



**Tutorial: BlackBerry Object API
Application Development**

Sybase Unwired Platform 2.1

ESD #2

DOCUMENT ID: DC01214-01-0212-01

LAST REVISED: March 2012

Copyright © 2012 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase trademarks can be viewed at the Sybase trademarks page at <http://www.sybase.com/detail?id=1011207>. Sybase and the marks listed are trademarks of Sybase, Inc. ® indicates registration in the United States of America.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

Java and all Java-based marks are trademarks or registered trademarks of Oracle and/or its affiliates in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

Sybase Unwired Platform Tutorials	1
Task Flow	3
Getting Started with Unwired Platform	5
Installing Sybase Unwired Platform	5
Starting Sybase Unwired Platform Services	5
Starting Sybase Unwired WorkSpace	6
Connecting to Sybase Control Center	6
Learning Unwired WorkSpace Basics	7
Developing a BlackBerry Application	11
Installing the BlackBerry Java Plug-in for Eclipse	11
Generating Java Object API Code	11
Creating the BlackBerry Project	13
Configuring BlackBerry Application Properties	15
Copying Unwired Platform Files to Sample Project	15
Creating the User Interface	16
Creating a Launch Configuration for the Project	19
Testing the Device Application on the BlackBerry Simulator	21
Learn More About Sybase Unwired Platform	27
Index	29

Contents

Sybase Unwired Platform Tutorials

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

Tip: If you want to see the final outcome of a tutorial without performing the steps, the associated example project is available on SAP® SDN: <http://www.sdn.sap.com/irj/sdn/mobile?rid=/webcontent/uuid/40ea4956-b95c-2e10-11b3-e68c73b2280e>.

- 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*
- 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 Mobile Object API Application Development*
- Create a mobile business object, then develop a mobile workflow package that uses it:
 - *Tutorial: Mobile Workflow Package Development*

Task Flow

Use this tutorial to develop, deploy, and test a mobile device application on a simulator or an emulator.

Task	Goals	Steps
Getting started	<ul style="list-style-type: none"> • Install Sybase Mobile SDK and Sybase Unwired Platform Runtime. • Start Unwired Server and other platform services, if not already started. • Start Sybase Control Center. • Start Sybase Unwired WorkSpace, open the Mobile Development perspective, and become familiar with the views of the perspective and the Mobile Application Diagram. 	<ul style="list-style-type: none"> • Installing Sybase Unwired Platform • Starting Unwired Platform Services • Starting Sybase Unwired WorkSpace • Connecting to Sybase Control Center • (Optional) Learning Unwired WorkSpace Basics <p>These steps are prerequisites for the rest of this tutorial. You need to perform them only once.</p>
Developing database mobile business objects	<ul style="list-style-type: none"> • Create a mobile application project and a connection to the database. • Create two mobile business objects, and create a relationship between them. • Deploy the mobile business objects to Unwired Server. 	<p>Complete the <i>Tutorial: Mobile Business Object Development</i>, or obtain the completed example project.</p> <hr/> <p>Note: This tutorial is a prerequisite for the remaining steps. You need to perform it only once. If you want to download the final outcome of a tutorial without performing it, the associated example project is available on SAP® SDN: http://www.sdn.sap.com/irj/sdn/mobile?rid=/webcontent/uuid/40ea4956-b95c-2e10-11b3-e68c73b2280e</p>

Task Flow

Task	Goals	Steps
Developing a native device application	<ul style="list-style-type: none">• Install the BlackBerry Java Plug-in for Eclipse• Generate Java Object API code• Create the user interface• Test the device application on the BlackBerry simulator	<ul style="list-style-type: none">• Installing the BlackBerry Java Plug-in for Eclipse• Generating Java Object API Code• Creating the BlackBerry Project• Creating the User Interface• Creating a Launch Configuration for the Project• Testing the Device Application on the BlackBerry Simulator

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 Sybase Unwired Platform Runtime.

Before starting this tutorial, be sure you have all the requisite Unwired Platform components installed. For complete installation instructions, see the Sybase Unwired Platform documentation at <http://sybooks.sybase.com/nav/summary.do?prod=1289>.

- *Release Bulletin for Sybase Mobile SDK*
- *Installation Guide for Sybase Mobile SDK*
- *Release Bulletin for Runtime*
- *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 (Mobile Workflow) applications, and OData SDK applications.
- Sybase Unwired WorkSpace, the Eclipse-based development environment for MBOs and mobile workflows.

Starting Sybase Unwired Platform Services

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

How you start Unwired Platform services depend on the options you selected during installation. In some cases, you may need to manually start Unwired Platform services. Select **Start > Programs > Sybase > Unwired Platform > Start Unwired Platform Services**.

The Unwired Server services enable you to access the Unwired Platform runtime components and resources.

Starting Sybase Unwired WorkSpace

Start the development environment, where you can perform tasks that include creating mobile business objects (MBOs), managing database and server connections, developing Mobile Workflow applications, and generating Object API code.

