SYBASE®

An **SAP** Company

**Tutorial: BlackBerry Object API
Application Development**

Sybase Unwired Platform 2.2 SP02

Contents

Contents

Sybase Unwired Platform Tutorials

The Sybase® tutorials demonstrate how to develop, deploy, and test mobile business objects, device applications, and Hybrid App packages. You can also use the tutorials to demonstrate system functionality and train users.

- Learn mobile business object (MBO) basics, and use this tutorial as a foundation for the Object API application development tutorials:
  - *Tutorial: Mobile Business Object Development*

    Note: For all Object API tutorials, if you opt to use the Mobile Business Object example project instead of performing the Mobile Business Object Tutorial, you must deploy the mobile application project to Unwired Server as a prerequisite.
- Create native Object API mobile device applications:
  - *Tutorial: Android Object API Application Development*
  - *Tutorial: BlackBerry Object API Application Development*
  - *Tutorial: iOS Object API Application Development*
  - *Tutorial: Windows Object API Application Development*
  - *Tutorial: Windows Mobile Object API Application Development*
- Create a mobile business object, then develop a hybrid app package that uses it:
  - *Tutorial: Hybrid App Package Development*

Getting Started with Unwired Platform

Install and learn about Sybase Unwired Platform and its associated components.

Complete the following tasks for all tutorials, but you need to perform them only once.

Installing Sybase Unwired Platform

Install Sybase Mobile SDK and Unwired Platform Runtime.

Before starting this tutorial, install all the requisite Unwired Platform components. See the Sybase Unwired Platform documentation at *http://sybooks.sybase.com*:

- *Release Bulletin*
- *Installation Guide for Sybase Mobile SDK*
- *Installation Guide for Runtime*

1. Install these Unwired Platform Runtime components:
   - Data Tier (included with single-server installation)
   - Unwired Server
2. Install Mobile SDK, which includes:
   - Development support for native Object API applications, HTML5/JS Hybrid Apps, and native OData SDK applications.
   - Sybase Unwired WorkSpace, the Eclipse-based development environment for MBOs and Hybrid Apps.

Starting Sybase Unwired Platform Services

Start Unwired Server, Sybase Control Center, the sample database, the cache database (CDB), and other essential services.

The way in which you start Unwired Platform services depends on the options you selected during installation. You may need to manually start Unwired Platform services.
Select **Start > Programs > Sybase > Unwired Platform > Start Unwired Platform Services**.
The Unwired Platform services enable you to access the Unwired Platform runtime components and resources.

Starting Sybase Unwired WorkSpace

Start the development environment, where you can create mobile business objects (MBOs), manage enterprise information system (EIS) datasources and Unwired Server connections, develop Hybrid App applications (Hybrid Apps), and generate Object API code.

Select **Start > Programs > Sybase > Unwired Platform > Sybase Unwired WorkSpace**. The Sybase Unwired WorkSpace opens in the Mobile Development perspective. The Welcome page displays links to the product and information.

Next
To read more about Sybase Unwired WorkSpace concepts and tasks, select **Help > Help Contents**.

Connecting to Sybase Control Center

Open Sybase Control Center to manage Unwired Server and its components.

From Sybase Control Center, you can:

- View servers and their status
- Start and stop a server
- View server logs
- Deploy a mobile application package
- Register application connections
- Set role mappings
- Assign/Unassign a hybrid application to a device

For information on configuring, managing, and monitoring Unwired Server, click **Help > Online Documentation**.

1. Select **Start > Programs > Sybase > Sybase Control Center**.

   Note: If Sybase Control Center does not launch, make sure that the Sybase Control Center service is started in the Windows Services dialog.

2. Log in by entering the credentials set during installation.

   Sybase Control Center gives you access to the Unwired Platform administration features that you are authorized to use.

Learning Sybase Unwired WorkSpace Basics

Sybase Unwired WorkSpace features are well integrated in the Eclipse IDE. If you are unfamiliar with Eclipse, you can quickly learn the basic layout of Sybase Unwired WorkSpace and the location of online help.

