Advanced Java Application Development for the BlackBerry Smartphone

BlackBerry Academic Program
Prior to subscribing for, installing or using any Third Party Products and Services it is your responsibility to ensure that your airtime service provider has agreed to support all of their features. Some airtime service providers may not offer Internet browsing functionality with a subscription to BlackBerry® Internet Service. Check with your service provider for availability, roaming arrangements, service plans and features. Installation or use of Third Party Products and Services with RIM's products and services may require one or more patent, trademark, copyright or other licenses in order to avoid infringement or violation of third party rights. You are solely responsible for determining whether to use. Third Party Products and Services and if any third party licenses are required to do so. If required you are responsible for acquiring them. You should not install or use Third Party Products and Services until all necessary licenses have been acquired. Any Third Party Products and Services that are provided with RIM's products and services are provided as a convenience to you and are provided "AS IS" with no express or implied conditions, endorsements, guarantees, representations or warranties of any kind by RIM and RIM assumes no liability whatsoever, in relation thereto. Your use of Third Party Products and Services shall be governed by and subject to you agreeing to the terms of separate licenses and other agreements applicable thereto with third parties, except to the extent expressly covered by a license or other agreement with RIM.

Certain features outlined in this documentation require a minimum version of BlackBerry® Enterprise Server, BlackBerry® Desktop Software, BlackBerry® Device Software and/or BlackBerry Handheld Software and may require additional development or Third Party Products and Services for access to corporate applications.

Certain products mentioned in this documentation include software developed by the Apache Software Foundation (http://www.apache.org/) and/or licensed pursuant to Apache License, Version 2.0 (http://www.apache.org/licenses/). For more information, see the NOTICE.txt file included with the software. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The terms of use of any RIM product or service are set out in a separate license or other agreement with RIM applicable thereto. NOTHING IN THIS DOCUMENTATION IS INTENDED TO SUPERSEDE ANY EXPRESS WRITTEN AGREEMENTS OR WARRANTIES PROVIDED BY RIM FOR PORTIONS OF ANY RIM PRODUCT OR SERVICE OTHER THAN THIS DOCUMENTATION.
Chapter 1
Introduction to advanced mobile application development for the BlackBerry smartphone

Objectives

- Discuss mobile application development for BlackBerry® smartphones
- Describe advanced application design considerations for BlackBerry smartphones
- Describe the function of the BlackBerry® Java Plug-in for Eclipse®
- Describe troubleshooting steps for BlackBerry smartphone mobile applications
- Discuss deployment options for mobile applications
- Discuss public key license requirements

This chapter outlines design considerations, software tools, and deployment options for advanced application development for the BlackBerry smartphone. The chapter reviews the BlackBerry Java Plug-in for Eclipse, and how to use it, as well as how to troubleshoot the BlackBerry Java Plug-in for Eclipse.
What is advanced mobile application development?

This textbook describes how to create mobile applications with advanced features, that run on BlackBerry smartphones.

The following topics are discussed in this chapter:

- creating advanced graphical user interfaces
- developing highly secure applications,
- integrating Bluetooth® wireless technology into your applications,
- using server push technology
- using client/server push technology
- using application control
- optimizing your code

The BlackBerry smartphones are manufactured by RIM, and are Java®-based devices: all applications on the smartphones are created using Java. BlackBerry smartphones support MIDP 2.0/CLDC 1.1.

The Java APIs on BlackBerry smartphones enable you to develop client applications that provide the following capabilities:

- customizable user interfaces
- local data storage on the BlackBerry smartphone
- event listening and system interfaces
- secure wireless transport via HTTP and TCP
- network coverage and seamless roaming support
Introduction to advanced mobile application development for the BlackBerry smart-

Advanced application design considerations

This section describes considerations you must keep in mind when designing mobile applications for the BlackBerry smartphone.

Programming for BlackBerry smartphones

BlackBerry smartphones provide a Java® ME wireless environment that supports client/server applications. You can create BlackBerry Java Applications that provide a sophisticated interface, including advanced options for data entry and searching, and support for multithreading, internationalization, network communication, and local data storage. Applications can communicate with networks using standard TCP and HTTP connections, regardless of the underlying wireless network.

You can also create a BlackBerry Java Application that integrates with core BlackBerry smartphone applications, such as the message list, organizer applications, phone, and browser, for a seamless user experience.

Design principles for BlackBerry smartphones

Design your BlackBerry smartphone applications to balance user experience against battery life. BlackBerry smartphones differ from computers in many ways, and have the following limitations:

- small screen size
- displays a limited number of characters
- displays only one screen at a time
- limited processor speeds
- limited available memory
- limited battery life
- use wireless networks that have a longer latency period than standard LANs

BlackBerry smartphone users use applications differently than they would use applications on a computer. On BlackBerry smartphones, users expect to find information quickly. For example, a CRM system can provide a massive amount of information, but users require only a small amount of that information at one time.

Design your applications for BlackBerry smartphones to be as consistent as possible with the design of other BlackBerry smartphone applications.

Consider the following guidelines:

- Use or extend existing UI components where possible so that your application can inherit the default behavior of the component.
- Follow the standard navigation model as closely as possible so that BlackBerry smartphone users can make full use of the keyboard and trackball.
- Make all actions available from the menu. Verify that the actions available in the menu are relevant to users’ current context.
- Simplify data selection and presentation to display only the information that users need at any moment.
- Display information in a way that makes effective use of the small screen.

Consider using the core applications on the BlackBerry smartphone or the BlackBerry Smartphone Simulator to learn more about the navigation model and best practices for designing your application UI.

