

# Adobe's view of Flex and its commitments to Flex in the future

## Table of contents

- 1: Adobe Flex
- 2: Adobe's vision for Flex
- 4: Adobe runtime support of Flex
- 4: Adobe Flash Platform
- 5: Adobe tooling support for Apache Flex
- 5: Summary
- 5: For more information

Adobe® believes that Flex® is the best solution for enterprise and data-centric application development today, and that moving Flex into a community-driven open source project ensures the continued development and success of Flex for years to come. In this new model, the community will be responsible for the maintenance and evolution of the Flex SDK while Adobe will continue to evolve tooling and runtimes. This document provides Adobe's view of Flex, its contributions to the open source project, and its commitments to Flex in the future.

## Adobe Flex

Adobe Flex is a software development kit (SDK) for developing rich Internet applications (RIAs) targeting the Adobe Flash® Player in desktop browsers, Adobe AIR® for desktop, or Adobe AIR for mobile devices. The SDK provides a library of components collectively referred to as the Flex framework written in ActionScript® 3, the object-oriented programming language for the Flash® Platform. In addition to a compiler and debugging tools, the SDK defines MXML, a declarative XML syntax that developers can use to create and maintain complex user interfaces.

## History of Adobe Flex

In the early 2000s, web application developers had been struggling with lack of cross browser support of JavaScript and Cascading Style Sheets, the stateless nature of Hypertext Transport Protocol, and the limitations of HTML. Companies and developers were looking for ways to deliver better user experiences to improve customer interactions. Macromedia® helped define the concept of rich Internet applications (RIAs) as web applications that combine the functionality of desktop software applications with the broad reach and low-cost deployment of web applications resulting in more intuitive, responsive, effective user experiences ([adobe.com/products/central/whitepaper/central\\_wp.pdf](http://adobe.com/products/central/whitepaper/central_wp.pdf)). RIAs also reduced the number of page refreshes, required less bandwidth, and offered the unique ability to offload application processing to the client. At that time, RIAs targeted the Macromedia Flash Player within the browser, but Macromedia was also looking ahead to deploying RIAs across multiple devices, including real-time communications in RIAs, and providing distributed data storage.

Deploying to Flash Player in the browser, rather than the browser itself, eliminated the problems web application developers were experiencing while improving application design and usability and creating rich user experiences. Although developers were using Flash MX to build RIAs, its very limited code editor and the existence of a timeline made Flash MX feel awkward to developers and disrupted their standard workflows.

Released in 2004 by Macromedia, Flex 1.0 was a server-based product intended to provide enterprise application developers a way to deploy web applications to the nearly ubiquitous Flash Player while allowing them to use the traditional application development workflows they were accustomed to. Flex 2, released by Adobe after the acquisition of Macromedia, was no longer a server-based product. Flex® Builder™ 2, an integrated development environment (IDE) built upon Eclipse, allowed developers to write and compile an application locally and then deploy it to a web server. At this time ActionScript 3 was also released.

Adobe continued developing Flex, releasing Flex 3.0, Flex 4.0, Flex 4.5, and most recently Flex 4.6 on November 30, 2011. Over the years, new components have been added to the Flex framework. The component skinning architecture was redesigned. The IDE, renamed Flash® Builder®, improved developer

productivity with new code refactoring and formatting features, new debugging features, and the addition of both a profiler and Network Monitor. The 4.5 and 4.6 releases added support for mobile application development.

## Target use cases for Flex

Flex is used to build a wide variety of dynamic, data-driven applications that leverage the advantages of targeting the Flash Platform. Although Flex can and has been used to build smaller-scale applications, it has gained particular popularity in the development of larger-scaled and enterprise applications. At the enterprise level, Flex has been used to build hospital and medical information systems, reporting applications at Fortune 100 companies, wealth and asset management applications at banks, dashboard applications for education administrators, information distribution systems for the military, and applications delivering subscription and premium video.

Although many enterprise and data-centric applications provide a high-level of user interactivity and a rich user experience, the requirements of enterprise applications are frequently different from those of smaller-scale or consumer-based applications. Flex provides a unique set of benefits for enterprise developers, making it the appropriate tool of choice for building enterprise applications, such as:

- Optimal performance under high-stress conditions over long periods of time can be maintained.
- Other technologies such as JEE, PHP, and .NET can be integrated with a Flex application.
- The Flex framework allows for the internationalization and localization of the application for deployment in multiple languages and across continents.
- Testing throughout the development process can be done using the many tools designed specifically for Flex applications to perform unit, integration, functional, and other types of testing.
- Supporting tools provide build automation and continuous integration of Flex applications.
- The individual classes associated with a Flex project are easy to track and maintain in a revision control system.

## Flex and HTML

We at Adobe believe that when developing large, complex, rich applications, there are significant and sustainable advantages to using Flex over any other alternative technology. HTML innovation is moving at a very rapid pace and this will provide developers with more choices when selecting an appropriate technology for projects. Adobe is investing in helping advance HTML and believes it is an important technology for interactive development. But, we also recognize that Flex has advantages which may make it the best solution in the enterprise space for years to come. Some of the specific advantages of Flex for enterprise application development include:

- Flex offers complete feature-level consistency across multiple platforms, browsers, and devices.
- The Flex component set and programming model makes it extremely productive when building complex user interfaces.
- ActionScript 3 is a mature object-oriented programming language, suitable for large application development.
- Supporting tools offer a productive development environment with respect to code editing, debugging, profiling, and testing.

