

Part I: Getting Started

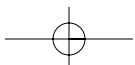
Chapter 1: Introducing Flex 2.0

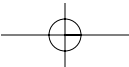
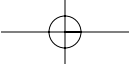
Chapter 2: Introducing Flex Builder 2.0

Chapter 3: Flex 2.0 Basics

Chapter 4: Using Flex Builder 2.0

COPYRIGHTED MATERIAL





1

Introducing Flex 2.0

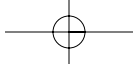
Flex 2.0 is arguably the most important new Internet development tool set released to date. It will help speed the migration of the Internet to the Web 2.0 that we have been promised for so long now. Not only has Adobe introduced an extremely easy-to-use development environment in Flex Builder 2.0, but it has also released an SDK allowing for Flex Rich Internet Applications to be developed with absolutely no fees. So what are Rich Internet Applications? Let's start with a definition.

Rich Internet Applications

A *Rich Internet Application* (RIA) is an application that runs in the traditional browser but utilizes an intermediate layer that can bypass the traditional page refresh that has been a standard of most current Web applications. The most common tools that can achieve this intermediate layer include JavaScript used in Asynchronous JavaScript and XML (Ajax) applications, as well as Flex or Flash using the Flash Player. Other RIA solutions include OpenLaszlo (which utilizes the Flash Player as well as Ajax), XUL (which is dependent on a Mozilla-compatible browser), and the Windows Presentation Foundation (which is part of the Microsoft .NET Framework 3.0).

OpenLaszlo

Like Flex, OpenLaszlo can compile to SWF and, because it is open source, it is also free for developers to use. OpenLaszlo applications are built using JavaScript, as well as an Extensible Markup Language (XML) based programming language named LZX, and compiled to SWF. It is very similar to Flex, and because it was available for free, it garnered a larger amount of attention during the days of Flex 1.5. Because OpenLaszlo compiles to SWF, it enjoys the same write-once-run-anywhere feature that is synonymous with Flex and Flash applications. The one advantage to OpenLaszlo over Flex is that the next version of OpenLaszlo (code-named *legals*) will allow write-once-and-compile to either an SWF or a dynamic HTML (DHTML), or Ajax, application. The major disadvantages of OpenLaszlo are that it has a smaller set of built-in components than Flex, and it has always been at least one full version behind in optimization for the most current Flash Player.



Part I: Getting Started

Ajax

Ajax is a combination of HTML or Extensible HTML (XHTML), Cascading Style Sheets (CSS), and JavaScript to create an application-type feel from a standard Web page request. CSS and JavaScript can be used to trigger updates to the visual aspect of the page, and XMLHttpRequests can be used to fetch data in the background of the page. The combination of these techniques allows a single Web page to change its appearance and update its data without any additional page requests from the server. There are many limitations to this type of RIA, including compatibility issues with browser versions, as well as differences in (or lack of) JavaScript support necessary for the RIA to perform as expected.

XUL

XML User Interface Language (XUL) is an XML-based language developed by the Mozilla project. It is a tag-based language that contains many predefined widgets (buttons, radio buttons, and so on) that will be rendered only by a Mozilla-compatible browser such as Firefox. Although it can also be used in the creation of RIAs, XUL has been most widely used in the creation of Mozilla applications and extensions.

Windows Presentation Foundation

Windows Presentation Foundation (WPF) consists of an engine and a framework that will be preinstalled in Windows Vista, and introduces Extensible Application Markup Language (XAML) as a new tag-based language. XAML is very similar to MXML because it is a tag-based XML language with attributes whereby each tag creates an object model class. XAML tags are fully extendable, allowing developers to create custom classes. XAML, along with C# (which is the programming language), would correspond to MXML and ActionScript 3 in Adobe Flex.

WPF is used for both traditional desktop applications and browser-based RIAs. With Microsoft's backing, WPF and XAML are sure to make a large contribution to Web 2.0 and RIA. Microsoft has stated that it will also introduce Windows Presentation Server Everywhere (WPF/E), which will provide support for other browsers and platforms. One major advantage of WPF is that it will support three dimensions out of the box, which is something that has been lacking in the Flash Player. A disadvantage of WPF is that it does not run within the Flash Player, and it will be a long time before Microsoft can expect the same penetration of its runtime as the Flash Player.

