# PROJECT: CREATING YOUR FIRST APP USING FLASH CS5

In the first section of the book we looked at setting up your Flash environment to work with Android. You also looked at design considerations you should bear in mind when developing Android apps. The goal of this project is to bring these two things together.

During this project you will apply the following:

- Set up a default AIR for Android file
- Develop background image details for the Android app
- Work with embedded text
- Create icons for your project
- Test your application on your Android device

The goal of this section is to validate how easily you can build your Android applications. There should be no heartaches when it comes to Android development and I think you are going to be very pleased with how fast you pick it up.

## Setting up Your Development Environment

Throughout this book you will go through the steps needed to create a new Android application. Following this project, I am going to make an assumption that you know enough about the default setup, and will not need me to run through this process each time. Phew, you won't need to keep hearing me say, "Download the Android SDK…." We can just focus on the fun stuff.

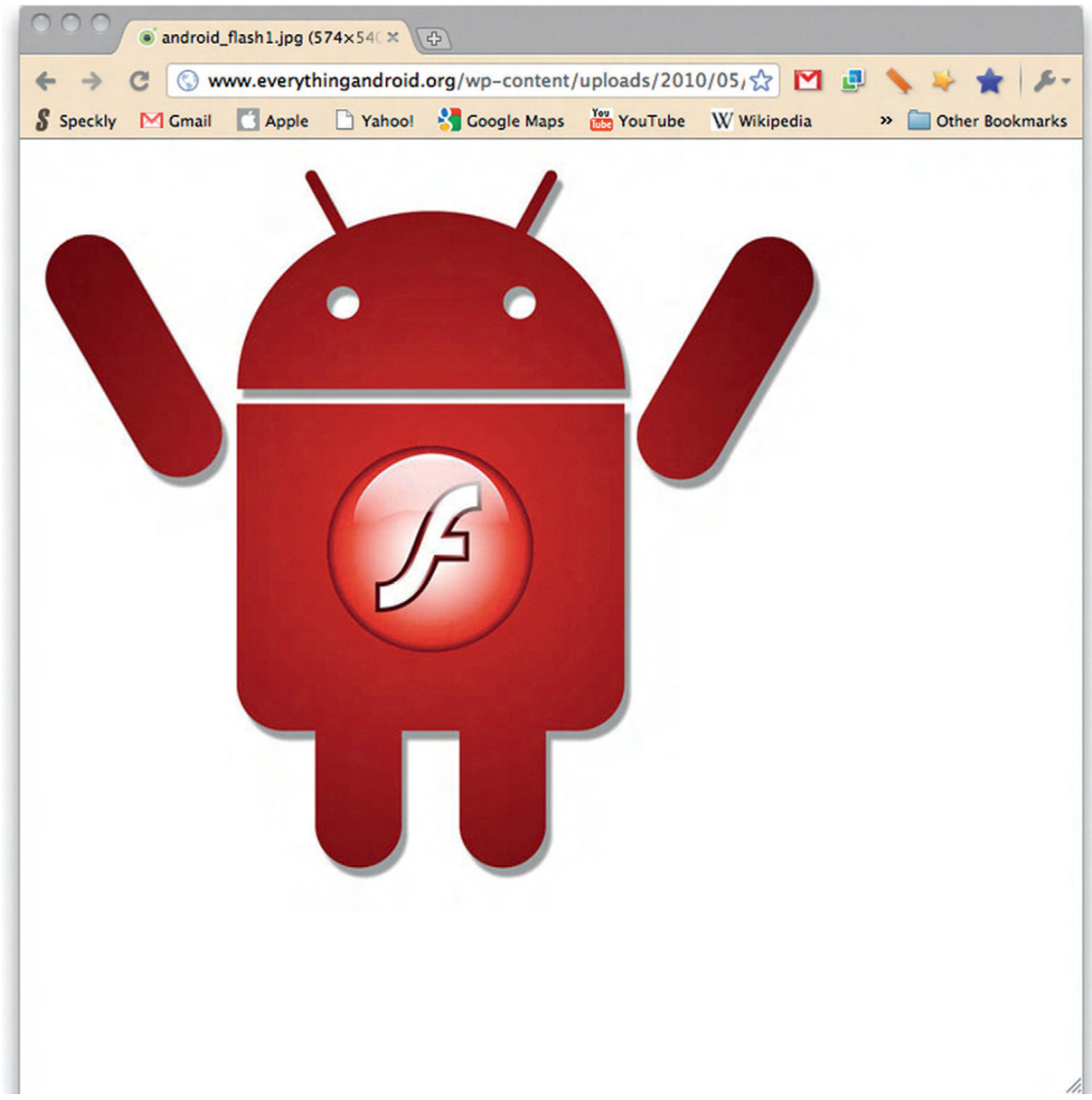
For now, let's step through the whole process.

Before you get started you will need a physical Android device to test with. This is essential for your development in this book. You can either buy an unlocked phone that is not connected to a carrier or drop the pennies to buy your own Android phone from any of the many mobile carriers. Remember, your phone must be

running Android 2.2. There are a lot of cheap Android phones on the market that are running Android 1.6. AIR and Flash are supported only on Android 2.2 and later (Figure 1.1Proj).

Your development environment also needs to be either Windows or Macintosh OS X 10.5+.

**Figure 1.1Proj** Android 2.2, codename Froyo, is Flash friendly.



With your Android 2.2 device in hand, let's set up your development environment.

1. Start by going to the Android development site at *http://developer.android.com/index.html*, as shown in Figure 1.2Proj.
2. Select the SDK tab along the top of the page.

**Figure 1.2Proj** All the Android code you need is at developer.android.com.

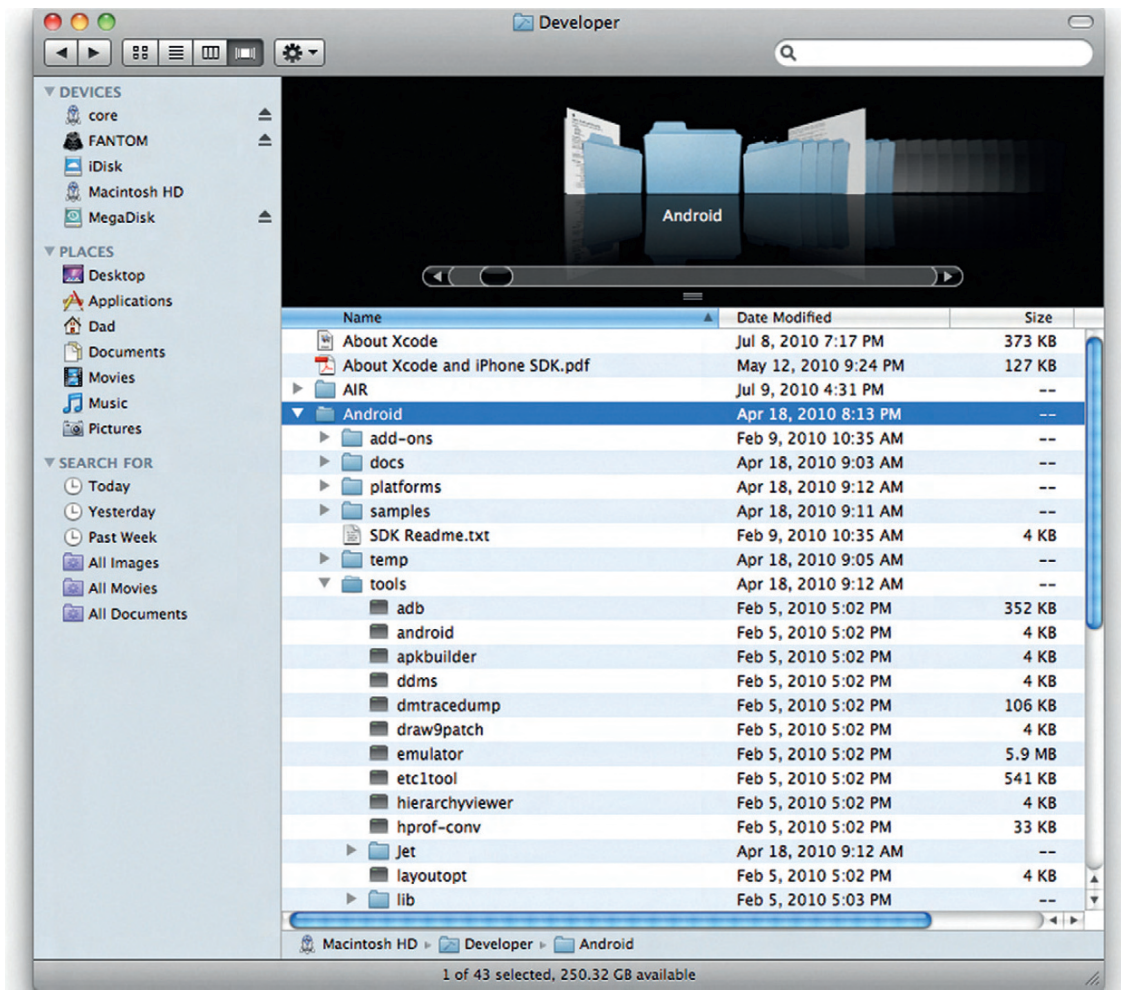
3. You will need to download either Windows or Mac OS X (Intel) versions of the SDK. At this time, there is not a Linux version of Flash that allows you to develop AIR for Android apps. (Figure 1.3Proj).
4. The Android SDK will download in a ZIP file. Save this to your computer.