### Release cycles and versions

All BlackBerry smartphones include the BlackBerry® Device Software and the BlackBerry® Java® Virtual Machine.

To determine the version of the BlackBerry Device Software installed on a BlackBerry smartphone, in the BlackBerry smartphone Options, click About. You can upgrade the BlackBerry Device Software. For example, you can upgrade a BlackBerry smartphone with BlackBerry Device Software version 4.0 to BlackBerry Device Software version 4.1.

The version of the BlackBerry Device Software installed on a BlackBerry smartphone determines the version of the BlackBerry Java Plug-in for Eclipse that you must use to develop applications that can run on the smartphone. Use a version of the BlackBerry Java Plug-in for Eclipse that matches the lowest level of BlackBerry Device Software on which you want to run your application. If you use a version of the BlackBerry Java Plug-in for Eclipse that is more recent than the BlackBerry Device Software version, application errors can occur.
BlackBerry Java Plug-in for Eclipse

The BlackBerry Java Plug-in for Eclipse is a fully integrated development and simulation environment for building a BlackBerry Java Application for BlackBerry smartphones. Using the BlackBerry Java Plug-in for Eclipse, you can build applications using the Java programming language and the extended Java APIs for BlackBerry smartphones.

When creating your mobile application, you can use existing development tools to optimize code for efficient performance, provide network and data connectivity, provide security, and ensure compatibility with a range of BlackBerry smartphones.

The BlackBerry Java Plug-in for Eclipse includes the following development tools:

- BlackBerry IDE
- BlackBerry Smartphone Simulator
- Java ME APIs and BlackBerry APIs
- sample applications

The BlackBerry Java Plug-in for Eclipse includes a full suite of editing and debugging tools that are optimized for the development of a BlackBerry Java Application. The BlackBerry Smartphone Simulator is designed to simulate user interfaces and user interaction, network connections, email services, and wireless data synchronization.

The BlackBerry Java Development Environment Component Package includes the following development tools for development within third-party IDEs such as NetBeans® or Eclipse®:

- **RAPC**: Use this command prompt compiler to compile .java files and .jar files into .cod files that you can run in the BlackBerry Smartphone Simulator or on a BlackBerry smartphone.
- Javaloader: Use this tool to add or update an application on a BlackBerry smartphone for testing, and to view information about application .cod files.
- BlackBerry Signature Tool: Use this tool to send code signature requests to the BlackBerry® Signing Authority Tool.
- Preverify Tool: Use this tool to partially verify your classes before you load your application onto a BlackBerry smartphone.
- **JDWP**: Use this tool to debug applications using third-party integrated development environments.
- MDS-CS Simulator: Use this tool to debug applications using simulated mobile data system connections.
- ESS: Use this tool to debug applications using simulated email messages.
Java ME and Java APIs for BlackBerry smartphone

Java ME is an industry standard platform that defines common Java APIs for different types of wireless and embedded devices. A Java ME application on a BlackBerry smartphone runs in the BlackBerry JVM, which provides all of the runtime services to the applications and performs functions such as typical memory allocations, security checks, and garbage collection.

The Java ME MIDP standard addresses the API and BlackBerry JVM needs of a constrained wireless device with a graphical user interface.

The BlackBerry smartphone supports the Java ME MIDP standard as defined in JSR 118. The Java ME MIDP standard provides a core group of Java APIs that any BlackBerry smartphone can support, regardless of the underlying operating system. You can use the MIDP standard API to build a Java application that runs on many different types of BlackBerry smartphones.

Support for standard Java APIs

The BlackBerry smartphone and the BlackBerry Java Plug-in for Eclipse support the Java ME MIDP standard, which provides a core group of Java APIs that you can use to develop wireless device applications. The BlackBerry smartphone and the BlackBerry Java Plug-in for Eclipse support the following JSRs (some require specific versions of the BlackBerry Device Software):

- JSR 30: Connected Limited Device Configuration version 1.0 (supported on BlackBerry smartphones with BlackBerry Device Software version 4.0 or earlier)
- JSR 37: Mobile Information Device Profile version 1.0 (supported on BlackBerry smartphones with BlackBerry Device Software version 4.0 or earlier)
- JSR 75: Portable Optional Packages for the J2ME Platform support for the PIM APIs only and the File Connection API for Java ME (supported on BlackBerry smartphones with BlackBerry Device Software version 4.2 or later)
- JSR 82: Java APIs for Bluetooth wireless devices
- JSR 118: Mobile Information Device Profile version 2.0
- JSR 120: Wireless Messaging API (WMA) version 1.1
- JSR 135: Mobile Media APIs (MM API) version 1.1
- JSR 139: Connected Limited Device Configuration version 1.1
- JSR 172: J2ME Web Services
- JSR 177: Security and Trust Services API for J2ME (SATSA)
- JSR 179: Location API for Java ME
- JSR 185: Java Technology for the Wireless Industry (JTWI)
- JSR 205: Wireless Messaging API 2.0
- JSR 211: Content Handler API
- JSR 226: Scalable 2D Vector Graphics API for Java ME
- JSR 238: Mobile Internationalization API
Support for Java API extensions

