Introduction

The Adobe Integrated Runtime (AIR) is a new product from Adobe, released in the spring of 2008. It allows web developers and web designers to use the skills they already have to build rich desktop applications. Revolutionary in many ways, AIR is a cross-operating system run-time that is light-weight yet powerful. Many naysayers might comment that AIR gives you a capability that is already covered in competing products (you can build desktop applications with other tools). But AIR excels in its easy-to-learn application programming interface (API) and in its capability to take existing web applications and add powerful desktop features that are either impossible to do from a browser or prone to browser differences, making them difficult to code. AIR applications, often called rich desktop applications, are springing up all over the Internet. Take a moment and use your favorite search engine to search “Adobe AIR applications”. As of this writing, more than 159,000 results come back with these search keywords (double-quoted) on Google. AIR applications can be written in Adobe Flex, Adobe Flash, HTML, in a combination of HTML/JavaScript, or AJAX—the same skills you use today. In the next 24 hours, after reading Sams Teach Yourself Adobe AIR Programming in 24 Hours, you can start building for the desktop today. It’s that easy.

This book gives you the knowledge, from a professional developer's perspective, of just how easy it is to program for AIR. From installing the AIR runtime to leveraging popular integrated development environments (IDEs), from creating your AIR applications to packaging and deploying them to end users, from debugging AIR applications to profiling them for improving performance, each hour presents you with new, important concepts, reinforced with a workshop, hands-on section at the end of each chapter.

The first 10 hours of chapters introduce you to getting started with AIR, and include the following topics:

- In Hour 1, “Taking in a Deep Breath of AIR,” you’ll learn about the pieces that make up AIR, who benefits from AIR’s easy-to-grasp learning curve, a short history lesson about AIR, and just how big AIR is expected to be.

- In Hour 2, “Ready to Install AIR,” safety considerations for the AIR runtime are discussed, along with differences for your operating system (Windows, Mac OS X, and Linux), and how to install the AIR runtime on the different operating systems.

- In Hour 3, “Harnessing the AIR SDK,” you learn about the AIR software development kit (AIR SDK), including using the Adobe AIR Debug Launcher (ADL), using the Adobe AIR Developer Tool (ADT), and creating a badge (seamless) installer for AIR application deployment.
In Hour 4, “Preparing to Build Your First AIR Application,” you learn about the Adobe suite of tools recommended for building AIR applications, including Flex Builder 3, Flash CS3 Professional, and Dreamweaver CS3, as well as a third-party IDE, Aptana Studio.

In Hour 5, “Writing Your First AIR Application with Flex 3,” you see how to use Flex Builder 3 to build an AIR application. You will walk through building a Hello AIR World application, testing it, and packaging it for deployment.

In Hour 6, “Writing an AIR Application with HTML,” you see how HTML and HTML/JavaScript or AJAX can be used to build AIR applications.

In Hour 7, “Writing an AIR Application with Flash CS3 or Dreamweaver CS3,” you see how to use Flash CS3 Professional and Dreamweaver CS3 to build AIR applications, test them, and package them for deployment.

In Hour 8, “Writing an AIR Application with PDF Integration,” you learn how to add PDFs to your AIR applications and how these integrated PDFs can be controlled within the AIR API.

In Hour 9, “Debugging AIR Applications,” you learn about debugging your AIR applications, including debugging with the Flex Builder Debugger, the Flash Debugger, the AIR HTML Introspector, and even some third-party debugging tools.

In Hour 10, “Distributing Your AIR Application,” you see additional information about packaging and distributing your AIR applications, including suggestions on where to distribute them.

The next nine chapters introduce you to the AIR APIs and include the following topics:

In Hour 11, “Opening Doors with AIR,” you discover how to open one or more windows for your AIR application’s user interface.

In Hour 12, “Screening an AIR Application,” you learn how to control the screens and the screen real estate that your AIR applications utilize.

In Hour 13, “File This API with AIR,” you get a first-hand look at the AIR File API and how you can interact with the files and folders on the end user’s file system.

In Hour 14, “Drag and Drop, Copy and Paste in AIR,” you learn the capabilities of the AIR API related to drag and drop and copy and paste, two must-have capabilities for a professional desktop application.

In Hour 15, “AIR and the Natives,” you see how to build native menus on Windows and the Mac OS.
In Hour 16, “Connectivity with AIR Networking,” you learn how to detect an end user’s connectivity to a network or the Internet and how to transition from a connected to a non-connected state in your AIR applications.

In Hour 17, “Local Data in an AIR Application,” you discover how you can leverage a SQLite local database as an AIR application’s data repository.

In Hour 18, “Encryption and Security in AIR,” you learn about AIR’s encryption and security features.

In Hour 19, “AIR and User Notifications,” you see how AIR can access the operating system’s notification capabilities to build a full-featured desktop application.

The final five chapters discuss some important development-related topics for many AIR applications as follows:

In Hour 20, “Leveraging Server-Side Features for AIR,” you see examples of connecting an AIR desktop application to a server-side infrastructure.

In Hour 21, “Building an AIR Application,” you see a step-by-step walkthrough of building an AIR application that can help you continue learning more about the AIR API.

In Hour 22, “Signing AIR Applications,” you learn more about the digital code-signing of AIR applications.

In Hour 23, “Updating AIR Applications After Deployment,” you discover strategies for updating AIR applications, including suggestions for dynamic updates using a form of pull technology.

In Hour 24, “Where Do I Go from Here?” you read some suggestions on how to continue learning about and improving your understanding of AIR.

Appendix A includes details about downloading an AIR Flex 3 API as an AIR application and how to re-create the most recent AIR Flex 3 API as an AIR application.