**Figure 1.3Proj** The latest Android SDKs can be downloaded for Windows, Mac, and Linux.

5. Create a new folder in the root of your main hard drive and name the file Developer.
6. Open and extract the files in the Android SDK to the Developer folder you just created (Figure 1.4Proj).

At this point you have all the files you need from Google. Let's direct our attention to Flash CS5. To get started with AIR for

**Figure 1.4Proj** Extract the files for the Android SDK to your local hard drive.



Android you will need the latest version of Flash CS5. When you have Flash CS5 installed, jump over to *http://labs.adobe.com/technologies/air2/android/* to download and install the Android MXI extensions for Flash CS5 (Figure 1.5Proj).

**Figure 1.5Proj** Adobe's AIR for Android files can be downloaded from its labs site.

Close Flash CS5 before starting your first AIR for an Android project. Follow these steps to get up and running:

1. Open Flash CS5.
2. From Create from Template, open the splash screen and select AIR for Android (Figure 1.6Proj).
3. Choose the default AIR for Android template from the New from Template window (Figure 1.7Proj).
4. Select the OK button. Your default, blank Android file is ready for you. At this point you now work on the fun bit of creating your Flash content (Figure 1.8Proj).
5. Save your file as AndroidWelcomeMessage.xfl.

**Figure 1.6Proj** The AIR for Android template has all the settings you need for your Android app.

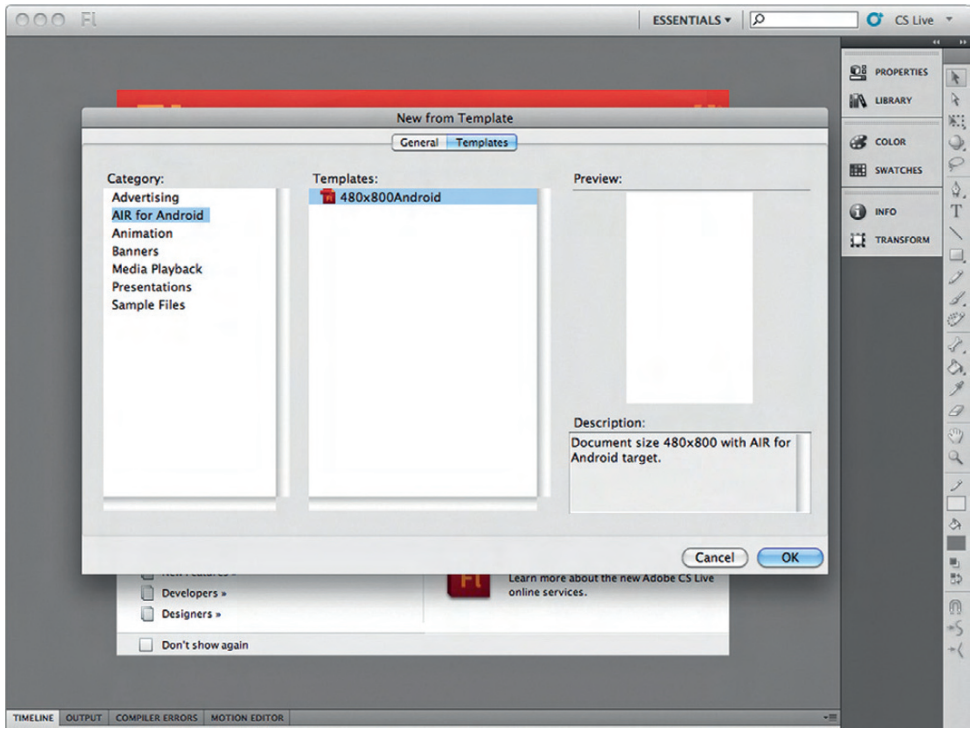



**Figure 1.7Proj** Android templates you can choose from.

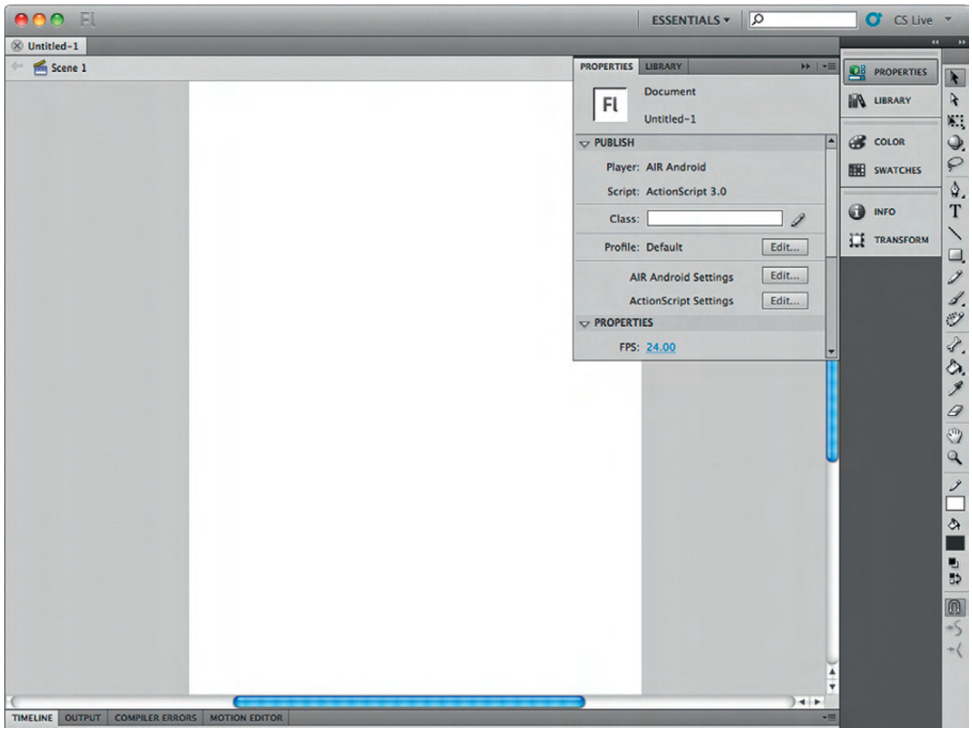



**Figure 1.8Proj** All you need to get started working with Android apps in Flash.

## Creating the Graphics

Let's get started on the graphics you need for your Flash movie. As mentioned in the previous chapter, the best format for creating graphics in Android apps is the PNG bitmap format. Luckily for you, one of the best PNG image editors is Adobe's Fireworks. Fireworks is packaged with the CS5 Web Suite of tools. This means you already have all the tools you need installed on your computer.