Select **Start > Programs > Sybase > Unwired Platform > 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 Unwired WorkSpace concepts and tasks, select **Help > Help Contents**.

Connecting to Sybase Control Center

Open the Sybase Control Center administration console 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

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

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

Note: If the Sybase Control Center service does not open, make sure that the service is started. See the *Installation Guide for Runtime*.

2. In Sybase Control Center, 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 Unwired WorkSpace Basics

Sybase Unwired WorkSpace features are well integrated in the Eclipse IDE. If you are not familiar with Eclipse, you can quickly learn the basic layout of 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 up and running.
 - Reopen the Welcome page by selecting **Help > Welcome**.
 - To close the Welcome page, click **X**.
 - To learn about tasks you must perform, select the **Development Process** icon.
- In 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	<p>Use this view to create Mobile Application projects, and review and modify MBO-related properties.</p> <p>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.</p>
Enterprise Explorer view	<p>A view that provides functionality to connect to various enterprise information systems (EIS), such as database servers, SAP® back ends, and Unwired Server.</p>

Window	Description
Mobile Application Diagram	<p>The Mobile Application Diagram is a graphical editor where you create and define mobile business objects.</p> <p>Use the Mobile Application Diagram to create MBOs (including attributes and operations), then define relationships with other MBOs. You can:</p> <ul style="list-style-type: none"> • Create MBOs in the Mobile Application Diagram using Palette icons and menu selections – either bind or defer binding to a data source, when creating an MBO. For example, you may want to model your MBOs before creating the data sources to which they bind. This MBO development method is sometimes referred to as the top-down approach. • Drag items from Enterprise Explorer and drop them (drag and drop) onto 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. <p>Each new mobile application project generates an associated mobile application diagram.</p>
Palette	<p>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.</p>
Properties view	<p>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.</p>
Outline view	<p>Displays an outline of the active file and lists structural elements. The contents are editor-specific.</p>
Problems view	<p>Displays problems, errors, or warnings that you may encounter. This is a valuable source for collecting troubleshooting information.</p>

Window	Description
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 with that code and sample files, and test the application's functionality on a simulator.

Prerequisites

- Install Sybase Unwired Platform Mobile SDK and Runtime as indicated in *Getting Started* on page 5.
- Complete *Tutorial: Mobile Business Object Development*, which provides the foundation tasks for this tutorial.
- Review the *Supported Hardware and Software* requirements for this release.

Task

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

Installing the BlackBerry Java Plug-in for Eclipse

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

1. Download the BlackBerry Java Plug-in for Eclipse (full installer) from <http://us.blackberry.com/developers/javaappdev/>.
2. Run the downloaded installer and follow the instructions to install the BlackBerry Java Plug-in for Eclipse. Accept the default installation folder: do not specify the Unwired WorkSpace Eclipse directory.

Generating Java Object API Code

Use the Generate Code wizard to generate object API code for the SUP101 mobile application project. The code generation creates the business logic, attributes, and operations for the mobile business objects 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 Unwired WorkSpace, open the SUP101 mobile application project.
In WorkSpace Navigator, right-click the SUP101 folder and select **Open in Diagram Editor**.
2. (Optional) If you are performing other tutorials, add a new folder to the project to organize the generated code for each device platform.
For example, in WorkSpace Navigator, expand SUP101 and under `Generated Code` add a RIM folder. The `Generated Code` directory was created during the MBO tutorial.
3. Right-click 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 page, select the **Customer MBO**, then click **Next**.
6. In the Configure options page, 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	Select default.
Page size	Select 1024.
Package	(Optional) Enter a unique name for the Java package.
Project path	Leave the default <code>\SUP101\Generated Code</code> , or browse to another folder you created for the device platform in Step 2.
Third-party jar file	You do not need to choose one for this tutorial. You will see a warning at the top of the page: The third-party class 'net.rim.device.api.system.ApplicationDescriptor' cannot be found. You can ignore this warning; it will not affect code generation.
Generate JavaDoc	Deselect for this tutorial.

You can ignore the error icons that appear in WorkSpace Navigator under `\Generated Code\src\SUP101`, 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 will 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 `\src\SUP101` folder.

Creating the BlackBerry Project

Create a new BlackBerry SUP101Sample project in the BlackBerry Java Plug-in for Eclipse. You add library resources to the project.