- To access the online help, select **Help > Help Contents**. Some documents are for Sybase Unwired Platform, while others are for the Eclipse development environment.
- The Welcome page provides links to useful information to get you started.
    - To close the Welcome page, click **X**.
    - Reopen the Welcome page by selecting **Help > Welcome**.
    - To learn about tasks you must perform, select the **Development Process** icon.
- In Sybase Unwired WorkSpace, look at the area (window or view) that you will use to access, create, define, and update mobile business objects (MBOs).

Window	Description
WorkSpace Navigator view	Use this view to create Mobile Application projects, and review and modify MBO-related properties.
	This view displays mobile application project folders, each of which contains all project-related resources in subfolders, including MBOs, datasource references to which the MBOs are bound, personalization keys, and so on.
Enterprise Explorer view	A view that provides functionality to connect to various enterprise information systems (EIS), such as database servers, SAP® back ends, and Unwired Server.

Window	Description
Mobile Application Diagram	The Mobile Application Diagram is a graphical editor where you create and define mobile business objects. Use the Mobile Application Diagram to create MBOs (including attributes and operations), then define relationships with other MBOs. You can: • Create MBOs in the Mobile Application Diagram using Palette icons and menu selections – either bind or defer binding to a datasource, when creating an MBO. For example, you may want to model your MBOs before creating the datasources to which they bind. This MBO development method is sometimes referred to as the top-down approach. • Drag and drop items from Enterprise Explorer to the Mobile Application Diagram to create the MBO – quickly creates the operations and attributes automatically based on the datasource artifact being dropped on the Mobile Application Diagram. Each new mobile application project generates an associated mobile application diagram.
Palette	The Palette is accessed from the Mobile Application Diagram and provides controls, such as the ability to create MBOs, add attributes and operations, and define relationships, by dragging and dropping the corresponding icon onto the Mobile Application Diagram or existing MBO.
Properties view	Select an object in the Mobile Application Diagram to display and edit its properties in the Properties view. While you cannot create an MBO from the Properties view, most development and configuration is performed here.
Outline view	Displays an outline of the active file and lists structural elements. The contents are editor-specific.

Window	Description
Problems view	Displays validation errors or warnings that you may encounter in addition to errors in the Diagram editor and Properties view. Follow warning and error messages to adjust MBO properties and configurations to avoid problems, and use as a valuable source for collecting troubleshooting information when reporting issues to Customer Service and Support.
Error Log view	Displays error log information. This is a valuable source for collecting troubleshooting information.

Developing a BlackBerry Application

Generate code for the BlackBerry platform, develop a BlackBerry device application using that code and sample files, and test the application's functionality on a simulator.

Prerequisites

Note: This tutorial was created using Sybase Unwired Platform 2.2 SP02, BlackBerry Java SDK 7.1.0.10, BlackBerry Java Plug-in (core) 2.0.0, and run on an BlackBerry SDK 9900 device emulator. If you use a different version, some steps may vary.

1. Complete the tasks in *Getting Started with Unwired Platform*.
2. Either:
   - create the MBO project by completing *Tutorial: Mobile Business Object Development*, or
   - download and deploy the MBO SUP101 example project (complete project files) from the SAP® Community Network: *http://scn.sap.com/docs/DOC-8803*.

   Note: If you upgrade Sybase Mobile SDK after completing the tutorial, you can convert the project to the current SDK by importing the earlier project into the Unwired Workspace and then accepting the confirmation prompt.
3. (Optional) To use as a reference and copy source code when completing this tutorial, download the BlackBerry SUP 101 example project (source code only) from the SAP® Community Network: *http://scn.sap.com/docs/DOC-8803*.
4. Download the supported versions of the BlackBerry Java Plug-in for Eclipse.
   See the *Supported Hardware and Software* guide at *http://sybooks.sybase.com/sybooks/ sybooks.xhtml*. Navigate to the appropriate version of Sybase Unwired Platform.

Task

Create a BlackBerry native application that communicates with the mobile business objects that are deployed to Unwired Server.

Installing the BlackBerry Java Plug-in for Eclipse

Install the supported version of BlackBerry Java Plug-in in the Sybase Unwired WorkSpace Eclipse environment.

