

## A 50 Minutes Walkthrough for Adobe Flash/Flex

Jingtao Wang, [jingtaow@cs.berkeley.edu](mailto:jingtaow@cs.berkeley.edu)

### Disclaimer

*This document is written by Jingtao Wang for the CS160 class at UC Berkeley in fall 2008. This document focuses on several products from Adobe because most students indicated their personal preferences on such technologies during an in-class survey conducted in September 2008. This document and the corresponding discussion session should not be interpreted as an endorsement to related Adobe products by the class instructor or the TA. Neither the instructor nor the TA has any direct or indirect financial relationships with Adobe. The instructor and the TA encourage students in CS160 to choose whatever environments that most group members are comfortable with for their group projects.*

### What's the difference between Adobe Flash and Adobe Flex?

Due to the popularity of YouTube and embedded advertisement banners, you have no doubt seen Adobe Flash animations on the Internet, and have some idea of what they can do. You may have also heard the buzzword RIA (Rich Internet Application) in recent years. Flash and Flex are tools for creating web based animations and RIAs; competing tools/environments include AJAX/DHTML, SilverLight, JavaFx and Java Applet.

There are two products from Adobe that can generate Flash animations/RIAs: Adobe Flash (the most recent version is CS 3, a.k.a. version 9.x), and Adobe Flex (the most recent version is 3.x). Both products can generate .swf files that can be embedded in web pages and played by Adobe Flash Player plugins in a web browser. Adobe Flash is a **story-board** style environment that is more designer friendly – a major portion of functions can be completed by designers via drag and drops over an editing stage and a timeline. Programmers can also add ActionScript code in a code editing window when necessary. Adobe Flex Builder is an **IDE** style environment that is more programmer/code friendly. Adobe Flex Builder doesn't have the interactive story board function provided by Adobe Flash, but it has a better code editor/debugger based on Eclipse, and a better UI library for building RIA applications. Adobe Flash and Adobe Flex can complement each other – the ActionScript code can be shared in both Flash and Flex projects if environment specific libraries are not used, modules created in Flash can be imported/accessed in Flex and vice versa.

### Why Flash/Flex?

In the context of serious games, you get the following advantages if you choose Flash/Flex (when compared with 'vanilla' programming environments such as C/C++, Java).

1. Built-in support for bitmap images, vector graphics, audio and video.
2. Built-in support for commonly used game functions such interactive sprite animation/clipping, tile scrolling and hit detection.

3. Easy to deploy and cross platform – the Flash player supports all major OSes (Windows, Mac OS, Linux, FreeBSD and Solaris) and browsers (IE, Firefox, Opera, Safari and Chrome). If you create a Flash Player 7.0 compatible game, you can even play it on Nintendo Wii, SONY PSP and many cell phones.

Some interesting websites that use and demonstrate capabilities of Adobe Flash/Flex

- Youtube.com
- The StreetView function in Google Maps
- The stock price visualizer in Yahoo Finance and Google Finance
- SlideRocket.com – a Flex-based Presentation creation/sharing RIA
- Buzzword.com – a Flex-based word processing RIA
- Blist.com – a Flex-based spreadsheet RIA
- Photoshop Express – <http://www.photoshop.com/express>
- Last.fm <http://Last.fm>
- Pandora Radio <http://www.pandora.com>

### Notes for Adobe Flash CS3

1. Source files include .fla and .as. You can embed ActionScript code snippet directly in .fla files, but the TA strongly encourages you to put any ActionScript code snippet longer than 15 lines into a separate .as file and then import the .as file from the .fla file. The target file is in .swf format.
2. There are two different flavors of ActionScript in use nowadays – 2.0 and 3.0. ActionScript 2.0 is accompanied with Flash 7.0. The syntax of ActionScript 2.0 is similar to JavaScript and is not fully object oriented. If you want to create an application that can run on SONY PSP, Nintendo Wii or some cell phones, you need to use ActionScript 2.0 at this time. ActionScript 3.0 is accompanied with Flash 9.0 (CS3) and Adobe Flex and it's a fully object oriented language. ActionScript 3.0 is significantly easier to learn and use than ActionScript 2.0; The performance of ActionScript 3.0 runtime is claimed to be 2-5 times faster than ActionScript 2.0, so ActionScript 3.0 is always a better choice if you do not have to run your application on game consoles and cell phones.
3. Key frames are stages/story-boards that users can manipulate or add interactive elements. Normal frames in Flash are automatically managed by the Flash environment. Key frames can be used to host different states of an application/game.