Recognizing these advantages, Adobe remains committed to enabling the success of all existing and new Flex projects.

## Adobe's vision for Flex

As Flex moves into a community-driven open source project, Adobe will make initial technical contributions and will also continue to provide a team of full-time Flex SDK engineers to contribute to the Apache project. While under this new model Adobe will provide fewer engineering resources than in the past, we are working with the Flex developer community to increase the total number of active contributors and resources.

## **Flex SDK contributions to Apache**

Adobe is currently in the process of contributing the core Flex SDK (including advanced data visualization components), automation libraries, AIR SDK binaries, documentation, and specifications to the Apache Flex Project. We will also contribute complete, but yet-to-be-released Spark components including ViewStack, Accordion, DateField, DataChooser, and an enhanced DataGrid.

## **Falcon compiler contribution to Apache**

Falcon 1.0 is the next generation compiler for ActionScript and is currently in development. Upon completion of the ActionScript portion of the compiler, Adobe will contribute Falcon 1.0 to the Apache Flex Project, which we expect will be in Q4 2012.

## **Falcon JS compiler contribution to Apache**

Falcon JS is an experimental ActionScript 3 compiler that targets JavaScript rather than the Flash Runtime. Adobe will contribute the prototype of Falcon JS source code to the Apache Flex Project after the completion and contribution of Falcon 1.0.

## **Testing contributions to Apache**

Adobe will contribute Mustella, a functional testing framework developed and used by Adobe to test the Flex SDK. Adobe is contributing it to the Apache Flex Project to facilitate the successful, continued, high-quality development of Flex.

## **BlazeDS contributions to Apache**

Adobe intends to contribute BlazeDS code to Apache to foster open development. We are currently working on the contribution details, and it is likely that we will contribute BlazeDS to the existing Apache Flex Project. The BlazeDS code is already available at [adobe.com/go/blazeds\\_source](http://adobe.com/go/blazeds_source) under the [GNU Lesser General Public License, Version 3](http://www.gnu.org/licenses/lgpl-3.0.html).

BlazeDS provides Java based server-side remoting and web messaging technology allowing Flex clients to exchange typed binary data with the server and receive notification from other clients and servers. The messaging service also allows Flex applications to exchange messages with other non Flex, JMS-enabled applications.

## **Engineering contributions to the Apache Flex Project**

Adobe will provide a team of Flex SDK engineers who will contribute to and support the Apache Flex Project full time, including work on new feature contributions. These engineers will also provide bug and security fixes for major issues in previously released Adobe versions of Flex.

## **Contributions of related technologies**

Adobe will not be contributing the AIR for Linux SDK, LCDS, or LCCS to Apache. Adobe is still investigating the viability of contributing TLF, BlazeDS.NET, Gravity, FXG, Squiggles, and OSMF.

## **Support contracts**

Adobe is fully committed to honoring support contracts for released versions of the Adobe Flex SDK and will continue to offer new support contracts for five years. Adobe does not have plans to provide support for Apache released versions of the Flex SDK. We anticipate that third parties will offer support for releases of Apache Flex.

## **Community support**

The Flex developer community is an active, vibrant community of highly skilled developers with a wealth of knowledge and experience. Moving Flex to an open source model means that the future of Flex is in the hands of this community. Adobe wants to help maintain this vital community and intends to work with the community throughout and beyond the transition to an Apache project.

Adobe will sponsor and provide funding for 360|Flex ([360flex.com](http://360flex.com)), the premiere conference dedicated to Flex. Adobe will also continue to support all of its user groups. No changes will be made, including application/reapplication, to the Adobe Community Professional (ACP) program until June 1, 2012. User groups are playing a significant role in the multi-city international tour being organized by Adobe (see [North American dates](#),

[European dates](#)). Adobe evangelists are scheduled to be at these meetings to facilitate open, face-to-face discussions with as many Flex developers as possible.

## Adobe runtime support of Flex

Flash Player 11.2 and Adobe AIR 3.2, which are anticipated to ship in the first quarter of 2012, will be tested with applications built using Adobe Flex 4.6. Adobe will test future releases of Flash Player and AIR against the Adobe Flex 4.6 SDK and maintain backwards compatibility for five years.

While Adobe will ensure that the Adobe Flex SDK 4.6 and prior will be supported in future versions of Flash Player and AIR, it will be the responsibility of the Apache Flex Project to test future versions of the Apache Flex SDK against released Adobe runtimes to ensure compatibility and proper functioning.

In the past, features were added to Flash Player and AIR specifically to support the needs of Flex applications. Going forward, features will be added to the runtimes to support Adobe's vision for the Flash Platform. The Apache Flex Project may choose to take advantage of those features; however, new features will not be added to the runtimes specifically to support the Apache project's efforts.