See *RIM BlackBerry Versions for Object API* at *http://sybooks.sybase.com/sybooks/ sybooks.xhtml*. Navigate to the appropriate version of Sybase Unwired Platform.

The BlackBerry Java Plug-in for Eclipse enables you to finish developing the BlackBerry application using smartphone-specific development, debugging, and simulation tools.

1. Confirm that your system meets the requirements at *https://developer.blackberry.com/ java/download/eclipse/.*

2. Start Sybase Unwired WorkSpace, then select **Help > Install New Software**.

3. In the Available Software window, click **Add**.

4. In the Add Repository dialog, enter `BlackBerry Plugin` for the name and `http:// www.blackberry.com/developers/jar/win/java` for the location. Click **OK**.

5. In the Available Software dialog, select **BlackBerry Java Plug-in (core)** and the appropriate version of the **BlackBerry Java SDK**, for example, 7.1.0.10, then click **Next**.

6. Review the items to be installed, then click **Next** again.

7. Accept the license agreements, then click **Finish**.

   Note: If you get a security warning about the authenticity or validity of the software, click **OK**.

8. When the installation completes, restart Sybase Unwired WorkSpace.

9. Click **Finish**.

Generating Java Object API Code

Use the Generate Code wizard to generate object API code for the SUP101 mobile application project. Code generation creates the business logic, attributes, and operations for the mobile business objects (MBOs) in the project.

Prerequisites
In Enterprise Explorer, you must be connected to both My Sample Database and My Unwired Server. Code generation fails if the server-side (runtime) enterprise information system (EIS) datasources referenced by the MBOs in the project are not running and available to connect to when you generate object API code.

Task

1. In Sybase Unwired WorkSpace, open the **SUP101** mobile application project.

   In WorkSpace Navigator, right-click the **SUP101** folder and select **Open in Diagram Editor**.

2. In WorkSpace Navigator, expand **SUP101**. Under **Generated Code** add a folder named `BlackBerry`.

   The `Generated Code` directory was created during the MBO tutorial.

3. Right-click anywhere in the SUP101 - Mobile Application Diagram and select **Generate Code**.

4. In the Generate Code wizard, click **Next** to select the default configuration and continue without any changes.

5. In the Select Mobile Business Objects window, select the **Customer** MBO, then click **Next**.

Ignore any warning about unresolved mobile business object dependencies.