Prerequisites

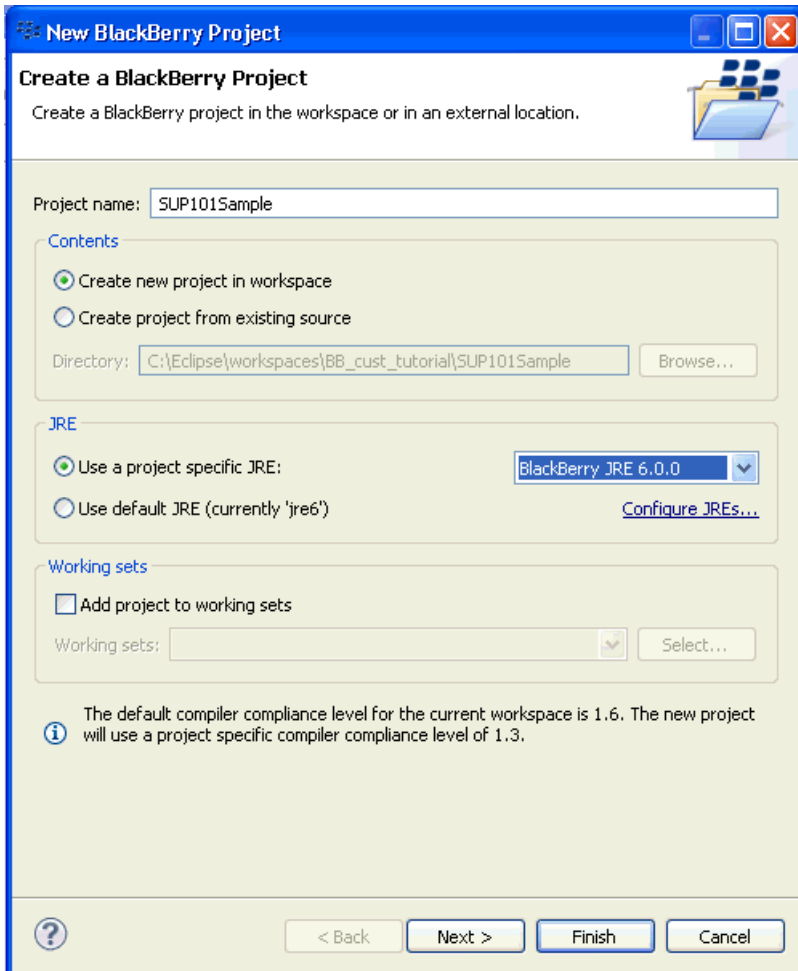
Install the BlackBerry Java Plug-in for Eclipse. Review the *Supported Hardware and Software* requirements to ensure you have the correct version.

Task

1. Start the BlackBerry Java Plug-in for Eclipse.

Note: When the Eclipse Workspace Launcher prompts you to select a workspace, do not choose one that you are using with Unwired WorkSpace. If necessary, select or create a new workspace.

2. Select **File > New > BlackBerry Project**.
3. In the New BlackBerry Project wizard, use these values and click **Next**.
 - Project name – enter `SUP101Sample`.
 - Use a project specific JRE – select **BlackBerry JRE 6.0.0**.

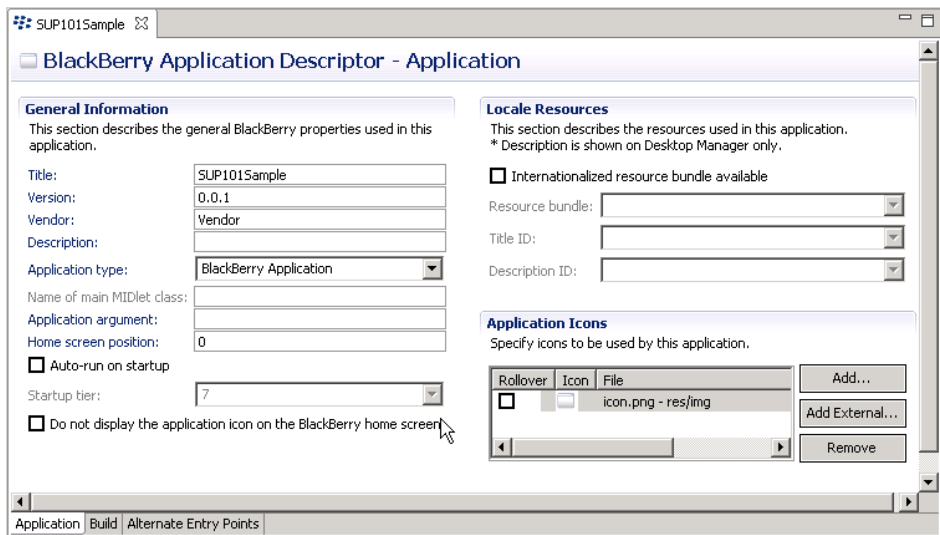


4. In the Java Settings page, modify the build path to point to the correct location for the `sup_client2.jar` and `UltraLiteJ12.jar` files for the project.
 - a) Click the **Libraries** tab.
 - b) Click **Add External Jars**.
 - c) Browse to `<UnwiredPlatform_InstallDir>\UnwiredPlatform\MobileSDK\ObjectAPI\BB`.
 - d) Select the two JAR files, then click **Open**.
 - e) Click **Finish**.