Appendix B includes a compilation of useful AIR-related links to help you continue to learn and improve your AIR applications.
In this hour, you’ll learn

- How to use Flex Builder 3 for AIR development
- How to use Flash CS3 Pro for AIR development
- How to use Dreamweaver CS3 for AIR development

How to Use Flex Builder 3 for AIR Development

Setting up Flex Builder 3 for AIR development is a simple matter because everything you need to develop AIR applications is included. Why would anyone want to use another development tool? One possible reason is that Flex Builder 3 might not be free and for smaller, cost-conscious firms, the relatively low upgrade cost might be too much.

If you are a student, a faculty member, or staff member of eligible education institutions, you might qualify for a free license for Flex Builder 3. Check for the requirements at www.flexregistration.com.

By the Way

Other considerations might make Flex Builder 3 a poor choice. If you are not planning to leverage Flex as your development language for your AIR applications, Flex Builder 3 is not the best choice for you. If you are a Flash developer, you should take a look at the Adobe AIR Update for Flash CS3 Professional. If you are a fan of Dreamweaver, give the Adobe AIR Extension for Dreamweaver a try. The next two sections in this hour discuss both of these alternatives to Flex Builder 3.
Okay, so you are still here and want to create your AIR applications with Flex-based code. Without a doubt, Flex Builder 3 is the de facto tool for quickly building Flex-based AIR applications. You should have already downloaded and installed the trial version of Flex Builder 3 (during the Exercises found at the end of Hour 1, “Taking a Deep Breath of AIR”). If not, do so now from www.adobe.com/cfusion/entitlement/index.cfm?e=flex3email.

Feel free to download and install the Eclipse plug-in version if you already have Eclipse installed. Otherwise, the stand-alone version works just as well.

As mentioned in Hour 2, “Ready to Install AIR,” the installation steps are outside the scope of this book. The goal for this section is to familiarize you with the AIR-related parts of Flex Builder 3 and not how to install the tool.

With Flex Builder 3 installed, fire it up and you see a main screen that looks like Figure 4.1. This screen is the Flex Start Page.

You can always get to this screen by looking under the Help menu and selecting Flex Start Page.

The Flex Start Page is a great place to start learning more about building web-based, rich Internet applications, so you can spend some time here later if you like. In the meantime, take a look at the built-in AIR development features found in Flex Builder 3. With Flex Builder 3 you can perform all of the following:

- Create a Flex-based project for coding your AIR application using a wizard-style user interface (UI). Figure 4.2 shows the first step of this wizard-style UI. The rest of the steps are covered in Hour 5, “Writing Your First AIR Application with Flex 3.”
Automatically create the AIR application descriptor, an Extensible Markup Language (XML)–based file that contains metadata information about how the AIR application is handled. This file is automatically generated and named according to the following convention: the AIR main application filename (sans the .mxml extension) plus -app.xml (for example, HelloWorld-app.xml), as shown in Figure 4.3.
Test your AIR application from the toolbar to ensure that you are satisfied with the functioning of your AIR application. Figure 4.4 shows this capability.

Use the Debug toolbar button to debug your AIR applications from within Flex Builder 3. Hour 9, “Debugging AIR Applications,” covers debugging in detail.

Profile the performance of your AIR application by clicking the Profile toolbar button.


Access abundant help within the Flex Builder 3 application. Figure 4.5 shows an example of searching for help for the topic, AIR.

As you can see, if you are going to use Flex-based code to build your AIR applications, the price of the Flex Builder 3 integrated development environment (IDE) might well be worth the expense.
How to Use Flash CS3 Pro for AIR Development

If you are a Flash developer and would like to continue building AIR applications in the familiar Flash IDE, you might best be served by downloading and installing the Adobe AIR update for Flash CS3 Professional. Versions for Flash CS3 Professional for Windows and for Flash CS3 Professional for Macintosh are currently available. You can find either version at www.adobe.com/support/flash/downloads.html.

Installing the AIR update for Flash CS3 Professional is not covered in this book. Please consult the Adobe website for information about the installation steps. As with the Flex Builder 3 discussion in the previous section, the purpose for this section is to discuss the AIR-related features that are enabled with this update. So what does this Flash update provide you? With the AIR update for Flash CS3 Professional you can perform all of the following:

- When you first start Flash CS3 Professional, you can now choose the Flash File (Adobe AIR) option (see Figure 4.6).

![New Menu Option](image)

Figure 4.7 shows an introductory screen about authoring AIR applications with Flash CS3 Professional. It tells you where the AIR-specific menus are located, how to test and debug your AIR applications, and where you can find more help about authoring for an AIR application.
Automatically create the AIR application descriptor, an Extensible Markup Language (XML)-based file that contains metadata information about how the AIR application is handled. This file is automatically generated and named according to the following convention: the AIR main application filename (sans the .mxml extension) plus -app.xml (for example, HelloWorld-app.xml), as shown in Listing 4.1.

**LISTING 4.1** The HelloWorld-app.xml Created by Flash CS3 Pro

```
<?xml version ="1.0" encoding="utf-8" ?>
<application xmlns="http://ns.adobe.com/air/application/1.5">
  <id>com.adobe.example.HelloWorld</id>
  <version>1.0</version>
  <filename>HelloWorld</filename>
  <description></description>
  <name>HelloWorld</name>
  <copyright></copyright>
  <initialWindow>
    <content>HelloWorld.swf</content>
    <systemChrome>standard</systemChrome>
    <transparent>false</transparent>
    <visible>true</visible>
  </initialWindow>
  <icon>
  </icon>
  <customUpdateUI>false</customUpdateUI>
  <allowBrowserInvocation>false</allowBrowserInvocation>
</application>
```

Two new menu choices are available from the Commands menu: AIR - Application and Installer Settings, which is used to make changes to the AIR application descriptor file via a graphical user interface [GUI] and AIR - Create AIR.
File, which is used to physically create the .air file for your AIR application. The use of these menu choices is discussed fully in Hour 7, “Writing an AIR Application with Flash CS3 or Dreamweaver CS3.”