6. In the Configure Options window, specify these values and click **Finish**.

Option	Description
Language	Select **Java**.
Platform	Select **Java ME for BlackBerry**.
Unwired server	Select **My Unwired Server**.
Server domain	Accept default value.
Page size	Accept default value of 1024.
Package	Enter com.mycorp.sup101.bb.mbo.
Project path	Enter \SUP101\Generated Code \BlackBerry.
Third-party jar file	You do not need to choose one for this tutorial. You see a warning at the top of the page: The dependent third-party class 'net.rim.device.api.sys-tem.ApplicationDescriptor' cannot be found. You can ignore this warning; it does not affect code generation.
Generate Metadata Classes	Select for this tutorial.
Generate JavaDoc	Unselect for this tutorial.
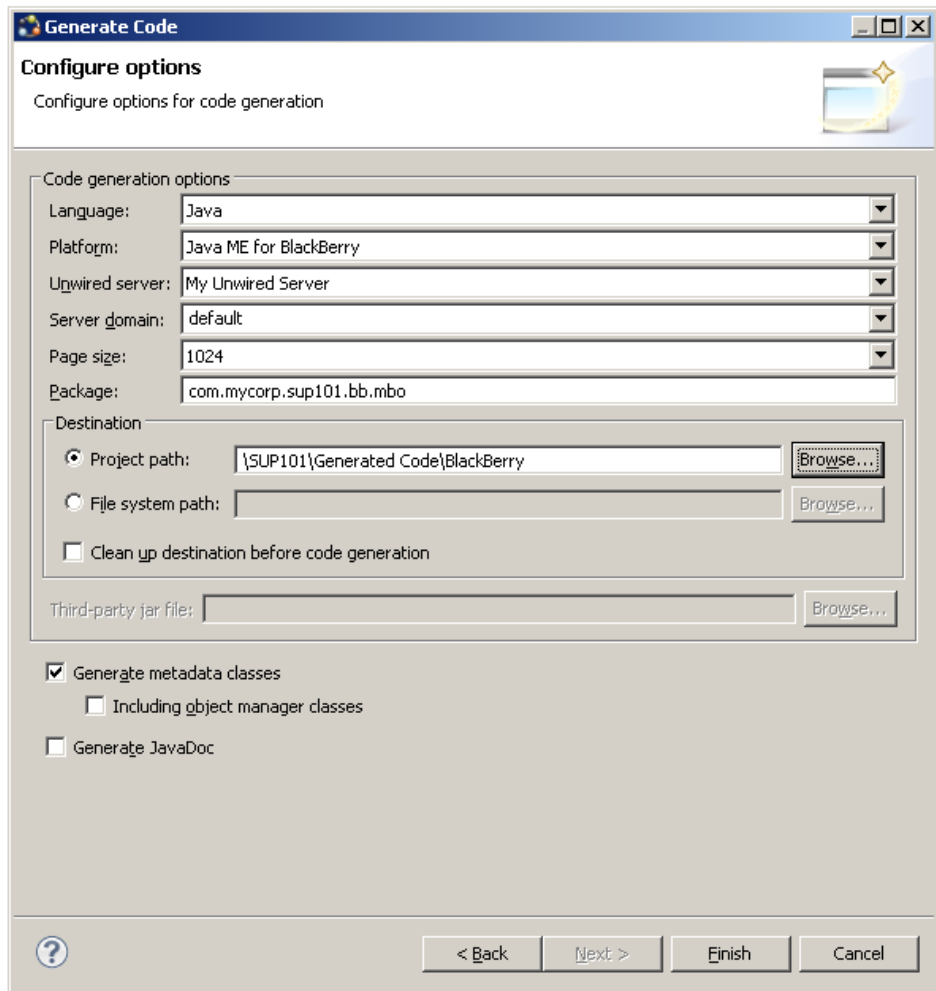
You can ignore the error icons that appear in WorkSpace Navigator under
`SUP101\Generated Code\BlackBerry\src\`, and the error in the Problems
view: `net cannot be resolved`.

The error occurs because the package Java file called by `SUP101DB.jar` is not yet in the
project build path. The errors resolve later, when you build the application in the
BlackBerry Java Plug-in for Eclipse.

7. In the Success dialog, click **OK**.
In the `Generated Code` directory, you see a `\BlackBerry\src\` folder.

Creating the BlackBerry Project
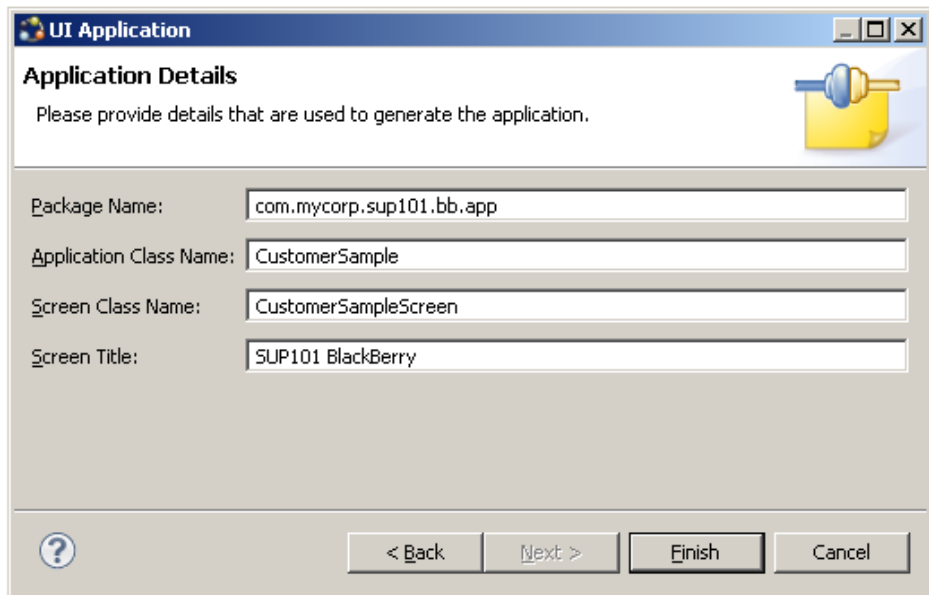
Create a new BlackBerry project in Unwired WorkSpace. Add library resources to the project and set other application properties.

