

## HTML5 E O FUTURO DAS FERRAMENTAS PARA INTERNET RICA

Alexandre VARASCHIN  
Faculdade de Tecnologia de Jahu  
alexvars@hotmail.com

Fabricio Kazumi SATO  
Faculdade de Tecnologia de Jahu  
fabricioksato@hotmail.com

Aparecida ZEM-LOPES  
Faculdade de Tecnologia de Jahu  
cida.zem@gmail.com

### RESUMO

Quando a Internet foi concebida contava apenas com páginas web simples e estáticas, mas após meados dos anos 90 a Web se popularizou assim como a demanda por inovações. Na época os navegadores não puderam acompanhar essa evolução, surgindo assim a ideia do uso de *plug-ins*. A finalidade dos *plug-ins* é trabalhar em conjunto com o navegador, executando tarefas específicas, possibilitando assim métodos de interação, animação e programação para suprir suas limitações e as do HTML4. Tais ferramentas agregam mais funcionalidades às aplicações Web e passaram a ser conhecidas como *Rich Internet Application* - Aplicação de Internet Rica (RIA), tendo o *Adobe Flash* e o *Microsoft Silverlight* como principais exemplos. O HTML5 surge com a proposta de suprir uma das grandes deficiências da Web atual, a dependência de *plug-ins* para reprodução de conteúdos multimídias, com o foco na interoperabilidade entre as diversas

plataformas e dispositivos, e utilizando tecnologias abertas e padronizadas. Por meio de pesquisas em diferentes fontes de literatura e sites técnicos, este trabalho foi criado com o objetivo de discutir o futuro das ferramentas de desenvolvimento de aplicações para Internet. O cenário mais provável é que as ferramentas atuais continuarão coexistindo com as novas, pois possuem recursos ainda não contemplados pela nova tecnologia, além do vasto conteúdo já criado e em funcionamento. Contudo, espera-se que o HTML5 evolua muito e conquiste cada vez mais espaço em aplicações web nos próximos anos.

**Palavras-chave:** HTML5, Adobe Flash, Internet Rica, Web.

### ABSTRACT

When the internet was conceived just had simple and static web pages. However, in the middle of the 90's, web became popular as well as the demand for innovations. At

that time navigators could not follow that evolution, thus the idea of using plug-ins emerged. The purpose of the plug-ins is to work alongside the navigator, executing specific tasks and though, making possible the interaction methods, animation and programming to supply their limitations as well as the HTML4 ones. Such tools aggregate more functionalities to the Web applications and they are known then as Rich Internet Application (RIA), and the *Adobe Flash* and the *Microsoft Silverlight* are their main examples. HTML5 appears with the proposal of supplying one of the greatest lacks of the current Web, the plug-ins dependence for reproduction of multimedia contents, with the focus in the interoperability between the several platforms and devices and using open and standardized technologies. Through researches in different literature sources and technical sites, this paper aims to discuss the future of the development tools of applications for Internet. The most probable scenery is that the current tools will continue coexisting with the new ones, because they possess resources which are not contemplated by new technologies, besides the vast content in operation that has been created. However, it is expected that HTML5 enhances a lot and accomplishes more and more space in web applications in the coming years.

**Keywords:** HTML5; Adobe Flash; Rich Internet; Web.

## INTRODUÇÃO

Quando foi concebida, a Internet contava com um pequeno número de *sites* desenvolvidos principalmente para troca de informações entre pesquisadores. As páginas eram simples e estáticas, contendo basicamente textos e poucas imagens. Havia diversas limitações na linguagem HTML, nos navegadores e na reduzida banda de transmissão. Mas conforme a Web foi se popularizando começou a aumentar o interesse em se criar páginas mais elaboradas e interativas, aumentando assim a necessidade por novas tecnologias

(HARRIS, 2013; LALLI, BUENO e ZACHARIAS, 2008).

Em meados dos anos 1990, multiplicava-se o número de sites na Internet, assim como a quantidade de desenvolvedores produzindo páginas, o que levou à necessidade de inovações e melhores recursos tecnológicos. Os navegadores da época não puderam acompanhar essa evolução, e ao invés de se modificarem para que pudessem exibir conteúdos diferentes, surgiu a ideia do uso de *plug-ins*. Os *plug-ins* são programas que trabalham em conjunto com o navegador, executando tarefas específicas, as quais o navegador, sozinho, não conseguiria. Normalmente são criados para trabalhar com um tipo de arquivo próprio e são fáceis de serem baixados e instalados (HARRIS, 2013).