If you do not have Fireworks installed you can download a 30-day evaluation copy from Adobe.com. The project you are going to build is very simple: You will create an icon of the Android logo that you can select.

Go to the accompanying website for this book, www.visualizetheweb.com/flashmobile, to download the files used in this book. Project 1 will have a graphics file labeled google_android.png. You can open this image with Fireworks.

The image is fine as is (Figure 1.9Proj). You will, however, need three icons for your final app. Let's go ahead and create those now.

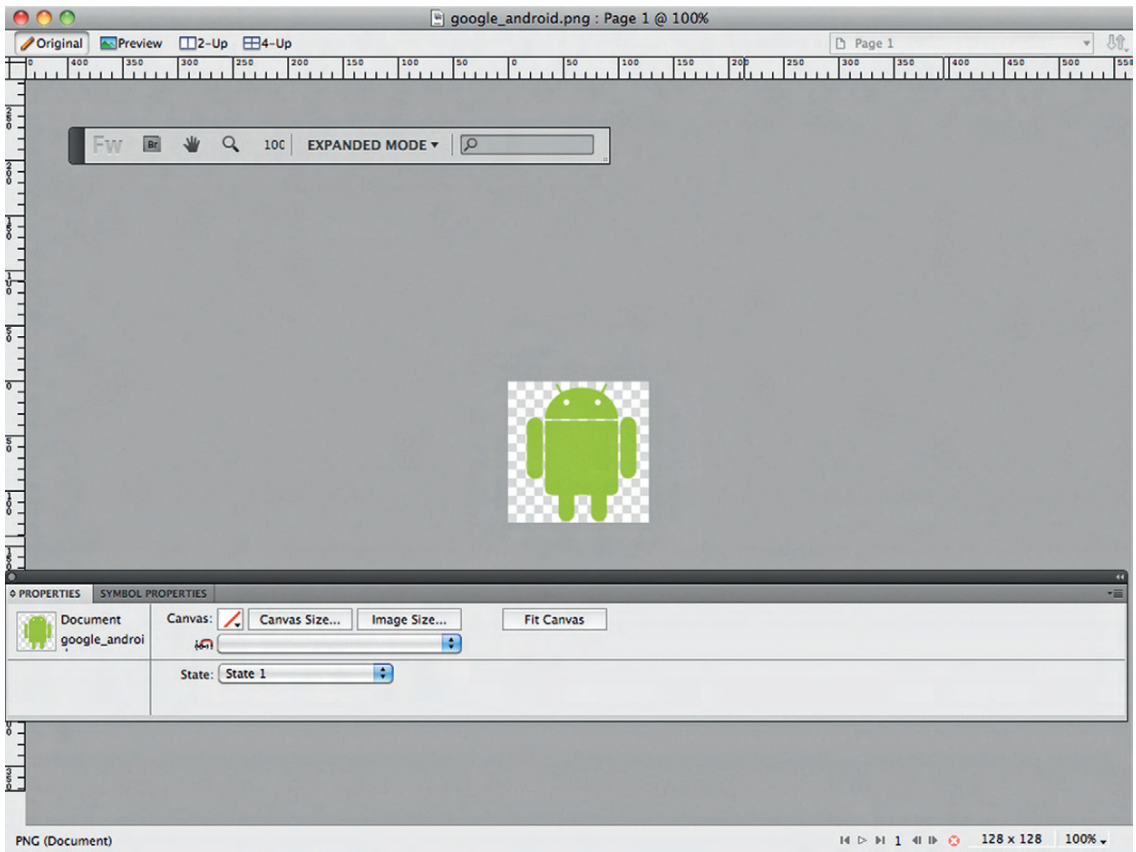



**Figure 1.9Proj** A PNG graphic of the Android logo in Adobe's Fireworks.

1. The three icons you need to create are 72 × 72, 48 × 48, and 36 × 36 pixels.
2. Select the Android logo. Open the properties panel. Change the X and Y properties to 0.
3. Change the size of the logo to width 52 and height 72 pixels.
4. Select Modify → Canvas Size. The Canvas Size screen opens. Change the width and height to 72 × 72 pixels (Figure 1.10Proj).
5. Save your file as Android_logo_72.png.
6. Repeat this process for 48 × 48 and 36 × 36 pixel icons, and name the files Android_logo_48.png and Android_logo_36.png, respectively. There is no magic to how the files are numbered. This is just an easier way to remember what each file does.
   At this point you have all of the graphics you need for your first application.