Access a new publish profile, Adobe AIR 1.5. This profile is found by selecting Publish Settings within the File menu, or by using the keyboard shortcut, Ctrl+Shift+F12, and then choosing the Flash tab, as depicted in Figure 4.8.

![Figure 4.8](image.png)

Receive abundant help within the Flash CS3 Pro application. Figure 4.9 shows an example of searching for help for the topic, CS3 for Adobe AIR.

As you can tell, if you are going to use the Flash IDE to build your AIR applications, the Adobe AIR update for CS3 Professional is an excellent tool for the job.
How to Use Dreamweaver CS3 for AIR Development

If you are an HTML, JavaScript, or AJAX developer, the Adobe AIR extension for Dreamweaver CS3 allows you to develop, test, and package your AIR applications directly within Adobe Dreamweaver CS3. The Adobe AIR extension is an MXP file and you use the Adobe Extension Manager to install it. The MXP file is available at www.adobe.com/products/air/tools/ajax/.

Installing the Adobe AIR extension for Dreamweaver CS3 is not covered in this book. Please consult the Adobe website for information if you have questions about the installation. You can access documentation for this extension at www.adobe.com/go/learn_dw_air.

As with the two previous sections, the purpose for this section is to discuss the AIR-related features that are enabled with this extension. With the Adobe AIR extension for Dreamweaver CS3, you can perform all of the following:

- Automatically create the AIR application descriptor, an XML-based file that contains metadata information about how the AIR application is handled. This file is automatically generated and named according to the following...
convention: the AIR main application filename (sans the .mxml extension) plus -app.xml (for example, hello_world-app.xml), as shown in Listing 4.2.

**LISTING 4.2 The hello_world-app.xml Created by Dreamweaver CS3**

```xml
<?xml version="1.0" encoding="utf-8" ?>
<application xmlns="http://ns.adobe.com/air/application/1.5">
  <filename>testSite</filename>
  <copyright>2008</copyright>
  <description>Hello World from DW CS3</description>
  <customUpdateUI>false</customUpdateUI>
  <name>Hello World</name>
  <id>testSite</id>
  <version>3</version>
  <initialWindow>
    <content>hello_world.html</content>
    <height>600</height>
    <width>800</width>
    <systemChrome>standard</systemChrome>
    <transparent>false</transparent>
    <visible>true</visible>
  </initialWindow>
  <icon>
    <image128x128>01fig10.png</image128x128>
  </icon>
</application>
```

Access two new Site menu options, AIR Application Settings (used to make changes to the AIR application descriptor file via a GUI and Create AIR File (used to physically create the .air file for your AIR application).

Choose a new preview by clicking the Preview toolbar icon and selecting Preview in Adobe AIR. This allows you to test your AIR application from within Dreamweaver CS3.

Access help within the Dreamweaver CS3 application. Figure 4.10 shows an example.

As you have seen, if you plan to develop AIR applications with HTML and JavaScript or AJAX and are currently using Dreamweaver CS3, the Adobe AIR extension for Dreamweaver CS3 might just be a perfect match for you.

Another busy hour and you saw three of the IDEs for leveraging your current skills to build AIR applications. Other choices are available in this arena, but with Flex Builder 3, Flash CS3 Pro, and Dreamweaver CS3, you will be using the tools that the creator of the AIR runtime, Adobe, built precisely for you to create AIR applications.
Before starting the next chapter, you should review what you have learned in the following workshop.

**Workshop**

The workshop contains a Q&A section, quiz questions, and an exercise to help reinforce what you’ve learned in this hour. If you get stuck, the answers to the quiz can be found after the questions.

**Q&A**

**Q. Why would you select one IDE over the other?**

**A.** Depending on your experience level, one IDE might just fit your developer hand better. Flex Builder 3 is a likely IDE for Flex developers who want to build AIR applications; Flash CS3 Pro is the best choice for Flash developers who want to build AIR applications; Dreamweaver CS3 and Aptana Studio are two choices for HTML/JavaScript or AJAX-style AIR development.
Q. Which IDE is ready to go out of the box (without needing any updates or extensions)?

A. Flex Builder 3 was designed from the ground up to be the de facto standard for building Flex-based AIR applications. Flash CS3 Pro requires an update and Dreamweaver CS3 requires an extension to be able to leverage AIR development features.

Q. Can I use the Flex Builder 3 Eclipse plug-in version to build AIR applications, or do you need the Flex Builder 3 standalone version?

A. You can build AIR applications with either version of the Flex Builder 3 software. There are no differences in the AIR development features (or any other Flex features for that matter) in either version.

Q. Can I name the application descriptor file differently?

A. The AIR application descriptor created automatically by all three of the Adobe IDEs is named after the main application name (without any file extension) plus -app.xml (for example, for a main application named hello_world.html, the AIR application descriptor is named hello_world-app.xml).

Quiz

1. What does an AIR application descriptor file contain?

2. The Adobe AIR extension for Dreamweaver CS3 is used for what type of AIR applications?

3. How do you create the AIR application descriptor file?

4. What version of AIR are the three Adobe IDEs targeted for?

Answers

1. The AIR application descriptor contains metadata that describes the AIR application and includes such things as the application name, the description, copyright, application version, AIR runtime version, system chrome, window size and initial position, and in some cases application icon information. Listing 4.3 shows the application descriptor created by completing the Dreamweaver CS3 AIR Application Settings GUI.
LISTING 4.3 The *hello_world-app.xml* Created by Dreamweaver CS3

```
<?xml version="1.0" encoding="utf-8" ?>
<application xmlns="http://ns.adobe.com/air/application/1.5">
  <filename>testSite</filename>
  <copyright>2008</copyright>
  <description>Hello World from DW CS3</description>
  <customUpdateUI>false</customUpdateUI>
  <name>Hello World</name>
  <id>testSite</id>
  <version>3</version>
  <initialWindow>
    <content>hello_world.html</content>
    <height>600</height>
    <width>800</width>
    <systemChrome>standard</systemChrome>
    <transparent>false</transparent>
    <visible>true</visible>
  </initialWindow>
  <icon>
    <image128x128>01fig10.png</image128x128>
  </icon>
</application>
```

2. The Adobe AIR extension for Dreamweaver CS3 is used to build HTML, HTML/JavaScript, or AJAX-based AIR applications.

3. You do need to create the AIR application descriptor file. All three of the Adobe IDEs automatically generate this file. You can certainly modify it with any text editor, but creating it from scratch is not necessary.