Por meio desse tipo de tecnologia, as aplicações Web deixaram de ser limitadas pela capacidade do navegador, principalmente na interação do usuário com a página, que era muito reduzida apenas com o uso de controles HTML. Portanto, poderia ser implementado no lado cliente tudo que a plataforma oferecia por meio de *plug-ins*: mais métodos de interação, animações e programação. Além disso, com o processamento local, reduziu-se a quantidade de chamadas ao servidor e a necessidade de recarregamento da página (WENZ, 2008).

A aplicação que rodava no navegador agregava mais funcionalidades tornando-a mais rica. O termo Aplicação de Internet Rica (*Rich Internet Application* ou RIA) foi difundido por Allaire (2002) em documento que enfatizava os benefícios da nova versão do *plug-in* Flash Player. Passando a definir assim as aplicações Web, quando essas aparentam e se comportam mais como aplicações *desktop* do que páginas de Internet (VOSSEN e HAGEMANN, 2007).

A última versão do HTML4 foi implementada há mais de uma década e não possui especificação sobre recursos de mídia rica, animação e interatividade. O HTML5 traz a evolução para atender aos requisitos

que os usuários esperam de uma navegação moderna, incluindo a habilidade de reproduzir áudio e vídeo, exibir gráficos e controles ricos, com métodos de interação mais avançados. Ao contrário do HTML4, essa nova versão engloba várias tecnologias interconectadas, para prover mais riqueza e capacidade aos navegadores sem a necessidade do uso de *plug-ins* (LIKNESS, 2012).

Este trabalho foi desenvolvido por meio de pesquisas em diversas fontes de literatura, entre elas: livros técnicos, artigos e sites, além da análise de seus autores sobre o cenário atual e futuro no desenvolvimento Web.

Na revisão bibliográfica, procurou-se descrever as ferramentas atuais e novas, estudando suas características e seu uso no desenvolvimento de aplicações ricas para internet. Os resultados foram obtidos com base nesse estudo, somado à análise dos autores referenciados.

Mesmo o HTML5 não sendo ainda um padrão definitivo, a chegada dessa nova tecnologia não decreta o fim imediato das outras, pois segundo Vossen e Hagemann

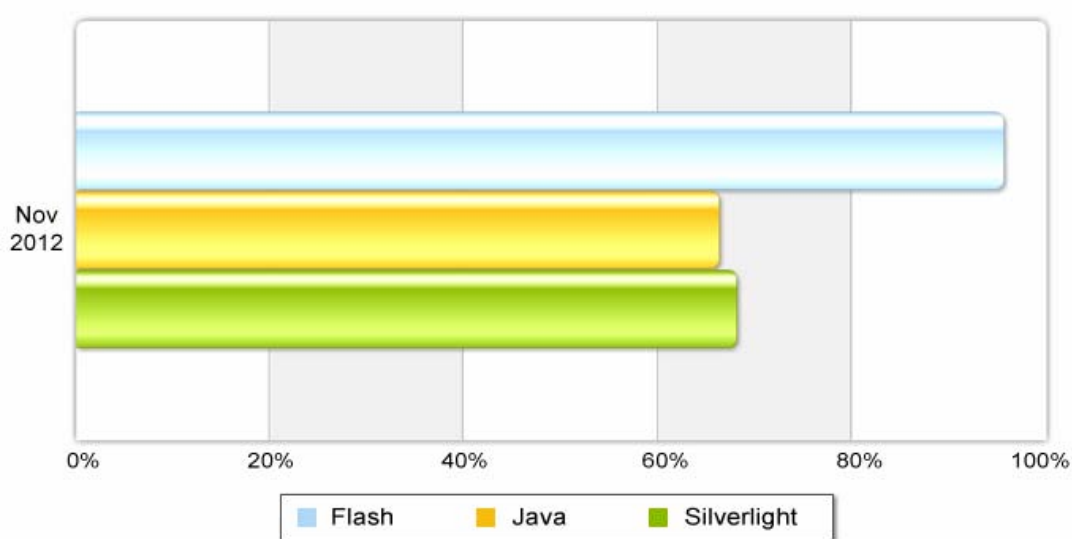
(2007, p. 137, tradução nossa) “[...] o desenvolvimento da Internet é evolucionário, não revolucionário”.

O objetivo deste artigo é mostrar as principais ferramentas para Aplicações de Internet Rica (RIA) utilizadas no presente, apresentar as novas tecnologias do HTML5 e discutir a relação que elas podem ter no futuro. Seu conteúdo está distribuído da seguinte maneira: são apresentadas as principais ferramentas utilizadas atualmente; em seguida é abordado o HTML5 e suas tecnologias; são mostradas algumas ferramentas novas para desenvolvimento Web; é discutido o futuro das tecnologias tradicionais e novas; e após esses conteúdos e discussões, são feitas as considerações finais desse trabalho.