4. Storyboard-generated animation in Flash is layer based, not element/component based. If a user needs to move two on-screen elements in different directions at the same time, put them in different layers. Layering is a powerful method to organize on-screen elements, if you are using Flash to create your games, it's not uncommon to have more than 15 user created layers. You are very likely to use the "mask layer" feature if you are creating a game.
5. Flyout menus on the Tool panel can be accessed by press and hold the left mouse button.
6. Both JSON and XML can be used to exchange data with a server (this technique also applies to Adobe Flex).
7. If you are using ActionScript 2.0, most likely your in game sprite will be a sub-class of the MovieClip class. If you are using ActionScript 3.0, most likely your in game sprite will be a sub-class of Sprite. The class library for 3.0 is light years better than the class library for 2.0
8. If your application/game is big, please consider using ARP as your application framework.

### **Notes for Adobe Flex Builder 3**

1. You can either install the standalone Adobe Flex Builder (a rebranded Eclipse IDE) or install the Flex Builder plugin to your existing Eclipse IDE. The TA prefers the second approach.
2. There are two different types of source files in a Flex project - .mxml files and .as file. .mxml files are xml files used to define the whole (or part of) the user interface. .as files are ActionScript files. .mxml files will be compiled to .as files when the project is built.
3. Adobe Flex only supports ActionScript 3.0.
4. There is no key frame in Flex (actually there are only two frames in any .swf generated by Flex). A user can either use mx:states or mx:ViewStack to manage different modes in an application/game (mx is the name space for official libraries included in Flex).
5. You are strongly encouraged to use FlexUnit to create your test cases if your application is non-trivial.
6. If your application is big, please consider using Cairgorm as your application framework.
7. You may want to use FxSpy if you are familiar with using tools like Windows Spy or Firebugs in debugging.
8. If you want to add a sub-class of Sprite as a child of Flex related UIComponents, please try `myuicomponent.rawChildren.addChild()` rather than `myuicomponent.addChild()`

### **Recommended Readings**

#### **Free Tutorials**

1. Using Flash [http://livedocs.adobe.com/flash/9.0/UsingFlash/flash\\_cs3\\_help.pdf](http://livedocs.adobe.com/flash/9.0/UsingFlash/flash_cs3_help.pdf)
2. Programming ActionScript 3.0  
[http://livedocs.adobe.com/flash/9.0/main/flash\\_as3\\_programming.pdf](http://livedocs.adobe.com/flash/9.0/main/flash_as3_programming.pdf)
3. Flex 3 Developer Guide [http://livedocs.adobe.com/flex/3/devguide\\_flex3.pdf](http://livedocs.adobe.com/flex/3/devguide_flex3.pdf)
4. Building and Deploying Adobe Flex 3 Applications  
[http://livedocs.adobe.com/flex/3/build\\_deploy\\_flex3.pdf](http://livedocs.adobe.com/flex/3/build_deploy_flex3.pdf)

### Non-Free Books

1. Colin Moock, Essential ActionScript 3.0, O'Reilly, ISBN 0596526946, 2007 (Free web access link from Berkeley campus <http://uclibs.org/PID/112821> )
2. Joey Lott, et al, ActionScript 3.0 Cookbook, O'Reilly, 2006 (Free web access link <http://uclibs.org/PID/110214> )
3. Alaric Cole, Learning Flex 3 : Getting up to Speed with Rich Internet Applications, Adobe Dev Library, ISBN 0596517327, June 2008.
4. Gary Rosenzweig, ActionScript 3.0 Game Programming University, Que, ISBN: 0789737027,2007

### Important Web Resources

1. Free Adobe Flex Builder 3 for Education <https://freeriatools.adobe.com/flex/>
2. Tweeners : A third party tweening library and examples <http://code.google.com/p/tweener/>
3. TweenLite – Yet another tweening library <http://blog.greensock.com/>
4. A huge set of “how-to” style examples with code snippets <http://flexexamples.com/>
5. Some Flash examples, including games <http://www.adobe.com/devnet/flash/samples/>
6. Goto and Learn (tips and examples for learning Flash) <http://www.gotoandlearn.com/>
7. Flash, Flex, and ActionScript related resources website <http://www.actionscript.org>
8. ARP, an open source framework for creating RIA applications in Flash/Flex  
<http://osflash.org/projects/arp>
9. Cairngorm, the adobe official open source framework for creating RIA application in Flex  
<http://labs.adobe.com/wiki/index.php/Cairngorm>
10. A large library of open source flash games  
<http://www.flashadvisor.com/movie/index.php?viewCat=24>
11. Benchmark websites for DHTML, Flash/Flex, Silverlight etc related RIA websites  
<http://www.bubblemark.com/> and <http://www.craftymind.com/guimark/>
12. FxSpy – An Open source runtime property checker/modifier for Flex  
<http://code.google.com/p/fxspy/>