4. All three Adobe IDEs are targeted, as of this writing, at the AIR 1.5 runtime.

By the Way

The AIR runtime, as of this writing, is at version 1.5.

Exercises

Take a look at Aptana Studio. You should explore what it offers because you might find that it is a better fit for you over what Dreamweaver CS3 offers. You can download it at www.aptana.com/studio.
Index

AC_FL_RunContent( ) function, 38-39, 169, 171
Add Shortcut Icon to My Desktop checkbox (Installing Application progress dialog), 164
addItem( ) method, 257
addSubmenu( ) method, 257
addSubmenuAt( ) method, 257
ADL (AIR Debug Launcher), 29-30, 151-153
Adobe AIR 1.0 publish profile (Flash CS3 Professional), 51
Adobe AIR Already Installed dialog, 163
Adobe AIR Developer Center, 388-390
Adobe AIR Marketplace, 391
Adobe AIR update (Flash CS3), 117
Adobe Apollo Camp (AIR development), 10
Adobe XML News Aggregator, 390
ADT (AIR Developer Tool), 31-34
could not generate timestamp messages, 373
packaging release builds via, 177
adt -help command, 32
Advanced Settings dialog (Flash CS3), 118
aggregators (news), Adobe XML News Aggregator, 390
AIR (Adobe Integrated Runtime), 7-8
API classes, accessing via JavaScript, 102-106
application descriptor files.
See application descriptor files
application IDs, 59
application installation dialog, 16
application sandbox, 16
beneficiaries of, 9
code-signing certificates, 17
development of, 10
documentation, downloading, 387-388
downloading, 25
Dreamweaver CS3-based application creation, 122-128
features of, 8
Flash CS3-based application creation, 116-122
Flex Builder 3-based application creation, 57-59
application code, test, repeat process, 73-87
creating application descriptors, 60, 62-68
testing applications, 48
AIR (Adobe Integrated Runtime)

forums, 388
HTML-based application creation, 94-100
nonapplication sandbox, 16
popularity of, 12
runtime
Linux installations, 19, 21
Mac OS X installations, 18-19, 21
safety considerations, 15, 17
testing installations, 19
Windows installations, 18-20
security model, 16
timestamping, 17
AIR - Application and Installer Settings command (Flash CS3 Professional), 50, 118
AIR - Create AIR File command (Flash CS3 Professional), 50
AIR 1.0/Flex 3 API application reference, 395-396
AIR Application and Installer Settings dialog (Dreamweaver CS3), 127
AIR application descriptors
Dreamweaver CS3, creating via, 52-53
Flash CS3 Professional, creating via, 50
Flex Builder 3, creating via, 47
modifying, 190-193
AIR Application Is Already Installed dialog, 165
AIR Application Settings option (Dreamweaver CS3), 53, 125
AIR codes
digital certificates, 309-311
publisher ID display example, 310
signing, 308
timestamping, 308-309
AIR Debug Launcher, 151
AIR File Contents screen (Export Release Build Wizard), 89
AIR HTML Introspector (Adobe), 155, 157
AIR installation dialog (Dreamweaver CS3), 128
AIR Project Wizard (Aptana Studio)
accessing, 94
Application XML Properties screen, 94, 96
Create an Adobe AIR Project screen, 94
Import JavaScript Library screen, 96-97
AIR SDK, 8, 12
ADL (AIR Debug Launcher), 29-30
ADT (AIR Developer Tool), 31-32, 34
badge installers, 35-39, 166
components of, 24
downloading, 387
instructional PDF files, 387-388
Linux installations, 24
Mac OS X installations, 22-23
packaging release builds via, 177
Windows installations, 22
AIR.Capabilities API class, 105
air.File API class, 106
air.FileMode API class, 106
airFileStream API class, 106
air.System API class, 105
AIRAliases.js file, 102-105
airversion parameter (flashvars), 39
API classes, accessing via JavaScript, 102-106
Apollo Camp (AIR development), 10
application descriptor files
initial application window, assigning properties to, 66-67
installation folders, assigning to, 68
naming, 99
new file type associations, registering, 68-69, 71-72
program menu folders, assigning to, 68
troubleshooting, 63
updates, 68
application IDs, 59
application descriptor files, 64
AIR-based AIR applications, 96
application installation dialog, 16
application menus, 259, 261
application sandbox, 16
Application XML Properties screen (AIR Project Wizard), 94, 96
applicationComplete events
application menus, 259
creationComplete events versus, 259
encrypted data, storing, 306
applicationComplete tag (XML), 74
applicationCreation events
context menus, 267
dock icon menus, 264
window menus, 262
applications, building
coding
displaying archived information, 353-355
displaying information, 349, 351-352
file selection, 347-348
initializing applications, 348
monitoring Internet connections, 360-362
PDF integration, 358-360
replaying archived information, 356-358
retrieving server-side data, 362-365
validation coding, 349
decision-making process, 343
feature selection, 345
infrastructure diagrams, 345-346
appname parameter (flashvars), 39
appurl parameter (flashvars), 39
Aptana Studio
AIR Project Wizard, 94-97
Export Adobe Air Package wizard, 115
HTML-based AIR application creation, 94-100
New Project Wizard, Select a Wizard dialog, 94
archived information (application build process)
displaying, 353-355
replaying, 356-358