<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Interface APIs</td>
<td>Create screens, menu items, and all the components of the user interface.</td>
</tr>
<tr>
<td>Persistent Data Storage APIs</td>
<td>Store custom data locally within your application.</td>
</tr>
<tr>
<td>Networking and I/O APIs</td>
<td>Establish network connections and read or write data to a server-side application.</td>
</tr>
<tr>
<td>Event Listeners</td>
<td>Respond to BlackBerry smartphone user or system initiated events on a BlackBerry smartphone.</td>
</tr>
<tr>
<td>Application Integration APIs</td>
<td>Integrate your application with the existing BlackBerry smartphone email services, phone, calendar, contacts, browser, camera, media player, and task list applications.</td>
</tr>
<tr>
<td>Additional Utilities</td>
<td>Perform operations including data encryption and compression, XML parsing, Bluetooth connectivity, and location-based services.</td>
</tr>
</tbody>
</table>

BlackBerry smartphone solutions

BlackBerry smartphone users can use either the BlackBerry* Enterprise Server or the BlackBerry* Internet Service, or they can use both on the same BlackBerry smartphone. Be sure that you understand the differences between the BlackBerry Enterprise Server and the BlackBerry Internet Service (and which types of users you plan to support), because the user types you choose can impact which modes of transport you use and how you manage data synchronization.

BlackBerry Enterprise Solution

The BlackBerry Enterprise Server is part of the BlackBerry* Enterprise Solution, and operates behind an organization's firewall. The BlackBerry Enterprise Server provides a wireless gateway for BlackBerry smartphone users in the organization to access corporate email servers and organizer data, and provides the following services:

- data encryption and compression
- BlackBerry smartphone management and monitoring utilities
- simplified application provisioning
- authenticated gateway for intranet access from a BlackBerry Java Application
BlackBerry MDS

To allow a BlackBerry Java Application access to resources behind an organization's firewall, the BlackBerry Enterprise Server includes the BlackBerry® Mobile Data System. The BlackBerry MDS provides HTTP and socket connections for BlackBerry Java Applications, which allow BlackBerry smartphones to communicate with application and web servers behind an organization's firewall without additional VPN software. Applications that send data using the BlackBerry Enterprise Server as a gateway can capitalize on the simplified enterprise connectivity, data encryption and compression, and wireless network-independence that the BlackBerry Enterprise Solution offers. BlackBerry MDS also provides an open interface, allowing server-side applications behind an organization's firewall to push content to applications on BlackBerry smartphones.

BlackBerry Internet Service

BlackBerry smartphone users who are not associated with a BlackBerry Enterprise Server can use the BlackBerry Internet Service.

The BlackBerry Internet Service is an email and Internet service for BlackBerry smartphones that is designed to provide users with automatic delivery of email messages, wireless access to email attachments, and access to Internet content.

The BlackBerry Internet Service includes support for direct HTTP and TCP/IP Internet connectivity from a third-party BlackBerry Java Application.

You must be part of the RIM ISV Alliance program to use the BlackBerry Internet Service, and your application must be approved for access.

BlackBerry Java Plug-in for Eclipse features

The BlackBerry Java Plug-in for Eclipse extends the Eclipse framework and enables you to develop Java applications optimized for BlackBerry smartphones. Within the framework of the Eclipse IDE, the BlackBerry Java Plug-in for Eclipse provides tools to design, debug, optimize, and localize mobile applications. The BlackBerry Java Plug-in for Eclipse also provides the ability to simulate the integration between mobile applications, BlackBerry smartphone services, and external data sources.

You can create efficient mobile applications using existing development tools to optimize code for efficient performance, network and data connectivity, highly secure encryption, and compatibility with a range of BlackBerry smartphones.
The BlackBerry Java Plug-in for Eclipse version 1.0 is delivered as a standard plug-in to Eclipse, enabling you to create BlackBerry applications from within the Eclipse framework.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Benefit</th>
</tr>
</thead>
<tbody>
<tr>
<td>support for multiple BlackBerry Device Software versions</td>
<td>You can change the target BlackBerry Device Software version so that you can develop applications that make use of various APIs and BlackBerry smartphone capabilities.</td>
</tr>
<tr>
<td>code assist integration</td>
<td>You can make use of code assist that adjusts to available APIs based on the target BlackBerry Device Software version.</td>
</tr>
<tr>
<td>extended Java debugging</td>
<td>You can debug BlackBerry Applications using object, memory, and profiling views.</td>
</tr>
<tr>
<td>preprocessing support</td>
<td>You can specify preprocessing directives using fully integrated Eclipse compiler support.</td>
</tr>
<tr>
<td>integration with BlackBerry Smartphone Simulators</td>
<td>Development tools are integrated with the simulators, making it easier for you to access the simulators.</td>
</tr>
</tbody>
</table>

**Minimum system requirements**

Make sure that your system meets minimum hardware and software requirements. For more information about minimum system requirements, visit the BlackBerry Developer Zone.

**Troubleshooting**

Several methods are available to search for and correct errors in your application, using either the BlackBerry Smartphone Simulator or a BlackBerry smartphone.

In most situations, you can debug your application on the BlackBerry Smartphone Simulator, as it provides better tools for identifying issues. You can identify certain issues more easily by debugging on the BlackBerry smartphone; for example: making network connections through a wireless service provider WAP gateway.
1. Which of the following advanced mobile application features are discussed in this textbook?
   A. creating advanced interfaces
   B. developing highly secure applications,
   C. integrating Bluetooth wireless technology into your applications,
   D. using server push technology
   E. using application control
   F. all of the above
   G. none of the above

2. Which of the following provides a full suite of editing and debugging tools that are optimized for the development of a BlackBerry Java Application.
   A. The BlackBerry Smartphone Simulator
   B. The BlackBerry Java Plug-in for Eclipse
   C. The BlackBerry smartphone

3. Which of the following provides HTTP and socket connections for BlackBerry Java Applications?
   A. The BlackBerry Smartphone Simulator
   B. The BlackBerry Java Plug-in for Eclipse
   C. The BlackBerry MDS
   D. Java ME
Introduction to advanced mobile application development for the BlackBerry smart-

Answers

1. F
2. B
3. C
Using the BlackBerry Java Plug-in for Eclipse

Use the procedures in this section to manage the BlackBerry Java Plug-in for Eclipse.