### RSLs for Apache-released versions of Flex

Now that Flex is a community-driven project, it is no longer appropriate nor practical for Adobe to sign the resulting Apache Flex RSLs. This means that when using an Apache Flex release, framework RSLs will not be cached globally by Flash Player, but rather per domain in the web browser.

## Adobe Flash Platform

Adobe is actively developing the Flash Runtimes and the next versions of Flash® Professional. Flash Platform technology will continue to evolve with a focus on gaming and premium video. The Flash Player Roadmap white paper with details about the future innovations in the Flash Platform will be published shortly. You will be able to read it at [adobe.com/go/flashplatform\\_whitepapers](http://adobe.com/go/flashplatform_whitepapers) after it is published.

### Adobe Flash Player for desktop browsers

Adobe is committed to continuing development of Adobe Flash Player for desktop browsers. The latest major release—Flash Player 11—introduced dozens of new features. Among the features planned for future versions of Adobe Flash Player on the desktop are concurrency, mouse lock support, telemetry support, audio API improvements, and new data types in ActionScript 3.

Accessibility support in Flash Player will continue to be maintained but no new accessibility features or enhancements are expected at this time.

### Adobe AIR for desktops

Adobe is also committed to continuing support for Adobe AIR applications running on the desktop. We are actively working on the next version of Adobe AIR for desktop. The recent major release—Adobe AIR 3.0—added support for captive runtimes allowing developers to package and distribute Adobe AIR applications without requiring users to first have the AIR runtime installed. It is anticipated that Adobe AIR 3.2 will be released shortly. Concurrency is targeted to be delivered in a subsequent release of Adobe AIR.

The AIR 3.x SDK is not currently available on Linux and there are no plans for it to be made available on Linux platforms in the future.

### Adobe Flash Player for mobile

Adobe will no longer develop Flash Player in the mobile browser to work with new mobile device configurations (chipset, browser, OS version, etc.) following the recent release of Flash Player 11.1 for Android and BlackBerry PlayBook. We will continue to provide critical bug fixes and security updates for existing device configurations.

### Adobe AIR for mobile

Adobe is committed to the support and development of Adobe AIR for mobile on current and future devices and OS updates including iOS 5, iPhone 5, iPad 3, and Android 4 (Ice Cream Sandwich). AIR will continue to be

the delivery mechanism for Flash content and applications deployed as stand-alone applications. RIM plans to continue to support and develop AIR for the BlackBerry PlayBook OS.

## Adobe Flash Player and AIR for Windows 8

For information on support in future operating systems, please refer to the Flash Player Roadmap white paper which will be published shortly. You will be able to read it at [adobe.com/go/flashplatform\\_whitepapers](http://adobe.com/go/flashplatform_whitepapers) after it is published.

## Adobe tooling support for Apache Flex

### Flash Builder

Development of Flash Builder continues. Adobe plans to maintain support for Flex projects in updates to Flash Builder 4.x, including additional work to ensure Apache Flex based SDKs can work within Flash Builder. Adobe will evaluate integrating an Apache-derived Falcon MXML compiler into Flash Builder if and when a suitable compiler is available from the open-source project. In order to better support future Apache-derived Flex SDKs, Design View, Data Centric Development tools, and Flash Catalyst workflows will be removed in updated 4.x versions of Flash Builder.

### Flash Catalyst

Flash Catalyst CS5.5 is the last release of Flash® Catalyst®. At this time, it will not be updated to work with Flex SDK 4.6 or any Apache Flex releases nor will it be included in future versions of the Adobe Creative Suite software offerings.

### Flash Professional

Adobe is not changing its commitment to or the release plans of Flash Professional. The next two versions, code-named "Reuben" and "Hellcat", are currently on track. Reuben is planned to ship in 2012. There are no plans to support the Apache Flex SDK in Flash Professional.

## Summary

Adobe believes that Flex is the best solution for enterprise and data-centric application development today and is moving Flex into a community-driven open source project to ensure the continued development and success of Flex for years to come. We are currently in the process of contributing the core Flex SDK, automation libraries, AIR SDK binaries, and documentation to the Apache Flex Project. We will also be contributing Falcon, Falcon JS, Mustella, and BlazeDS.

In addition to these contributions, Adobe is providing a team of full-time Flex SDK engineers who will contribute to and support the Apache Flex project. These Adobe engineers will work directly with the highly skilled Flex developer community to maintain, support, and evolve the Flex SDK. We remain committed to enabling the success of all existing and new Flex projects.

## For more information

Flash Player Roadmap white paper: [www.adobe.com/go/flashplatform\\_whitepapers](http://www.adobe.com/go/flashplatform_whitepapers)

BlazeDS source code: [www.adobe.com/go/blazeds\\_source](http://www.adobe.com/go/blazeds_source)

Flex User Group 2012 Tour – North America: [www.adobe.com/go/flex\\_ug\\_2012\\_tour\\_na](http://www.adobe.com/go/flex_ug_2012_tour_na)

Flex User Group 2012 Tour – Europe: [www.adobe.com/go/flex\\_ug\\_2012\\_tour\\_eu](http://www.adobe.com/go/flex_ug_2012_tour_eu)



Adobe