## FERRAMENTAS ATUAIS

A maior parte das ferramentas de desenvolvimento para aplicações Web ricas faz o uso de *plug-ins*, rodando em paralelo aos navegadores para suprir suas limitações e as do HTML4. A Figura 1 mostra a porcentagem de navegadores que possuem *plug-ins* desse tipo instalados, fornecendo suporte às plataformas.

**Figura 1** – Abrangência das plataformas RIA nos navegadores em uso



Fonte: Adaptado de STAT OWL (2012)

## Flash

O Adobe Flash se consolidou no mercado há um bom tempo e é uma das ferramentas mais utilizadas para desenvolver aplicações Web, devido aos seus ricos recursos de desenho, multimídia e programação (MCLEAN, 2011).

Anteriormente chamado de *Future Splash Animator*, a ferramenta começou como um simples pacote de desenho, que depois passou a reproduzir animações em páginas Web. Logo, a Macromedia comprou a ideia da ferramenta, e no final de 1996 foi lançada a primeira versão do Flash. Nos anos seguintes, a ferramenta progrediu além da animação e incorporou a habilidade de programação. O Flash deu então início à definição de aplicação de internet rica (RIA), pois fornecia riqueza na aplicação Web para o cliente com gráficos, sons e interatividade, mais a capacidade de interação assíncrona com o servidor. No ano de 2005, a Macromedia foi comprada e incorporada a Adobe Systems, dando continuidade ao desenvolvimento da ferramenta (LEGGETT, DE BOER, JANOUSEK, 2007).

Apesar de suportar diversos formatos de imagens, o Flash trabalha basicamente com desenho em gráfico vetorial, cujas imagens são geradas a partir de algoritmos que descrevem os aspectos geométricos do gráfico. O principal benefício de gráficos desse tipo é que eles não perdem qualidade quando são manipulados ou redimensionados (MCLEAN, 2011).

A linguagem de programação ActionScript é um dos elementos principais do Flash, responsável por proporcionar dinamicidade e interatividade nas aplicações, e que evoluiu em conjunto com a ferramenta. É uma linguagem orientada a objetos muito similar ao Java e C#, trabalhando com herança de classes, interfaces, exceções e nível de acesso a métodos e membros (LEGGETT, DE BOER e JANOUSEK, 2007).

O Flash possui mais de 15 anos de mercado, com um histórico de rápida inovação no desenvolvimento de aplicações Web, bem à frente do que tradicionalmente o HTML4 era capaz. Mas mesmo sendo possível criar animações e interatividade básicas com o HTML5, o Flash se apresenta, além disso, como uma ferramenta poderosa para criação de jogos de alta qualidade, soluções avançadas de vídeo e outros usos específicos (LABRECQUE, 2012).

O desempenho de uma aplicação do lado cliente depende do poder de processamento que seu equipamento dispõe. Normalmente aplicações usuais são executadas pelo Flash Player de forma satisfatória, mas para criação de jogos ou aplicativos móveis pode ser necessário o uso técnicas de desenvolvimento que minimizam o uso de recursos, otimizando assim a performance da ferramenta. (GLADSTIEN, 2012)

## Silverlight

O Microsoft Silverlight, anteriormente conhecido como *Windows Presentation Foundation / Everywhere* (WPF/E), é um software criado pela Microsoft para o desenvolvimento e execução de aplicações ricas, com recursos e propostas similares ao Adobe Flash, sendo uma tecnologia independente de browser. Seu ambiente de execução está disponível por meio de um *plug-in* para navegadores Web que executam tanto para o sistema operacional Windows quanto para o Mac OS X (LIKNESS, 2012).

A ferramenta utiliza um formato declarativo de conteúdo baseado em XAML (*eXtensible Application Markup Language*), modelo de programação baseado em JavaScript, com suporte a vários tipos de arquivos multimídias (ANDERSON, 2012).

O Silverlight faz parte de um conjunto de ferramentas para desenvolvimento chamado Visual Studio, capaz de construir aplicações Web ASP.NET, serviços Web XML, aplicações desktop e aplicativos para dispositivos móveis. Um dos poderes do Silverlight é também a sua capacidade de

trabalhar com vídeo, sendo desenvolvido para gerar novas soluções de transmissão de vídeo streaming de alta qualidade (SPAKI et al., 2008).

Existem várias semelhanças entre o Silverlight e o Flash, já que ambos rodam no navegador por meio de *plug-ins*, fornecendo uma plataforma de aplicações do lado cliente. Há algumas características diferentes entre eles, como a arquitetura e ambiente de desenvolvimento, mas numa visão geral um é o maior concorrente do outro (WENZ, 2008; ANDERSON, 2012).