Configuring the BlackBerry Java Plug-in for Eclipse

1. In the Eclipse workspace, click Window > Preferences.
2. Expand BlackBerry JDE.
3. Select Installed Components.
4. In the Components section, select an installed component from the drop-down list.
5. Click OK until the Preferences window disappears.

The BlackBerry Java Plug-in for Eclipse supports preprocessing, which provides a conditional compilation capability, allowing you to more easily create versions of your application for each BlackBerry smartphone. You can turn on preprocessing for your applications by updating the Eclipse configuration file.

To enable application preprocessing

1. Navigate to Eclipse/configuration.
2. Open the config.ini file for editing.
3. Add the following line:
   
   `osgi.framework.extensions=net.rim.eide.preprocessing.hook`

4. Save the config.ini file.

If you turn on preprocessing after a build, you must remove the project from the Project menu before you build the project again.

Note:

To create a BlackBerry application for a specific version of BlackBerry Device Software, you must use a matching version of the Eclipse Software.

To create an application for a specific version of the BlackBerry Device Software, perform the following steps:

1. Open the Eclipse workspace.
2. On the Window menu, select Preferences.
3. Expand BlackBerry JDE.
4. Select **Installed Components**.

5. In the **Components** field, select a Java SDK plug-in component.

6. Click **OK**.

**Removing the BlackBerry Java Plug-in for Eclipse**

If you used the installer for the BlackBerry Java Plug-In for Eclipse, perform the following steps to remove the plug-in:

1. Close any instances of Eclipse.

2. In the file system, navigate to the **Eclipse installation directory**.

3. Navigate to the **plugins** folder.

4. Remove the following files and folders:
   - net.rim.eide.bootstrapper_*.jar
   - net.rim.eide.componentpack_*
   - net.rim.eide.doc_*.jar
   - net.rim.eide.preprocessing.hook_*.jar
   - net.rim.eide_*.jar

5. Navigate to the **features** folder.

6. Remove the following folders:
   - net.rim.EclipseJDE_ *
   - net.rim.eide.feature.componentpack*

   If you enabled preprocessing, complete steps 7-9.

7. Navigate to the **configuration** folder.

8. Open the config.ini file for editing, and remove the following line:
   
   osgi.framework.extensions=net.rim.eide.preprocessing.hook

9. Save the config.ini file.
Methods for testing applications

You can use the following methods to test BlackBerry smartphone applications:

- testing applications on the BlackBerry Smartphone Simulator
- testing applications on the BlackBerry smartphone

Testing applications on a BlackBerry Smartphone Simulator

The BlackBerry Smartphone Simulator permits you to run BlackBerry smartphone applications on your computer. The BlackBerry Smartphone Simulator includes the BlackBerry smartphone applications that are typically available on BlackBerry smartphones and permits you to load and test your own applications. You can simulate and test various connectivity and state changes using the BlackBerry Smartphone Simulator.

The BlackBerry Smartphone Simulator runs the same Java code that the BlackBerry smartphone runs, so the BlackBerry Smartphone Simulator provides an accurate environment for testing how applications function on a BlackBerry smartphone. The BlackBerry Java Plug-in for Eclipse includes current versions of the BlackBerry Smartphone Simulator.

For more information about using the BlackBerry Smartphone Simulator, see *The BlackBerry Smartphone Simulator Development Guide*.

Testing applications on a BlackBerry smartphone

After you test your application on the BlackBerry Smartphone Simulator, you can install your application on a BlackBerry smartphone. If your application uses signed APIs, you require code signing keys. After you install the application on the BlackBerry smartphone, you can open the application and test its functionality and performance. For debugging purposes, you can attach your BlackBerry smartphone to the BlackBerry IDE and use the debugging tool to step through your application code. The BlackBerry IDE can be useful if you are trying to identify problems that are difficult to simulate, such as problems with networking or with Bluetooth wireless technology.
Introduction to advanced mobile application development for the BlackBerry smart-

Quiz

1. The BlackBerry Java Plug-in for Eclipse supports _____________________, which provides a conditional compilation capability, allowing you to more easily create versions of your application for each BlackBerry smartphone? *Fill in the blank.*

2. The ________________________________ includes the BlackBerry smartphone applications that are typically available on BlackBerry smartphones and permits you to load and test your own applications. *Fill in the blank.*
Chapter 1

Answers

1. preprocessing

2. BlackBerry Smartphone Simulator
Deploying mobile applications

There are a number of methods you can use to deploy applications on the BlackBerry smartphones.