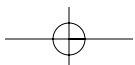
History of RIA

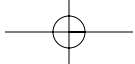
Although the concept of a Web application that performs more in line with the traditional desktop application has been around for years, RIAs were first introduced by Macromedia in March 2002.

Benefits of RIA

RIAs offer many benefits over traditional Web applications. Following are a few of the many advantages of RIAs:

- RIAs offer a richer interface that provides a more engaging user experience without the need for page reloads.
- RIAs offer real-time feedback and validation to the user, triggered by user events.





Chapter 1: Introducing Flex 2.0

- ❑ The look and feel of a traditional desktop application can be accomplished with an RIA.
- ❑ RIAs also can include a full multimedia experience, including audio and video.
- ❑ RIAs have capabilities such as real-time chat and collaboration that are either very difficult or simply impossible with traditional Web applications.

Overview of Flex

Flex was developed to create a development environment that more closely resembled the traditional development environment utilized by other programming languages. The goal was to take the already tremendously successful Flash visual development environment and open it up to programmers who are more comfortable with a code-based model.

In 2004, Macromedia released Flex 1.0, which caused a buzz in the world of RIAs because it was the first time that it was possible to compile Flash SWF files from code. In 2005, Flex 1.5 was released, which was a big improvement over Flex 1.0 and included many new features. A few of the notable new features included new charting components, updates to the datagrid component, and new component-skinning techniques for more control over the look and feel of Flex 1.5 applications.

Flex 1.5 was very successful, but was still targeted to the mid- to large-size company, and it was priced that way. For smaller companies and developers, Flex 1.5 was simply out of reach because of Macromedia's pricing model. In 2006, Adobe released Flex 2.0 and announced that it was now possible to create and distribute Flex 2.0 applications absolutely free. This has created a large influx of developers into the Flex community, and shows tremendous growth potential.

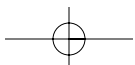
Flex 2

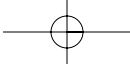
Flex 2 is a tremendous upgrade over its predecessor. It includes a new version of the MXML tag-based language, as well as a new version of ActionScript. Adobe has built the Flex 2 Framework to be fully extendable, allowing developers to create custom components that extend or combine base components using either MXML or ActionScript 3.

Flex 2 is built to run within the all-new Flash 9 Player, which in itself includes huge performance increases over previous versions. Flex 2 includes a new development environment built on the Open Source Eclipse integrated development environment (IDE) named Flex Builder 2, the free Flex 2 SDK, as well as the new Flex Data Services (FDS) application server. Flex 2 also offers a new set of charting components that are not part of the basic free versions of the Flex SDK but can be purchased separately or as part of a Flex Builder package.

Flash Player 9

In May 1996, a small company named FutureWave Software released a product that it called FutureSplash Animator. FutureSplash was released as a browser extension using the new plug-in API from Netscape. Ironically, FutureWave attempted to sell its technology to Adobe, but at the time, Adobe wasn't interested. Macromedia was interested in FutureSplash, and, in December 1996, acquired FutureWave Software. Macromedia renamed FutureSplash as Flash Player 1.0.





Part I: Getting Started

Following is a timeline of key events in this history of the Flash player:

- ❑ 1997 — FutureSplash is renamed and released as Macromedia Flash 1.0.
- ❑ 1997 — Flash 2 adds buttons, the Library, sound support, and color tweens.
- ❑ 1998 — Flash 3 adds new animations and alpha transparency.
- ❑ 1999 — Flash 4 adds MP3 audion streaming and motion tweens.
- ❑ 1999 — 100-millionth download of the Flash Player.
- ❑ 2000 — Flash 5 adds ActionScript 1.
- ❑ 2000 — Flash Player installs reach the 92 percent penetration mark.
- ❑ 2002 — Flash Player 6 adds support for Flash remoting, Web services, video, shared libraries, and components.
- ❑ 2003 — Flash Player 7 introduces ActionScript 2.0 and streaming audio and video.
- ❑ 2005 — Flash Player 8 adds filter effects, GIF and PNG images, bitmap caching, new video codec (On2 VP6), file upload and downloading, and FlashType.
- ❑ 2006 — Flash Player 9 is released and introduces ActionScript 3, E4X XML parsing, regular expressions, and binary sockets.

The introduction of Flash Player 9 in 2006 represents the most significant release of any Flash Player to date. The incorporation of ActionScript 3 brings true object-oriented programming to Flex 2 and the upcoming Flash 9 authoring tool. Flash Player 9 boasts performance increases of up to ten times over Flash Player 8. Flash Player 9 is based on the ECMAScript standard and adds new features, including ECMAScript for XML (E4X), regular expressions, a standardized event model, and binary sockets.

