



April 21, 2010

## Does HTML 5 Herald The End Of RIA Plug-Ins? Not Really

The Pending Demise Of Popular Rich Internet App Platforms Is Wildly Overstated

by **Jeffrey S. Hammond**

with John R. Rymer and Adam Knoll

### EXECUTIVE SUMMARY

Will HTML 5 make rich Internet application (RIA) technologies such as Adobe Flash/Flex and Microsoft Silverlight obsolete? For at least the next five years, the answer is a definite “no”; inconsistent implementations of the draft HTML 5 specification and immature tooling make building HTML 5 apps that work consistently across browsers and operating systems a real challenge. Furthermore, this “either/or” scenario is driven only by vendor politics, not by developer realities. Ultimately, HTML 5 and RIA platforms will be complementary technologies, and enterprise development shops will need to invest in both approaches to deliver expressive applications that combine reach and richness.

### ADOPTION OF RIA PLATFORMS IS ON THE RISE; WILL HTML 5 HALT IT?

Over the past two years, Forrester clients have adopted rich Internet application (RIA) platforms because they demonstrate clear benefits, including improved customer satisfaction, higher Web conversion rates, and reduced application maintenance costs. Then along came HTML 5, which promises to provide all of the benefits of RIA platforms but in an open, standard way supported by any browser. No plug-in downloads required! The question: Should developers turn away from RIA platforms now and embrace HTML 5?<sup>1</sup>

In short: No, and here’s why.

### The Early Winners Are ASP.NET Ajax, Plug-Ins, And Open Source Ajax

Forrester’s latest Enterprise And SMB Software Survey, North America And Europe, Q4 2009, shows that the number of development decision-makers whose companies use RIA platforms to deliver applications is up significantly, from 26% in 2008 to 34% in 2009 (see Figure 1). At the same time, the overall number of inquiries Forrester receives regarding RIA platforms is down by 15%. What’s going on here? It’s simple: As clients have transitioned from early adopters to aggressive exploiters of RIA technology, we no longer get questions about the viability of RIA platforms per se. Inquiries now tend to focus on the relative strengths and weaknesses of individual platforms.

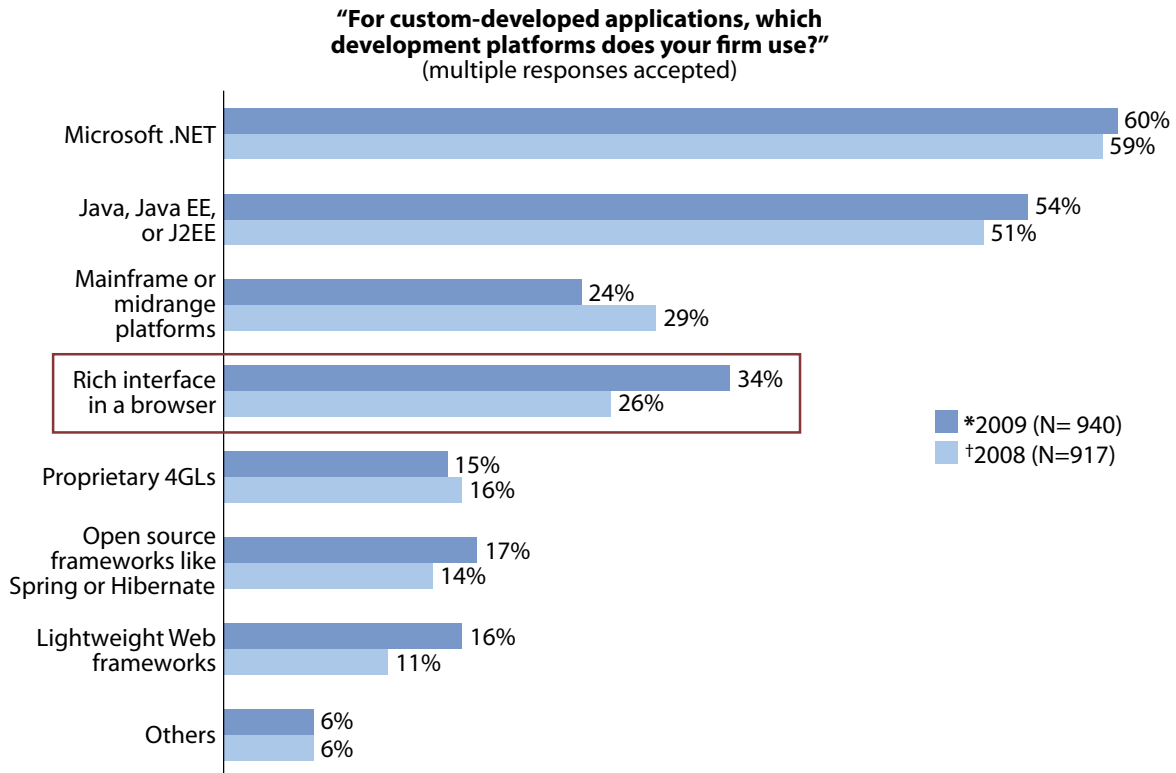
In the Forrester/Dr. Dobb’s Global Developer Technographics® Survey, Q3 2009 and the Eclipse Community Survey 2009, we found that there is a strong correlation between the server platform technology a shop uses and the RIA strategy it chooses to employ. Specifically:



#### Headquarters

Forrester Research, Inc., 400 Technology Square, Cambridge, MA 02139 USA  
Tel: +1 617.613.6000 • Fax: +1 617.613.5000 • [www.forrester.com](http://www.forrester.com)

**Figure 1** Rich Internet Application Platform Adoption Is Up In 2009



Base: platform software decision-makers at North American and European enterprises and SMBs who use development platforms

\*Source: Forrester’s Enterprise And SMB Software Survey, North America And Europe, Q4 2009

†Source: Forrester’s Enterprise And SMB Software Survey, North America And Europe, Q4 2008

56768

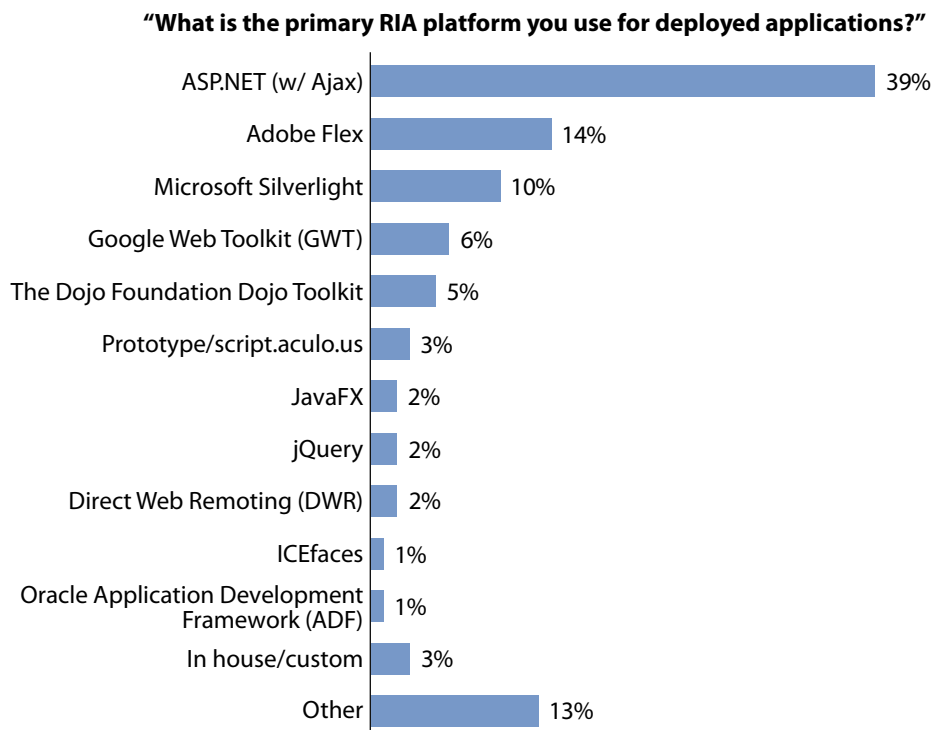
Source: Forrester Research, Inc.