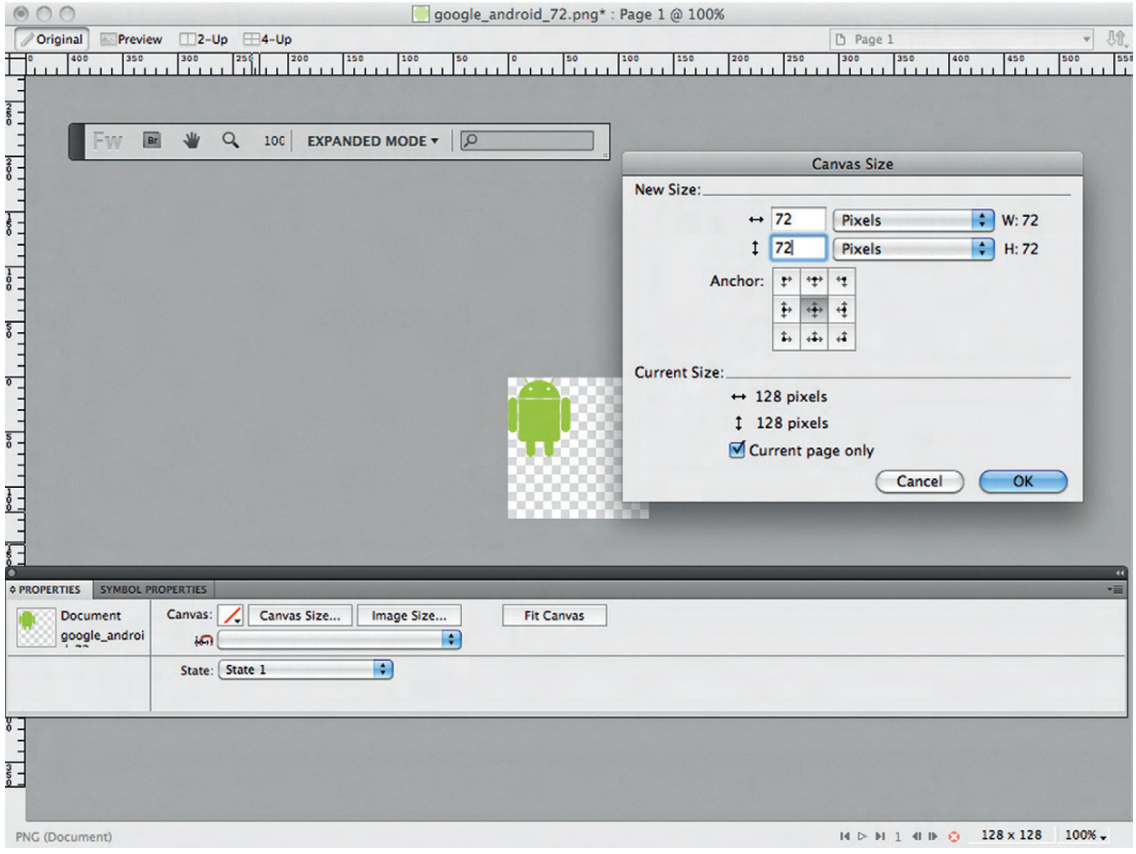


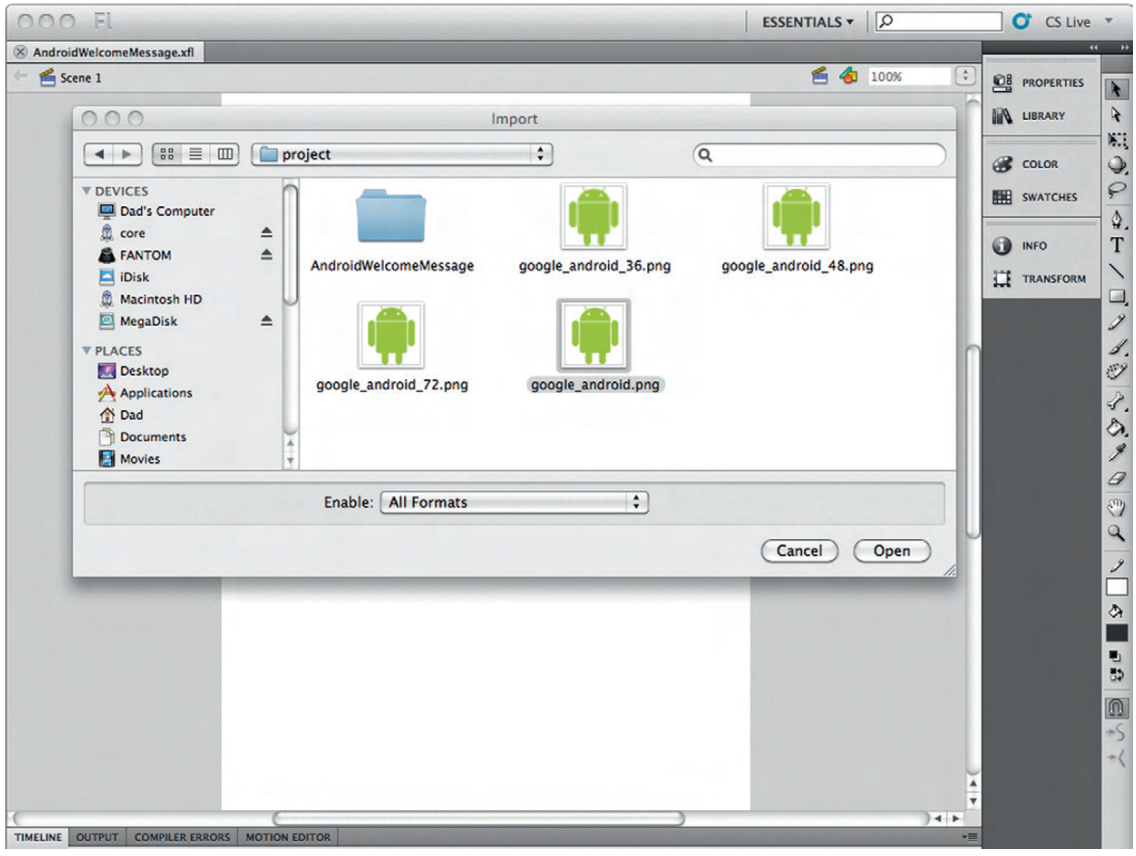

**Figure 1.10Proj** Use Fireworks to create the image icons for the final application.

## Building an Application

Now, let's get down to the fun part: building applications.

1. Open Flash CS5, if you do not still have it open, and open the AndroidWelcomeMessage.xfl Flash movie.
2. Select File → Import → Important to Stage…
3. The Import window will open. Navigate to the folder containing your images. Select google_android.png, as shown in Figure 1.11Proj.
4. The Import Fireworks Document window opens. Select Import as a single flattened bitmap. Choose OK (Figure 1.12Proj).

**Figure 1.11Proj** Import the PNG file you need for the application.



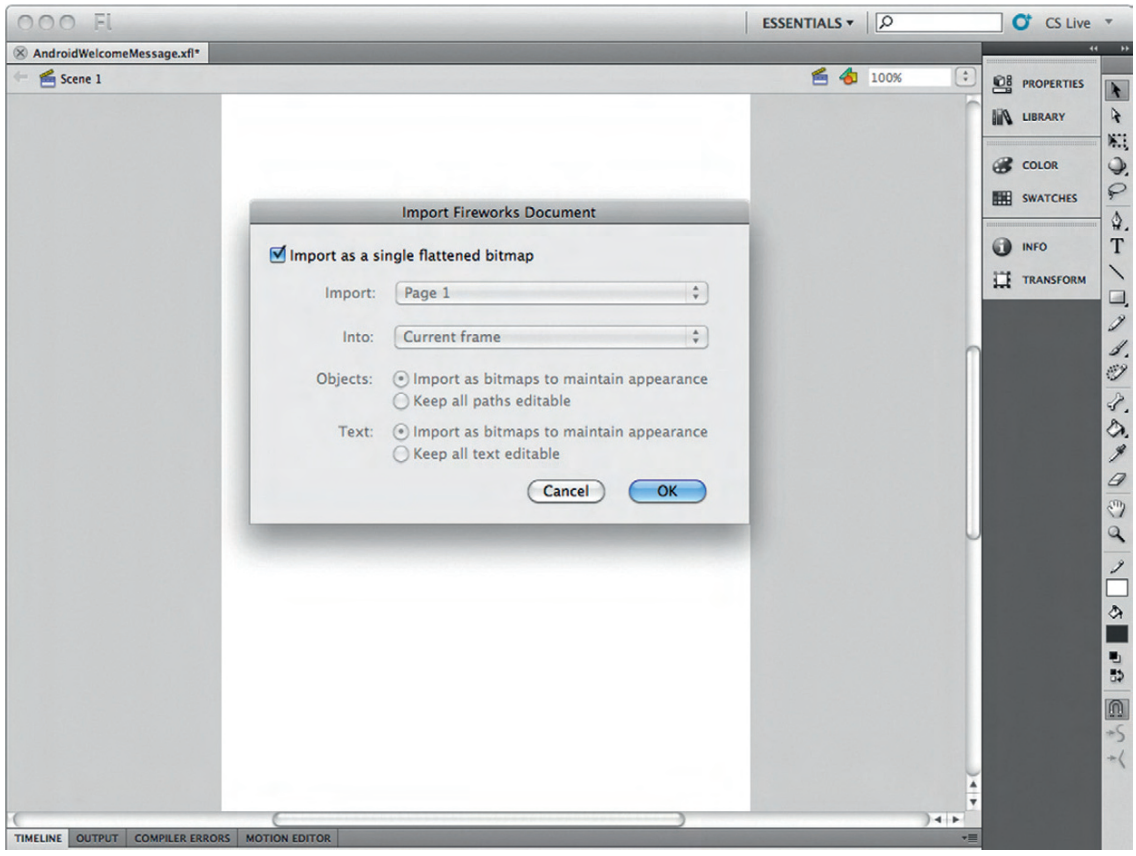



**Figure 1.12Proj** Flatten the imported Fireworks PNG image.

5. Select CTRL+K (Windows) or CMD+K (Mac) to open the Align panel. Select Align to stage and center the imported image (Figure 1.13Proj).
6. Change the pointer tool to the text tool.
7. On the Stage, below the Android image, draw a rectangular text region. Open the properties panel and change the text type to TLF, read only, set the color to black and the font size to 40. Don't forget to change the line setting to multiline.
8. Name the new text field myText (Figure 1.14Proj).

