

BlackBerry Push Service SDK

Version: 1.0.0.5

Getting Started Guide

Contents

1	Overview of the BlackBerry Push Service SDK.....	3
2	System requirements: development environment.....	5
3	Planning the installation of the BlackBerry Push Service SDK development environment.....	6
4	Demonstrating the functionality of the BlackBerry Push Service SDK.....	7
5	Prerequisites for using the BlackBerry Push Service SDK.....	8
6	Installing a database management system.....	9
	Install and configure a MySQL database.....	9
	Create a server instance.....	10
	Configure the server instance.....	10
	Create the database schema.....	11
	Create a user account for the database schema.....	12
	Create a new connection to the database schema.....	13
	Create the database tables manually.....	13
7	Installing the BlackBerry Push Service SDK and Apache Tomcat.....	15
	Prerequisites: Installing and configuring the BlackBerry Push Service SDK and Apache Tomcat.....	15
	Installing and configuring the BlackBerry Push Service SDK and Apache Tomcat.....	15
	Configuration settings for the BlackBerry Push Service SDK setup application.....	16
	Install and configure Apache Tomcat manually.....	17
8	Configuring the server-side applications.....	20
	Configure the server-side applications manually.....	20
9	Verify that the server-side applications work.....	22
	URLs for the DebugPortal helper application.....	22
	URLs for the sample Push Initiator.....	23
	URLs for the PushSDK helper application.....	24
10	Adding the configuration settings for the push application to the database.....	26
	Add the configuration settings for the push application to the database.....	26
	Configuration settings for the push application.....	26

11	Installing and configuring the sample push-enabled application on a device.....	29
	Prerequisites: Installing the sample push-enabled application on a device.....	29
	Install the sample push-enabled application on a device.....	29
	Configure the sample push-enabled application.....	30
	Configuration settings for the sample push-enabled application.....	30
12	Sending test pushes and using the Acknowledgment component.....	32
	Configure a server-side application to receive result notification messages.....	32
	Result notification messages.....	33
	Send a test push through the sample Push Initiator.....	33
	Send a test push through the DebugPortal helper application.....	34
13	Configuring your own push application.....	35
	Configuration settings for your push application.....	35
	Configure the cache settings for the server-side applications.....	36
	Using a sample application as a starting point for your own application.....	37
	Create an Eclipse project using a sample application as a template.....	37
	Configure an Eclipse project.....	38
14	Glossary.....	39
15	Legal notice.....	40

Overview of the BlackBerry Push Service SDK

1

The BlackBerry® Push Service SDK contains a set of Java® libraries you can use to develop push applications. It consists of a server-side library that exposes the push functionalities of the Push Proxy Gateway at a higher level so that you can focus on developing the features of your push application, instead of spending time to learn how to accomplish infrastructure related tasks.

The libraries are designed to handle complex use cases and implement best practices, thereby removing the need for you to write boilerplate code. They provide a consistent programming model that you can follow to develop Push Initiators. The server-side library is componentized. There are dependencies among the components and choosing one component might mean you need to use one or more of the other components. Typical usage involves either using the components that provide low-level API functionalities or using the components that provide high-level API functionalities.

You can configure the BlackBerry Push Service SDK for public mode or enterprise mode. Public mode allows you to develop push applications that use the BlackBerry® Push Service in the BlackBerry® Infrastructure to send push content to anyone who subscribes to your push service (you must register with Research In Motion to use the BlackBerry Push Service). Enterprise mode allows you to develop push applications that use the BlackBerry® Enterprise Server to send push content to members of your organization.

To create a complete push solution, you must develop both the Push Initiator (that is, the application that submits push requests to the Push Proxy Gateway) on the server side, and a push-enabled application (that is, the application on the BlackBerry device that receives push content) on the client-side. A client-side library (the Push API, included in the `net.rim.blackberry.api.push` package) is available in the BlackBerry Java Development Environment 5.0 or later to allow you to develop push-enabled applications for the BlackBerry device. This API functions as a companion to the BlackBerry Push Service SDK, providing similar functionalities for the client side as the BlackBerry Push Service SDK provides for the server side. If you use the BlackBerry Push Service SDK to develop your Push Initiator, and intend to use the BlackBerry® Internet Service as the Push Proxy Gateway, you must use the Push API to create your client application.

The BlackBerry Push Service SDK solution provides the following:

Component	Description
server-side library	The server-side library provides the following components: Commons, PAP, Push, Push Application Management, Subscription, and Acknowledgment.
client-side library	Included in BlackBerry JDE 5.0 or later, the <code>net.rim.blackberry.api.push</code> package provides push-enabled applications, that use the BlackBerry Push Service SDK, with connectivity management functionalities.
helper applications	These applications are complete web applications or web application templates that demonstrate or can be modified to use the functionalities of the BlackBerry Push Service SDK server-side library: DebugPortal, PushSDK, and PAPNotify.

Component	Description
sample applications	The sample Push Initiator and client-side sample application demonstrate the use of the BlackBerry Push Service SDK.

System requirements: development environment

2

Item	Requirement
servlet container	any JSP 2.1 servlet container (Apache Tomcat™ 6 or later is suggested) that implements Java® Servlet Specification version 2.5
database management system	a MySQL® database or an Oracle® database (MySQL Community Server 5.1.34 or later is suggested)
Java® development environment	any integrated development environment with JRE™ 1.6 or later (Eclipse® is suggested)

Planning the installation of the BlackBerry Push Service SDK development environment

3

To develop and test your push application, you need to set up the BlackBerry® Push Service SDK and the necessary third-party components.

When developing the Push Initiator and the push-enabled application, if you use Eclipse® as your Java® development tool, you have the benefit of being able to use the same IDE for developing both applications. Developing a push-enabled application requires the BlackBerry APIs and the BlackBerry® Smartphone Simulator, which can be easily set up using the BlackBerry® Java® Plug-in for Eclipse®.

The BlackBerry® Push Service SDK server-side library utilizes third-party libraries. So, you need to import the third-party libraries in your development environment along with the server-side library components. Some of the server-side library components require a persistent store. Therefore, depending on the components you are going to be using in your development, you might need to configure a database for an RDBMS implementation of those components. Similarly, depending on the components you are going to be using in your development, you might need to install and configure a servlet container.

If you want to use the sample application or use any of the helper applications, you will need to install and configure both a database and a servlet container.

A setup application is available to automate some parts of the installation and configuration of the database and the servlet container, as well as extracting the library components. Some instructions for manual installation and configuration are also available.

For more information about the BlackBerry Push Service, visit <http://na.blackberry.com/eng/developers/javaappdev>.

Demonstrating the functionality of the BlackBerry Push Service SDK

4

The BlackBerry® Push Service SDK setup application is designed to help you set up a development environment so that you can develop an understanding of the BlackBerry® Push Service framework and its components.

After you install and configure a database and the BlackBerry Push Service SDK you can configure the sample Push Initiator and other helper applications, install a sample push-enabled application on a BlackBerry device, and begin pushing content through the push framework.

Prerequisites for using the BlackBerry Push Service SDK

5

To use the BlackBerry® Push Service SDK, be aware of the following considerations:

- You must identify the following requirements for your push application:
 - whether it will use the BlackBerry® Internet Service or a BlackBerry® Enterprise Server as the Push Proxy Gateway to push content
 - the number of daily pushes
 - the size of the file or content that the Push Initiator submits to the push framework
- If your push framework uses the BlackBerry® Internet Service as the PPG to push content, you must register your push application with Research In Motion. To register, visit www.blackberry.com/profile/?eventId=push_api.
- To create a complete push solution, you must develop the following components:
 - the Push Initiator (that is, the application that submits push requests to the Push Proxy Gateway) for the server side
 - a push-enabled application (that is, the application on the BlackBerry device that receives push content) for the client-side

Installing a database management system

6

Instructions are provided for installing and configuring a MySQL® database management system. The BlackBerry® Push Service SDK also supports the use of an Oracle® database.

Install and configure a MySQL database

The following instructions are intended only for computers that are running Windows® and describe configuration settings for installing a MySQL® database that you can use to quickly set up the BlackBerry® Push Service SDK so that you can see its functionality. Depending on your organization's push requirements, you might change some of these settings when you develop and configure your own push application.