Configuring BlackBerry Application Properties

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

1. In Package Explorer, expand the **SUP101Sample** project.
2. Open the `BlackBerry_App_Descriptor.xml` file.
3. Select the **Application** tab.
4. In the Title, enter `SUP101Sample`.
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. Click **File > Save** to save the `SUP101Sample` descriptor file.

Copying Unwired Platform Files to Sample Project

Copy Unwired Platform BlackBerry runtime files and generated object API code to the `SUP101Sample` project. You had created the code previously in the Generate Code wizard.

1. In Windows Explorer, browse to `<UnwiredPlatform_InstallDir>\UnwiredPlatform\MobileSDK\ObjectAPI\BB`.
2. Copy these files: `CommonClientLib.cod`, `MessagingClientLib.cod`, `MocaClientLib.cod`, `sup_client2.cod`, and `UltraLiteJ12.cod`.

3. Paste the copied COD files into the BlackBerry Java Plug-in simulator directory, located in `<BlackBerry_Eclipse_InstallDir>\plugins\net.rim.ejde.componentpack<version>\components\simulator`.
4. In Windows Explorer, go to the workspace folder you set for Sybase Unwired WorkSpace, and copy the generated code files.
For example, go to `C:\Documents and Settings\user1\SUP-wksp\SUP101\Generated Code\src` and copy the SUP101 folder.
5. In the BlackBerry Application Development workspace, go to the SUP101Sample project and paste the folder into the src directory.
For example, go to `C:\Documents and Settings\user1\BB-wksp\SUP101Sample\src` and paste the SUP101 folder.
In Package Explorer, you see the SUP101 and SUP101.intrnl folders.

If the copied folders do not appear, click **Refresh** or press **F5**.

6. Modify the `Customer.java` file so the `KeywordFilterField` displays the data properly.
 - a) In Package Explorer, expand the `SUP101Sample\src\SUP101` folder.

Tip: If you do not see the SUP101 package, refresh the project.

- b) Open the `Customer.java` file.
- c) After the line: `/** End code region: JSON methods */`, add this code:

```
public String toString() { return getFname() + " " +
    getLname(); }
```
- d) Select **File > Save**.

Creating the User Interface

Add sample code from the ZIP archive to the SUP101Sample application. The files provide the functionality and layout of the user interface.

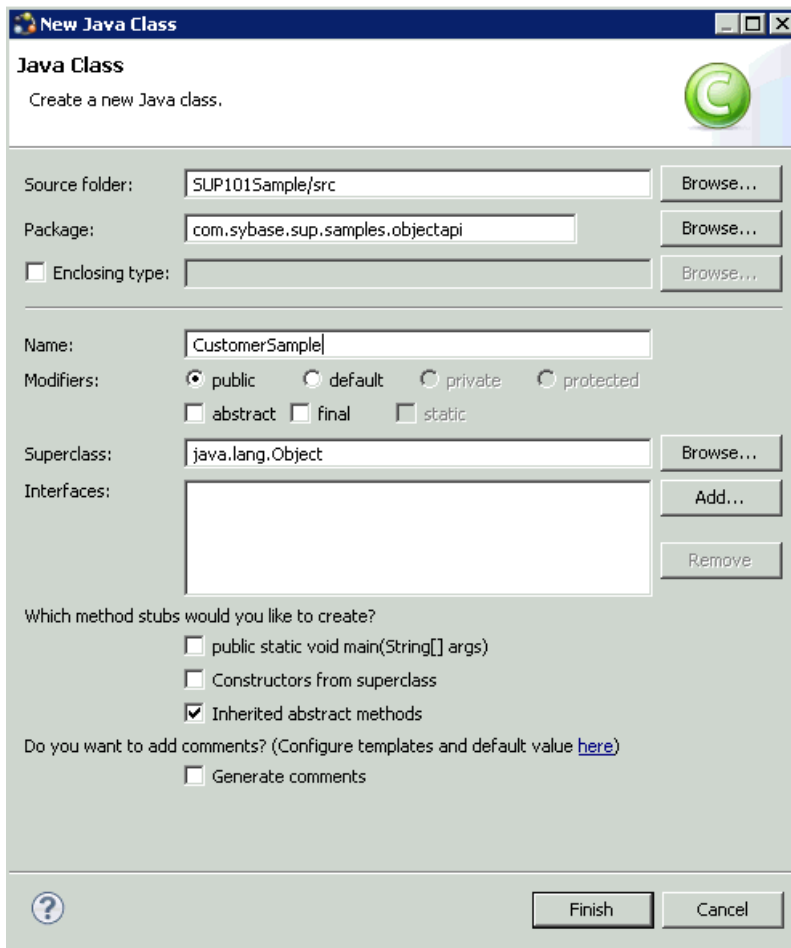
Prerequisites

Obtain text files from the `SUP_BB_Custom_Dev_Tutorial_code.zip` file. The text files contain code snippets that you need to build the user interface. You will create these classes for the interface: `CustomerSample`, `CustomerSampleScreen`, `CustomerList`, `CustomerDBCcallback`, and `ObjectAPIUtil`.