Prerequisites

To help create your project, and in a subsequent topic, build the user interface, download the SUP101 BlackBerry Object API (2.2 SP02) example project from the SAP Community Network (SCN) Web site at *http://scn.sap.com/docs/DOC-8803*.

Task

1. Start Unwired WorkSpace.
2. Select **File > New > Project**.
3. In the New Project window, select **BlackBerry Project** and click **Next**.
4. In the Create a BlackBerry Project window, enter SUP101BlackBerry for the project name and click **Next**.
5. In the Java Settings page, modify the build path to the sup_client2.jar and UltraLiteJ12.jar files:
    a) Click the **Libraries** tab.
    b) Click **Add External JARs**.
    c) Browse to *SUP_HOME*\MobileSDK22\ObjectAPI\BB.
    d) Select the two JAR files, then click **Open**.
    e) Click the **Order and Export** tab.
    f) Make sure the sup_client2.jar is selected.
6. Click **Next**.
7. In the Templates window, accept the default BlackBerry Application and click **Next**.
8. In the Application Details window, enter these values and click **Finish**.
    • Package Name – com.mycorp.sup101.bb.app
    • Application Class Name – CustomerSample
    • Screen Class Name – CustomerSampleScreen
   • Screen Title – SUP101 BlackBerry

9. If prompted to use a different perspective, click **No**.

Configuring BlackBerry Application Properties

Modify the BlackBerry application description to define the general BlackBerry properties used in an application.

1. In WorkSpace Navigator, expand the **SUP101BlackBerry** project.

2. Double-click the BlackBerry_App_Descriptor.xml file.

3. Select the **Application** tab.

4. In the Title, enter SUP101 BlackBerry.

5. (Optional) Enter a version, vendor, and description.
You can review the other options in the Application Descriptor window, such as general attributes, resources, and build settings of the project. For this tutorial, leave all other settings unchanged.
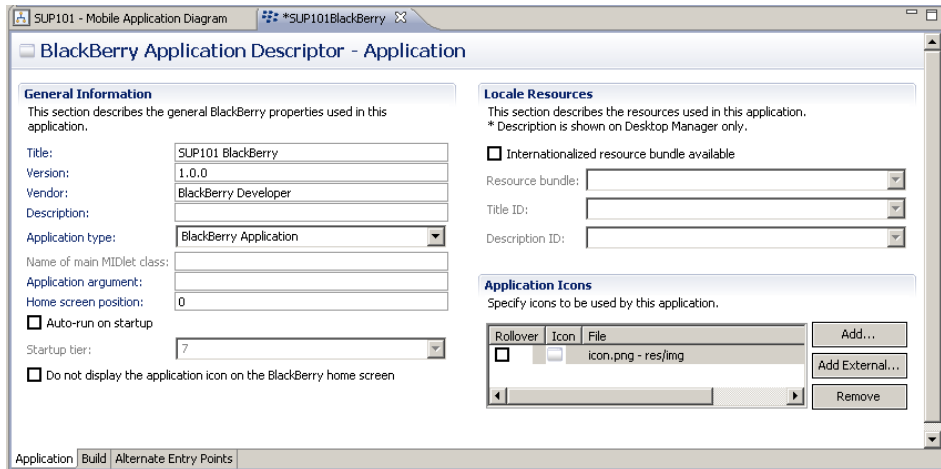
6. Select **File > Save**.

## Copying Unwired Platform Files to the Sample Project

Copy Unwired Platform BlackBerry runtime files and the object API code you generated using the Generate Code wizard to the SUP101BlackBerry project.

