



Interview with Allen Levine: BLX's component for Adobe® LiveCycle™ Forms



Allen Levine
Business Loan Express

Business Loan Express, LLC (BLX), is one of the largest non-bank small business and commercial lenders in the United States. BLX has developed many applications to streamline the loan process, including a ColdFusion component that leverages the Adobe® LiveCycle™ Forms web service to facilitate PDF document rendering in an environment with no in-house Java expertise. Adobe Enterprise Developer Relations (DR thereafter) interviewed Allen Levine (AL below), developer of the component, about this project.

DR: Tell us a bit about yourself as a developer.

AL: I am the webmaster for BLX, responsible for development and support for the company's public website. I've been doing web development for 13 years. My development background started with PASCAL, DELPHI, and then Visual Basic. Since 1998, I've been primarily focused on web development and ColdFusion.

DR: How long have you been using Adobe LiveCycle software?

AL: Only a couple of months. I started to play with LiveCycle software in June 2005.

DR: What's the background for this project?

AL: BLX is constantly streamlining our application process to improve customer service and turnaround time. In addition, we work closely with our brokers to provide them with the resources they need to support their customers. This project addresses both of these items directly as well as provides some additional benefits.

DR: What are your technical considerations for selecting the ColdFusion/LiveCycle architecture for the project?

AL: As a ColdFusion developer, I would naturally like to develop this component in the ColdFusion environment. I investigated several document rendering solutions. However, most of them required the use of obscure third-party plug-ins, which I did not want for our public website. Adobe Reader® is the only plug-in that I would consider because it is basically a standard now and it is freely available for many platforms.

DR: What does the component exactly do?

AL: The component uses the LiveCycle Forms 7.0 web service to render documents in a variety of formats such as HTML, XHTML, PDF, and PDF Form. The web service works by passing a gzipped, base64 encoded XDP file and returning a gzipped base64 encoded PDF file. The web service allows for merging of XML data into the PDF file, which is the primary function of this component. The most important feature is that I am prepopulating PDF Forms, not just a document. An important distinction of this component is that it can preserve the Form fields, conduct validation, and do other things.

DR: Why do you need such a component?

AL: Adobe LiveCycle Forms 7.0 is a J2EE application that currently supports JBoss®, WebLogic™, and WebSphere®. Because I am running a ColdFusion environment and need the ability to merge data to render a prepopulated PDF form file, the J2EE platform was a concern. Our organization is primarily a Microsoft® house and there is no in-house Java expertise. So, the only way that I can get buy in is to limit the amount of Java expertise that is required to support and leverage the technology.

That being said, my goal was to install the turnkey installation for JBoss and Windows and not have to do any Java programming, including creating or deploying any applications on the Adobe server. Basically, install the server software and never touch it again.

DR: Does the component meet your expectations?

AL: Yes, it does exactly what I set out to do. It handles all the necessary steps of implementing the LiveCycle Forms web service to render PDF and merge XML if desired, all using the default Turnkey Solution installation. 98% of the code is written in ColdFusion. There are only one or two places where I used Java code for the gzip and ungzip routines which are done in memory. Normally you could utilize reference. But the Java classes are embedded in the component. Actually, everything you need is contained in one component.

DR: Do you know that LiveCycle Forms 7.0 itself has several methods that you can use to invoke the Form Server Module?

AL: Yes, I know. There are a few options in front of me. To sum up, I can locally use an EJBClient object, remotely use an EJBClient object, and use SOAPClient object, Microsoft. NET client assembly, and FormServerFactory class. However, after spending quite a bit of time trying to leverage these options, I decided that the only solution that will fulfill my requirements was to use the web service.

DR: Why do you say so?

AL: First of all, except the Microsoft .NET client assembly, all the methods require Java expertise, which is not available in our organization. Second, although .NET is supported, as a ColdFusion developer I think it is a too complicated starting point for me for the project.

DR: As I understand, in the component you use SOAP messages to call the LiveCycle Forms web service. This is a complicated task. How do you handle SOAP messaging?

AL: Yes, the SOAP requests and responses requirements are somewhat complex. Because the SOAP messages contain multiple base64 encoded, gzipped strings, you have to handle binary data, data formatting routines, and gzip, parse the SOAP responses, and process the base64 encoded, gzipped XML documents contained in the parsed SOAP message.