Before you begin:

Download and install Microsoft® .NET Framework 3.5 or later from www.microsoft.com/downloads/.

1. Download and run the Windows® MSI Installer option for MySQL® Community Server 5.1 from www.mysql.com.
2. In the setup wizard, click the **Typical** setup type and follow the instructions on the screen.
3. Select the **Configure MySQL Server now** check box and click **Finish**.
4. In the configuration wizard, follow the instructions on the screen and use the default configuration settings, except for the following settings:
 - a. For the approximate number of concurrent connections to the server, click **Online Transaction Processing (OLTP)**.
 - b. For the default character set, click **Best Support for Multilingualism**.
 - c. For the Windows options, click the **Include Bin Directory in Windows PATH** check box.
5. Download the MySQL Connector/J 5.1 from www.mysql.com and extract the files.
If you plan to use the option to install the distribution files and Apache Tomcat™ in the BlackBerry® Push Service SDK setup application, when you perform the installation, you will need to point to the `mysql-connector-java-x.x.x.x-bin.jar` file (where `x.x.x.x` is the version number) in this folder.
6. Download and install the MySQL Workbench from www.mysql.com.
7. If you plan on using the BlackBerry® Push Service SDK setup application on a server other than the server that you installed MySQL on, perform the following actions:
 - a. On the server that you plan to use the setup application, download the Windows ZIP archive option for MySQL Community Server 5.1 from www.mysql.com and extract the files to a location on your server.
 - b. In the installation folder for MySQL, navigate to the `bin` folder and copy the filepath.
 - c. Append the filepath that you copied to the value for the `PATH` environment variable.

After you finish:

Create a server instance.

Create a server instance

The following instructions describe configuration settings for creating a server instance that you can use to quickly set up the BlackBerry® Push Service SDK so that you can see its functionality. Depending on your organization's push requirements, you might change some of these settings when you develop and configure your own push application.

Before you begin:

[Install and configure a MySQL database.](#)

1. On the **Home** screen for the MySQL® Workbench, in the **Server Administration** section, click **New Server Instance**.
2. In the **Create a new server instance** wizard, follow the instructions on the screen and use the default configuration settings except for the settings described in steps 3 - 5.
3. To specify the host machine for the database server, click the **Localhost** option.
4. To specify the operating system of the host machine, click the **Windows (MySQL 5.1 Installer Package)** option.
5. To create the database connection, perform the following actions:
 - a. On the **Database Connection** screen, click **Open Connection Manager**.
 - b. Click **New**.
 - c. In the **Stored Connections** section, click **New Connection 1**.
 - d. In the **Connection Name** field, type a name for the connection type (for example, **Admin**).
 - e. In the **Hostname** field, type **localhost**.
 - f. In the **Port** field, type **3306**.
 - g. In the **Username** field, type **root**.
 - h. In the **Password** field, type the password that you specified when you installed and configured the MySQL database.
 - i. Click **Close**.

After you finish:

[Configure the server instance.](#)

Configure the server instance

The following instructions describe configuration settings for a server instance that you can use to quickly set up the BlackBerry® Push Service SDK so that you can see its functionality. Depending on your organization's push requirements, you might change some of these settings when you develop and configure your own push application.

Before you begin:

Create a server instance.

1. On the **Home** screen for the MySQL® Workbench, in the **Server Administration** section, double-click the server instance you created.
2. On the **Configuration** tab, click the **Advanced** tab.
3. Select the **Def. Char Set** checkbox and specify a default character set for the server instance (UTF-8 is suggested).
4. Click **Apply**.
A window appears on the screen.
5. In the window that appears, click **View File**.
6. Find the line `sql -mode="STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION"` and replace it with `sql -mode="TRADITIONAL"` to cause the MySQL database to behave like a traditional SQL database management system (for example, the database throws an error instead of a warning when an input value is invalid).
7. Click **Apply**.
8. On the **Startup** tab, click **Stop Server**.
9. After the server stops, click **Start Server** to allow the new configuration to take effect.

After you finish:

Create the database schema.

Create the database schema

The following instructions describe configuration settings for creating the database schema that you can use to quickly set up the BlackBerry® Push Service SDK so that you can see its functionality. Depending on your organization's push requirements, you might change some of these settings when you develop and configure your own push application.

Before you begin:

Configure the server instance.

1. On the **Home** screen for the MySQL® Workbench, in the **SQL Development** section, double-click the database connection that you created when you created the server instance for the database.
2. Right-click the **Schemata** section, and click **Create Schema**.
3. In the **Name** field, specify a schema name (for example, pushsdk).
4. In the **Collation** drop-down list, select a default collation type.
If you use UTF-8, for a case sensitive collation type, specify **utf8_bin**. For a case-insensitive collation type, specify **utf8_general_ci**.
5. Click **Apply** twice.
6. Click **Finish** to apply the changes to the server instance.

7. Right-click the **Schemata** section and click **Refresh All** to display the new schema.

After you finish:

[Create a user account for the database schema.](#)

Create a user account for the database schema

The following instructions describe configuration settings for creating a user account that you can use to quickly set up the BlackBerry® Push Service SDK so that you can see its functionality. Depending on your organization's push requirements, you might change some of these settings when you develop and configure your own push application.

The user account that you set up in this task is the user account that you use to access the database tables through a Push Initiator.

Before you begin:

[Create the database schema.](#)

1. On the **Home** screen for the MySQL® Workbench, in the **Server Administration** section, double-click the server instance that you created.
2. Click the **Accounts** tab and click **Add Account**.
3. Click the **Login** tab and specify a login name and password for the user.
4. In the **Limit Connectivity to Hosts Matching** field, type one or more names and host machines that the user can connect from (for example, localhost).
5. Click **Apply**.
6. On the **Schema Privileges** tab, in the **Users** column, click the user account you created.
7. Click **Add Entry**.
8. In the **Host** section, select the **Selected host** option.
9. In the **Selected Host** drop-down list, click the host machine that you want to define the privileges for (for example, localhost).
10. In the **Schema** section, select the **Selected schema** option.
11. In the **Selected schema** list, click the database schema that you created and that you want to define the privileges for.
12. Click **Select "ALL"** to assign the user all privileges for this database schema.
13. Click **Save Changes**.

After you finish:

[Create a new connection to the database schema.](#)

Create a new connection to the database schema

The following instructions describe configuration settings for creating a new connection to the database schema that you can use to quickly set up the BlackBerry® Push Service SDK so that you can see its functionality. Depending on your organization's push requirements, you might change some of these settings when you develop and configure your own push application.

Before you begin:

[Create a user account for the database schema.](#)

1. On the **Home** screen for the MySQL Workbench, in the **SQL Development** section, click **New Connection**.
2. In the **Connection Name** field, type a name for the connection.
3. In the **Hostname** field, type the name of the host machine that the server instance is installed on (for example, localhost).
4. In the **Port** field, type **3306**.
5. In the **Username** and **Password** fields, specify the login credentials that you created for the database schema.
6. In the **Default Schema** field, type the name of the database schema that you created.

After you finish:

[Install and configure the BlackBerry Push Service SDK and Apache Tomcat.](#)

Create the database tables manually

If you plan to create the database tables by using the BlackBerry® Push Service SDK setup application, you can skip this task.

If you choose not to let the setup application create the database tables, the setup application displays the SQL scripts so that you can create them manually. You can also navigate to the `pushsdk_schema_mysql.sql` file after you run the setup application to retrieve the SQL scripts. When you configure your own push application, you can substitute your own SQL scripts for the ones provided in the BlackBerry Push Service SDK distribution package.

Before you begin:

Run the BlackBerry Push Service SDK setup application using either installation option (the distribution package with Apache Tomcat™ or the distribution package without Apache Tomcat).

1. On the **Home** screen for the MySQL Workbench, in the **SQL Development** section, double-click the new connection that you created.
2. On your computer, navigate to the folder containing the database scripts, double-click `pushsdk_schema_mysql.sql`, and copy the contents of the file (for Windows®, the default location is `C:/Program Files/BPSS/pushsdk/database-scripts/mysql`).
3. On the **SQL Editor** screen, in the **SQL Statements** section, paste the contents of `pushsdk_schema_mysql.sql`.
4. Click **Execute**.