1. In Windows Explorer, browse to *SUP_HOME*\MobileSDK22\ObjectAPI\BB.

2. Copy the UltraLiteJ12.cod file.

3. Paste the copied COD file into the BlackBerry Java Plug-in simulator directory, located in *SUP_HOME*\MobileSDK22\Eclipse\plugins \net.rim.ejde.componentpack<*version*>\components\simulator.

4. Generate application key file SUP.key with BlackBerrySigningAuthorityTool_1.0.exe, and copy it to project root folder.

   See BlackBerry Password Based Code Signing Authority at: *http:// supportforums.blackberry.com/t5/Java-Development/Protect-persistent-objects-from- access-by-unauthorized/ta-p/524282.*

5. In WorkSpace Navigator, select the **SUP101** project and copy the com folder under \Generated Code\BlackBerry\src.

6. In WorkSpace Navigator, go to the **SUP101BlackBerry** project and paste the com folder in to the src directory. Select **Yes to All** to copy over existing folders.

Creating the User Interface

Copy the Java code files, which provide the functionality and layout of the user interface, from the SUP101 BlackBerry Project example project archive to the SUP101 BlackBerry project.

1. In Windows Explorer, browse to the directory where you saved the SUP101 BlackBerry Project example project file.
2. Copy these Java files:
   - `CustomerDBCallback.java` – implements the CallbackHandler to demonstrate how to track changed entities in the onSynchronize callback method.
   - `CustomerList.java` – populates the customer list.
   - `CustomerSample.java` – creates the main customer application.
   - `CustomerSampleScreen.java` – creates the customer screen.
   - `ObjectAPIUtil.java` – governs how the application initializes and synchronizes data in the background.
3. In WorkSpace Navigator, go to **SUP101BlackBerry** and expand `\src\com\mycorp\sup101\bb\app`, then paste the copied Java files, copying over any existing files.
4. Modify the host name or IP address in the `ObjectAPIUtil.java` file to point to the Unwired Server.
   a) In WorkSpace Navigator, expand the **SUP101BlackBerry** project.
   b) Under the `\src\com\mycorp\sup101\bb\app` folder, double-click the `ObjectAPIUtil.java` file.
   c) Modify the host name or IP address, and verify the username and password are valid.
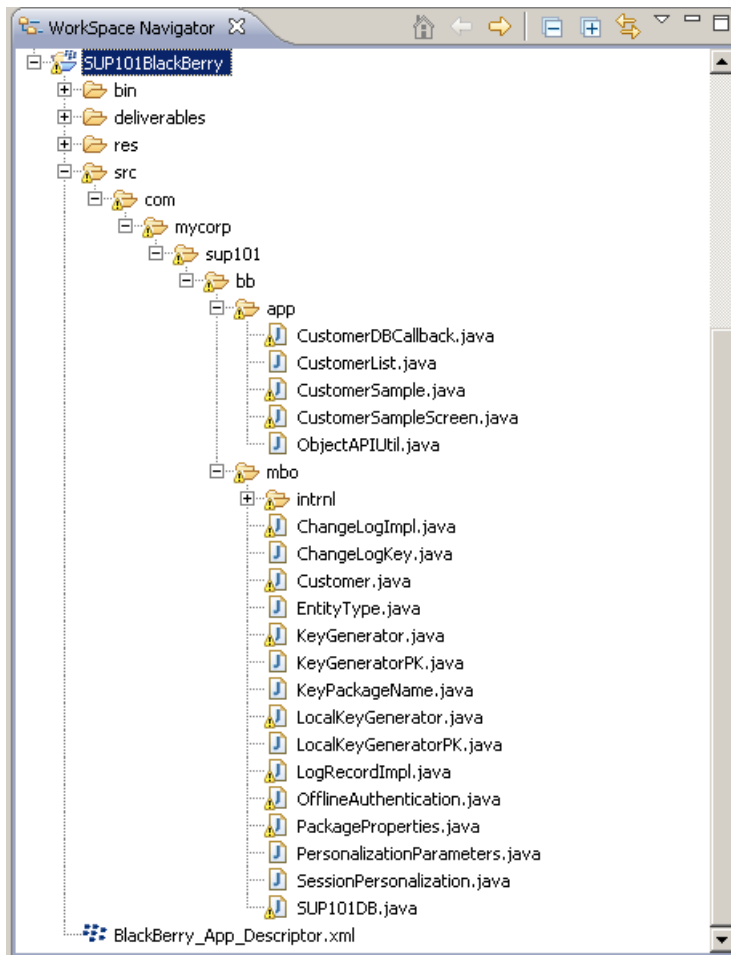
Creating a Launch Configuration for the Project

Create a new launch configuration for the SUP101BlackBerry project. The configuration specifies how the application launches, and defines the target BlackBerry platform.