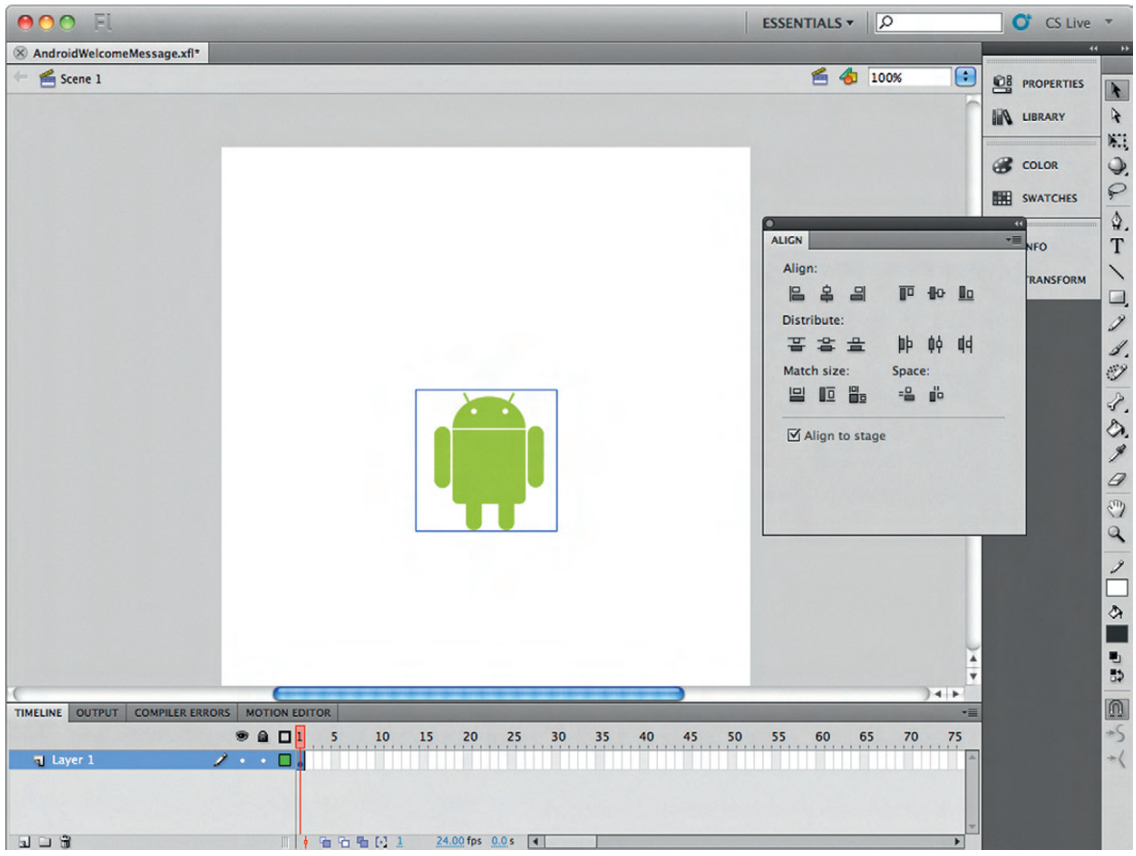



**Figure 1.13Proj** You can use all the image manipulation tools in Flash, such as Align, in your Android apps.

9. Right-click the Android image on the Stage. Select Convert to Symbol.
10. Name the new symbol android_image as shown in Figure 1.15Proj.
11. The image is now a movie clip. With the Android movie clip still selected, open the Properties panel. Name the movie clip android_Btn (Figure 1.16Proj).
12. Open the Actions panel.