- If you are viewing this guide online from the Sybase Product Documentation web site, click [SUP_BB_Custom_Dev_Tutorial_code.zip](#) to access the ZIP archive containing the text files.
- If you are viewing this guide as a PDF, go to the Sybase Product Documentation Web site at <http://sybooks.sybase.com/nav/summary.do?prod=1289>. Click the link for the appropriate Sybase Unwired Platform version. Then, navigate to this topic in the tutorial, and click the link for the ZIP file to access the text files.

Task

1. In Package Explorer, right-click **SUP101Sample**, and select **New > Package**.
2. In the New Java Package wizard, use these values and click **Finish**.
 - Source folder –verify SUP101Sample/src appears.
 - Name –enter `com.sybase.sup.samples.objectapi`.
3. In Package Explorer, you will create one new Java class for the application in the `com.sybase.sup.samples.objectapi` package, and then copy the remaining four classes into the package.
 - CustomerDBCcallback.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.
 - a) Right-click the **com.sybase.sup.samples.objectapi** package and select **New > Class**.
 - b) In the Java Class page, for the Name, enter one of the Java class names, for example: `CustomerSample` and click **Finish**.



In the SUP101Sample, under the src
\`com.sybase.sup.samples.objectapi` package, you see the new Java
class: `CustomerSample.java`.

- c) Browse to the directory where you saved the ZIP file.
- d) Copy the content of the Java text file with the same name:
`CustomerSample.java`.
- e) In Package Explorer, open the new Java class with the same name:
`CustomerSample.java`.
- f) Paste the copied text code into the new Java class: `CustomerSample.java`. Copy
over any existing content.

```

package com.sybase.sup.samples.objectapi;

import java.util.Vector;
import net.rim.device.api.system.Characters;
import net.rim.device.api.ui.Graphics;
import net.rim.device.api.ui.UiApplication;
import net.rim.device.api.ui.XYRect;
import net.rim.device.api.ui.component.BasicEditField;
import net.rim.device.api.ui.component.KeywordFilterField;
import SUP101.Customer;
import SUP101.SUP101DB;
import com.sybase.collections.ObjectList;

/**
 * @author bdeng
 *
 * Customer sample main application.
 *
 * Note: For the sake of simplicity, this sample application may not
 * leverage resource bundles and resource strings. However, it is
 * STRONGLY recommended that application developers make use of the
 * localization features available within the BlackBerry development
 * platform to ensure a seamless application experience across a variety
 * of languages and geographies. For more information on localizing your
 * application, please refer to the BlackBerry Java Development
 * Environment Development Guide.
 */

```

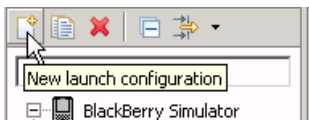
- g) Select **File > Save**.
The package contains errors until you create the other classes. Ignore these errors and continue.
- h) For the remaining four classes, copy the Java files from the ZIP archive to the SUP101Sample\src\com.sybase.sup.samples.objectapi package.
The compilation errors in the Java files resolve when you finish creating the classes.
4. Modify the HOST name or IP address in the ObjectAPIUtil.java file to point to the Sybase Unwired Server.
 - a) In Package Explorer, expand the **SUP101Sample** project.
 - b) Under the \src\com.sybase.sup.samples.objectapi package, double-click the ObjectAPIUtil.java file.
 - c) Modify the HOST name or IP address, and **Save**.