B
badge installers
AIR application installations via web page, 166-170
AIR SDK, 166
creating, 35-39
badges (application descriptor files), toggling browser invocation, 68
BlazeDS, leveraging messaging, 339-340
bouncing Dock icon (Mac OS), 325-327
bounds property, screens, 199, 205
BridgeSound class, 79-81, 84-85
custom windows, creating, 190-195
customizing flashvars parameters, 39

C
CA (certificate authorities)
digital certificates
AIR codes, 309-311
obtaining, 178, 368-371
selecting, 367
Capabilites.screenResolutionX property, 206
Capabilites.screenResolutionY property, 206
certificates. See digital certificates
CFC (ColdFusion component)
HTTP communication, leveraging, 337-338
remoting, leveraging, 334-335
server-side data, retrieving (application build process), 362-365
web services, leveraging, 336-337
Charles debugging tool, 154
checkConnection( ) function, 82, 84
checkLogin( ) function, 307
CheckSO( ) function, 306
copy-and-paste API, 247
Clipboard class, 246
clipboard data, deferring creation/rendering of, 249
ClipboardFormats class, 246
ClipboardTransferMode class, 246
code example, 250, 252
copy events, 247-248
paste events, 249-250
copyright node (XML), 65
could not generate timestamp messages, 373
Create a Flex Project screen (New Flex Project Wizard), 57
Create AIR File option (Dreamweaver CS3), 53
Create an Adobe AIR Project screen (AIR Project Wizard), 94
Create New section (Flash CS3), 116
creationComplete events
application menus, 259
applicationComplete events versus, 259
custom user interfaces
customizing flashvars parameters, 38

How can we make this index more useful? Email us at indexes@samspublishing.com
Debug toolbar button (Flex Builder 3), 48
debugging
   ADL, 29-30, 151-153
   AIR HTML Introspector (Adobe), 155, 157
   Charles, 154
   Flex Builder Debugger (Adobe), 143-151
   ServiceCapture, 153-154
   third-party tools, 153-154
decision-making process (application build process), 343
deploying AIR applications, updating post deployment
   manual updates, 380-381
   remote updates, 382-383
   Updater class, 377-378, 381
description node (XML), 65
descriptor files. See application descriptor files
desktop AIR applications
   Dreamweaver CS3, creating via, 122, 128
      descriptor modification, 125, 127
      HelloWorld.html file, 123, 125
      self-signed certificate creation, 125-126
   Flash CS3, creating via, 116, 121
      configuring settings, 118
      HelloWorld.fla file, 116
      installing applications, 120
      site creation, 122
   HTML, creating via, 94
      application IDs, 96
      default HelloWorld.html file, 97-99
      importing JavaScript libraries, 96-97
      packaging applications, 100
      testing applications, 99-100
developers, web resources, 397-398
   Adobe AIR Developer Center, 389-390
   Adobe AIR Marketplace, 391
   Adobe XML News Aggregator, 390
   AIR forums, 388
diagrams (application build process), 345-346
dictionary class, 217-218, 352
digital certificates
   AIR codes, 309-311
   CAs, selecting, 367
   obtaining, 178, 368-369, 371
   publisher ID, leveraging, 372
   Digital Signature dialog
      (Dreamweaver CS3), 126
   Digital Signature dialog
      (Flash CS3), 120
   Digital Signature screen (Export Release Build Wizard), 88
directories, 225
   existence of, testing for, 225
   File objects in, 225
      reading/writing to, 226
      testing for existence of, 226-227
   missing directories, 225
   storage directories, 226
displaying information (application build process), 349, 351-355
Dock icon (Mac OS), bouncing, 325-327
Dock icon menus, 264-265
doDrag() method, 237
downloading
   AIR, 25
   Air 1.0/Flex 3 API application reference, 395-396
   AIR applications, 161-163
   AIR documentation, 387-388
   AIR SDK, 387-388
   Downloads tab (Adobe Air Developer Center), 389
dPAPI (Data Protection API), 302
drag-and-drop API (application programming interface), 235
   drag events, 236-242
   drop events, 243-246
   NativeDragManager class, 235-236, 238
   NativeDragOptions class, 237
dreamweaver CS3, 113
   AIR Application and Installer Settings dialog, 127
   AIR application descriptors, creating, 52-53
   AIR Application Settings menu, 53, 125
   AIR development, 52-54
   AIR installation dialog, 128
   Create AIR File option, 53
desktop application creation via, 122-128
   Digital Signature dialog, 126
   Flash CS3 versus, 114
   packaging release builds via, 173, 175
   Site menu, 122
   TextPad versus, 114
drop events (drag-and-drop API), 243-246
dynamic CSS, 107
encrypted data
   DPAPI, 302
      storing, 301-307
   eval() function, security concerns with, 108, 110
   Export Adobe AIR Package wizard (Aptana Studio), 115
   Export Release Build Wizard (Flex Builder 3), 87-89
feature selection (application build process), 345
file API (application programming interface), 213
code example, 221-222
File class, 214-221
FileEvent class, 214
Flash CS3
    Adobe AIR update, 117
    Advanced Settings dialog, 118
    Command menu, 118
    Create New section, 116
desktop application creation via, 116-122
    Digital Signature dialog, 120
    Dreamweaver CS3
    versus, 114
    Publish AIR File button, 120
    radio button selection in, 120
    Select Icon Images
    button, 118
    splash screen, accessing, 116
    TextPad versus, 114
    Timestamp check box, 120
    Use Custom UI for Updates
    check box, 118

Flash CS3 Professional
    Adobe AIR 1.0 publish
    profile, 51
    AIR - Application and Installer
    Settings command, 50
    AIR - Create AIR File
    command, 50
    AIR application descriptors,
    creating, 50
    AIR development, 49-51
    Flash File (Adobe AIR)
    option, 49
    packaging release builds
    via, 176