## Java

A plataforma Java, introduzida pela Sun Microsystems, foi projetada para permitir o desenvolvimento de aplicações portáteis de alto desempenho para a mais ampla variedade possível de arquiteturas de computação. As aplicações web do Java, as chamadas *applets*, apresentam independência de plataforma e podem ser executadas em qualquer navegador da Internet. As *applets* são escritas na forma de programação orientada a objetos (POO) e voltadas para o ambiente de rede, podendo ser facilmente incluídas em um documento HTML, em conjunto com outros elementos multimídia como figuras, vídeos e sons. (GONÇALVES e CANESIN, 2002; JAVA, 2013).

O JavaFX é outra ferramenta baseada em Java lançado no final de 2008, possui princípios de portabilidade e interoperabilidade entre sistemas operacionais que se comportam de forma consistente, permitindo que desenvolvedores de aplicações criem e implantem, facilmente, aplicações ricas para Internet (LOPES e TAVARES, 2010). A ferramenta permite que os desenvolvedores usem qualquer biblioteca Java em suas aplicações. Dessa forma, os desenvolvedores podem expandir seus recursos em Java e aproveitar a tecnologia de apresentação que o JavaFX fornece para criar aplicações ricas para web (JAVA, 2013).

## HTML5, CSS E JAVASCRIPT

O HTML5 começou a ser desenvolvido em 2006 pelo *World Wide Consortium* (W3C) em conjunto com a *Web Hypertext Application Technology Working Group* (WHAT-WG). Uma das grandes deficiências da Web atual é a dependência de *plug-ins* para a reprodução de conteúdos multimídias, disponibilizados por empresas proprietárias (SEBBEN e GUEDES, 2011). Um dos principais objetivos do HTML5 é facilitar, para o desenvolvedor, a manipulação das características dos objetos de maneira que seja transparente para o usuário final (FERREIRA e EIS, 2013).

Pelo fato do HTML ser apenas uma linguagem de marcação, o navegador deve interpretá-lo juntamente com o CSS e o JavaScript para produzir páginas mais aprimoradas. O HTML mantém toda estrutura e conteúdo da página, o CSS é responsável pela formatação dos elementos e a linguagem de programação JavaScript torna a página mais dinâmica e interativa (MEYER, 2010).

O *Cascading Style Sheets* (CSS) permite formatar as informações entregues pelo HTML. Essa informação pode ser qualquer objeto: imagem, texto, vídeo, áudio ou qualquer outro elemento criado. Assim como no HTML5, o CSS3 é na verdade uma mistura de *tags* antigas e novas (SANDERS, 2011).

Na programação utiliza-se o JavaScript, uma linguagem orientada a objetos baseada em ECMAScript, executada pelo navegador que acessa uma página Web, tornando-a mais dinâmica e interativa. Em resumo, JavaScript serve para tratar os comportamentos ou manipular elementos específicos na estrutura do HTML, inclusive estilos CSS. O JavaScript também faz parte da tecnologia Ajax (*Asynchronous JavaScript and XML*) que possibilita interação em tempo real com o lado servidor sem a necessidade de recarregar a página (MEYER, 2010; LALLI, BUENO e ZACHARIAS, 2008).

Além de novos elementos e recursos de multimídia, uma das características mais esperadas do HTML5 é o elemento que proporciona funções de desenho, chamado *Canvas*. Mas para obter o máximo possível do *Canvas* se faz necessário JavaScript e CSS3 que, por isso, desempenham um papel importante no HTML5 (SANDERS, 2011). O elemento *Canvas* é uma *Application Programming Interface* (API) destinada a delimitar uma área para renderização gráfica de elementos visuais em tempo real, permitindo desenhar via JavaScript, utilizando o conceito de bitmap, pixel a pixel (SEBBEN e GUEDES, 2011).

O HTML5 também dará suporte ao padrão SVG (*Scalable Vectorial Graphics*), formato aberto e gratuito recomendado pelo W3C, sendo uma grande aposta para uma possível revolução no desenvolvimento de interfaces para internet (SEBBEN e GUEDES, 2011). O SVG é uma linguagem para marcação de gráficos vetoriais, baseado em XML acessível via DOM (*Document Object Model*), com ele o conteúdo é acessível a leitores de tela, e possui gráfico escalável, que não perde resolução ou serrilha ao redimensionar (FERREIRA e EIS, 2013).