Creating a Launch Configuration for the Project

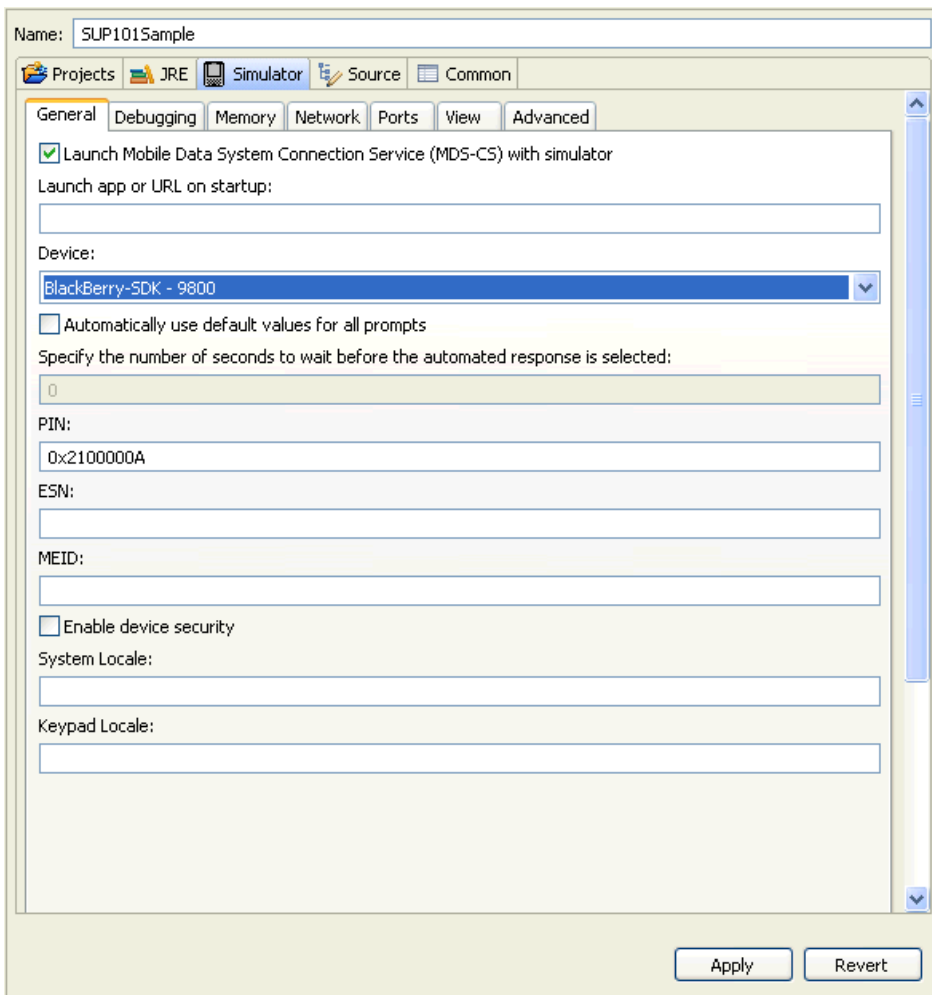
Create and define a new launch configuration for the SUP101Sample project. The configuration defines how the application will launch and the target BlackBerry platform.

1. In Package Explorer, right-click the **SUP101Sample** project, and select **Run As > Run Configurations**.
2. In the Run Configurations wizard, select **BlackBerry Simulator** and the New launch configuration icon.

Developing a BlackBerry Application



3. In the Projects tab, select the **SUP101Sample** project.
4. For the Name, enter: SUP101Sample.
5. In the JRE tab, select the Runtime JRE to use, in this case, **BlackBerry JRE 6.0.0**.
6. In the **Simulator** tab, select **Launch Mobile Data System Connection Service (MDS-CS) with simulator**. Then select the **BlackBerry-SDK - 9800** as the device.



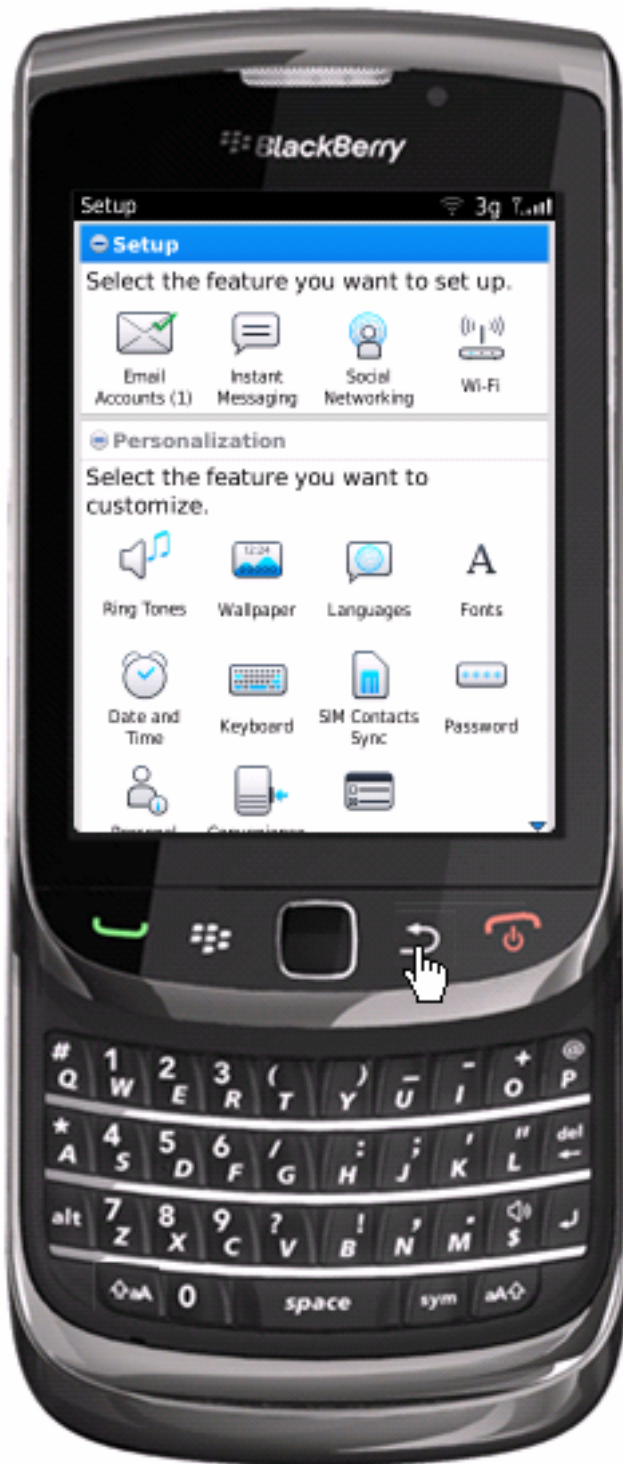
7. Click **Apply**, then **Close**.