Flash File (Adobe AIR) option
(Flash CS3 Professional), 49

Flash Remoting, 153
flash.display.Screen class,
199-201, 203
flashvars parameters,
customizing, 38
flashvars property, 169-170

Flex Builder 3
    AIR application descriptors
    assigning information to,
    64-65
    assigning installation
    folders to, 68
    assigning program menu
    folders to, 68
    assigning properties
to initial application
window, 66-67
    creating, 47, 60, 62-63
custom user interfaces, 68
    registering new file type
associations, 68-72
toggling browser
    invocation, 68
    updates, 68

AIR applications
    BridgeSound class, 81,
    84-85
code, test, repeat
    process, 73-87
    ConnectionStatus class,
    81-83
creating, 57, 59
custom user interfaces,
    86-87
initApp( ) function, 78-79
packaging release builds,
    87-88, 90
    Script tag, 75-78, 81
    startHelloTimer( )
    function, 80
    SWF files, 66
testing, 48
timer functions, 79-80
    WindowedApplication
tag, 74
    AIR development, 45, 48
    Debug toolbar button, 48
    Export Release Build Wizard,
    87-89
    Flex Start Page, 46
    launching, 57
    New Flex Project Wizard,
    57-58
    packaging release builds via,
    171, 173
    Profile toolbar button, 48
    AIR development, 45, 48
    Debug toolbar button, 48
    Export Release Build Wizard,
    87-89
    Flex Start Page, 46
    launching, 57
    New Flex Project Wizard,
    57-58
    packaging release builds via,
    171, 173
    Profile toolbar button, 48
    AIR development, 45, 48
FlexBuilder Debugger (Adobe),
    143-151
    Flex Start Page (Flex Builder 3), 46
    FlexNativeMenu class, 258-259

How can we make this index more useful? Email us at indexes@samspublishing.com
flight information gathering application build example

flight information gathering application build example coding
displaying archived information, 353-355
displaying information, 349, 351-352
file selection, 347-348
flight searches, 349
initializing applications, 348
monitoring Internet connections, 360-362
PDF integration, 358-360
replaying archived information, 356-358
retrieving server-side data, 362-365
validation coding, 349
decision-making process, 343
feature selection, 345
infrastructure diagrams, 345-346
forums (AIR-related), 388
full-screen mode (screens), 208, 210

G–H
GeoTrust, timestamping, 18
getSound( ) function, 79-80, 85
Getting Started tab (Adobe Air Developer Center), 389
Go Fly application build example.
See flight information gathering application build example
hasFormat( ) method, 250
height node (XML), 67
HelloWorld.fla file, 116
HelloWorld.html file, 97-99, 123, 125
highlighting Windows Taskbar, 318
Home tab (Adobe Air Developer Center), 389
HTML (Hypertext Markup Language)
displaying archived information, 353-355
displaying information, 349, 351-352
file selection, 347-348
flight searches, 349
initializing applications, 348
monitoring Internet connections, 360-362
PDF integration, 358-360
replaying archived information, 356-358
retrieving server-side data, 362-365
validation coding, 349
decision-making process, 343
feature selection, 345
infrastructure diagrams, 345-346
forums (AIR-related), 388
full-screen mode (screens), 208, 210

I
icons, creating system tray icons, 323-325
id strings, 64
IDE (integrated development environments), choosing, 113, 116
imageurl parameter (flashvars), 39
Import JavaScript Library screen (AIR Project Wizard), 96-97
installing
installation files, creating
AIR SDK, 177
ADT, 177
Dreamweaver CS3, 173, 175
Flash CS3 Professional, 176
Flex Builder 3 (Adobe), 171, 173
installing
installation folders, assigning application descriptor files, 68
installing
AIR applications
one-off approach, 161-165
via web pages, 166-170
AIR runtime
Linux installations, 21
Linux suggested minimum configurations, 19
Mac OS X installations, 19, 21
Mac OS X suggested minimum configurations, 18-19
testing installations, 19
Windows installations, 19-20
Windows suggested minimum configurations, 18-19
AIR SDK
Linux installations, 24
Mac OS X installations, 22-23
Windows installations, 22
Installing Application progress dialog, 164

init( ) function
application menus, 260
dock icon menus, 264
FlexNativeMenu class, 258
local SQL databases, 295
window menus, 262
initApp( ) function, 74, 78-79
"Initial content not found" error messages (ADL), 151
initializing applications (application build process), 348
installation folders, assigning application descriptor files, 68
installing
AIR applications
one-off approach, 161-165
via web pages, 166-170
AIR runtime
Linux installations, 21
Linux suggested minimum configurations, 19
Mac OS X installations, 19, 21
Mac OS X suggested minimum configurations, 18-19
testing installations, 19
Windows installations, 19-20
Windows suggested minimum configurations, 18-19
AIR SDK
Linux installations, 24
Mac OS X installations, 22-23
Windows installations, 22
Installing Application progress dialog, 164

init( ) function
application menus, 260
dock icon menus, 264
FlexNativeMenu class, 258
local SQL databases, 295
window menus, 262
initApp( ) function, 74, 78-79
"Initial content not found" error messages (ADL), 151
initializing applications (application build process), 348
installation folders, assigning application descriptor files, 68
installing
AIR applications
one-off approach, 161-165
via web pages, 166-170
AIR runtime
Linux installations, 21
Linux suggested minimum configurations, 19
Mac OS X installations, 19, 21
Mac OS X suggested minimum configurations, 18-19
testing installations, 19
Windows installations, 19-20
Windows suggested minimum configurations, 18-19
AIR SDK
Linux installations, 24
Mac OS X installations, 22-23
Windows installations, 22
Installing Application progress dialog, 164