After you finish:

- If you did not specify the application ID of your push-enabled application when you ran the setup application, [configure the Push Initiator manually](#).
- [Verify that the Push Initiator work](#).

Installing the BlackBerry Push Service SDK and Apache Tomcat

7

Prerequisites: Installing and configuring the BlackBerry Push Service SDK and Apache Tomcat

- Install and configure a database management system.
- If you use a MySQL® database, you must also obtain a Java® database connector for MySQL. If you use an Oracle database, the BlackBerry® Push Service SDK provides the Java database connector.
- Verify that you have the login information for the user account(s) that you created when you installed and configured the database.
- If you use the BlackBerry® Internet Service as the Push Proxy Gateway to push content, verify that you have the application ID and the PPG base URL which you received during the registration process.
- Verify that the HTTP port and HTTPS port that you plan to use for Apache Tomcat™ are open for incoming requests in the server's firewall and are not already in use.

Installing and configuring the BlackBerry Push Service SDK and Apache Tomcat

After you register to use the BlackBerry® Push Service, you can install and configure parts of the BlackBerry Push Service SDK development environment with a setup application. When you run the setup application, you can choose to install only the distribution files, or the distribution files and Apache Tomcat™. If you install only the distribution files, you will have to configure your development environment manually. If you specify the option to install the distribution files and Apache Tomcat, the setup application can help you perform the following actions:

- install the BlackBerry Push Service SDK distribution files
- install Apache Tomcat
- configure and deploy three server-side applications (the sample Push Initiator and the DebugPortal and PushSDK helper applications)
- specify the HTTP port, HTTPS port, and the default character encoding for Apache Tomcat
- specify the database server instance for the BlackBerry Push Service SDK applications and create the tables for the database
- specify the location of a MySQL® Java® database connector to allow the BlackBerry Push Service SDK to integrate with the database
- configure your organization's push application to use either the BlackBerry® Enterprise Server or the BlackBerry® Internet Service as the Push Proxy Gateway for pushing content
- specify a location to store logging information for the server-side applications

Even if you specify the option to install the distribution files and Apache Tomcat, you can still configure some components of the BlackBerry Push Service SDK manually if you choose to do so. If you do not create the tables for the database while using the setup application, you can still [create the database tables manually](#) after you install the BlackBerry Push Service SDK. You can also [configure the server-side applications manually](#) if you do not specify the application ID of your push application.

After the installation and configuration is complete, you should [verify that the server-side applications work](#) to see whether all the components of the BlackBerry Push Service SDK are installed and configured correctly.

Configuration settings for the BlackBerry Push Service SDK setup application

Setting	Description
HTTP port	This setting specifies the HTTP port number for the Apache Tomcat™ server. The default port number is 8080.
HTTPS port	This setting specifies the HTTPS port for the Apache Tomcat™ server. The default port number is 8443.
Default Character Encoding	This setting specifies the default character encoding type for HTTP requests. The default character encoding type is UTF-8.
Server Host	This setting specifies the name of the host machine that you created the server instance on.
Port	This setting specifies the port number that the server instance uses.
User with query/insert/update permissions	This setting specifies the user name and password for a user account that you created to work with this database schema (can be the same as the user with table creation permissions).
User with table creation permissions	This setting specifies the user name and password for a user account that you created to work with this database schema (can be the same as the user with query/insert/update permissions).
PPG Base URL	<p>This setting specifies the base URL for the Push Proxy Gateway.</p> <p>If your push application uses the BlackBerry® Internet Service as the PPG to push content, you receive the PPG Base URLs when you register your push application with Research In Motion. After your initial registration, you receive a PPG Base URL for an evaluation environment that you can use to send test pushes while you are developing and configuring your push application. The evaluation environment's PPG Base URL is https://pushapi.eval.blackberry.com. After your push application is ready for production, you can register to use the production environment and start pushing content to subscribers. The PPG Base URL for the production environment is https://pushapi.region.blackberry.com, where <i>region</i> is one of the following:</p> <ul style="list-style-type: none"> • na (North America, Latin America, South America, Africa) • eu (Europe) • ap (Asia-Pacific)

Setting	Description
	If your push application uses the BlackBerry® Enterprise Server as the PPG to push content, you would specify the server that BlackBerry Enterprise Server is installed on and the port number that it is configured to work with (for example, <code>http://localhost:8080</code>).
Push Application ID	This setting specifies the ID for your push-enabled application. If your push application uses the BlackBerry Internet Service as the PPG to send pushes, you receive this ID when you register your push application with RIM. If your push application uses the BlackBerry Enterprise Server as the PPG to push content, you can create your own ID. The ID for your push application is a required parameter for many of the BlackBerry® Push Service SDK APIs.
Should application reliability be used for pushes to this application?	This setting specifies that a push is only confirmed as being successful when the push-enabled application acknowledges that the content arrives. For push applications that do not specify application level reliability, a push is considered as successful as soon as the content reaches the BlackBerry device, even if the push-enabled application no longer works or is removed.

Install and configure Apache Tomcat manually

If you use the option to install the distribution files and Apache Tomcat™ in the BlackBerry® Push Service SDK setup application, you can skip this task.

You can configure your Apache Tomcat installation to use a character encoding type for HTTP GET and HTTP POST requests that do not specify their own character encoding, even though it is best practice to specify the character encoding type of your HTTP request in the Content-Type header. To allow HTTPS to work, you can obtain an SSL certificate from a certificate authority and add it to the JDK keystore or you can create a self-signed certificate and add it to the JDK keystore.

To allow the use of the BlackBerry Push Service SDK server-side applications (sample Push Initiator and DebugPortal and PushSDK helper applications) you must copy the WAR files for the server-side applications from the BlackBerry Push Service SDK distribution package into the webapps folder in the Apache Tomcat installation folder.

Before you begin:

[Install and configure a database management system.](#)

1. Download the .zip installation file for Apache Tomcat 6 from <http://tomcat.apache.org>, and extract the files.
2. Specify a default character encoding type for HTTP GET and HTTP POST requests by performing the following actions:
 - a. Navigate to the **bin** folder in the installation folder for Apache Tomcat and open **catalina.bat** (for UNIX®, **catalina.sh**) in a text editor.
 - b. Locate the JAVA_OPTS environment variable and specify the value `-Dfile.encoding="UTF-8"` (where UTF-8 is the default character encoding).

3. In the folder for Apache Tomcat, in the **conf** folder, open **server.xml** in a text editor.
4. To configure Apache Tomcat so that the query component of the URI for an HTTP GET request is treated with the correct character encoding, perform one of the following actions:

Task	Steps
Set a character encoding type to be used for the query component of the URI.	<p>In the server.xml file, locate the Connector XML tag with the <code>protocol="HTTP/1.1"</code> attribute, and add the <code>URIEncoding="encoding_type"</code> attribute .</p> <pre><Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" URIEncoding="UTF-8" /></pre> <p>Replace <i>UTF-8</i> with your desired character encoding type.</p>
Specify that the query component of the URI has the same character encoding type as the body of the HTTP request.	<p>In the server.xml file, locate the Connector XML tag with the <code>protocol="HTTP/1.1"</code> attribute, and add the <code>useBodyEncodingForURI="true"</code> attribute .</p> <pre><Connector port="8080" protocol="HTTP/1.1" connectionTimeout="20000" redirectPort="8443" useBodyEncodingForURI="true" /></pre>

5. In the **server.xml** file, locate and uncomment this Connector XML tag to enable HTTPS:

```
<Connector port="8443"
  protocol="HTTP/1.1"
  SSLEnabled="true"
  maxThreads="150"
  scheme="https"
  secure="true"
  clientAuth="false"
  sslProtocol="TLS" />
```

6. In the **server.xml** file, locate and uncomment this Valve XML tag to enable web access logging:

```
<Valve className="org.apache.catalina.valves.AccessLogValve"
  directory="logs"
  prefix="localhost_access_log."
  suffix=".txt"
  pattern="common"
  resolveHosts="false"/>
```