<table>
<thead>
<tr>
<th>Deployment option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlackBerry® Desktop Manager</td>
<td>The BlackBerry smartphone user downloads the application to a PC, and then installs it using a USB connection.</td>
</tr>
<tr>
<td>Javaloader</td>
<td>The BlackBerry smartphone user downloads the application to a PC, and then installs it using a USB connection. Javaloader is for developer use only, and is not intended for end users.</td>
</tr>
<tr>
<td>BlackBerry Application Web Loader</td>
<td>The BlackBerry smartphone user downloads the application using a web browser, and uses BlackBerry Application Web Loader to install the application over a USB connection.</td>
</tr>
<tr>
<td>Over the air</td>
<td>The BlackBerry smartphone user installs the application directly from your web site.</td>
</tr>
<tr>
<td>BES administration</td>
<td>The server pushes the application to the BlackBerry smartphone.</td>
</tr>
<tr>
<td>BlackBerry App World™ storefront</td>
<td>Your application is published by RIM, and is accessible to BlackBerry smartphone users through the RIM App World web site.</td>
</tr>
<tr>
<td>Virtual Preload</td>
<td>When a smartphone registers on a network, the carrier sends application icons, which then appear on the BlackBerry smartphone users’ home page. BlackBerry smartphone users can click on the icons, and the applications download automatically.</td>
</tr>
<tr>
<td>Application Center</td>
<td>BlackBerry smartphone users can use Application Center to discover and download applications, or to receive notifications of new applications that are available for download. Application Center is available on BlackBerry Device Software version 4.7 and higher.</td>
</tr>
</tbody>
</table>

Before you deploy

Before you deploy your application, consider the following:

- If you are using APIs that require signing, make sure that you sign your application.
• Verify whether your application requires a specific Trust Level be configured in the BlackBerry smartphone Application Control Permissions. Smartphone users can configure Application Control Permissions to deny access to resources your application needs, such as Internet connectivity or keypad input.

• Verify whether IT Policies can interfere with the operation of your application. IT Policies take precedence over Application Control Policies, so even if a BlackBerry smartphone user allows a feature, the feature can still be blocked if IT policy disables it.

**BlackBerry Desktop Manager**

BlackBerry Desktop Manager is an application that BlackBerry smartphone users can install on their Windows or Macintosh® computer. Users can then use the BlackBerry Desktop Manager to install applications on the BlackBerry smartphone using a USB cable.

![Figure 2.1 BlackBerry Desktop Manager](image)

**Preparing an application for deployment using BlackBerry Desktop Manager**

To prepare your application to be deployed using BlackBerry Desktop Manager, you must create two files using Eclipse.

• **COD.** Each time you run or debug your application, Eclipse builds a new version of the COD file.
Introduction to advanced mobile application development for the BlackBerry smart- 

- ALX. To create an ALX file in Eclipse, right-click on your project, and then in the shortcut menu select Generate ALX.

Eclipse saves both the ALX and the COD files in your Workspace folder; you must distribute both files to BlackBerry smartphone users.

**Loading an application using BlackBerry Desktop Manager**

1. Launch the BlackBerry Desktop Manager.
2. Click Application Loader
3. On the taskbar, click Start > Add/Remove Applications
4. Click Browse, and then navigate to the ALX file for the application you want to install.
5. Click Next.
6. Click Finish

**Javaloader**

Javaloader is a command prompt utility that provides low level control over the smartphone.

**Caution:**
Javaloader is a developer-only deployment method, and is not designed to distribute applications to BlackBerry smartphone users. Improper use can cause data loss, or can prevent the smartphone from working.

Javaloader is bundled with the BlackBerry Java Plug-in for Eclipse; you can launch it from the Eclipse/bin folder, or from the Start menu.

**Note:**
Add the path to the Javaloader to your system path, so you can more readily access Javaloader from any folder.

**Loading an application using Javaloader**

1. At the command prompt, type javaloader –u load yourfilename.cod.
2. Press ENTER.
   If the file already exists on the smartphone, the existing file is overwritten.
   The smartphone restarts.

**Viewing the list of installed applications**

1. At the command prompt, type `javaloader –u dir`.
2. Press ENTER.

**Removing an application**

1. At the command prompt, type `javaloader –u erase yourfilename.cod`.
2. Press ENTER.

You can use Javaloader to perform various other tasks, including debugging, capturing screen shots, and copying applications from a BlackBerry smartphone to your PC. For more information about these options, see Javaloader help.

You can also remove an application while it is running.

1. At the command prompt, type `javaloader -u erase -f filename.cod`.
2. Press ENTER.

The -f switch removes the application even if it is running. This can cause the BlackBerry smartphone to restart.

**Viewing Javaloader help**

1. At the command prompt, type `javaloader`.
2. Press ENTER.
BlackBerry Application Web Loader

The BlackBerry smartphone user can download the application using a web browser, and use BlackBerry Application Web Loader to install the application over a USB connection.

![Diagram of BlackBerry Application Web Loader](image)

**Figure 2.2** Preparing an application for deployment using Web Loader

To deploy your application using BlackBerry Application Web Loader, you must configure a web site where BlackBerry smartphone users can access the files.

1. Download the BlackBerry Application Web Loader from the BlackBerry web site, and install the BlackBerry Application Web Loader on your web server. For more information about how to install the BlackBerry Application Web Loader on your web server, visit: www.blackberry.com/developers/downloads/webloader.

2. Load the following files onto your web server:
   - .jad file
   - .cod file
   - AxLoader.cab

3. Associate the BlackBerry Application Web Loader with the application .jad file.

Loading an application using the BlackBerry Application Web Loader

BlackBerry smartphone users must have ActiveX installed to use the BlackBerry Application Web Loader.

1. Open Internet Explorer 5.0 or later, and load the web site.
2. Connect the BlackBerry smartphone using the supplied USB cable.

3. Type the BlackBerry smartphone password (if configured).

4. Click **Load**.

![Figure 2.3 Loader page for Web Loader](image)

For sample Application Web Loader code, see “Appendix A: Sample BlackBerry Application Web Loader HTML code” on page 34.

