocesso de Planejamento, Orçamento e Finanças*. In: *Panorama da Interoperabilidade no Brasil*, 117-136, MPOG, Brasília, 2010.

DADOS ABERTOS PARA A DEMOCRACIA NA ERA DIGITAL – *IV Congresso Internacional de Software Livre e Governo Eletrônico*, Brasília, Fundação Alexandre de Gusmão, 2011.

FERRER, Florência et al. *Gestão Pública eficiente: impactos econômicos de governo inovadores*. Rio de Janeiro: Elsevier, 2007. pp.26-27.

GOVERNO BRASILEIRO – COMITÊ EXECUTIVO DE GOVERNO ELETRÔNICO. *Documento de Referência e-PING – Padrões de Interoperabilidade de Governo Eletrônico*. Brasília, 2005.

HOESCHL, Hugo César. *Introdução ao Governo Eletrônico*. Instituto i3G. Disponível em: <<http://www.i3g.org.br/editora/livros/introducaogovernoeletronico.pdf>>.

LANDERSBERGEN JE, D; WOLKEN JR, G. *Realizing tre promise: government information systems anda de fourth generation of information technology*. Public Administration Review, march/april, 2001.

POGGI, Eduardo (2008): “*Modelo de Madurez de la Interoperabilidad*”. Monografía presentada em el II Simposio de Informática Estatal, 37 AIO. Rosario, Argentina, Septiembre de 2008.

VERNADAT, F. B. *Enterprise modeling and integration: principles and applications*. London: Chapman & Hall, 1996.



Formato	15,5 x 22,5 cm
Mancha gráfica	10,9 x 17cm
Papel	pólen soft 80g (miolo), cartão supremo 250g (capa)
Fontes	Gentium Book Basic 14/15 (títulos), Chaparral Pro 11,5/15 (textos)



Desde 2008, o Consemi é um evento estratégico no estímulo ao debate de temas da política e gestão de tecnologias em Software Livre, permeado pelos conceitos de promoção da cidadania, interoperabilidade dos sistemas de governo, compartilhamento do conhecimento e adoção de normas e padrões.

Em 2009, expôs o imenso acervo de soluções e de códigos desenvolvidos e compartilhados pela Esfera Pública. O evento mostrou que, com a lógica do Software Livre, os recursos podem ser mais bem aproveitados.

Além disso, os conceitos de liberdade e direito de acesso ao conhecimento foram discutidos por mais de 4 mil participantes. Representantes de 17 países (África do Sul, Argentina, Chile, Coreia do Sul, Cuba, Egito, Equador, França, Índia, Indonésia, Malásia, Paraguai, São Tomé e Príncipe, Uruguai, Venezuela e Zimbábue) participaram dessa edição do evento.

Em 2010, o Consemi inovou ao trazer para dentro do governo a discussão sobre Computação em Nuvem e a evolução dos serviços de governo eletrônico a partir dessa nova proposta de infraestrutura tecnológica. A cooperação internacional e a defesa do software livre como melhor alternativa técnica para a Administração Pública continuaram presentes no evento.

Em 2011, o tema Dados Abertos trouxe o debate da liberação de informações públicas para o âmago do governo. Instituições públicas e grupos interessados dedicaram-se durante três dias ao compartilhamento de conhecimento em torno da questão, que ainda era embrionária no país.

A edição 2012, realizada pela primeira vez fora de Brasília, aconteceu em Belém e abordou a “Mobilidade Digital”, um advento que vem alterando a forma como os indivíduos usam e se apropriam das tecnologias. O governo, de maneira a inovar nos serviços públicos aos cidadãos, já começou a pensar e disponibilizar aplicativos voltados para os dispositivos móveis. Como ampliar a atuação na área de governo eletrônico, tendo a portabilidade como elemento essencial para melhor atender às pessoas, foi o grande foco de debates da quinta edição.