O surgimento dessa nova tecnologia tem aberto um novo mercado de jogos na internet, tornando possível o desenvolvimento por meio de tecnologias abertas e livres de licenças proprietárias. O uso da API JavaScript para a construção de figuras geométricas utilizando como área para desenho o elemento *Canvas*, embutido no código HTML da página, possibilita a criação de jogos que rodam direto no navegador de maneira leve e prática (HENRIQUES et al., 2011).

Outra melhoria trazida pelo HTML5 é a inclusão de uma série de elementos que ajudam na definição e organização dos

setores do documento, agregando maior semântica ao conteúdo e simplificando o trabalho de sistemas de busca. (FERREIRA e EIS, 2013).

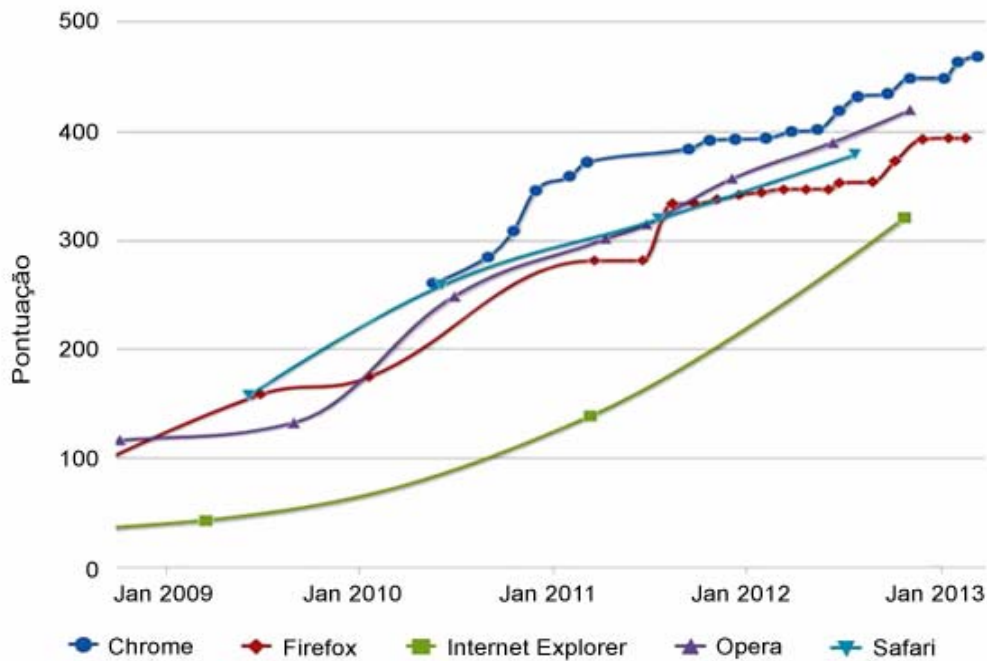
O HTML5 é um padrão Web de propósito geral, mas muitos dos seus recursos são destinados para tornar as aplicações Web mais parecidas com as aplicações *desktop*. Além disso, seus usuários podem interagir com os aplicativos e jogos, por meio de diferentes dispositivos, sem a necessidade de instalação prévia ou uso de algum *plug-in*. (PRADO, 2012)

Atualmente, a maioria dos navegadores já implementam parcialmente as recomendações estabelecidas pelo grupo WHAT-WG, porém a Web ainda não está preparada para uma migração repentina e total para essa nova linguagem. Nem todos os desenvolvedores conseguiram se atualizar em relação às novidades e padrões, e provavelmente nem todos os usuários aceitariam trocar seu navegador (BATISTA, 2009). Outro fator preocupante está relacionado à retro compatibilidade com versões antigas de *browsers* como Internet Explorer, que não possuem praticamente nenhum suporte ao HTML5 (FERREIRA e EIS, 2013).

A especificação atual do HTML5, apesar de funcional, ainda não está finalizada. Os navegadores estão adicionando novos recursos ativamente, e eventuais alterações poderão ocorrer, mas a tecnologia está evoluindo de maneira rápida e prática para trazer melhorias para plataforma Web (LUBBERS, ALBERS e SALIM, 2011).

Na Figura 2 é mostrada a evolução constante dos navegadores em adicionar e melhorar os recursos para suporte ao HTML5. A pontuação máxima de 500 significa que todas as especificações foram atendidas nos testes.

**Figura 2 - Pontuação dos navegadores no suporte ao HTML5**



**Fonte:** Adaptado de THE HTML5 TEST (2013)

## NOVAS FERRAMENTAS

Tanto o HTML5, como o CSS e JavaScript utilizam código aberto e em formato texto, permitindo o desenvolvimento em qualquer editor de textos. Mas as grandes empresas já começaram a lançar ferramentas profissionais para se trabalhar com essas novas tecnologias.