7. Create a self-signed certificate keystore to allow HTTPS to work by running one of the following commands in the command prompt:
 - If your system runs Windows®, use `%JAVA_HOME%\bin\keytool -genkey -alias tomcat -keyalg RSA`.
 - If your system runs UNIX®, use `$JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA`.
8. Specify a password for the keystore and provide the requested information.
9. To allow the use of the server-side applications (the sample Push Initiator and the DebugPortal and PushSDK helper applications) perform the following actions:
 - a. Navigate to the **sample-app** folder and copy the WAR file for the sample application (for Windows, the default location is C:\Program Files\BPSS\pushsdk\sample-app).
 - b. Paste the WAR file into the **webapps** folder in the Apache Tomcat installation folder.
 - c. Navigate to the **helper-apps** folder and copy the WAR files for the DebugPortal and PushSDK helper applications (for Windows, the default location is C:\Program Files\BPSS\pushsdk\helper-apps).
 - d. Paste the WAR files into the **webapps** folder in the Apache Tomcat installation folder.
 - e. Extract the WAR files that you pasted in the **webapps** folder.
 - f. If you use MySQL for the database, navigate to the folder for MySQL Connector/J 5.1 and copy the JAR file for the Java® database connector.
 - g. Navigate to the Apache Tomcat installation folder and paste the JAR file into the `webapps\app_name\WEB-INF\lib` folder (where *app_name* is one of sample-app, debug-portal, or pushsdk) for each server-side web application that you want to deploy.
 - h. Navigate to the Apache Tomcat installation folder, and in the `webapps\app_name\WEB-INF\classes` (where *app_name* is one of sample-app, debug-portal, or pushsdk), double-click the **PushSDK.properties** file, and update it with the information from your database.

After you finish:

[Configure the sample Push Initiator and the DebugPortal and PushSDK helper applications manually.](#)

Configuring the server-side applications

8

If you install the BlackBerry® Push Service SDK by using the option to install Apache Tomcat™, you can configure the server-side applications (the sample Push Initiator and the DebugPortal and PushSDK helper applications) to work with a specific push-enabled application during the installation process.

To configure the server-side applications you must provide the application ID for your push application. The application ID depends on the type of Push Proxy Gateway that you use to push content. If you use the BlackBerry® Internet Service as the PPG, when you register with Research In Motion, you receive a unique application ID. If you use a BlackBerry® Enterprise Server as the PPG, you can create your own application ID. If you do not have the application ID during the installation process, you can manually configure the server-side applications after you run the setup application.

Configure the server-side applications manually

If you configured the server-side applications (the sample Push Initiator and the DebugPortal and PushSDK helper applications) by using the BlackBerry® Push Service SDK setup application, you can skip this task.

If you do not have the application ID during the installation process, you can manually configure the server-side applications by adding the application ID of the push application to the appropriate files in the Apache Tomcat™ installation folder. Depending on how you installed Apache Tomcat, the property that you add the application ID to varies. If you installed Apache Tomcat by using the setup application, the property that you change the value of is `sampleapp.appid`. If the installation was performed manually, the property that you change the value of is `appid`.

Before you begin:

[Install and configure the BlackBerry Push Service SDK and Apache Tomcat.](#)

1. Navigate to the `webapps\sample-app\WEB-INF\classes` folder in the Apache Tomcat installation folder and open **sample-app.properties** in a text editor.
2. Locate the `sampleapp.appid` property and replace `sampleapp.appid` with the application ID for your device application.
3. Open **acknowledgement-context.xml** in an XML editor.
4. Locate the `<entry key="{ application.ID} ">` XML tag (where *application.ID* is `sampleapp.appid` or `appid`) and replace *application.ID* with the application ID for your device application.
5. Save and close **acknowledgement-context.xml**.
6. Navigate to the `webapps\pushsdk\WEB-INF\classes` folder in the Apache Tomcat installation folder and open **acknowledgement-context.xml** in an XML editor.
7. Locate the `<entry key="{ application.ID} ">` XML tag (where *application.ID* is `sampleapp.appid` or `appid`) and replace *application.ID* with the application ID for your device application.
8. Save and close **acknowledgement-context.xml**.

9. Navigate to the webapps\debug-portal\WEB-INF\classes folder in the Apache Tomcat installation folder and open **acknowledgement-context.xml** in an XML editor.
10. Locate the `<entry key="{ application.ID} ">` XML tag (where *application.ID* is `sampleapp.appid` or `appid`) and replace *application.ID* with the application ID for your device application.
11. Save and close **acknowledgement-context.xml**.

After you finish:

[Verify that the server-side applications work.](#)

Verify that the server-side applications work

9

Before you configure a push-enabled application and start pushing content to it, you can verify that the server-side applications work correctly by accessing the URLs for the applications in a web browser.

Before you begin:

- Install and configure the BlackBerry Push Service SDK and Apache Tomcat.
 - If you installed Apache Tomcat manually, or if you did not specify the application ID when you ran the setup application, configure the server-side applications manually.
1. Verify that the value of the CATALINA_HOME environment variable is the filepath for the installation folder for Apache Tomcat by performing the following actions:
 - a. Right-click **My Computer** and click **Properties**.
 - b. On the **Advanced** tab, click **Environment Variables**.
 - c. In the **System variables** section, click **CATALINA_HOME**. Click **Edit**.
 - d. Verify that the variable value is the filepath for the Apache Tomcat installation folder.
 2. Navigate to the **bin** folder in the Apache Tomcat installation folder and double-click **startup.bat** (for UNIX®, **startup.sh**) to start Apache Tomcat.

To stop Apache Tomcat, double-click **shutdown.bat** (for UNIX, **shutdown.sh**).
 3. Navigate to the **logs** folder in the Apache Tomcat installation folder and verify that the log files do not contain any errors.
 4. In the address field of a web browser, type a URL for a server-side application and verify that the results are correct.
 5. Repeat step 4 for each URL.

After you finish:

Add the configuration settings for the push application to the database.

URLs for the DebugPortal helper application

The following table provides the URLs to use to access the DebugPortal helper application, and the results of accessing the URLs in a web browser without using any request parameters. Though the URLs display errors when you access them without the proper parameters, this indicates that the DebugPortal helper application is installed correctly, and is not performing as expected only because the the proper request parameters are missing.

These URLs are designed for a server that is installed locally and configured to use the default port number. Change the host name and port number to match the configuration of your organization's push server.

URL	Description
https://localhost:8443/debug-portal	<ul style="list-style-type: none"> the URL for the main page of the DebugPortal helper application displays a web interface for administering and testing your push-enabled applications
https://localhost:8443/debug-portal/subscribe	<ul style="list-style-type: none"> the URL for subscribing a BlackBerry® device user to a push application returns an rc=10002 error
https://localhost:8443/debug-portal/unsubscribe	<ul style="list-style-type: none"> the URL for unsubscribing a device user from a push application returns an rc=10002 error
https://localhost:8443/debug-portal/suspend	<ul style="list-style-type: none"> the URL for suspending a device user's subscription returns an rc=10002 error
https://localhost:8443/debug-portal/resume	<ul style="list-style-type: none"> the URL for resuming a device user's subscription returns an rc=10002 error
https://localhost:8443/debug-portal/notification	<ul style="list-style-type: none"> the URL for receiving result notification messages that describe the status of a push to a subscriber returns an Internal Server Error

URLs for the sample Push Initiator

The following table provides the URLs to use to access the server-side sample application, and the results of accessing the URLs in a web browser without using any request parameters. Though the URLs display errors when you access them without the proper parameters, this indicates that the sample Push Initiator is installed correctly, and is not performing as expected only because the the proper request parameters are missing.

These URLs are designed for a server that is installed locally and configured to use the default port number. Change the host name and port number to match the configuration of your organization's push server.