**Over the air**

Use over the air distribution to enable BlackBerry smartphone users to download and install the application from your web site using the BlackBerry smartphone wireless connection.
When you deploy your application using over the air, BlackBerry smartphone users can use BlackBerry Browser to view information about the application, download and install the application, and configure application permissions.

![GoogleMaps.jad](image)

*Figure 2.4 Over the air distribution*

### Preparing your application for over the air deployment

To use over the air deployment, you must first create a web site where BlackBerry smartphone users can download your application.

1. Load the following files onto your web server:
   - Yourapplication.cod
   - Yourapplication.jad

2. On your web server, configure three MIME types:
   - .jad files: `text/vnd.sun.j2me.app-descriptor`
   - .cod files: `application/vnd.rim.cod`
   - .jar files (optional): `application/java-archive`

3. Optionally, you can detect what smartphone is connected so that you can provide a unique application for each model of BlackBerry smartphone. You can detect what BlackBerry smartphone is connected by checking HTTP user-agent or profile header.

   **User-Agent:**
   `BlackBerry8320/4.2.2 Profile/MIDP-2.0 Configuration/CLDC-1.1 VendorID/100 Profile: http://www.blackberry.net/go/mobile/profiles/uaprof/8320/4.2.2.rdf`
**BlackBerry Enterprise Server**

A BlackBerry Enterprise Server can push applications to connected BlackBerry smartphones. The BlackBerry Enterprise Server provides a web interface that you can log in to from any computer and deploy applications to specific BlackBerry smartphone users or groups. You can also use the BlackBerry Enterprise Server to control permissions.

The management and administration of the BlackBerry Enterprise Server server is beyond the scope of this discussion; for more information, visit na.blackberry.com/eng/services/server.

**BlackBerry App World**

You can use BlackBerry App World to distribute your applications to BlackBerry smartphone users who visit the site.

To publish your application in BlackBerry App World, you must first create a vendor account, for which you must have a PayPal® account. For more information, visit na.blackberry.com/eng/developers/app-world.jsp.

RIM must approve applications submitted for BlackBerry App World before they can be published on BlackBerry App World.

Blackberry App World supports BlackBerry smartphones running OS version 4.2.0 or higher; therefore, BlackBerry App World is supported on the following BlackBerry smartphones:

- BlackBerry® Bold™ 9000 smartphone
- BlackBerry® Storm™ smartphone
- BlackBerry® Pearl™ Flip Series
- BlackBerry® Curve™ 8300 Series
- BlackBerry® Curve™ 8900 smartphone
- BlackBerry® 8800 Series
- BlackBerry® Pearl™ Series

**Virtual Preload**

You can work with carriers to have them virtually preload applications on BlackBerry smartphones. The carrier does this by sending icons to the Home screen of users whose BlackBerry smartphones register with the network. When the BlackBerry smartphone user clicks the icons, the smartphone downloads the associated application.

Contact your carrier to find out more about deploying your applications using Virtual Preload.
**Instant Load**

Instant Load deployment is similar to Virtual Preload, however, no user action is required. Instant Load applications are automatically and transparently installed when a BlackBerry smartphone activates on the network.

Contact your carrier to find out more about deploying your applications using Instant Load.

**Application Center**

Application Center is available on BlackBerry smartphones with BlackBerry Device Software version 4.7 and higher. BlackBerry smartphone users can use it to discover and download applications, or to receive notifications of new applications that are available for download.

![Application Center](image)

*Figure 2.5 Application Center*

Contact your carrier to find out more about deploying your applications using Application Center.
1. Which of the following are possible deployment methods for BlackBerry smartphone applications? *Choose all that apply.*
   A. Over the Air
   B. BlackBerry App World™ storefront
   C. CD by Mail
   D. BlackBerry Desktop Manager
   E. all of the above
   F. none of the above

2. Which of the following is correct?
   A. You can use Javaloader to develop scripts that you distribute to BlackBerry smartphone users. These scripts enable users to automatically download applications.
   B. Javaloader provides utilities to load your Java COD files into various versions of the BlackBerry simulator for testing.
   C. Javaloader is a developer-only deployment method, and is not designed to distribute applications to BlackBerry smartphone users.

3. Which of the following can you use to enable BlackBerry smartphone users to download your application directly from your web site. *Choose all that apply.*
   A. BlackBerry Enterprise Server
   B. BlackBerry Application Web Loader
   C. Over the air
   D. BlackBerry virtual web deployment
   E. none of the above
Answers

1. E
2. C
3. A, B and C
Public key license requirements

You must indicate if your application requires a license key when you submit the application to BlackBerry App World™. The license key models supported include Static, Single, Pool, or Dynamic.

- In the Static model, a license key is not required to run the application. With the Single model, a single key is used to unlock the application.
- With the Pool model, you must send a pool of serial numbers that are handed out one at a time to each end user.
- With the Dynamic model, BlackBerry App World server performs an HTTP connection to your web site to generate a license key. Your web site can generate a dynamic license key based on your predetermined algorithm; for example, based on the user email address.

In all cases, the license key is provided to the user by the BlackBerry App World client.