O Edge é uma nova ferramenta da Adobe, a mesma empresa que produz o Flash, que permite o desenvolvimento de aplicações para Web utilizando o conjunto de tecnologias HTML5, CSS e JavaScript. O Edge procura oferecer as mesmas funcionalidades de animação e interatividade que o Flash oferece, com interface e maneira de trabalhar semelhantes, mantendo a integração com os outros programas da suíte Adobe. Várias funções e conceitos foram transferidas de uma ferramenta para a outra, inclusive o conceito de linha do tempo (*timeline*), na espera de que muitos desenvolvedores que usarão o Edge já tenham conhecimento em Flash. O conteúdo criado na ferramenta não depende de

nenhum *plug-in* e pode, portanto, ser visto em qualquer navegador ou dispositivo móvel com suporte à tecnologia HTML5. Para o funcionamento da aplicação gerada pelo Edge, é necessária a distribuição de certas bibliotecas JavaScript, dentre elas a jQuery e a Adobe Edge Runtime (LABRECQUE, 2012).

Também lançada pela Adobe, o Wallaby é uma ferramenta de exportação de conteúdo, que permite converter projetos criados no Flash para o HTML5, aproveitando assim seus avançados recursos de desenho e animação. Apesar de ser uma boa alternativa para seus usuários, apresenta algumas limitações na otimização de código, principalmente em projetos mais complexos (MCLEAN, 2011).

Já a Microsoft disponibilizou o Lightswitch, uma ferramenta de desenvolvimento rápido de aplicações (RAD) com banco de dados utilizando o mínimo de código. Integrado ao Visual Studio 2012, é uma solução voltada ao desenvolvimento de aplicações simples. Inicialmente o Lightswitch utilizava o Silverlight para criação de interfaces com o



cliente, mas a versão mais recente já possui suporte ao HTML5 (CILIBERTI, 2013).

Outras ferramentas de desenvolvimento web da Microsoft como o WebMatrix também já oferecem suporte ao HTML5, além da versão mais recente do seu navegador, o Internet Explorer 10, o que demonstra a crescente expansão da nova tecnologia. Segundo CILIBERTI (2013, p. 17, tradução nossa) “Embora a Microsoft tenha se comprometido a continuar apoiando Silverlight até 2021, a estratégia RIA da Microsoft mudou para HTML5”.

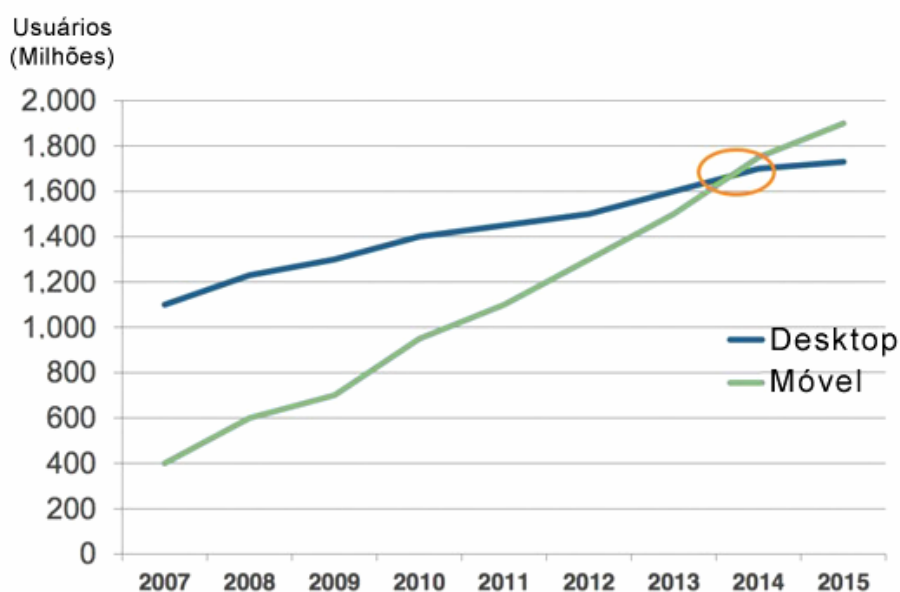
Pelo fato das tecnologias do HTML5 não serem proprietárias, começa-se perceber o

aparecimento de um número maior de ferramentas disponíveis, criadas por empresas menores ou por comunidades, e muitas voltadas ao desenvolvimento para dispositivos móveis.