URL	Definition
https://localhost:8443/sample-app	<ul style="list-style-type: none"> the URL for the main page of the sample Push Initiator displays a web interface for starting and stopping pushes to a push-enabled application

URL	Definition
https://localhost:8443/sample-app/subscribe	<ul style="list-style-type: none"> the URL for subscribing a BlackBerry® device user to a push application returns an rc=10002 error
https://localhost:8443/sample-app/unsubscribe	<ul style="list-style-type: none"> the URL for unsubscribing a device user from a push application returns an rc=10002 error
https://localhost:8443/sample-app/suspend	<ul style="list-style-type: none"> the URL for suspending a device user's subscription returns an rc=10002 error
https://localhost:8443/sample-app/resume	<ul style="list-style-type: none"> the URL for resuming a device user's subscription returns an rc=10002 error
https://localhost:8443/sample-app/notification	<ul style="list-style-type: none"> the URL for receiving result notifications that describe the status of a push to a subscriber returns a 500 error with an exception that indicates that the Push ID is null

URLs for the PushSDK helper application

The following table provides the URLs to use to access the PushSDK helper application, and the results of accessing the URLs in a web browser without using any request parameters. Though the URLs display errors when you access them without the proper parameters, this indicates that the PushSDK helper application is installed correctly, and is not performing as expected only because the the proper request parameters are missing.

These URLs are designed for a server that is installed locally and configured to use the default port number. Change the host name and port number to match the configuration of your organization's push server.

URL	Definition
https://localhost:8443/pushsdk/subscribe	<ul style="list-style-type: none"> the URL for subscribing a BlackBerry® device user to a push application returns an rc=10002 error
https://localhost:8443/pushsdk/unsubscribe	<ul style="list-style-type: none"> the URL for unsubscribing a device user from a push application returns an rc=10002 error
https://localhost:8443/pushsdk/suspend	<ul style="list-style-type: none"> the URL for suspending a device user's subscription

URL	Definition
	<ul style="list-style-type: none">• returns an rc=10002 error
https://localhost:8443/pushsdk/resume	<ul style="list-style-type: none">• the URL for resuming a device user's subscription• returns an rc=10002 error
https://localhost:8443/pushsdk/notification	<ul style="list-style-type: none">• the URL for receiving result notifications that describe the status of a push to a subscriber• returns a 500 error with an exception that indicates that the Push ID is null

Adding the configuration settings for the push application to the database

10

Before you can push content to a push-enabled application, you must add the configuration settings for the push application to the database.

If you use the BlackBerry® Internet Service as the Push Proxy Gateway to push content, you establish the configuration settings when you register your push application with Research In Motion. If you use the BlackBerry® Enterprise Server as the PPG to push content, you can determine most of these configuration settings yourself.

There are two options for adding the configuration settings for the push application to the database. You can add the settings using the web interface that is provided in the DebugPortal helper application, or you can use the BlackBerry® Push Service SDK APIs to hardcode the settings into your Push Initiator. For the purpose of testing the functionality of the BlackBerry® Push Service SDK, using the DebugPortal is optimal.

Add the configuration settings for the push application to the database

Before you begin:

Verify that the server-side applications run correctly.

1. In a web browser, type the URL for the main page of the DebugPortal helper application.
2. On the **Push Application Management** tab, click **Add new application**.
3. On the **Add push application** screen, specify the [configuration settings for the push application](#).

After you finish:

Install the sample push-enabled application on a device.

Configuration settings for the push application

Setting	Description
Id	This setting specifies the application ID for the push application. You must verify that the application ID that you specify here matches the application ID that you specified when you configured the server-side applications.
Name	This setting specifies a text string that you can use to identify your push application.
Version	The setting specifies the version number of the push application.

Setting	Description
Description	This setting specifies a description of the push application.
Username	This setting specifies either the application ID that you received when you registered the push application with Research In Motion (if the BlackBerry® Internet Service is the PPG for pushing content) or the user name sent with each push (if a BlackBerry® Enterprise Server is the PPG for pushing content and if authentication is required for each push).
Password	This setting specifies either the password that you specified when you registered the push application with RIM (if the BlackBerry Internet Service is the PPG for pushing content) or the password that is sent with each push (if a BlackBerry Enterprise Server is the PPG for pushing content and if authentication is required for each push).
Push listen port	This setting specifies the BlackBerry device port number that content is pushed through.
Status	This setting specifies whether the push application is currently active or inactive.
Max daily quota (MB)	This setting specifies the size limit of content, in megabytes, that a push application can push in one day to all subscribers.
Consecutive failed push threshold	This setting specifies the maximum number of consecutive pushes that the PPG can send to a device that fail before pushes to that device are suspended (only for push applications that have a result notification URL). Otherwise, this field can be left empty.
Default push live time (in seconds)	This setting specifies the length of time in seconds, after a push request is accepted by the PPG, that the PPG attempts to push content to a device before the push request expires.
Application level reliability	<p>This setting specifies whether or not the push application uses application level reliability (a push is confirmed as being successful when the push-enabled application acknowledges that it has arrived).</p> <p>If the push application does not use application level reliability, a push is confirmed as being successful when the push reaches the device. Even if the push-enabled application is deleted or is not running on the device, the push is still considered to be successful.</p>
Store push requests	<p>This setting specifies whether or not to store the details of a push.</p> <p>To save the details of a push request, you must supply a result notification URL. When you push content to a push-enabled application on a device, the details of the push are sent back to the Push Initiator which stores the information in the database.</p> <p>The database stores push request information in the <code>BPSS_PUSH_REQUEST</code> and <code>BPSS_PUSH_REQUEST_DETAIL</code> database tables. The <code>BPSS_PUSH_REQUEST</code> table stores information about the push such as the size of the push, the date that the push is sent,</p>

Setting	Description
	and the number of devices that are pushed to. The <code>BPSS_PUSH_REQUEST_DETAIL</code> database table stores information about the status of a push to the individual device users (result notification messages).
Result notification url	<p>This setting specifies the URL for the Push Initiator that handles result notifications and is mandatory if you specify that you want to store the details of push requests.</p> <p>If the PPG for pushing content is the BlackBerry Internet Service, you provided the base of the result notification URL (for example, <code>https://localhost:8443/sample-app</code>) when you registered your push application with RIM. This base of the result notification URL is set in the database of the PPG. In this case, you only specify the extension of the URL that handles the acknowledgements (for example, <code>/notification</code>).</p> <p>If the PPG for pushing content is the BlackBerry Enterprise Server, you must specify the full URL of the server-side push application that handles result notifications (for example, <code>https://localhost:8443/sample-app/notification</code>). Any of the sample Push Initiator, DebugPortal helper application, and PushSDK helper application can handle result notifications.</p>
Last notifications enabled	<p>This setting specifies whether or not the server-side application listens for the last notification to arrive after content is pushed to one or more devices. For example, if content is pushed to five devices, a <code>LastNotificationListener</code> object listens for when the server-side application receives the fifth and last notification. This feature is only available if you specify that you want to store the details of push requests.</p> <p>Though the ability to track the most recent notification might be useful, using this feature can affect the performance of a push application.</p>
Auto-delete push requests	This setting specifies whether or not to delete all of the entries in the database that are related to a push after the server-side application receives the last notification for that push. This feature is only available if you specify that you want to store the details of push requests.
Service class	This setting specifies the service level of the push application (the BlackBerry Push Plus service or the BlackBerry Push Essentials service).
Bypass subscription	This setting specifies whether or not subscriptions to a push application are required. This setting is available only if the PPG for pushing content is the BlackBerry Enterprise Server.
Start date	This setting specifies the date that a push application can begin receiving pushed content.
Expiry date	This setting specifies the date that a push application can no longer receive pushed content.
Added by (username)	This setting specifies the user name of the user that adds the configuration settings for the push application to the database.

Installing and configuring the sample push-enabled application on a device

11

Prerequisites: Installing the sample push-enabled application on a device

- Add the configuration settings for the push application to the database.
- Verify that the BlackBerry® device that you want to install the sample push-enabled application on is synchronized with the BlackBerry® Desktop Manager on the computer that you are using.
- If you plan to install the sample push-enabled application by using a computer that is not the computer that you installed the BlackBerry® Push Service SDK on, copy the .cod and .alx files for the sample push-enabled application to the computer that you are using.

Install the sample push-enabled application on a device

The sample push-enabled application is designed to receive pushed content only from push applications that use the BlackBerry® Internet Service as the PPG and it may not run correctly on BlackBerry devices running a version of BlackBerry® Device Software earlier than 4.2.