Dynamic licensing

1. In the BlackBerry App World vendor portal, click the Licensing tab.
2. Choose the license type (Free, Paid, or Try & Buy).
3. If you chose Free, skip to step 4.
   OR
   If you chose either Paid or Try&Buy, make a selection in the License Key Model (Static, Single, Pool, or Dynamic).
   Static means that no license key is needed to run the application. For trials, a check box on the release indicates that the software is a trial version.
   Single means you enter a single key into the portal to unlock the application.
   Pool means that you send RIM a pool of serial numbers that are handed out one at a time to each customer.
   Dynamic, means that the App Store server performs an HTTP connection to your web site when it is time to generate a license key, your web site generates a dynamic license key based on a predetermined algorithm, for example based on the PIN number, phone number or email address.
4. Upload your application with a dynamic license model, and provide a web address where the license is generated.

When a user purchases your application, the following events take place:

1. App Store collects information from the end user.
2. App Store server contacts the vendor server to obtain a License Key.
3. The License Key is generated.
4. The License Key is stored in the App Store server.
5. User downloads the application using the App Store Client.

6. The application is registered based on Key from App Store server.

**Purchase requests**

When a purchase request occurs, the vendor portal sends the following request to the HTTP URL you provided when you uploaded your application:

```
POST /pathfromdeveloper HTTP/1.1
Content-Type: application/www-url-encoded
Content-Length: 120
Host: hostfromdeveloper

PIN=12341234&email=customeremail@email.com&product=product&version=1.2&transactionid=123&test=false
```

The vendor portal expects the following response:

```
HTTP/1.1 200 OK
Content-Type: application/www-url-encoded
Content-Length: 20
key=ABCDEFGHIJK
```

The PIN is passed in hexadecimal format.
1. BlackBerry App World supports which of the following license key models? Circle all that apply.
   
   A. Static
   
   B. Symmetric
   
   C. Single
   
   D. Pool
   
   E. Asymmetric
   
   F. Dynamic
   
   G. BlackBerry App World does not support license keys.
Answers

A, C, D, F
The BlackBerry smartphones are manufactured by RIM, and are Java-based devices. All applications on BlackBerry smartphone are created using Java. BlackBerry smartphones support MIDP 2.0/CLDC1.1.

BlackBerry smartphones provide a Java ME wireless environment that supports client/server applications. You can create BlackBerry Java Applications that provide a sophisticated interface, including advanced options for data entry and searching, and support for multithreading, internationalization, network communication, and local data storage. Applications can communicate with networks using standard TCP and HTTP connections, regardless of the underlying wireless network.

You can also create a BlackBerry Java Application that integrates with core BlackBerry smartphone applications, such as the message list, organizer applications, phone, and browser, for a seamless user experience.

The BlackBerry Java Plug-in for Eclipse is a fully integrated development and simulation environment for building a BlackBerry Java Application for BlackBerry smartphones. Using the BlackBerry Java Plug-in for Eclipse, you can build applications using the Java programming language and the extended Java APIs for BlackBerry smartphones.

When creating your mobile application, you can use existing development tools to optimize code for efficient performance, provide network and data connectivity, provide security, and ensure compatibility with a range of BlackBerry smartphones.

The BlackBerry Java Plug-in for Eclipse includes a full suite of editing and debugging tools that are optimized for the development of a BlackBerry Java Application. The BlackBerry Smartphone Simulator is designed to simulate user interfaces and user interaction, network connections, email services, and wireless data synchronization.

You can deploy your application using several methods, including over the air deployment, where a BlackBerry smartphone user goes to your web site and installs the application from it, BlackBerry Desktop Manager or Javaloader deployment, where the user downloads and installs the application using a PC and USB connection, or using BlackBerry Application Web Loader, which installs the application over a USB connection from the web site. Alternatively, you can use BlackBerry Enterprise Server administration to push the application to the BlackBerry smartphone, or distribute your application through the BlackBerry App World, or using Virtual Preload.
1. List the development tools that are part of the BlackBerry Java Plug-in for Eclipse.

2. The following Java APIs are supported for BlackBerry smartphone mobile application development. Describe each one.
   - User Interface APIs
   - Persistent Data Storage APIs
   - Networking and I/O APIs
   - Event Listeners
   - Application Integration APIs
   - Additional Utilities

3. List, and briefly describe, at least four methods you can use to deploy your BlackBerry smartphone applications.

4. Briefly describe the benefits of using BlackBerry App World to deploy your applications.
Appendix A: Sample BlackBerry Application
Web Loader HTML code

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
<head>
<title>Loader page for Hello World Application</title>
</head>
<body bgcolor="#FFFFFF" lang=EN-CA link=blue vlink=navy alink="#FF0000"
onload="update();">
<p><h1>Hello World Application</h1></p>
<p>A simple application that displays "Hello World!" on the screen.</p>
<hr>
<center>
<div style="table:margin-left:1.5em; font:x-small 'Tahoma';
margin-right:1.5em; align:center">
<table border="0" cellpadding="0" cellspacing="0" width="100%">
<tbody>
<tr><td align="center">
<h2>Loader page for Hello World Application</h2>
</td></tr>
</tbody></table>
</div>
</center>
Introduction to advanced mobile application development for the BlackBerry smart-

</style>

<!-- Use this element to specify the required version of the BlackBerry Application Web Loader -->

<OBJECT ID="AxLoader"
CLASSID="CLSID:DAF7E6E6-D53A-439a-B28D-12271406B8A9"
CODEBASE='../AxLoader.cab#version=1,0,0,16'>
</OBJECT>

<div align='center' style='border=2px solid gray'>
<br/>
<b><h1>Hello World Application</h1></b>
<h2 id='title'></h2>
<div id='message' align='center'>
<br/>
</div>
</div>
<br/>
<br/>
<br/>
<br/>
<div id='sections'>
<div id='password' align='center' style="display:'none'">
<!-- Use this element to provide a user password to the BlackBerry Application Web Loader. The web page does not handle the password directly. -->

<OBJECT ID="AxLoaderPassword" CLASSID="CLSID:DAF7E6E7-D53A-439a-B28D-12271406B8A9"></OBJECT>
<br/>
<br/>
<a class='bd' onmousedown="providePassword()"
onkeydown="if(window.event.keyCode==13){providePassword();}">Enter</a>
</div>
</div>
<br/>
<div id='load' align='center' style="display:none">
Your BlackBerry smartphone may reset after loading.