## RESULTADOS E DISCUSSÃO

Talvez o maior motivo de impedimento para os *plug-ins* e ascensão do HTML5 seja o aumento do acesso à Internet por meio de dispositivos móveis como *tablets* e *smartphones*. Na Figura 3 pode-se observar esse aumento que, a partir de 2014, irá superar o acesso por computadores de mesa (*desktops*).

**Figura 3 - Previsão do número de usuários da Internet**



**Fonte:** Adaptado de COMSCORE e MORGAN STANLEY (2012)

No geral, as aplicações em Flash rodam sem grandes problemas em diversas plataformas, tanto que a Adobe relata que seu *plug-in* está instalado em 99% dos computadores. Porém, quando se trata de plataformas móveis o cenário é diferente, já que muitas delas impõem restrições ao Flash Player, chegando até o ponto de bani-lo completamente, como acontece no iOS da Apple (LABRECQUE, 2012).

Em abril de 2010, Steve Jobs, em nota publicada no site da Apple, mostrava sua aversão ao Flash. No texto, ele dizia “[...] a era móvel é sobre dispositivos com baixo consumo de energia, interfaces de toque e padrões abertos da web – todas as áreas em que o Flash é insuficiente”. E concluindo Jobs escreve “Novos padrões abertos criados na era móvel, como o HTML5, irão triunfar em aparelhos móveis (e PCs também).



Talvez a Adobe devesse se focar mais em criar ótimas ferramentas HTML5 para o futuro [...]”.

Contudo, o padrão HTML5 que ainda está em desenvolvimento, não foi completamente ratificado pelo W3C, e pode demorar algum tempo para que isso ocorra. Navegadores diferentes tendem a renderizar o HTML de forma diferente, além de possuírem níveis diferentes de compatibilidade com seus vários recursos. Além disso, muitos usuários ainda utilizam navegadores antigos que não oferecem nenhum suporte à nova tecnologia (ANDERSON, 2012).

A grande força do HTML5 é seu alcance, visto que para uma plataforma de desenvolvimento se tornar popular deve ser acessível pelo maior número de usuários e rodar nos mais diferentes dispositivos possíveis. Outro mérito nesse aspecto são as comunidades de desenvolvedores que continuam surgindo e crescendo (ANDERSON, 2012).

Porém, segundo MCLEAN (2011, p. X, tradução nossa) “Se um destas tecnologias substituirá completamente a outra é desconhecido; parece mais provável que elas simplesmente continuarão coexistindo”. E de acordo com TRETOLA (2011, tradução nossa) “O maior equívoco é achar que você pode usar o HTML5 para fazer tudo o que pode ser feito com o Adobe Flash”.

Muitas vezes o Flash e o HTML podem ser usados de maneira complementar. Já existem numerosos sites que utilizam o HTML como base, deixando os dados em formato padronizado e aberto aos mecanismos de busca, com o Flash por conta de prover animações ou interatividade mais sofisticadas (MCLEAN, 2011).

## CONSIDERAÇÕES FINAIS

A Internet evoluiu e continua evoluindo muito, está entrando agora na era da mobilidade, do código aberto e semântica na Web. Os *tablets* e *smartphones* oferecem uma nova interface com o usuário e o mesmo conteúdo deve estar disponível para

qualquer dispositivo. O HTML5 surge com a proposta de desenvolver aplicações com foco na interoperabilidade entre as diversas plataformas e dispositivos, com o uso de tecnologias abertas e padronizadas. (VOSSSEN e HAGEMANN, 2007; JOBS, 2010).

Porém, deve demorar a ser decretado o fim da era dos *plug-ins* para aplicações Web. Ferramentas como o Flash, Silverlight, Java entre outros, possuem vários anos de existência e adquiriram nesse tempo numerosos recursos que o HTML5 ainda não contempla. Além disso, existe um vasto conteúdo que já foi criado e está em uso nessas plataformas, o qual não pode ser simplesmente descartado.

Ambas as tecnologias devem continuar caminhando juntas por um bom tempo. Espera-se que o HTML5 evolua bastante e possa conquistar cada vez mais espaço nos próximos anos. No entanto, é difícil prever quando ou mesmo se as ferramentas de *plug-in* deixarão de existir por completo algum dia.