1. In WorkSpace Navigator, right-click the **SUP101BlackBerry** project, and select **Run As > Run Configurations**.

2. Select **BlackBerry Simulator** and the **New launch configuration** icon.
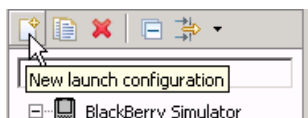
3. In the Projects tab, select the **SUP101BlackBerry** project.

4. In the Name field, enter `SUP101BlackBerry`.

5. In the JRE tab, select the **Project JRE (BlackBerry JRE *Version*)**.

6. In the **Simulator** tab, select **Launch Mobile Data System Connection Service (MDS-CS) with simulator**. Then select **BlackBerry-SDK - *Version*** as the device.
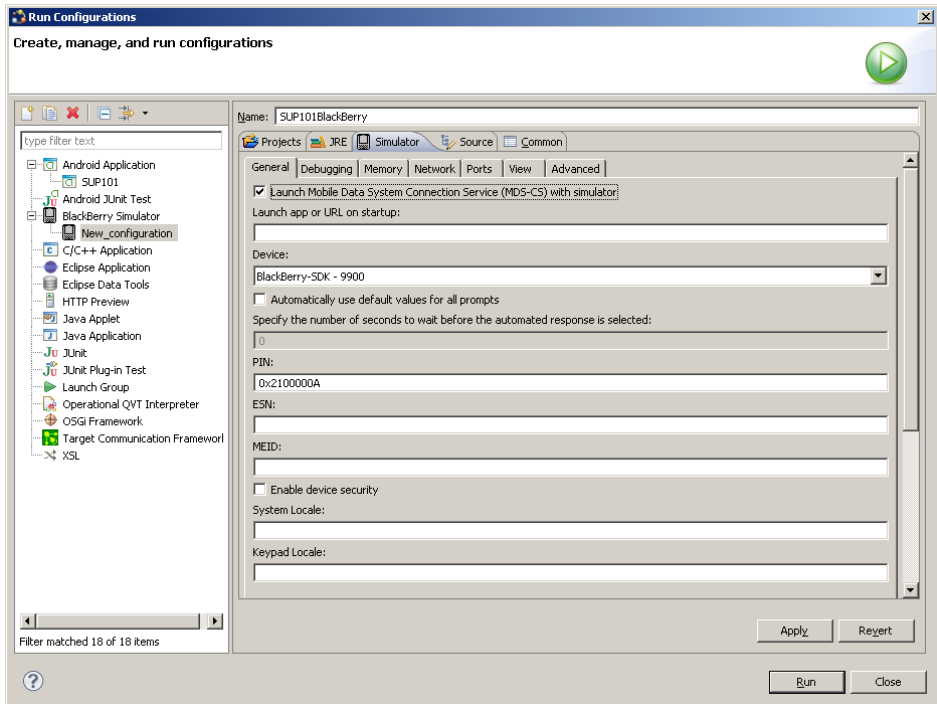
If you are running the BlackBerry JDE 7.0 with the BlackBerry Java Plug-in v1.5, you must launch the Mobile Data System (MDS) manually for the BlackBerry simulator to run.

MDS is a framework that offers security, wireless connectivity, manageability options, and development methods. You need it to run the simulator. It is a known issue, which has been fixed in BlackBerry JDE 7.1, that the simulator cannot launch MDS.

a) From a command prompt, go to `<Eclipse_InstallDir>\plugins \net.rim.ejde.componentpack7.0.0_7.0.0.33\components \MDS`.

b) Start MDS by entering:

```
start run.bat
```

c) Press any key to bypass this message.

```
ERROR: Your java version does not meet the minimum
requirement of 1.6.0_20. Press any key to continue...
```

MDS starts, and you can now launch the BlackBerry simulator. Leave the command window open.