When you install the sample push-enabled application on a device, you must install the version of the sample application that is compatible with the BlackBerry Device Software version that is running on the device. You can find the .alx and .cod files for the sample application in the client-side sample application folder (for Windows®, the default location is C:\Program Files\BPSS\pushsdk\client-sample-app).

1. Connect a device to a computer by using a USB cable.
2. Start BlackBerry® Desktop Manager.
3. If the device requires a password, specify the password and click **OK**.
4. In BlackBerry Desktop Manager, click **Application Loader**.
5. In the **Add/Remove Applications** section, click **Start**.
6. Click **Browse**.
7. Navigate to the location of the .alx files for the sample push-enabled application and perform one of the following actions:
 - If your device runs BlackBerry Device Software 5.0 or later, double-click the **PushAPISample5.0-X.X.X.X.alx** file (where X.X.X.X is the BlackBerry Push Service SDK version).
 - If your device runs BlackBerry Device Software 4.6 or later but earlier than 5.0, double-click the **PushAPISample4.6-X.X.X.X.alx** file (where X.X.X.X is the BlackBerry Push Service SDK version).
 - If your device runs BlackBerry Device Software 4.2 or later but earlier than 4.6, double-click the **PushAPISample4.2-X.X.X.X.alx** file (where X.X.X.X is the BlackBerry Push Service SDK version).
8. Select the **Push API Sample** check box and click **Next** to install the sample push-enabled application.

After you finish:

Configure the sample push-enabled application.

Configure the sample push-enabled application

When you configure the sample push-enabled application, make sure that you specify the URL for the server-side application (either the sample Push Initiator or the DebugPortal helper application) that you want to use to receive the subscription request from the BlackBerry® device.

Before you begin:

Install the sample push-enabled application on a device.

1. On the Home screen of the device that you installed the sample push-enabled application on, click **Push API Sample**.
2. Press the **Menu** key.
3. Click **Settings**.
4. Change the [configuration settings for the sample push-enabled application](#) according to the specifications of your push application.
5. Press **Back** and save the changes.
6. Press the **Menu** key.
7. Click **Register**.
8. Type a user name and password and click **OK**.
The message, "Request to register executed successfully" appears.

After you finish:

- To receive result notification messages for your pushes, [configure a server-side application to receive result notification messages](#).
- Send a test push through either the [sample Push Initiator](#) or the [DebugPortal helper application](#).

Configuration settings for the sample push-enabled application

Setting	Description
Port	<ul style="list-style-type: none"> • specifies the port number on the BlackBerry® device that content is pushed through • must match the value of the Push listen port that you specified when you added the settings for the push application to the database

Setting	Description
App ID	<ul style="list-style-type: none">• specifies the application ID for the push application• must match the application ID that you configured your Push Initiator with
BPAS URL	<ul style="list-style-type: none">• specifies an HTTP version of the PPG base URL that you received when you registered your push application with Research In Motion (for example, if you received a PPG Base URL of https://pushapi.eval.blackberry.com, the BPAS URL is http://pushapi.eval.blackberry.com)
Content Provider URL	<ul style="list-style-type: none">• specifies the URL for the server-side application that receives the subscription request from the device (for example, https://localhost:8443/sample-app to receive subscription requests with the sample Push Initiator, or https://localhost:8443/debug-portal to receive subscription requests with the DebugPortal helper application)
App Acknowledgement	<ul style="list-style-type: none">• specifies whether or not the push application supports application-level reliability

Sending test pushes and using the Acknowledgment component

12

You can send test pushes to the sample push-enabled application by using either the sample Push Initiator or the DebugPortal helper application. If you use the DebugPortal helper application to send a test push, you can specify the text or file that you want to push.

If you register to use the BlackBerry® Push Plus service, you can use the Acknowledgment component in your Push Initiator to capture messages regarding whether a push to a BlackBerry device is successful or unsuccessful. These messages, which are called result notification messages, are sent from the PPG to the BlackBerry® Push Service SDK for each BlackBerry device that content is pushed to.

When a Push Initiator that uses the Acknowledgment component receives a result notification message, the application calls the appropriate acknowledgment listener for that message (for example, a successful push, an unsuccessful push, or the last notification to return from a group of pushes) and performs the appropriate actions. In the case of the Push Initiators that are packaged with the BlackBerry Push Service SDK, the applications listen for all notification messages and displays the messages in the Apache Tomcat™ console window. When you develop your own Push Initiator, you can specify the notifications that your application listens for, and the actions performed on receiving such notifications. For example, on receiving a notification for an unsuccessful push, an implementation of the `FailureNotificationListener` class in your application can call a method that pushes the content to that user again.

If your push application uses the BlackBerry® Internet Service as the PPG, when you register your push application with Research In Motion, you provide the base URL for the Push Initiator that handles result notification messages. When you add the settings for the push application to the database, you must specify the result notification URL for the Push Initiator to allow it to receive the result notification messages. You can use any of the sample Push Initiator, DebugPortal helper application, or the PushSDK helper application to handle result notification messages.

For more information about the Acknowledgment component, see the *BlackBerry Push Service SDK Development Guide*.

Configure a server-side application to receive result notification messages

You can receive notification messages regarding the status of a push if your server-side application uses the Acknowledgment component. The Acknowledgment component is only available if you register to use the BlackBerry® Push Plus service.

You can view [result notification messages](#) in the console window for Apache Tomcat™. If you configure the server-side application to store information on push requests, you can verify that the information is being inserted and updated by viewing the message status in the `BPSS_PUSH_REQUEST_DETAIL` database table.

Before you begin:

- [Configure the sample push-enabled application.](#)
1. In a web browser, type the URL for the main page of the DebugPortal helper application.

2. On the **Push Application Management** menu, click **List applications**.
3. In the **Id** column, click the ID for your push-enabled application.
4. Depending on the PPG that your push application uses, perform one of the following actions:
 - If your push application uses the BlackBerry® Internet Service as the PPG, in the **Result notification url** field, type **/notification**.
 - If your push application uses the BlackBerry® Enterprise Server as the PPG, in the **Result notification url** field, type the full URL for the server-side web application that handles notifications (for example, <https://localhost:8443/sample-app/notification>).
5. To store information on push requests in the database, select the **Store push requests** check box.
6. Click **Update application**.

After you finish:

Send a test push through either the [server-side sample application](#) or the [DebugPortal helper application](#).

Result notification messages

When one of the BlackBerry® Push Service SDK server-side applications (the sample Push Initiator and DebugPortal and PushSDK helper applications) receives a result notification message, the appropriate listener processes the result notification and displays a message in the Apache Tomcat™ console window which identifies the type of notification.

Message	Description
AnyNotificationListener	<ul style="list-style-type: none"> • indicates receipt of a result notification message of any type
FailureNotificationListener	<ul style="list-style-type: none"> • indicates receipt of a result notification message for an unsuccessful push
SuccessNotificationListener	<ul style="list-style-type: none"> • indicates receipt of a result notification message for a successful push

Send a test push through the sample Push Initiator

Before you begin:

- [Configure the sample push-enabled application](#).
 - If you want to receive notification messages regarding the status of a push, [configure a server-side application to receive result notification messages](#).
1. In a web browser, type the URL for the main page of the sample Push Initiator.
 2. Verify that the **Application Id** field contains the correct ID for your push-enabled application.
 3. Perform one of the following actions to begin pushing messages:

- To send identical messages to all active subscribers, click **Start General Push**.
 - To send a unique message to each active subscriber, click **Start Personalized Push**.
4. To stop pushing content, click **Stop**.

Send a test push through the DebugPortal helper application

Note: There is an 8kb size limit for each individual push.

Before you begin:

- [Configure the sample push-enabled application](#).
- If you want to receive notification messages regarding the status of a push, [configure a server-side application to receive result notification messages](#).