## REFERÊNCIAS

- ANDERSON, C. **Pro Business Applications with Silverlight 5**, 2.d. New York: Apress, 2012.
- BATISTA, D. C. T. **O Impacto do HTML5 no Desenvolvimento para a Internet**. Universidade Tecnológica Federal do Paraná, Cornélio Procopio, 2009.
- CILIBERTI, J. **ASP.NET MVC 4 Recipes: A Problem-Solution Approach**. New York: Apress, 2013.
- COMSCORE; MORGAN STANLEY. 2012. Disponível em: <<http://www.businessinsider.com/mobile-will-eclipse-desktop-by-2014-2012-6>>. Acesso em: 15 ago. 2013.
- FERREIRA, E.; EIS, D. **Curso de HTML5**. Disponível em: <<http://www.w3c.br/Cursos/CursoHTML5>>. Acesso em: 10 jun. 2013.
- GLADSTIEN, K. **Flash Game Development in a Social, Mobile, and 3D World**. Boston: Cengage Learning, 2013.

GONÇALVES, F. A. S.; CANESIN, C. A. **Java applets para um software educacional distribuído em eletrônica de potência**. SBA Controle & Automação, vol.13, n°3. Campinas, 2002.

HARRIS, T. **Como funcionam as animações para a Web**. Disponível em:

<<http://informatica.hsw.uol.com.br/animacoes-para-a-web.htm>>. Acesso em: 15 mai. 2013.

HENRIQUES, A.; VARGAS, M.; AUAD, T.; KNOP, I. **O navegador como plataforma para jogos: Uma experiência extracurricular para desenvolvimento de software**. SBC - Proceedings of SBGames, Salvador, 2011.

JAVA-Oracle. **Learn About Java Technology**. Disponível em:

<<http://www.java.com/en/about/>> Acesso em: 15 mai. 2013.

JOBS, S. **Thoughts on Flash**. 2010. Disponível em: <<http://www.apple.com/hotnews/thoughts-on-flash>>. Acesso em: 16 mai. 2013.

LABRECQUE, J. **Adobe Edge QuickStart Guide**. Birmingham: Packt Publishing, 2012.

LALLI, F. M.; BUENO, F.; ZACHARIAS, G. K. **Evolução da Programação Web**. Faculdade Comunitária de Campinas Unidade III, Campinas, 2008.

LEGGETT, R.; De BOER, W.; JANOUSEK, S. **Foundation Flash Applications for Mobile Devices**. Berkeley: Apress, 2007.

LIKNESS, J. **Designing Silverlight Business Applications: Best Practices for Using Silverlight Effectively in the Enterprise**. Boston: Addison-Wesley, 2012.

LOPES, J. S.; TAVARES, R. O. **JavaFx: Uma Abordagem ao Desenvolvimento de Aplicações RIA**. Universidade Fumec Faculdade de Ciências Empresariais, Belo Horizonte, 2010.

LUBBERS, P.; ALBERS, B.; SALIM, F. **Pro HTML5 Programming**, 2.ed. New York: Apress, 2011.

MCLEAN, I. L. **Creating HTML5 Animations with Flash and Wallaby**. Sebastopol: O'Reilly Media, 2011.

MEYER, J. **The Essential Guide to HTML5: Using Games to learn HTML5 and JavaScript**. New York: Apress, 2010.

PRADO, E. F. **Introdução ao Desenvolvimento de Games com GWT e HTML5**. SBC - Proceedings of SBGames - Universidade Federal de São Carlos, São Carlos, 2012.

SANDERS, B. **Smashing HTML5: Técnicas para a Nova Geração da Web**. Porto Alegre: Bookman, 2011.

SEBBEN, N.; GUEDES, A. L. **Gráficos Vetoriais Escaláveis na criação de interfaces de sistemas para a Internet**. Desenvolvimento de Sistemas para Internet - Universidade do Oeste de Santa Catarina, São Miguel do Oeste, 2011.

SPAKI, E.; ALVES, E.; FARIAS, J. A. L.; LINO NETO, J.; ALTHMANN, M. F.; DURÃES, R.; KONO, R. **Desenvolvendo para web usando o Visual Studio 2008**. Tijuca: Brasport, 2008.

STAT OWL. 2012. Disponível em: <[http://www.statowl.com/plugin\\_overview.php](http://www.statowl.com/plugin_overview.php)>. Acesso em: 01 mai. 2013.

THE HTML5 TEST. 2013. Disponível em: <<http://html5test.com/results/desktop.html>>. Acesso em: 15 jul. 2013.

TRETOLA, R.; et al. **HTML5, Flash and RIAs: 18 Industry Experts Have Their Say**. 2008. Disponível em: <<http://active.tutsplus.com/articles/roundups/html5-and-flash-17-industry-experts-have-their-say/>>. Acesso em: 16 mai. 2013.

VOSSSEN, G.; HAGEMANN, S. **Unleashing Web 2.0: From Concepts to Creativity**. Burlington: Morgan Kaufmann Publishers, 2007.

WENZ, C. **Essential Silverlight 2 Up-to-Date**. Sebastopol: O'Reilly Media, 2008.