

Adobe Flex: Interview with Nicholas Hamming - Software Analyst / Developer at Accreon

Jacob Sanford - President, Elemental Codeworks Inc.

Introduction

Adobe Flex is a technological ensemble provided by Adobe to aid in the development of cross-platform rich web applications based on adobe Flash. It was presented originally due to that fact that in the past, developers had difficulty adapting to the 'animation metaphor' upon which Flash was originally designed.

A developer that has recently adopted the use of Flex in their projects is Accreon (<http://www.accreon.com>), a business consulting and information technology service provider based in Fredericton, New Brunswick. I spoke to their Software Developer and my friend Nicholas Hamming recently on the topic of Adobe Flex.

Interview

JS: Hello Nick, would you like to introduce yourself to the readers?

NH: I'm a Software Analyst / Developer at Accreon (<http://www.accreon.com>), one of Atlantic Canada's largest full service technology and management services companies. I've mostly been working on building web applications, and have been able to explore many of its related aspects: system architecture, security, application design, and process analysis.

I'm currently working on establishing Flex as an additional/optional presentation layer to the web applications we build using one of our long-standing "Solution Development" tools, T2M (which stands for Time-To-Market).

JS: For people unaware, could you give us an overview of the Adobe Flex suite and how it relates to Adobe's well-known Flash product.

NH: Flex is an open source SDK (free as in beer) built on Flash that allows the developer to create SWF files that will run in Adobe's Flash player. Flex is a meld of both pure ActionScript, and XML (MXML) for declarative markup.

Flex is mainly intended for building rich-client Internet applications (RIA) that run on the web (via the Flash player). It is not for server-side logic/processing.

JS: With a rise in popularity of 'Online Desktop' applications versus traditional HTML page-based solutions, considerations of standardization and end-user accessibility become a real concern.

Adobe's Flash product has obviously won it's way into the status-quo for video delivery and media streaming, but how does the performance of a 'compiled' database-driven flash application compare to a similar Javascript implementation?

NH: Flex/Flash is based on ActionScript, which is a scripting language based on ECMAScript (much like JavaScript). However, unlike JavaScript, the ActionScript in a compiled Flex SWF file is processed in the Flash player instead of the web browser. This allows the Flex application to always look, feel, and act the same regardless of which browser (or Operating System) is used.

One of the main facets of performance for almost every web application is transferring data between the server and the client (browser). For most existing web applications, this would be the time it takes for the browser to receive the HTTP response (whether that response contains basic HTML or the response to an AJAX/JSON call). Regardless of what is in the HTTP response, it is always delivered as ASCII characters (plain text).

One exciting aspect to using Flex (aside from the shiny new graphics) is the ability to leverage Adobe's Action Message Format (AMF) for transferring data between the Flash player and the server. AMF is a *binary* file format that is used to represent a serialized ActionScript object. With the data being transferred in a binary format, transmission times can be 10x faster! Visit James Ward's popular AMF benchmark application for your own test-drive.
(<http://www.jamesward.com/blog/2007/04/30/ajax-and-flex-data-loading-benchmarks/>)

However, because Flex is relatively young (Flex 1.0: March 2004; Flex 3.0: February, 2008), many standards are yet to be forged.

JS: That having been said, there seems to be a strong opinion that contrasting Flex and AJAX is flawed - similar to comparing apples and oranges. Is this true? Do these technologies compliment each other or should they be viewed as competition?

NH: There certainly are some glaring differences between AJAX and Flex: browser vs. Flash player; ASCII vs. AMF; fragmented framework market vs. clear, single-company driven direction.

Typically, these two technologies are viewed as competition, or at least that you have to pick between them, but heck, if you wanted to dive into Flex and not remove your trusted AJAX hat then maybe you'd be interested in the Flex Ajax Bridge

[\[http://livedocs.adobe.com/flex/3/html/ajaxbridge_2.html\]](http://livedocs.adobe.com/flex/3/html/ajaxbridge_2.html) (it's part of the Flex SDK).

JS: Earlier this year, there was a well publicized story where Steve Jobs announced at a Shareholder Meeting that Flash/Flex applications were unsuitable for the iPhone.

This brings us back to standardization - some people believe that since Javascript is interpreted differently across different browsers and operating systems that it is the least accessible of the two products. Others voice that the lack of universal flash support across non-Windows systems creates a weakness that will inhibit the growth of Flex. Where do you stand?