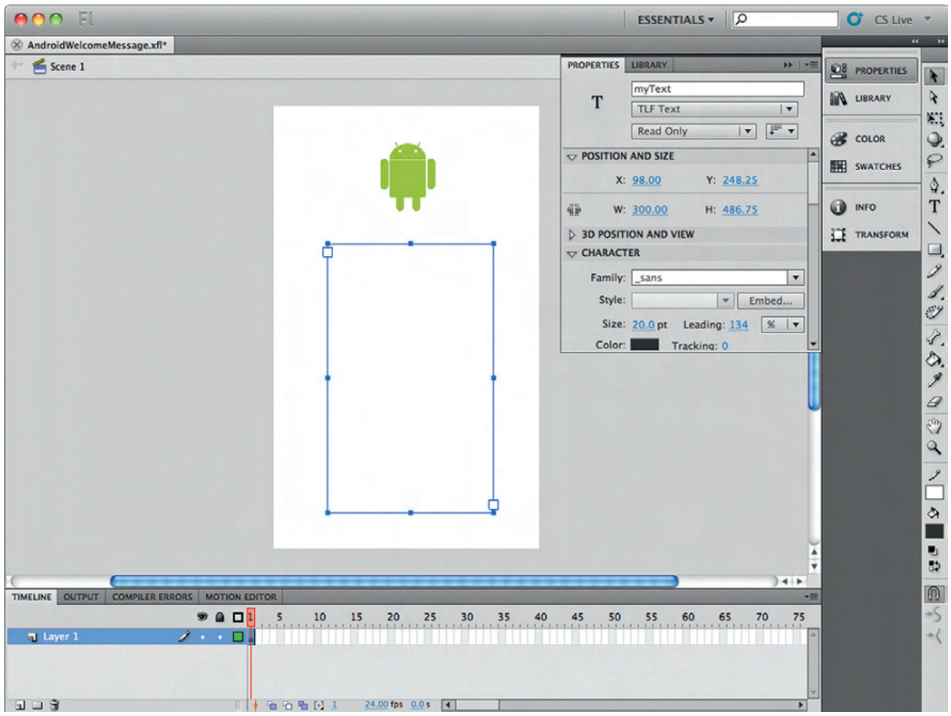



**Figure 1.14Proj** You can use the new TLF text on Android phones.

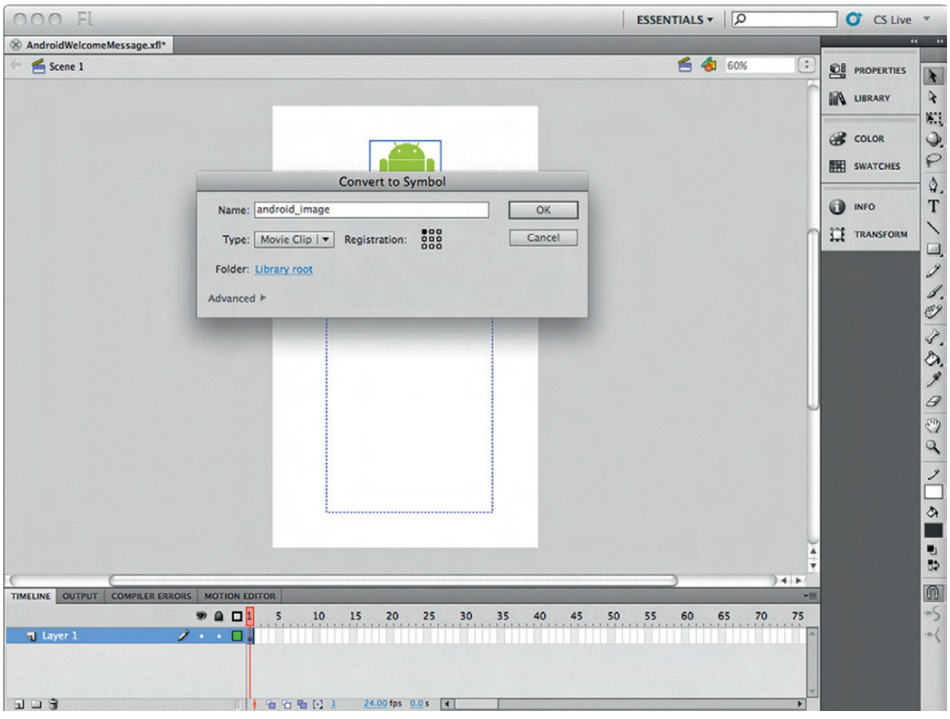



**Figure 1.15Proj** As you might expect, you can use Flash symbols in your Android apps.

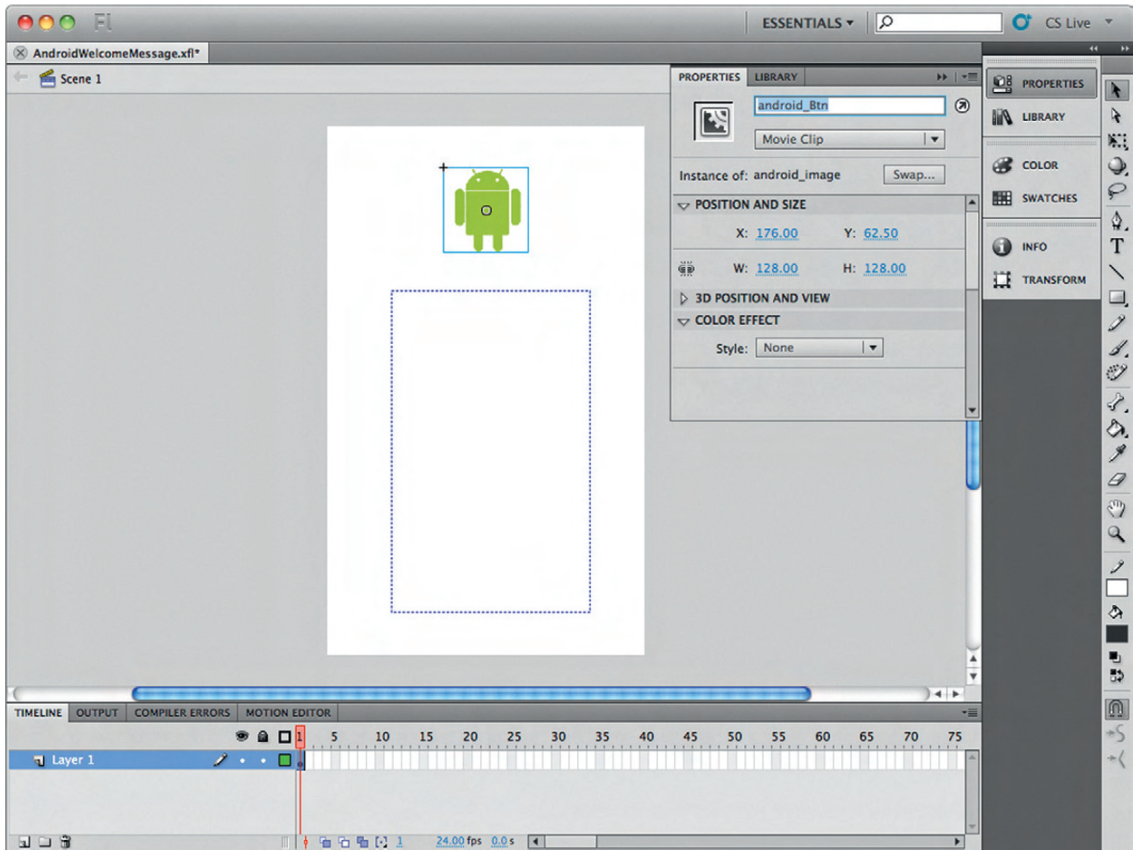



**Figure 1.16Proj** A named movie clip can be referenced in ActionScript.

13. Select Frame one from the timeline. In the Actions panel add the following ActionScript (Figure 1.17Proj):

```
var theDate:Date = new Date( );
var day = theDate.toLocaleDateString();
android_Btn.addEventListener(MouseEvent.CLICK, onClick);
function onClick(event:MouseEvent):void
{
myText.text = "Welcome to Android App development using
Flash CS5. \n \nThe date of your first app is: " + day;
}
```

14. The first line of this script defines a date object; the second line captures the date as a string object.
15. Line three associates a mouse click event with the android_btn object on the stage. A single tap is treated the same as a single click on the mouse.
16. Line 6 generates a message that is posted to the text object when you press the Android icon.
17. Press CTRL+ENTER (Windows) or CMD+ENTER (Mac) to test the application. Click on the Android icon to reveal a message, as shown in Figure 1.18Proj.