Testing the Device Application on the BlackBerry Simulator

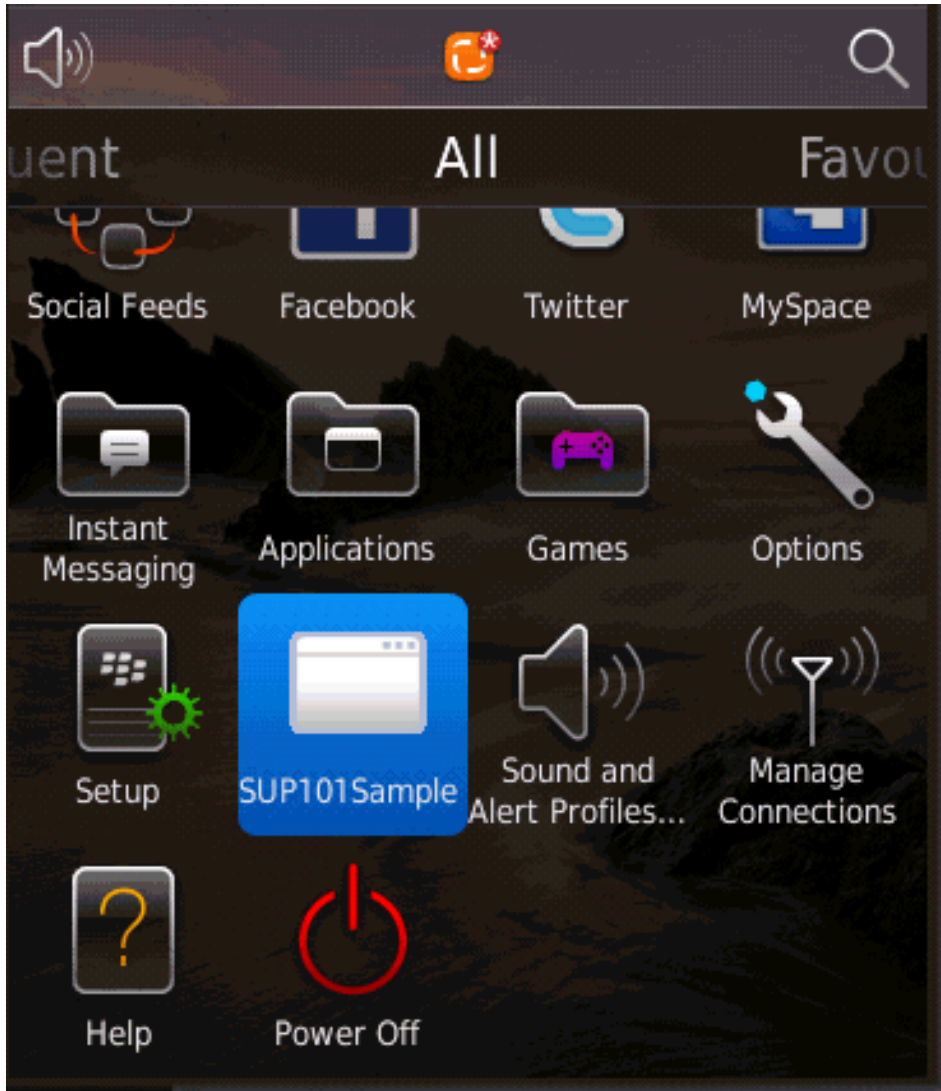
Run the SUP101Sample application on the BlackBerry simulator, and change customer information to update the interface.

1. In Package Explorer, right-click the **SUP101Sample** project 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 running the simulator, cancel the Setup screen.



2. On the main window, click **All** to access the applications screen, then scroll until you see the SUP101Sample application.