- **.NET platform developers adopt ASP.NET Ajax and Microsoft Silverlight.** In the Forrester/ Dr. Dobb’s Global Developer Technographics® Survey, Q3 2009, we asked 347 app dev professionals who use RIA platforms to identify their primary RIA platform (see Figure 2). The overwhelming choice was ASP.NET with Ajax. Of the 39% of developers who use ASP.NET with Ajax, more than 90% of them use .NET on the server as their primary application server framework. .NET adoption on the server is not quite as high for Silverlight users: Only 80% use .NET as their primary application server framework, and almost 9% use an Apache Tomcat/ Silverlight combination.
- **Adobe Flex is more likely to be used by Java developers.** We see the exact opposite adoption trend when it comes to the Adobe Flash/Flex framework. Only about one in five Flex developers uses .NET as the primary application server framework. Flex developers are more likely to use Apache Tomcat (23%), Red Hat JBoss (17%), IBM WebSphere (8.3%), or no app server at all (17%).

- **OSS Ajax is crowding out commercial Ajax vendors.** While there are numerous commercial Ajax vendors providing excellent platform solutions, developers are tilting toward leaner, open source frameworks such as The Dojo Foundation Dojo Toolkit and Google Web Toolkit (GWT). In both the Forrester/Dr. Dobb's Global Developer Technographics® Survey and a survey performed by the Eclipse organization last year, Dojo and GWT were the most frequently adopted OSS Ajax frameworks — coming in at 5% and 6%, respectively, in the Dr. Dobbs survey and tied at 14% of Eclipse users each in the Eclipse survey (see Figure 3).

These trends underline a key hurdle that HTML 5 technology must overcome to be a ready substitute for today's RIA platform options; users expect it to be as low cost as the other options, but to be of use it must also integrate with Java and .NET server technology. Even if HTML 5 turns out to be a great spec when it reaches Candidate Recommendation state in 2012, it's not clear that this alone will be enough to reverse current RIA adoption trends.

**Figure 2** RIA Adoption By Platform — Forrester/Dr. Dobbs Developer Technographics® Survey



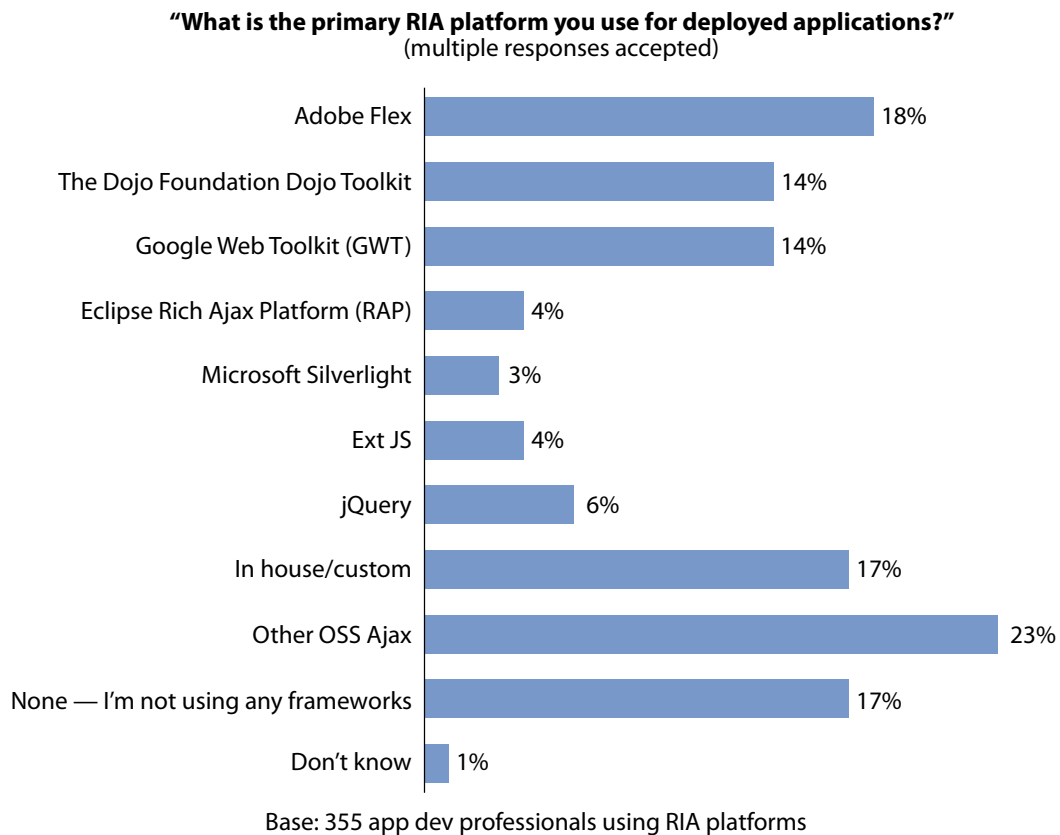
Base: 347 app dev professionals who use RIA platforms  
(percentages do not total 100 because of rounding)