init( ) function
application menus, 260
dock icon menus, 264
FlexNativeMenu class, 258
local SQL databases, 295
window menus, 262
initApp( ) function, 74, 78-79
"Initial content not found" error messages (ADL), 151
initializing applications (application build process), 348
installation folders, assigning application descriptor files, 68
installing
AIR applications
one-off approach, 161-165
via web pages, 166-170
AIR runtime
Linux installations, 21
Linux suggested minimum configurations, 19
Mac OS X installations, 19, 21
Mac OS X suggested minimum configurations, 18-19
testing installations, 19
Windows installations, 19-20
Windows suggested minimum configurations, 18-19
AIR SDK
Linux installations, 24
Mac OS X installations, 22-23
Windows installations, 22
Installing Application progress dialog, 164

init( ) function
application menus, 260
dock icon menus, 264
FlexNativeMenu class, 258
local SQL databases, 295
window menus, 262
initApp( ) function, 74, 78-79
"Initial content not found" error messages (ADL), 151
initializing applications (application build process), 348
installation folders, assigning application descriptor files, 68
installing
AIR applications
one-off approach, 161-165
via web pages, 166-170
AIR runtime
Linux installations, 21
Linux suggested minimum configurations, 19
Mac OS X installations, 19, 21
Mac OS X suggested minimum configurations, 18-19
testing installations, 19
Windows installations, 19-20
Windows suggested minimum configurations, 18-19
AIR SDK
Linux installations, 24
Mac OS X installations, 22-23
Windows installations, 22
Installing Application progress dialog, 164

init( ) function
application menus, 260
dock icon menus, 264
FlexNativeMenu class, 258
local SQL databases, 295
window menus, 262
initApp( ) function, 74, 78-79
"Initial content not found" error messages (ADL), 151
initializing applications (application build process), 348
installation folders, assigning application descriptor files, 68
installing
AIR applications
one-off approach, 161-165
via web pages, 166-170
AIR runtime
Linux installations, 21
Linux suggested minimum configurations, 19
Mac OS X installations, 19, 21
Mac OS X suggested minimum configurations, 18-19
testing installations, 19
Windows installations, 19-20
Windows suggested minimum configurations, 18-19
AIR SDK
Linux installations, 24
Mac OS X installations, 22-23
Windows installations, 22
Installing Application progress dialog, 164
Internet connectivity, 273
HTTPStatusEvent class, 274-276
Loader class, 277-280
monitoring (application build process), 360-362
NetStatusEvent class, 276-277
ServiceMonitor class, 280-281
URLMonitor class, 282-284

J–K–L

JavaScript
AIR API classes, accessing, 102-106
coding practices, 108, 110
HTML-based AIR applications, importing libraries to, 96-97
security, 108, 110
leveraging server-side features
HTTP communication, 337-338
messaging, 339-340
remoting, 333-335
web services, 336-337
Linux
AIR runtime installations, 19, 21
AIR SDK installations, 24
Loader class, 277, 279-280
local data storage, 287
local shared objects, 289-291
local SQL databases, 292-297
writing data to file systems, 288-289
local shared objects, 289-291
local SQL databases, 292-297
logins, encrypted data storage, 307

M

Mac OS X
AIR runtime installations, 19, 21
suggested minimum configurations, 18-19
testing, 19
AIR SDK installations, 22-23
Dock icon, bouncing, 325-327
maximizable node (XML), 67
maxSize node (XML), 68
menus
application menus, 259, 261
context menus, 267
Dock icon menus, 264-265
pop-up menus, 268-269
System Tray icon menus, 266
window menus, 262-263
messagecolor parameter (flashvars), 39
messages
could not generate timestamp messages, 373
toast messages, creating, 319-323
messaging, leveraging, 339-340
minimizable node (XML), 67
minSize node (XML), 68
missing directories, creating, 228
monitorConnectionStatus( ) function, 81-82
monitoring Internet connections (application build process), 360-362
mx.core.Window class, creating windows via, 187-189

N

name node (XML), 65
nativeDragComplete event, 242
NativeDragManager class, 235-236, 238
NativeDragOptions class, 237
nativeDragStart event, 241

O–P

onConnectionChange( ) function, 78, 84
onError( ) function, 83
onTimer( ) function, 79, 81
openAsync( ) method, 229

How can we make this index more useful? Email us at indexes@samspublishing.com
packaging

HTML-based AIR applications, 100
release builds
ADT, 177
AIR SDK, 177
Dreamweaver CS3, 173, 175
Flash CS3 Professional, 176
Flex Builder 3 (Adobe), 171, 173
Flex-based AIR applications, 87-90
password field (encrypted data storage), clearing, 307
paste events (copy-and-paste API), 249-250
PDF files, AIR SDK instructional files, 387-388
PDF integration, AIR applications with, 133, 358-360
HTMLPDFCapability class constants with, 133-134
flex-based example of, 134-135
limitations of, 139-140
loading PDF content, 135, 137-138
play( ) function, 85
playSoundEffects( ) function, 81
pop-up menus, 268-269
positioning screens programmatically, 206-207
Profile toolbar button (Flex Builder 3), 48
program menu folders, assigning application descriptor files, 68
Publish AIR File button (Flash CS3), 120
publisher ID, leveraging, 372
publisher ID display example, 310
publishing
Adobe AIR Marketplace, 391
web resources, 391

Q–R

radio button selection in Flash CS3, 120
RDA (rich desktop applications), defining, 7
readBoolean method, 229
readByte method, 229
readBytes method, 229
readDouble method, 229
readFloat method, 229
readInt method, 230
readMultiByte method, 230
readObject method, 230
readShort method, 230
readUnsignedByte method, 230
readUnsignedInt method, 230
readUnsignedShort method, 230
readUTF method, 230
readUTFBytes method, 230
release builds, packaging
ADT, 177
AIR SDK, 177
Dreamweaver CS3, 173, 175
Flash CS3 Professional, 176
Flex Builder 3 (Adobe), 171, 173
Flex-based AIR applications, 87-90
remotely updating AIR applications, 382-383
remoting, leveraging, 333-335
replaying archived information (application build process), 356-358
resizable node (XML), 67
resolution (screens), 204-205
retrieving server-side data (application build process), 362-365
RIA (rich Internet applications), defining, 7