7. In the **Network** tab, verify that **Disable Registration** is selected. If it is not, the sample application cannot get data from Unwired Server.

8. Click **Apply**, then **Close**.

Testing the Device Application on the BlackBerry Simulator

Run the SUP101BlackBerry application on the BlackBerry simulator, and change customer information to update the interface.

1. In WorkSpace Navigator, right-click **SUP101BlackBerry** and select **Run As > BlackBerry Simulator**.
   It may take several minutes for the BlackBerry simulator's Setup screen to appear. If this is the first time you have run the simulator, cancel the Setup screen.

2. On the main window, click **All** to access the applications screen, then scroll until you see the SUP101BlackBerry application.

3. Click **SUP101 BlackBerry** to launch the application.

The application registers and synchronizes data from the server in the background.

During initialization, the system enables the operation to target change notifications using:

```
SynchronizationGroup
sg=SUP101DB.getSynchronizationGroup("default");
sg.setEnableSIS(true);
sg.save();
```

After the application initializes, the device application shows the SUP101BlackBerry application with a list of customer data. You can scroll through the customer list to see more data, search, and make changes. The data loads from the database on demand.

When the application queries the customer list, it uses a named query findAll. If the customer list is long, it uses a SUP101DB.executeQuery() API to get only columns that are needed, such as (fname, lname...), instead of the entire customer object; this results in better performance.

4. Select the customer to update.

5. Change the first name of the customer and click **submit**.

The **submit** button maps to the update operation of the customer mobile business object. The synchronize operation uses SUP101DB.beginSynchronize in the background so the user interface is not affected. When the application synchronizes, any pending operations are uploaded to Unwired Server.

Any back-end changes initiate notifications from the server. The device application uses a ChangeLog API, specifically ObjectList changeLogs = SUP101DB.getChangeLogs(query);, to query those managed items and update the user interface if needed.

The customer list appears, and shows the name change you made.

6. Close the simulator to stop the SUP101BlackBerry application.

Learn More About Sybase Unwired Platform

Once you have finished, try some of the other samples or tutorials, or refer to other development documents in the Sybase Unwired Platform documentation set.

Check the Sybase® Product Documentation Web site regularly for updates: *http:// sybooks.sybase.com/sybooks/sybooks.xhtml? id=1289&c=firsttab&a=0&p=categories* , then navigate to the most current version.

Tutorials
Try out some of the other getting started tutorials available on the Product Documentation Web site to get a broad view of the development tools available to you.

Example Projects
An example project contains source code for its associated tutorial. It does not contain the completed tutorial project. Download example projects from the SAP® Community Network (SCN) at *http://scn.sap.com/docs/DOC-8803*.

Samples
Sample applications are fully developed, working applications that demonstrate the features and capabilities of Sybase Unwired Platform.

Check the SAP® Development Network (SDN) Web site regularly for new and updated samples: *https://cw.sdn.sap.com/cw/groups/sup-apps*.

Online Help
See the online help that is installed with the product, or available from the Product Documentation Web site.

Developer Guides
Learn best practices for architecting and building device applications:

- *Mobile Data Models: Using Data Orchestration Engine* – provides information about using Sybase Unwired Platform features to create DOE-based applications.
- *Mobile Data Models: Using Mobile Business Objects* – provides information about developing mobile business objects (MBOs) to fully maximize their potential.

Use the appropriate API to create device applications:

- *Developer Guide: Android Object API Applications*
- *Developer Guide: BlackBerry Object API Applications*
- *Developer Guide: iOS Object API Applications*
- *Developer Guide: Windows and Windows Mobile Object API Applications*

- *Developer Guide: Hybrid Apps*

Customize and automate:

- *Developer Guide: Unwired Server Runtime > Management API* – customize and automate system administration features.

Javadoc and HeaderDoc are also available in the installation directory.

Index

T

U