Source: Forrester/Dr. Dobb's Global Developer Technographics® Survey, Q3 2009

56768

Source: Forrester Research, Inc.

**Figure 3** RIA Adoption By Platform — Eclipse Survey



Source: Eclipse Community Survey 2009

56768

Source: Forrester Research, Inc.

### HTML 5 Promises To Make RIA Plug-Ins Unnecessary

While it's possible that in the long run HTML 5 will become an acceptable substitute for some types of RIA applications, it's not there yet. Application development professionals shouldn't be overly concerned about their near-term investments in RIA platforms for three reasons:

- **As a draft standard, HTML 5 specifications are open to inconsistent implementation.** The 605-page World Wide Web Consortium (W3C) draft spec for HTML 5 may make for good bedtime reading (if you have insomnia), but it still leaves room for interpretation of ambiguities in how HTML 5 is rendered. As a result, when browsers render HTML 5 today, it's not hard to find significant differences in the resulting behavior.<sup>2</sup> Codec support is another challenge, because there's no agreed-upon standard for rich media across all popular browsers. The result is a situation similar to the early days of JavaScript programming, when developers would spend

hours chasing down inconsistent behaviors across browser and platform combinations. Over time, Forrester expects that these different behaviors will be reduced and perhaps isolated by cross-browser frameworks, but it will be a few years at a minimum before developers can expect consistent cross-browser/cross-platform behavior from HTML 5.

- **It will take years for HTML 5 to become a stable, ratified standard.** Browser vendors claim to have HTML 5 support today, but app dev pros should examine these claims closely, because in reality this means partial HTML 5 support. For example, Google Chrome, Opera, Firefox, and Apple Safari now support the <canvas> tag; however, Internet Explorer doesn't, and support for Scalable Vector Graphics (SVG) tags is all over the map.<sup>3</sup> Until IT rolls out the latest versions of these browsers, their partial support of HTML 5 will do developers little good. In the meantime, the piecemeal implementation of HTML will continue to create a moving target for developers, with no guarantees of real standardization until as late as 2020: the current expectation for when the draft specification will become a fully approved W3C standard. That's 10 years from now, and a lot can happen in the meantime.
- **It remains to be seen if the HTML 5 programming model will be more productive.** HTML 5 has significantly expanded capabilities that make it richer than ever before. But to take advantage of all the capabilities, developers need to know more than HTML: Cascading Style Sheets (CSS) for styling apps, JavaScript for behavior, and now SVG for performing vector operations and animations. In comparison, Flex developers need to know ActionScript, CSS, and MXML (Minimal XML), while Silverlight developers only need to know XAML (Extensible Application Markup Language) and C#. Design, development, and testing tools are also a productivity concern. Until tools and standard control libraries that support HTML 5 appear, developers will need to do their work with basic text editors instead of modern integrated development environments (IDEs). While that's fine for alpha developers, mainstream enterprise developers will want easy-to-use, what-you-see-is-what-you-get (WYSIWYG) tools before they will be fully comfortable working in an HTML 5 environment.