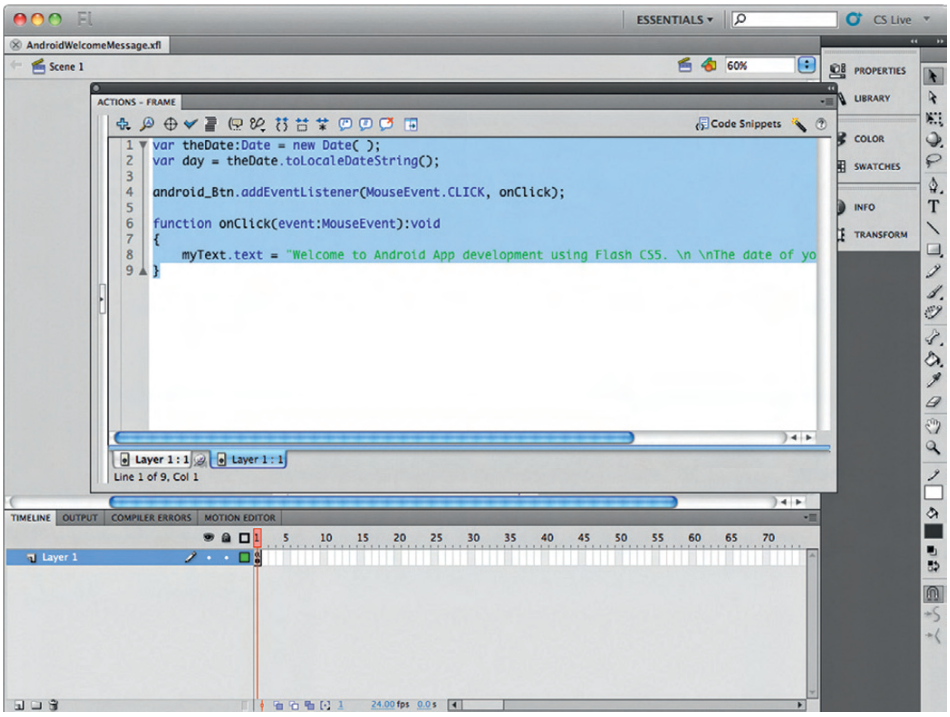



**Figure 1.17Proj** You will use a little ActionScript to test that AS3 will work on your Android phone.

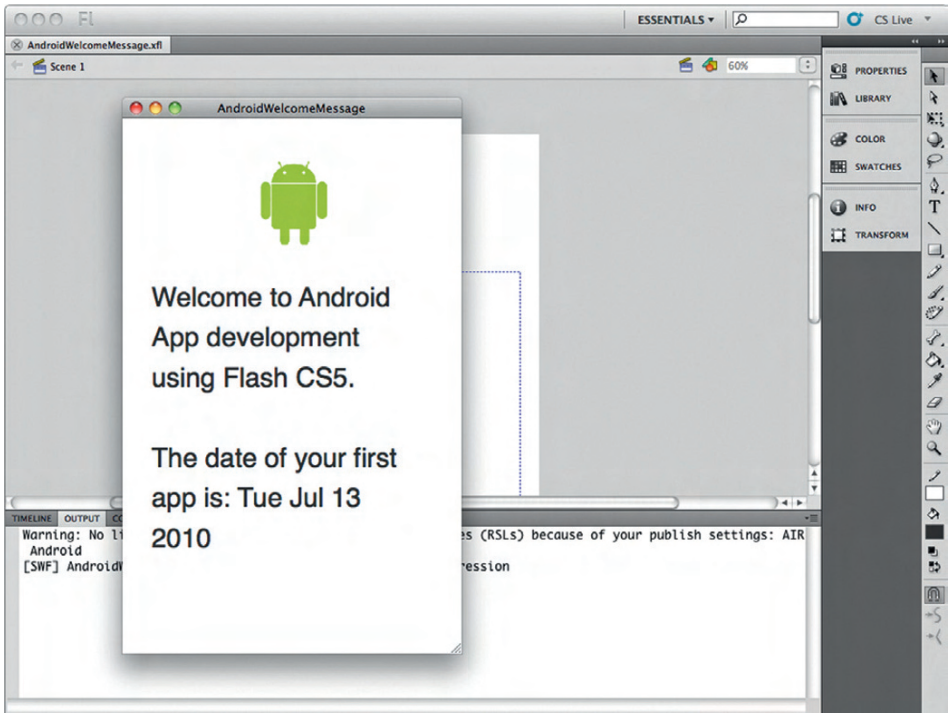



**Figure 1.18Proj** The Flash movie should work in Publish Preview mode.

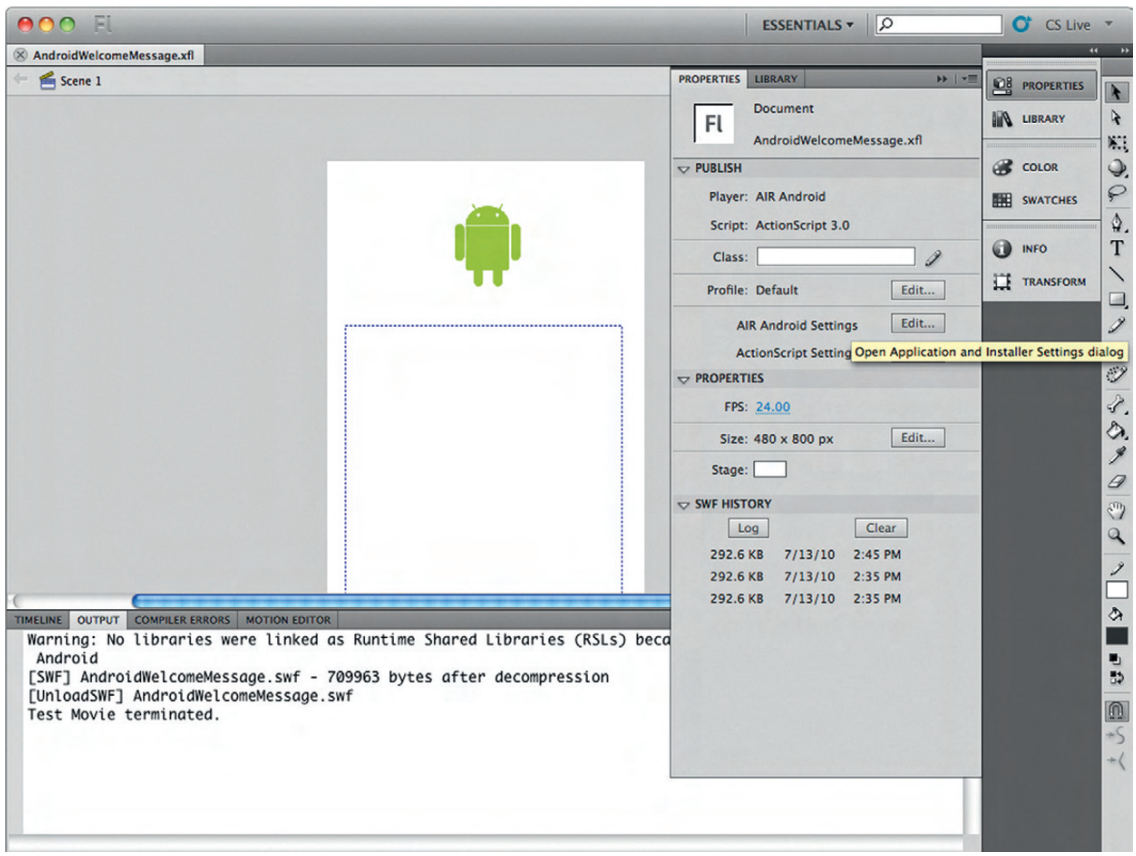
The Flash movie you have created is simple, but it contains all the elements of any large and complex movie: you have ActionScript and images, and you use the timeline. Now you need to publish your app as an Android solution.

## Running Your App on Your Android Phone

The final step is to publish your Flash movie as an Android application.

1. Select the Stage and open the Properties panel.
2. In the Publish section, select AIR Android Settings (Figure 1.19Proj).
3. The Application & Installer Settings window opens. For now, keep the default settings on the General tab (Figure 1.20Proj).

**Figure 1.19Proj** The AIR Android Settings control how you build your Android app.



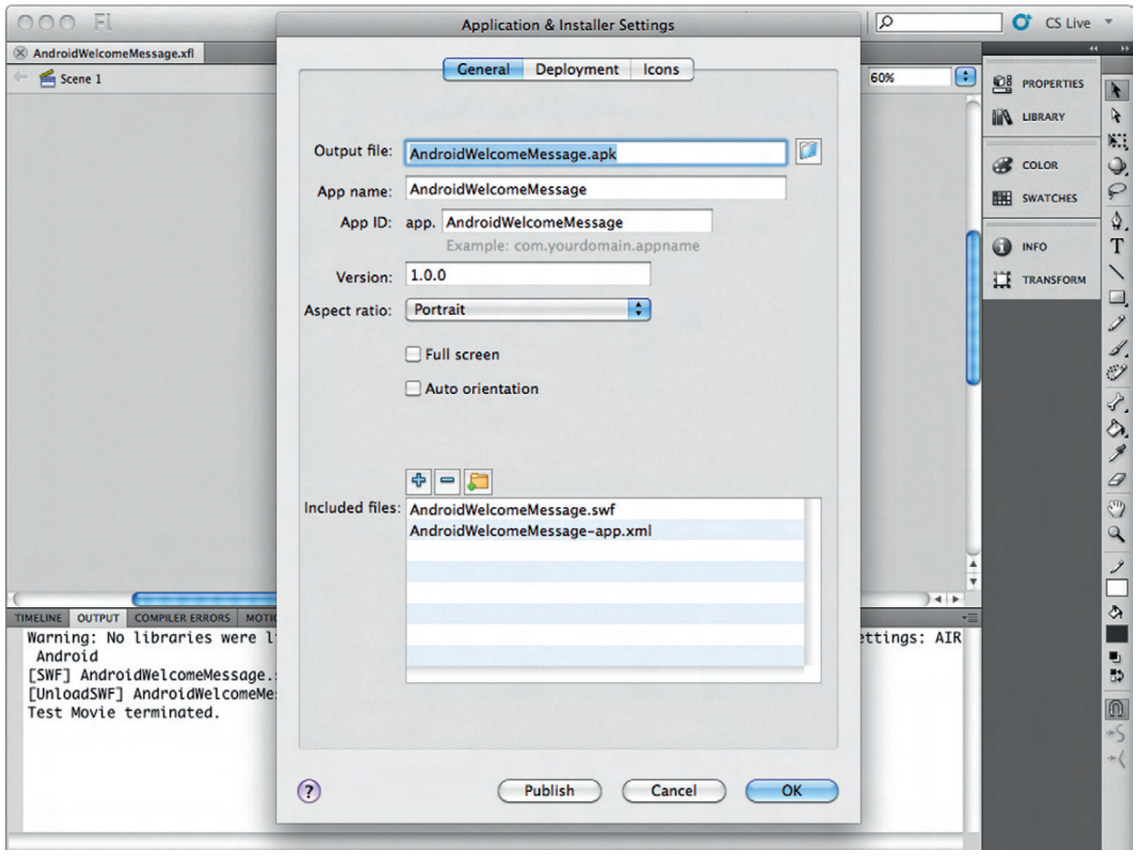



**Figure 1.20Proj** The name of your app and the filename are two of the settings on the General tab.

4. Select the Deployment tab, as shown in Figure 1.21Proj. The Certificate setting should be the same setting you created during the first chapter of the book.
5. Enter your password and check the Remember password for this session checkbox.
6. Connect your Android device to your computer. Select Install application on the connected Android device.
7. Check Launch application on the connected Android device.