S

Samples tab (Adobe Air Developer Center), 389
saveInfo( ) function, 307

screens
bounds property, 199, 205
coding example, 201-203
flash.display.Screen class, 199-201, 203
full-screen mode, 208, 210
positioning programmatically, 206-207
resolution, determining, 204-205
visibleBounds property, 199-207
windows versus, 199
Script tag (Flex-based AIR applications), 75-78, 81
seamless (badge) installers, creating, 35-39
security
AIR codes
digital certificates, 309-311
signing, 308
timestamping, 308-309
AIR security model, 16
code-signing certificates, 17
DPAPI, 302
encrypted data, storing, 301, 303-307
HTML, 108, 110
JavaScript, 108, 110
timestamping, 17
Select a Wizard dialog (Aptana Studio, New Project Wizard), 94
Select Icon Images button (Dreamweaver CS3), 125
Select Icon Images button (Flash CS3), 118
self-signed certificates, 32, 34
server-side data retrieval (application build process), 362-365
server-side features, leveraging HTTP communication, 337-338
messaging, 339-340
remoting, 333-335
web services, 336-337
server-side infrastructure diagrams (application build process), 345-346
ServiceCapture debugging tool, 153-154
ServiceMonitor class, 280-281
setDataHandler() method, 249
signatures
AIR codes, 308-311
CAs, selecting, 367
digital certificates
leveraging publisher ID, 372
obtaining, 178, 368-371
timestamping servers, 373
Site menu (Dreamweaver CS3), 122
soundCompleteHandler() function, 80
splash screen (Flash CS3), accessing, 116
SQL API (application programming interface)
local SQL databases, 292-297
SQLCollationType class, 292
SQLColumnNameStyle class, 292
SQLColumnSchema class, 292
SQLConnection class, 292
SQLError class, 292
SQLErrorEvent class, 292
SQLErrorOperation class, 293
SQLevent class, 293
SQLIndexSchema class, 293
SQLMode class, 293
SQLResult class, 293
SQLSchema class, 293
SQLSchemaResult class, 293
SQLStatement class, 293
SQLTableSchema class, 294
SQLTransactionLockType class, 294
SQLTriggerSchema class, 294
SQLUpdateEvent class, 294
SQLViewSchema class, 294
Start Application After Installation checkbox (Installing Application progress dialog), 164
startHelloTimer() function, 79-80
status bar notifications, creation, 315-317
statusChanged function, 83
storage directories, 226
storing
encrypted data, 301-306
clearing password field, 307
clearing username field, 307
logins, 307
local data, 287
local shared objects, 289-291
local SQL databases, 292-297
writing data to file systems, 288-289
SWF files (Flex-based AIR applications), 66
System Tray icon menus, 266
system tray icons, creating, 323-325
systemChrome node (XML), 66
Taskbar (Windows), highlighting, 318
testing
AIR applications (Flex Builder 3), 48
HTML-based AIR applications, 99-100
TextPad, Dreamweaver CS3 versus, 114
Timer class, 217
timer functions (Flex-based AIR applications), 79-80
Timestamp check box (Flash CS3), 120
timestamping, 17
AIR codes, 308-309
GeoTrust, 18
timestamping servers, 373
title node (XML), 66
toast messages, creating, 319-323
ToolTips (Flex-based AIR applications), 86-87
transparent node (XML), 66
troubleshooting application descriptor files, 63
Updated Adobe AIR dialog, 163
updates
to AIR applications
manual updates, 380-381
remote updates, 382-383
Updater class, 377-378, 381
application descriptor files, 68
URLMonitor class, 282-284
Use Custom UI for Updates check box (Flash CS3), 118
user interfaces
application descriptor files, 68
Flex-based AIR applications, 86-87

How can we make this index more useful? Email us at indexes@samspublishing.com
user notifications

Dock icon (Mac OS), bouncing, 325-327
status bar notifications, creating, 315-317
system tray icons, creating, 323-325
toast messages, creating, 319-323
Windows Taskbar, highlighting, 318
username field (encrypted data storage), clearing, 307

validation coding (application build process), 349
Variables view (Flex Builder Debugger), 146
VeriSign, timestamping, 18
version node (XML), 65
visible node (XML), 67
visibleBounds property, 199-207

web pages, AIR application installations via, 166-170
web resources for developers, 397-398
Adobe AIR Developer Center, 388-390
Adobe AIR Marketplace, 391
Adobe XML News Aggregator, 390
Air forums, 388
web services, leveraging, 336-337
width node (XML), 67
window menus, 262-263
WindowedApplication tag (Flex-based AIR applications), 74
Windows
AIR runtime installations, 19-20
suggested minimum configurations, 18-19
testing, 19
AIR SDK installations, 22
windows
closing via close() method, 189
creating
mx.core.Window class, 187-189
NativeWindow class, 183, 185-187
custom windows, creating, 190-195
screens versus, 199
Windows taskbar, highlighting, 318
wizards
AIR Project Wizard (Aptana Studio), 94-97
Export Adobe AIR Package wizard (Aptana Studio), 115
Export Release Build Wizard (Flex Builder 3), 87-89
New Flex Project Wizard (Flex Builder 3), 57-58
writeBoolean method, 231
writeByte method, 231
writeBytes method, 231
writeDouble method, 231
writeFloat method, 231
writeInt method, 231
writeMultiByte method, 231
writeObject method, 231
writeShort method, 231
writeUnsignedInt method, 231
writeUTF method, 231
writeUTFBytes method, 231

x node (XML), 67
XML
application descriptor files. See application descriptor files
applicationComplete tag, 74
copyright node, 65
creationComplete tag, 74
description node, 65
docfile node, 66
title node, 66
version node, 65
x node, 67
y node, 67

y node (XML), 67