1. In a web browser, type the URL for the main page of the DebugPortal helper application.
2. On the **Push Management** menu, click **Push**.
3. In the **Push Id** field, specify an ID for the push.
If you leave this field blank, the DebugPortal helper application generates a random ID for the push.
4. In the **Push Application** drop-down list, click the push application that you want to push to.
5. In the **Addresses** field, specify the user name that you created when you configured the sample push-enabled application.
6. Specify a file or type some text to push to the push-enabled application.
7. Click **Push** to push the content.
8. To check the status of the push, perform the following actions (only available with the BlackBerry® Push Plus service):
 - a. On the **Push Management** menu, click **Status Query**.
 - b. Specify the parameters for the query and click **Query**.If the push is successful, the status changes to 'DELIVERED'.

Configuring your own push application

13

When you run the BlackBerry® Push Service SDK setup application and use the option to install Apache Tomcat™, the setup application installs the distribution package and helps you configure the components in the BlackBerry Push Service SDK, such as the server-side applications and the database. The configuration that the setup application performs does not affect the configuration files in the distribution package. When you start to use the distribution package to develop and configure your own push application, you must configure these components manually. For Windows®, the default location of the configuration files is C:\Program Files\BPSS\pushsdk\configuration.

After you change the configuration files in the distribution package to fit your push requirements, you can import the files into the Java® project that contains your push application.

For more information about developing and configuring a push application, see the *BlackBerry Push Service SDK Feature and Technical Overview* and the *BlackBerry Push Service SDK Development Guide*.

Configuration settings for your push application

The following table defines a list of the properties that you might need to configure for your push application. For Windows® the default location of these files is C:\Program Files\BPSS\pushsdk\configuration.

Property	File name	Description
<code>\${bpss.driver.class.name}</code>	PushSDK.properties	specifies the driver class for the Java® database connector (com.mysql.jdbc.Driver for a MySQL® JDBC, or oracle.jdbc.driver.OracleDriver for an Oracle® JDBC)
<code>\${bpss.db.connection.url}</code>	PushSDK.properties	specifies the connection URL for the database (for example, jdbc:mysql://localhost:3306/pushsdk for a MySQL database or jdbc:oracle:thin:@localhost:1522:pushsdk for an Oracle database)
<code>\${bpss.db.username}</code>	PushSDK.properties	specifies the user name for connecting to the database
<code>\${bpss.db.password}</code>	PushSDK.properties	specifies the password for connecting to the database
<code>\${bpss.db.initialSize}</code>	PushSDK.properties	specifies the initial number of connections to the database that are available in the connection pool
<code>\${bpss.db.maxActive}</code>	PushSDK.properties	specifies the maximum number of active connections to the database that the connection pool can allocate at one time (a negative number makes the number of active connections limitless)

Property	File name	Description
<code>#{bpss.db.maxIdle}</code>	PushSDK.properties	specifies the maximum number of idle connections to retain in the connection pool
<code>#{bpss.db.maxWait}</code>	PushSDK.properties	specifies the maximum number of milliseconds that the connection pool waits for a connection to be returned when there are no available connections, before a request for a connection fails (a negative number makes the pool wait indefinitely)
<code>#{bpss.db.validationQuery}</code>	PushSDK.properties	specifies the query that the connection pool uses to validate connections
<code>#{bpss.db.testOnBorrow}</code>	PushSDK.properties	specifies whether to validate the database connections before obtaining them from the connection pool
<code>#{bpss.base.url}</code>	PushSDK.properties	specifies the base URL to the PPG
<code>#{log_location}</code>	log4j.xml	specifies the path to the directory to store the log files in
<code>#{appid}</code>	acknowledgement-context.xml	specifies the ID of a push-enabled application that a content provider wants to register notification listeners for

Configure the cache settings for the server-side applications

You can change the default values of the `multicastGroupAddress` and `multicastGroupPort` attributes in the `ehcache.xml` file for each server-side application to meet the requirements of your installation environment.

1. On your computer, navigate to `webapps\webapp\WEB-INF\classes` folder in the Apache Tomcat installation folder (where `webapp` is one of `sample-app`, `debug-portal`, or `pushsdk`), and open the **ehcache.xml** file in an XML editor.
2. Locate the `cacheManagerPeerProviderFactory` XML element and change the attribute values for `multicastGroupAddress` and `multicastGroupPort`.
3. Repeat steps 1 and 2 for each server-side application.
4. Save and close the **ehcache.xml** file.

Using a sample application as a starting point for your own application

The sample Push Initiator and the sample push-enabled application, along with the other components of the BlackBerry® Push Service SDK, form a basic push application that helps demonstrate the functionality of a push framework. The sample applications demonstrate examples and best practices that can help you develop your own push applications. You can run the sample applications in any IDE so that you can examine the code, and test their functionality. Once you become familiar with how the sample applications work, you can use them as a starting point for developing your own push application.

Both sample applications that come with the BlackBerry Push Service SDK distribution package include the source code, API reference documentation, and the WAR files.

Create an Eclipse project using a sample application as a template

1. On your computer, navigate to either the **sample-app** or **client-sample-app** folders depending on which sample application that you want to use (for Windows®, the default location of these folders is C:\Program Files\BPSS\pushsdk).
2. In the folder for the sample application, extract the JAR file for the sample application to a location on your computer.
3. Depending on the sample application that you extracted, perform one of the following actions:
 - If you are using the sample Push Initiator, in Eclipse®, on the **File** menu, click **New > Java Project**.
 - If you are using the sample push-enabled application, in Eclipse, on the **File** menu, click **New > Project**, expand **BlackBerry**, and double-click **BlackBerry Project**.
4. In the **Project name** field, specify a name for the project.
5. Click **Finish**.
6. In the Project Explorer window, expand the project you created.
7. Right-click the **src** folder, and click **Import**.
8. Expand **General**, and double-click **File System**.
9. Click **Browse**.
10. Navigate to the location where you extracted the sample application JAR file and click the folder containing the extracted sample application. Click **OK**.
11. Expand the folder you selected and navigate through the folder structure to the location of the source files.
12. Select the check box for each source file and click **Finish**.

After you finish:

Before you can compile and run the sample application, you must [configure the Eclipse project](#) to work with the BlackBerry Push Service SDK.

Configure an Eclipse project

To configure an Eclipse® project to work with the BlackBerry® Push Service SDK, you must import the BlackBerry Push Service SDK libraries, the third-party libraries, and the configuration files into your project. You can also import the Javadoc JAR files to enable the Javadoc comments to appear when you hover over a BlackBerry Push Service SDK class or method with your cursor.

1. In Eclipse, in the **Project Explorer** window, right-click the project and click **Properties**.
2. In the left pane, click **Java Build Path**.
3. Add the BlackBerry Push Service SDK libraries to the project by performing the following actions:
 - a. On the **Libraries** tab, click **Add External JARs**.
 - b. Navigate to the **components** folder and double-click a JAR file for a component that your push application requires (for Windows®, the default location is C:\Program Files\BPSS\pushsdk\components).
 - c. Repeat substeps a and b for each component that your push application requires.
4. To add the Javadoc comments for the BlackBerry Push Service SDK libraries to the project, perform the following actions:
 - a. Expand a BlackBerry Push Service SDK library that you added and click **Javadoc location**.
 - b. Click **Edit**.
 - c. Click **Browse**, navigate to the **Javadoc** folder and click a folder that corresponds to a library that you added (for Windows, the default location is C:\Program Files\BPSS\pushsdk\javadoc).
 - d. Click **OK**.
 - e. Click **OK**.
 - f. Repeat substeps a to e for each library.
5. Add the third party libraries to the project by performing the following actions:
 - a. On the **Libraries** tab, click **Add External JARs**.
 - b. Navigate to the third-party dependencies folder and double-click a JAR file for a third-party library that your push application requires (for Windows, the default location is C:\Program Files\BPSS\pushsdk\dependencies-lib).
 - c. Repeat substeps a and b for each third party library that your push application requires.
6. Add the configuration files to the project by performing the following actions:
 - a. In the **Project Explorer** window, right-click the project and click **Import**.
 - b. Expand **General** and double-click **Filesystem**.
 - c. Click **Browse**.
 - d. Navigate to the configuration folder and click **OK** (for Windows, the default location is C:\Program Files\BPSS\pushsdk\configuration).
 - e. Select the check box for each file in the **configuration** folder and click **Finish**.