### The Bottom Line: HTML And RIA Platforms Will Be Complementary Technologies

In the long run, HTML 5 will have a significant impact on how Web applications are built — but as a complementary technology to leading RIA platforms, not a replacement. Though you can use a saber saw or a circular saw to cut lumber, each has its optimum uses; this is also the case for HTML 5 and Adobe Flash/Adobe Flex/Microsoft Silverlight (see Figure 4).

**Figure 4** Sizing Up HTML 5 And Plug-In-Based RIAs

Flexi-point	HTML 5	RIA containers
<b>Reach:</b> Can it be used on a wide variety of operating systems and hardware platforms?	●	◐
<b>System requirements:</b> Does it need a minimum install of additional components to work?	●	◐
<b>Basic controls:</b> Can developers use it to build simple applications?	●	●
<b>Advanced controls:</b> Can developers use it to build advanced applications (3D, vectors)?	◐	●
<b>Readiness:</b> How widely available is the technology today?	○	●
<b>Consistent experience:</b> What level of cross-platform variability exists?	◐	●
<b>Disconnected use:</b> What level of support is provided for occasionally connected apps?	◐	◐
<b>Tools:</b> Are WYSIWYG design and development tools available?	○	●
<b>Mobile:</b> What smartphone OSes will be supported?	●	◐

Limited ○ Adequate ◐ Extensive ●

56768

Source: Forrester Research, Inc.

## RECOMMENDATIONS

### TRACK HTML 5'S EVOLUTION FOR TOMORROW, BUT INVEST IN RIA PLATFORMS TODAY

In some sense, the juxtaposition of HTML 5 and RIA plug-ins is really a manufactured controversy. Let's face it: An "us versus them" approach always sells more magazines, and controversy creates a level of marketing buzz that benefits firms that are trying to introduce new innovations. HTML 5 does have potential, and shops should track its evolution. In the meantime:

- **Track browser adoption of individual HTML 5 elements.** Browsers are slowly adding incremental support for individual elements of HTML 5, such as the <canvas> tag. Only when all major browsers implement support for a particular tag is it safe for application developers to consider using it as part of their app dev tool kit. In addition to watching for consistent tag support, development shops should also check for consistent browser support for UI controls, DRM, and video codecs if these will be part of the applications they plan to build.

- **Watch how quickly the standards bodies dot their I's and cross their T's.** There are currently two bodies pushing HTML 5 forward: the Web Hypertext Application Technology Working Group (WHATWG) and W3C. Although the organizations are working in sync, there is still much work to be done, and it's likely that there will be bumps in the road between now and 2012 (when it's projected that HTML 5 will reach candidate recommendation status). Conservative shops that would prefer to wait for a fully approved recommendation still have a while to wait, as that's not projected to happen until 2022.
- **In the meantime, use Ajax and plug-ins to get work done today.** If you need to deliver applications that work across all major browsers today, then it makes sense to continue to use Ajax frameworks such as Dojo Toolkit, GWT, and ASP.NET Ajax as well as plug-in-based technologies such as Adobe Flash and Microsoft Silverlight. These options aren't going anywhere; they work, and they are a low-cost way to get the job done.

## SUPPLEMENTAL MATERIAL

### Methodology

Forrester's Enterprise And SMB Software Survey, North America And Europe, Q4 2009, was fielded to 2,165 IT executives and technology decision-makers located in Canada, France, Germany, the UK, and the US from SMB and enterprise companies with two or more employees. This survey is part of Forrester's suite of Business Data Services studies. Forrester fielded the survey from September 2009 to November 2009. LinkedIn fielded this survey online on behalf of Forrester. Survey respondent incentives included gift certificates and research summaries. We have provided exact sample sizes in this report on a question-by question basis.