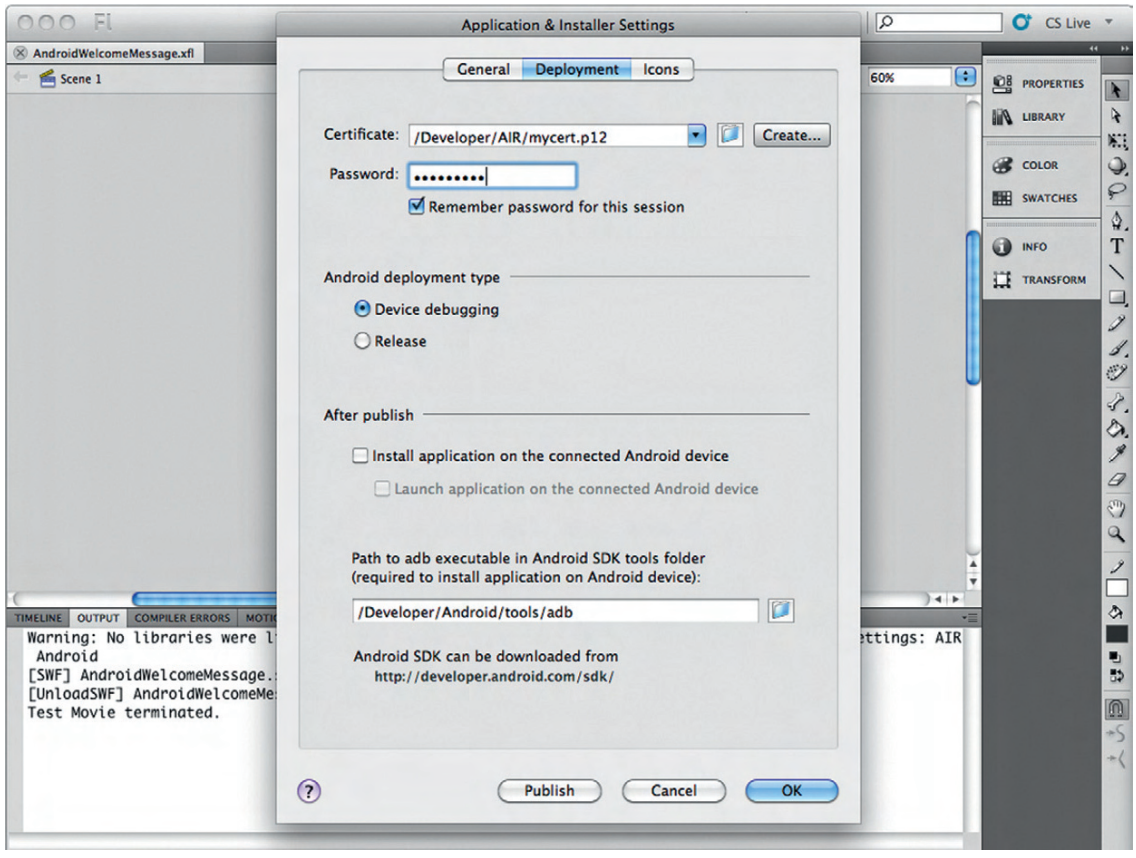



**Figure 1.21Proj** The AIR Android Settings control how you build your Android app.

8. Select the Icons tab. Select each icon in the list. Use the folder button to find and connect each icon you created earlier (Figure 1.22Proj).
9. Select the Publish button. The publishing process will take a couple of minutes depending on the speed of your computer. The final results will be a running app on your Android phone. Click the icon to bring up the message.

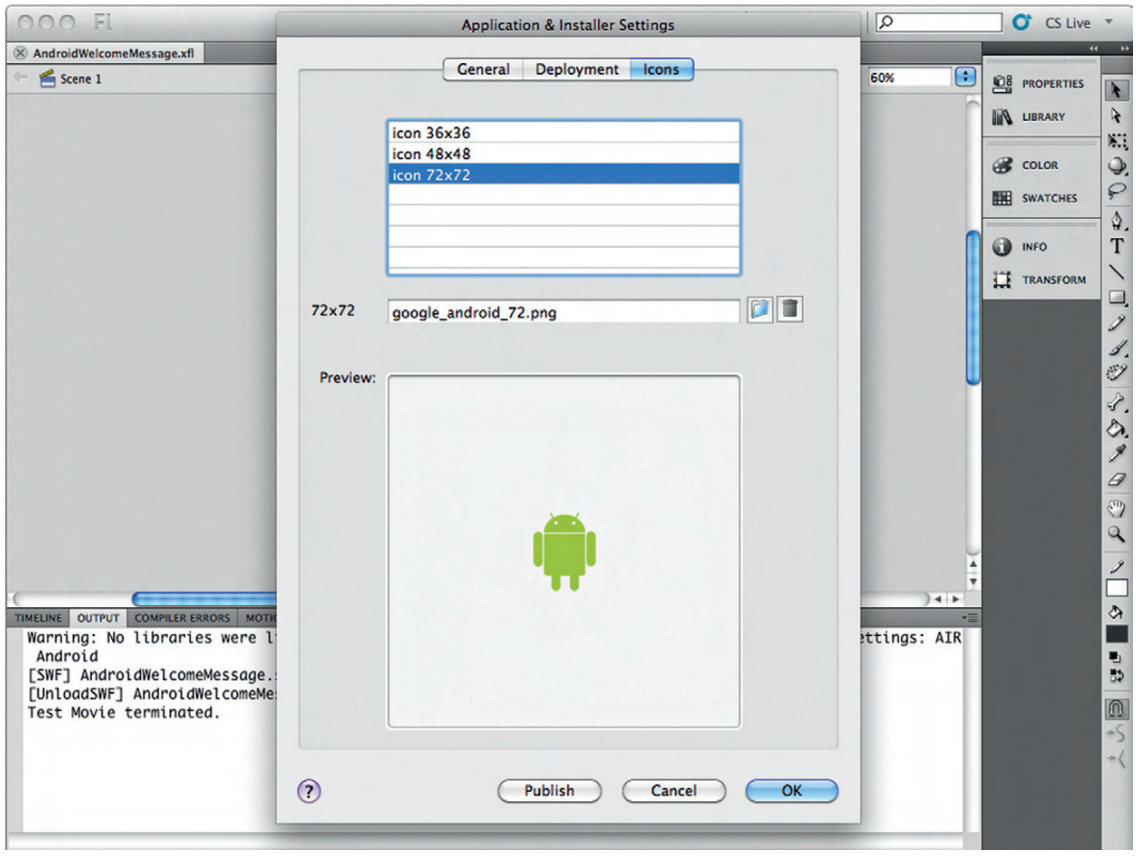



**Figure 1.22Proj** You can associate three different sized icons for your Android applications.

You have created your first complete Android application using Adobe's Flash CS5. In this chapter you learned how to install the Android SDK, update Flash CS5 with AIR for Android, and you created your first application that is now running on your Android phone. Well done!

In the next section you will expand on what can be accomplished in AIR for Android by leveraging animation, video, audio, and components.