Glossary

14

API

application programming interface

HTTP

Hypertext Transfer Protocol

HTTPS

Hypertext Transfer Protocol over Secure Sockets Layer

IPPP

Internet Protocol Proxy Protocol

PAP

Push Access Protocol

PPG

Push Proxy Gateway

Legal notice

15

©2010 Research In Motion Limited. All rights reserved. BlackBerry®, RIM®, Research In Motion®, SureType®, SurePress™ and related trademarks, names, and logos are the property of Research In Motion Limited and are registered and/or used in the U.S. and countries around the world.

Apache Tomcat is a trademark of The Apache Software Foundation. Eclipse is a trademark of Eclipse Foundation, Inc. Java is a trademark of Sun Microsystems, Inc. MySQL is a trademark of MySQL AB. Oracle is a trademark of Oracle corporation. All other trademarks are the property of their respective owners.

The BlackBerry smartphone and other devices and/or associated software are protected by copyright, international treaties, and various patents, including one or more of the following U.S. patents: 6,278,442; 6,271,605; 6,219,694; 6,075,470; 6,073,318; D445,428; D433,460; D416,256. Other patents are registered or pending in the U.S. and in various countries around the world. Visit www.rim.com/patents for a list of RIM (as hereinafter defined) patents.

This documentation including all documentation incorporated by reference herein such as documentation provided or made available at www.blackberry.com/go/docs is provided or made accessible "AS IS" and "AS AVAILABLE" and without condition, endorsement, guarantee, representation, or warranty of any kind by Research In Motion Limited and its affiliated companies ("RIM") and RIM assumes no responsibility for any typographical, technical, or other inaccuracies, errors, or omissions in this documentation. In order to protect RIM proprietary and confidential information and/or trade secrets, this documentation may describe some aspects of RIM technology in generalized terms. RIM reserves the right to periodically change information that is contained in this documentation; however, RIM makes no commitment to provide any such changes, updates, enhancements, or other additions to this documentation to you in a timely manner or at all.

This documentation might contain references to third-party sources of information, hardware or software, products or services including components and content such as content protected by copyright and/or third-party web sites (collectively the "Third Party Products and Services"). RIM does not control, and is not responsible for, any Third Party Products and Services including, without limitation the content, accuracy, copyright compliance, compatibility, performance, trustworthiness, legality, decency, links, or any other aspect of Third Party Products and Services. The inclusion of a reference to Third Party Products and Services in this documentation does not imply endorsement by RIM of the Third Party Products and Services or the third party in any way.

EXCEPT TO THE EXTENT SPECIFICALLY PROHIBITED BY APPLICABLE LAW IN YOUR JURISDICTION, ALL CONDITIONS, ENDORSEMENTS, GUARANTEES, REPRESENTATIONS, OR WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY CONDITIONS, ENDORSEMENTS, GUARANTEES, REPRESENTATIONS OR WARRANTIES OF DURABILITY, FITNESS FOR A PARTICULAR PURPOSE OR USE, MERCHANTABILITY, MERCHANTABLE QUALITY, NON-INFRINGEMENT, SATISFACTORY QUALITY, OR TITLE, OR ARISING FROM A STATUTE OR CUSTOM OR A COURSE OF DEALING OR USAGE OF TRADE, OR RELATED TO THE DOCUMENTATION OR ITS USE, OR PERFORMANCE OR NON-PERFORMANCE OF ANY SOFTWARE, HARDWARE, SERVICE, OR ANY THIRD PARTY PRODUCTS AND SERVICES REFERENCED HEREIN, ARE HEREBY EXCLUDED. YOU MAY ALSO HAVE OTHER RIGHTS THAT VARY BY STATE OR PROVINCE. SOME JURISDICTIONS MAY NOT ALLOW THE EXCLUSION OR LIMITATION OF IMPLIED WARRANTIES AND CONDITIONS. TO THE EXTENT PERMITTED BY LAW, ANY IMPLIED WARRANTIES OR CONDITIONS RELATING TO THE DOCUMENTATION TO THE EXTENT THEY CANNOT BE EXCLUDED AS SET OUT ABOVE, BUT CAN BE LIMITED, ARE HEREBY LIMITED TO NINETY (90) DAYS FROM THE DATE YOU FIRST ACQUIRED THE DOCUMENTATION OR THE ITEM THAT IS THE SUBJECT OF THE CLAIM.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, IN NO EVENT SHALL RIM BE LIABLE FOR ANY TYPE OF DAMAGES RELATED TO THIS DOCUMENTATION OR ITS USE, OR PERFORMANCE OR NON-PERFORMANCE OF ANY SOFTWARE, HARDWARE, SERVICE, OR ANY THIRD PARTY PRODUCTS AND SERVICES REFERENCED HEREIN INCLUDING WITHOUT LIMITATION ANY OF THE FOLLOWING DAMAGES: DIRECT, CONSEQUENTIAL, EXEMPLARY, INCIDENTAL, INDIRECT, SPECIAL, PUNITIVE, OR AGGRAVATED DAMAGES, DAMAGES FOR LOSS OF PROFITS OR REVENUES, FAILURE TO REALIZE ANY EXPECTED SAVINGS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF BUSINESS OPPORTUNITY, OR CORRUPTION OR LOSS OF DATA, FAILURES TO TRANSMIT OR RECEIVE ANY DATA, PROBLEMS ASSOCIATED WITH ANY APPLICATIONS USED IN CONJUNCTION WITH RIM PRODUCTS OR SERVICES, DOWNTIME COSTS, LOSS OF THE USE OF RIM PRODUCTS OR SERVICES OR ANY PORTION THEREOF OR OF ANY AIRTIME SERVICES, COST OF SUBSTITUTE GOODS, COSTS OF COVER, FACILITIES OR SERVICES, COST OF CAPITAL, OR OTHER SIMILAR PECUNIARY LOSSES, WHETHER OR NOT SUCH DAMAGES WERE FORESEEN OR UNFORESEEN, AND EVEN IF RIM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW IN YOUR JURISDICTION, RIM SHALL HAVE NO OTHER OBLIGATION, DUTY, OR LIABILITY WHATSOEVER IN CONTRACT, TORT, OR OTHERWISE TO YOU INCLUDING ANY LIABILITY FOR NEGLIGENCE OR STRICT LIABILITY.

THE LIMITATIONS, EXCLUSIONS, AND DISCLAIMERS HEREIN SHALL APPLY: (A) IRRESPECTIVE OF THE NATURE OF THE CAUSE OF ACTION, DEMAND, OR ACTION BY YOU INCLUDING BUT NOT LIMITED TO BREACH OF CONTRACT, NEGLIGENCE, TORT, STRICT LIABILITY OR ANY OTHER LEGAL THEORY AND SHALL SURVIVE A FUNDAMENTAL BREACH OR BREACHES OR THE FAILURE OF THE ESSENTIAL PURPOSE OF THIS AGREEMENT OR OF ANY REMEDY CONTAINED HEREIN; AND (B) TO RIM AND ITS AFFILIATED COMPANIES, THEIR SUCCESSORS, ASSIGNS, AGENTS, SUPPLIERS (INCLUDING AIRTIME SERVICE PROVIDERS), AUTHORIZED RIM DISTRIBUTORS (ALSO INCLUDING AIRTIME SERVICE PROVIDERS) AND THEIR RESPECTIVE DIRECTORS, EMPLOYEES, AND INDEPENDENT CONTRACTORS.

IN ADDITION TO THE LIMITATIONS AND EXCLUSIONS SET OUT ABOVE, IN NO EVENT SHALL ANY DIRECTOR, EMPLOYEE, AGENT, DISTRIBUTOR, SUPPLIER, INDEPENDENT CONTRACTOR OF RIM OR ANY AFFILIATES OF RIM HAVE ANY LIABILITY ARISING FROM OR RELATED TO THE DOCUMENTATION.

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 might not offer Internet browsing functionality with a subscription to the 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, and/or BlackBerry® Device Software.

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.

Research In Motion Limited
295 Phillip Street
Waterloo, ON N2L 3W8
Canada

Research In Motion UK Limited
Centrum House
36 Station Road
Egham, Surrey TW20 9LF
United Kingdom

Published in Canada