Forrester's Enterprise And SMB Software Survey, North America And Europe, Q4 2008, was fielded to 2,227 IT executives and technology decision-makers located in Canada, France, Germany, the UK, and the US from companies with two or more employees. This survey is part of Forrester's suite of Business Data Services studies. Forrester fielded the survey from December 2008 to February 2009. e-Rewards fielded this survey online on behalf of Forrester. e-Rewards provided incentives to survey respondents. We have provided exact sample sizes in this report on a question-by question basis.

Forrester's Business Data Services fields eight business-to-business technology studies in 19 countries each calendar year. For quality control, we carefully screen respondents according to job title and function. Business Data Services ensures that the final survey population contains only those with significant involvement in the planning, funding, and purchasing of IT products and services. Additionally, quotas are set for company size (number of employees) and industry as a means of controlling the data distribution and establishing alignment with IT spend calculated by Forrester analysts.

In addition to sampling error, one should bear in mind that the practical difficulties in conducting surveys can introduce error or bias into the findings of opinion polls. Other possible sources of error in polls are probably more serious than theoretical calculations of sampling error. These other potential sources of error include question wording, question ordering, and nonresponse. As with all survey research, it is impossible to quantify the errors that may result from these factors without an experimental control group, so we strongly caution against using the words “margin of error” in reporting any survey data.

These statements conform to the principles of disclosure of the National Council on Public Polls.

We have illustrated only a portion of survey results in this document. For access to the full data results, please contact [bds@forrester.com](mailto:bds@forrester.com)

Forrester in cooperation with Dr. Dobbs magazine fielded its Forrester/Dr. Dobbs Global Developer Technographics® Survey, Q3 2009 to 1,298 application development professionals. This online global survey was fielded in August 2009.

Exact sample sizes are provided in this report on a question-by-question basis. Panels are not guaranteed to be representative of the population. Unless otherwise noted, statistical data is intended to be used for descriptive and not inferential purposes.

If you're interested in joining one of Forrester's research panels, you may visit us at <http://Forrester.com/Panel>.

The Eclipse Foundation fielded its Eclipse Community Survey 2009 to 1,481 application development professionals. This online global survey was fielded in May 2009.

## ENDNOTES

- <sup>1</sup> For a small sample of some of the “will HTML 5 kill flash” articles, see: Paul Krill, “HTML 5: Could it kill Flash and Silverlight?” *InfoWorld*, June 16, 2009 (<http://www.infoworld.com/d/developer-world/html-5-could-it-kill-flash-and-silverlight-291>); Stephen Shankland, “HTML vs. Flash: Can a turf war be avoided?” CNET News, February 3, 2010 ([http://news.cnet.com/8301-30685\\_3-20000037-264.html](http://news.cnet.com/8301-30685_3-20000037-264.html)); and Michael V. Copeland, “Behind the Adobe-Apple cold war,” *FORTUNE*, January 29, 2010 (<http://brainstormtech.blogs.fortune.cnn.com/2010/01/29/behind-the-adobe-apple-cold-war/>).
- <sup>2</sup> As an example, at the day-two keynote at MIX10, Dean Hachamovitch showed how different browsers implement SVG transforms in different default ways. In one example, the Internet Explore (IE) 9 preview allowed rendered circles to spill outside a rendered rectangle, while other browser made them disappear. In another example, he showed how animation around blended corners of a rectangle created unforeseen artifacts in some browsers. To watch these differences, see: MIX10 The Next Web Now (<http://live.visitmix.com/MIX10/Sessions/KEY02>).

- <sup>3</sup> For an example of how browser support for SVG tags varies, see the W3Cs testing result at Codedread (<http://www.codedread.com/svg-support.php>), as run through the official W3C's SVG Test Suite Overview ([http://www.w3.org/Graphics/SVG/WG/wiki/Test\\_Suite\\_Overview](http://www.w3.org/Graphics/SVG/WG/wiki/Test_Suite_Overview)).