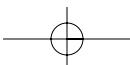
Flash Player 9 also includes better memory utilization, improved application initiation speed, as well as improvements in the debugger and error reporting.

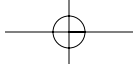
Flex Builder 2

Flex Builder 2 is the new development tool that (although it is not necessary for Flex development because any text editor can be used) offers the most complete development environment for rapidly creating Flex applications. Because Flex Builder 2 has been built on top of the mature Eclipse IDE, it will be very familiar to developers who have already been developing software in other languages using Eclipse as their tool of choice. Eclipse is a development environment that was built with extendability in mind, and this can also work to the benefit of Flex developers because it can be customized to suit the needs of the Flex developer. Chapter 2 provides a more comprehensive look into Flex Builder 2.

Flex Free SDK 2

The Flex Software Development Kit (SDK) is a free download from Adobe and includes the Flex framework (component class library), compiler, and debugger. Using the text editor of your choice, you can create the ActionScript and MXML files for your application, and then compile to SWF using the SDK. The SDK allows for data communication via Web services, HTTP services, and Flash remoting with ColdFusion as the back-end server.





Chapter 1: Introducing Flex 2.0

Flex Data Services (FDS)

FDS is the server version of Flex that must be installed under any Java 2 Enterprise Edition (J2EE) server. FDS is essentially the next generation of Flex 1.5, which was sold only as a server.

As well as a full commercial edition, the FDS server is available in an Express edition, which is free and available for commercial use (but is limited to one CPU without clustering).

FDS is made up of several components and capabilities, including the following:

- Flex Messaging Services (FMS)
- Publish-subscribe messaging
- Data push
- RPC services

Flex Messaging Services

FMS is one of the pieces that make up FDS, and it allows for the creation of applications that support real-time messaging, as well as collaboration. The messaging service has support for Java Message Service (JMS), as well as other existing messaging services, allowing for the creation of cross-platform chat applications.

Publish-Subscribe Messaging

FMS uses the producer/consumer publish-subscribe metaphor, which allows for the creation of co-browsing applications. To understand what is meant by a *co-browser application*, imagine a company's customer service representative being able to make edits to a form on the user's screen in real time while the user watches.

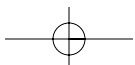
Data Push

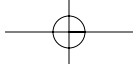
Data push is the capability for the server to push data changes to the client without any user interaction or polling of the servers. This can be critical when an application has hundreds or thousands of users connected, and they can all see changes to business-critical data in real time. This is a feature that is available only when using the FDS server.

RPC Services

Remote procedure call (RPC) services are the suite of services used to connect to external data. They include the following:

- WebService** — The `WebService` component can be used to access any Web service that complies with the WSDL 1.1 standard and returns data formatted as Simple Object Access Protocol (SOAP) messages over HTTP.
- HTTPService** — The `HTTPService` component can send HTTP GET, POST, HEAD, OPTIONS, PUT, TRACE, or DELETE requests. It does not support multipart requests.
- RemoteObjects** — The `RemoteObject` component uses Action Message Format (AMF) to transfer data that is a binary format and is the fastest of the RPC services. It is an ideal way to interact with server-side Java objects that are within the FDS server's source path. It can also be





Part I: Getting Started

used to connect to local or even remote ColdFusion servers. Both Java objects and ColdFusion components can be mapped to ActionScript objects, allowing for seamless integration between the server-side and client-side objects.

The `WebService` and `HTTPService` components are included for free with the Flex SDK, whereas the `RemoteObject` component is available only for use with the FDS server (Commercial or Free Express Edition) or ColdFusion 7.02.

Flex Charting

The Flex charting components are a set of rich charting components that enable easy development of professional dashboard and business intelligence (BI) systems. The charting components are sold as a stand-alone product or bundled with Flex Builder 2. A free trial of the charting components is available from Adobe. You can learn more about the charting components in Chapter 10.

Summary

This chapter defined RIA and gave some examples of the benefits of RIA, as well as the software options available for creating RIAs. It also provided an overview of Flex and its history, as well as the pieces that make up Flex. Most of the information on Flex covered in this chapter is covered in more depth later in this book.

Chapter 2 introduces the all-new Flex Builder 2 built on top of Eclipse.