Load

Again

```
function AxLoader::progressUpdate() {
    progress_bar.innerText = AxLoader.progress.toString() + "%";
    progress_bar.style.width = AxLoader.progress * 120 / 100;
}

function AxLoader::stateChanged() {
    update();
}

function AxLoaderPassword::onEnter() {
```
providePassword();
}

function releaseHold() {
    AxLoader.reset();
}

function providePassword() {
    AxLoader.submitPassword(AxLoaderPassword);
}

function show(section, titletext, html) {
    var count = sections.children.length;
    for (i=0; i<count; i++) {
        element = sections.children(i);
        if (element.id == section) {
            element.style.display = 'block';
        } else {
            element.style.display = 'none';
        }
    }
    title.innerHTML = titletext;
    message.innerHTML = html;
}

function update() {
    debugInfo.innerText = AxLoader.debugInfo;
    var state = AxLoader.state;
    if ("" + state == "undefined") {

show('message', 'Unable to create the loader Active X Control', "Please review your security settings and user privileges on this machine.");
return;
}

switch(AxLoader.state) {
  case 0: // The BlackBerry Application Web Loader cannot find the BlackBerry USB driver.
    show('message', 'No Driver', "The BlackBerry USB driver is not installed.");
    retry.style.display = 'none';
    break;
  case 1: // The BlackBerry Application Web Loader does not detect the BlackBerry smartphone.
    show('message', 'No BlackBerry Found', "To install the Hello World Application, connect your BlackBerry to the USB port on this computer.");
    retry.style.display = 'none';
    break;
  case 2: // The BlackBerry Application Web Loader requires a user password.
    show('password', 'Connected', "Type the BlackBerry smartphone password to continue.<br/>(" + AxLoader.passwordRetries + ") attempts remaining.");
    retry.style.display = 'none';
    AxLoaderPassword.focus();
    break;
  case 3: // The BlackBerry Application Web Loader successfully connected to the BlackBerry smartphone.
stantial.BlackBerryDeviceSoftware versions exist. -->
var version = AxLoader.version;
if (version.indexOf("3.6") != 0 &&
    version.indexOf("3.7") != 0 &&
    version.indexOf("3.8") != 0 &&
    version.indexOf("4.0") != 0 &&
    version.indexOf("4.1") != 0 &&
    version.indexOf("4.2") != 0 &&
    version.indexOf("4.2.1") != 0) {
    show('load', 'Connected', "Click the link to install the Hello World Application on your BlackBerry smartphone.");
    break;
}
show('load', 'Connected', "Click the link to install the Hello World Application on the BlackBerry smartphone (Version " + version + ").");
retry.style.display = 'none';
break;
case 4: // The BlackBerry Application Web Loader is installing an application on a BlackBerry smartphone.
    show('progress', 'Loading', "");
    retry.style.display = 'none';
    break;
case 5: // An error occurred while connecting to or installing on a BlackBerry smartphone.
    var remedy = "An unspecified error has occurred.";
    switch(AxLoader.error) {
        case 1: // ERROR_UNABLE_TO_CONNECT
remedy = "Unable to connect to smartphone. Close any other BlackBerry applications that are running, for example, the BlackBerry Desktop Software. Try again.";

break;

case 2: // ERROR_DEVICE_IS_FULL

remedy = "The smartphone is full. Remove some data and try again.";

break;

case 3: // ERROR_UNABLE_TO_DOWNLOAD

remedy = "Error downloading file. Check your Internet connection and try again.";

break;

case 4: // ERROR_UNABLE_TO_CREATE_LOCAL_COPY

remedy = "Unable to create a local copy of files. Verify that space is available for temporary files on your computer.";

break;

case 5: // ERROR_INVALID_FILE

remedy = "Attempt to load an invalid file. Try again.";

break;

case 6: // ERROR_LOAD_FAILED

remedy = "Error loading file. Try again.";

break;

case 7: // ERROR_OUT_OF_MEMORY

remedy = "Out of Memory. Close some applications and try again.";

break;

} 

show('message', 'Error', remedy);
retry.style.display = 'block';
break;

case 6: // The BlackBerry Application Web Loader successfully installed an application on a BlackBerry smartphone.
    show('message', 'Done', "Hello World Application has been loaded.");
    retry.style.display = 'block';
    break;
}
}

function loadFiles() {

    <!-- Load the appropriate .jad file based on the hardware ID and version number -->

    // loadJad accepts either a relative URL for the .jad file or the actual .jad file contents as a string.

    // The BlackBerry Web Application Loader supports BlackBerry .jad files only.

    Convert MIDlets to .cod files.

    var hwid = AxLoader.hwid;
    if ((hwid & 0x04000000) == 0x04000000) {
        AxLoader.loadJad("helloworld.jad");
    } else {
        AxLoader.load("helloworld_mono.jad");
    }
}

</script>
</div>
</body>
</HTML>