In the component these tasks are done through the half a dozen or so methods that I created. These methods handle all the required functions for creating, logging, sending, and receiving the SOAP processes required for utilizing the LiveCycle Forms web service.

DR: Can you provide an example of the methods and describe how it works?

AL: Take CreateRequest for example. As the name indicates, it creates a request for SOAP messages. In order to make it work, you first create the InputData XML document, which will be passed as the InputData argument and placed in the CaptureServerRequest XML document. Then, you create and gzip the CaptureServerRequest XML document. Finally, you encode as base64 the gzipped XML document using the ISO-8859-1 character encoding standard and save it as CaptureServerRequestXMLBase64Encoded variable.

DR: How is this LiveCycle Forms solution being used in your current environment?

AL: It is being used in several areas of our public and private websites. The major implementation is in our loan processing application for loan applicants. It lets multiple people fill out a web-based form with several questions. They are then presented with a Loan Application that fits their needs, prepopulated with their personalized information, application ID's, dates, etc. The user can then fill out the application, save it, and submit the document to the company via email or mail.

Another area is the secure log-in that allows company representatives to retrieve company documents from our sites. When users are logged in, they can download those documents that are automatically prepopulated with their profile information.

Our brokers are also benefiting from this solution. They can now provide us with their company logos and download marketing materials provided by us that are branded utilizing their company information and logos. This has been met with a lot of enthusiasm and we are excited to expand the personalized marketing solutions.

DR: While using this solution, what cool features have you observed?

AL: For the loan processing application, because the form is prepopulated with the applicant's information, we can now track the document from a download on the Internet through the entire loan process.

Another cool feature is that we can track the referrer information. So if the user came to our site from a marketing initiative such as a banner ad, email campaign, partner site, etc., when they submitted their applications over the web or via mail, we can easily determine the source. This is a major factor for us in determining the success of individual marketing efforts.

Also, because the application is delivered in this process, we only have one version of the application at any time, thus eliminating the issues of making changes to the application and receiving back older versions.

DR: What are the next steps for the project?

AL: Now that we've got a process that allows us to quickly prepopulate forms based on the data that we want, the more we use it, the more ideas we have come up with that can enhance our customer service, like the log-in feature of our site and the broker use case. We are planning on implementing these ideas.

DR: What are some of your technical observations about working with ColdFusion and LiveCycle together for a project?

AL: At first I was apprehensive about even trying to create a ColdFusion-based solution. Since much of what I wanted to accomplish was undocumented, I was not even sure it would be possible. After a few conversations with Marc Silsbe at Adobe's North American Sales Engineering, I determined that although not easy, there was a good chance that my idea would work. From a development standpoint, I found that the two applications worked well together, especially when debugging. Post development, I have found that the Adobe LiveCycle Forms is very stable and the performance is great. I have not had to restart the Adobe LiveCycle Forms server in production since it has been implemented.

DR: Based on your experience with this component, what advice do you have for other developers?

AL: There is nothing you cannot do with ColdFusion. In fact, leveraging ColdFusion and Adobe LiveCycle Forms has turned out to be a great investment. We are continually finding new ways to leverage the solution. Don't be afraid, dive right in.

ABOUT THE AUTHOR

Allen Levine is the webmaster for Business Loan Express, a non-bank small business and commercial lender. Allen has been developing web applications for 15 years and has been developing in ColdFusion since 1997. He can be reached at alevine@BLX.NET.

FOR MORE INFORMATION

For more information about this success story or Adobe LiveCycle software, please contact the Adobe Enterprise Developer Relations team at dev_info@adobe.com. If you would like to try out Adobe LiveCycle software, please join the *Adobe Enterprise Developer Program*, which provides access to the LiveCycle software, as well as one year of unlimited email based support.

Better by Adobe.™

Adobe Systems Incorporated
345 Park Avenue, San Jose, CA 95110-2704 USA
www.adobe.com

Adobe, the Adobe logo, Acrobat, the Adobe PDF logo, Reader, and LiveCycle are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Microsoft is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries. All other trademarks are the property of their respective owners.

© 2006 Adobe Systems Incorporated. All rights reserved.

2/06