NH: Adobe is working hard on universal Flash player support. Currently, Flash Player 10 is available for Windows, Macintosh, and 32-bit Linux. Pre-release versions for Solaris and 64-bit Linux platforms are available on Adobe Labs.

[\[http://labs.adobe.com/downloads/flashplayer10.html\]](http://labs.adobe.com/downloads/flashplayer10.html) There have been some rumblings of the Flash player eventually making its way onto the iPhone, but unfortunately that sounds a way off still [\[http://blog.wired.com/gadgets/2008/11/adobe-flash-on.html\]](http://blog.wired.com/gadgets/2008/11/adobe-flash-on.html).

One HUGE advantage Flex has for delivering rich-client applications (the kind that are exceptionally difficult to reach with JavaScript alone) is existing market penetration of the Flash player. [\[http://www.adobe.com/products/player_census/flashplayer/\]](http://www.adobe.com/products/player_census/flashplayer/) It's really hard to beat a steady 98% penetration rate.

[\[http://www.adobe.com/products/player_census/flashplayer/version_penetration.html\]](http://www.adobe.com/products/player_census/flashplayer/version_penetration.html) Any comparable RIA offering such as JavaFX or Silverlight still needs to get their audience to go out and get whatever runtime environment is needed.

JS: When it comes to 'Online Desktop' applications, Elemental Codeworks primarily produces end-user interfaces to ease the updates of database driven websites or general content creation. Other than the obvious improvement in media handling (which only mildly applies to this scenario), what do you feel would be the top 3 attractions to consider also adopting the Flex Platform into our workflow?

NH: (1) WYSIWYG content editing. The user can see construction progress in real-time, and how changes affect other sections of content.

(2) Rich-client Look, Feel, and Act - regardless of where the user is coming from. This provides you with the confidence that all users will experience your workflow in the same manner.

(3) Performance. Faster data transmissions and enabling selected parts of the user interface to be updated (instead of having to refresh the entire page).

JS: For a larger, more B2B oriented solutions such as those that Accreon produces, what features of the Flex development process do you feel appeal to your development team?

NH: Recently, I've been working on a project where we are helping a client move one of their existing applications from a client/server setup, using PowerBuilder, out onto the web. One expectation to manage is the differences between client/server applications, with their seemingly immediate responses, and web applications that traditionally require an entire page to be downloaded and refreshed for each action. AJAX and JSON have been the popular approaches to enable web applications to update only parts of a page, and to enable micro-actions. However, AJAX and JSON only get us part of the way to what we had with client/server. Flex reintroduces that feeling of being very close to our data. It's right there behind the screen, and we can interact with it immediately.

Oh, and everyone likes shiny things.

JS: Through our discussions, I've gotten the impression for you that you believe that Adobe releasing components of the Flex 3 SDK under the Mozilla Public License is an extremely positive move. Why? Why do you believe that Adobe chose to retain the closed-source of the Flex Builder IDE and LifeCycle Data services?

NH: For all the wonderful things that Adobe has done with releasing the Flex SDK and AMF as open source, they don't want to be throwing away their entire business model. It's not like they can draw upon some sort of ad-based revenue.

Maybe more importantly, I think its best that the core of how Flex applications are built and consumed (Flex Builder and Flash player) is still under the control of a single company with a very clear direction of where they want to bring Flex. A downside for open-source projects, that is difficult to be avoided without an overseeing entity, are spin-offs that fragment the community and dilute the offering.

As for the stifling price tag of LiveCycle Data Services, Adobe has also released most of it as BlazeDS, an impressive open-source alternative that will get your Flex application connected to its data.

JS: Any final thoughts?

NH: It seems to me that Adobe really wants to help people get started in the rich-client application arena, and (of course) to use Flex to get there. In my experience, it certainly has been an exciting journey, although not without its pitfalls and snares. There is the learning curve if you want to swim in the deep-end with ActionScript and MXML, and being on the bleeding edge of the technology is a highly dynamic and ever-changing field.

Overall, I've been thrilled to have the chance to expand, and learn, and apply something new.

Nicholas Hamming is a Software Analyst / Developer at Accreon (<http://www.accreon.com>), a business consulting and information technology service provider offering leadership that helps public and private organizations implement innovative technology solutions to increase business agility and improve performance.

Jacob Sanford is the founder and president of Elemental Codeworks Inc. (<http://www.elementalcodeworks.com>), a website development and promotion firm. Elemental Codeworks partners with new and established websites to develop and increase the popularity of their web presence through effective link building and buzz marketing. He can be reached via E-Mail at jake@elementalcodeworks.com.