3. Click the **SUP101Sample** to launch the application.

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

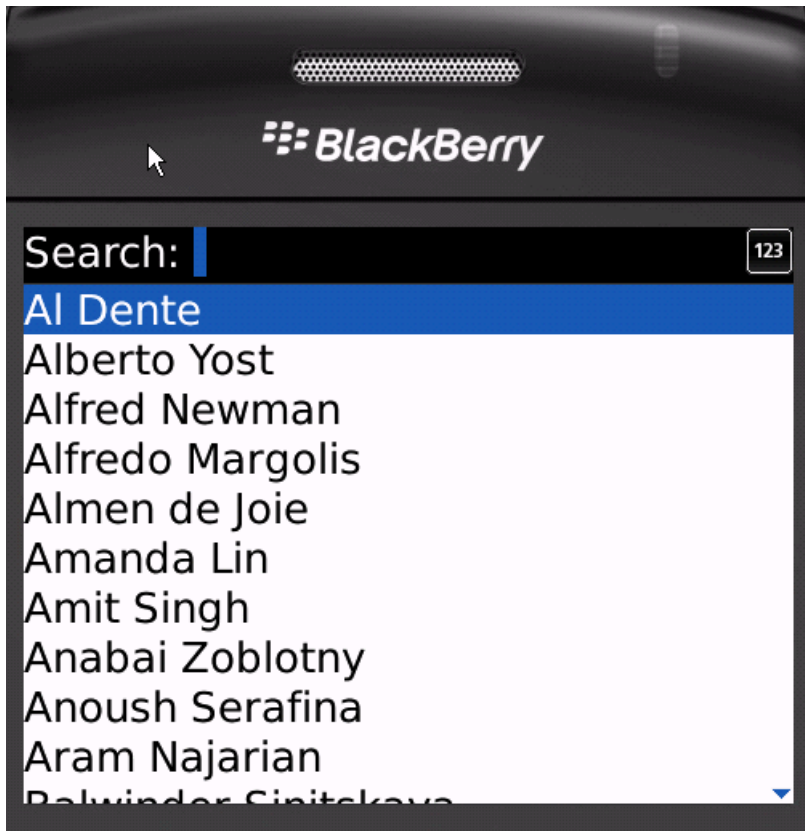
In the initialization process, the system enables the operation to target change notifications using:

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

```
sg.save();
```

After the application initializes, the device application shows the SUP101Sample 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. Focus on the customer list and in the **Search** bar, enter: dal.
The customer list is filtered and only customers with a first or last name beginning with 'dal' are shown, in this case, Daljit Sinnot.
5. Select the customer, **Daljit Sinnot**, and click the trackball.
The Update Customer screen for Daljit Sinnot appears.

6. Change the first name of the customer to abc and click **submit**.

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

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

The customer list appears with the name change you made.

7. Close the simulator to stop the SUP101Sample 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/nav/summary.do?prod=1289>, then navigate to the most current version.

Tutorials

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

Example Projects

Example projects are available for download, if you want the finished tutorial without going through the steps. Download example projects from: <http://www.sdn.sap.com/irj/sdn/mobile?rid=/webcontent/uuid/40ea4956-b95c-2e10-11b3-e68c73b2280e>.

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 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 how to develop mobile business objects (MBOs) to fully maximize their potential.

Learn about using the 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: Mobile Workflow Packages*

Customize and automate:

Learn More About Sybase Unwired Platform

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

Javadoc and HeaderDoc are also available in the installation directory.

Index

.cod files

- CommonClientLib.cod 15
- MessagingClientLib.cod 15
- MocaClientLib.cod 15
- sup-client2.cod 15
- UltraLiteJ12.cod 15

A

application properties 15

B

- BlackBerry application description 15
- BlackBerry Java Plug-in for Eclipse installing 11
- BlackBerry project, creating 13
- BlackBerry Simulator 21
- build path 13

C

- code generation 11
- CommonClientLib.cod 15
- CustomerDBCcallback.java 16
- CustomerList.java 16
- CustomerSample.java 16
- CustomerSampleScreen.java 16

D

descriptor file 13

E

example projects 1

J

JAR files

- adding 13
- sup-client2.jar 13
- UltraLiteJ12.jar 13

Java class, creating

- CustomerDBCcallback.java 16
- CustomerList.java 16
- CustomerSample.java 16
- CustomerSampleScreen.java 16
- ObjectAPIUtil.java 16

Java Object API code, generating 11

Java perspective 16

L

launch configuration 19

M

- MessagingClientLib.cod 15
- mobile business object tutorial 1
- Mobile Workflow package tutorial 1
- MocaClientLib.cod 15

O

- Object API tutorials 1
- ObjectAPIUtil.java 16

P

project build path 13

S

- samples, how to download 27
- sup-client2.cod 15
- sup-client2.jar 13
- SUP101Sample application
 - running 21
 - testing 21
 - updating data 21
 - viewing data 21
- Sybase Control Center, connecting to 6
- Sybase Mobile SDK installing 5
- Sybase Unwired Platform documentation resources 27

Index

- getting started 5
- installing 5
- Sybase Unwired WorkSpace
 - basics 7
 - how to access online help 7
 - starting 6

T

- task flow 3
- testing 21
- troubleshooting information 7
- tutorials 1

- tutorials, how to download 27

U

- UltraLiteJ12.cod 15
- UltraLiteJ12.jar 13
- Unwired Platform Runtime
 - installing 5
- Unwired Platform services 5
- Unwired WorkSpace basics 7
- user interface
 - creating 16