Previewing your Hybrid application

Overview

Learn how to preview your Hybrid application - either its common web resources (the `common` folder in the application structure), or the various environments.

- Previewing the application's common web resources
- Previewing the application in iOS
- Previewing the application in Android
- Previewing the application in Windows Phone 8
- Previewing the application in Windows 8
- Previewing the application in BlackBerry 10
- Previewing the application in Mobile Web and Desktop Browser

Previewing the common web resources

After the application is built and deployed, it can be previewed in the MobileFirst Operations Console Mobile Browser Simulator (MBS). You can open the MobileFirst Operations Console from either the MobileFirst CLI or MobileFirst Studio.

CLI

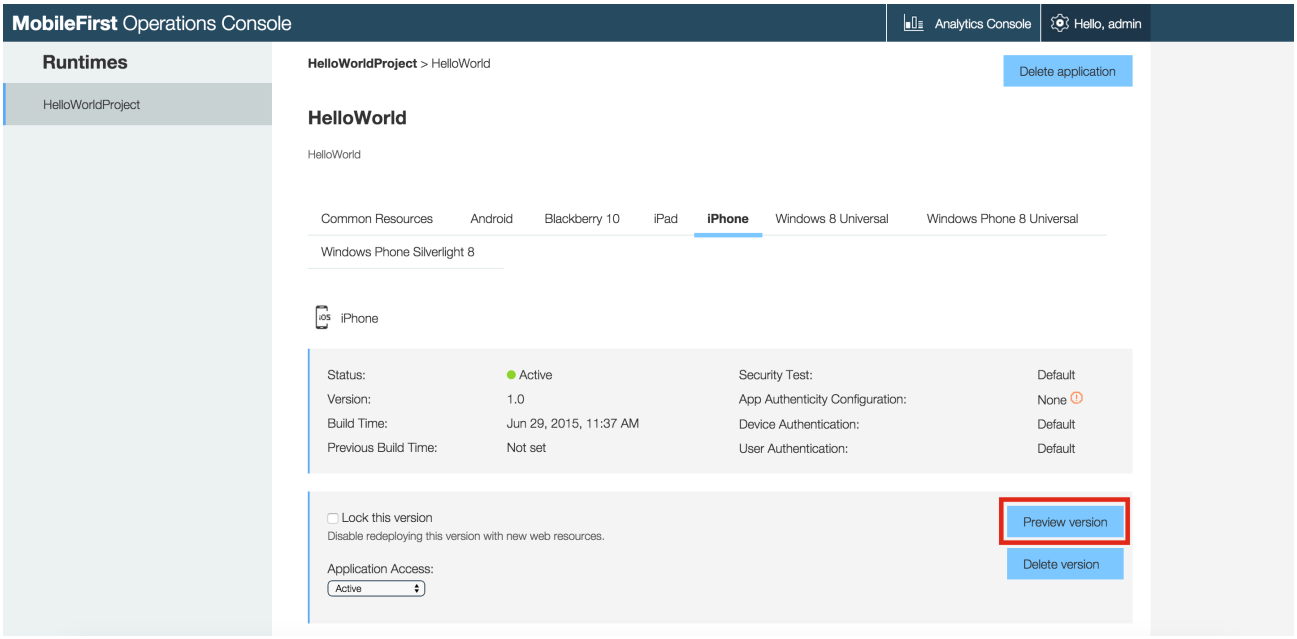
From a terminal window, run the following CLI (../../advanced-client-side-development/using-cli-to-create-build-and-manage-mobilefirst-project-artifacts/) command from the project folder: `mfp console`

Studio

Right-click the project and select **Open MobileFirst Operations Console** or browse to the MobileFirst Console. Typically the URL is `http://localhost:10080/worklightconsole`.

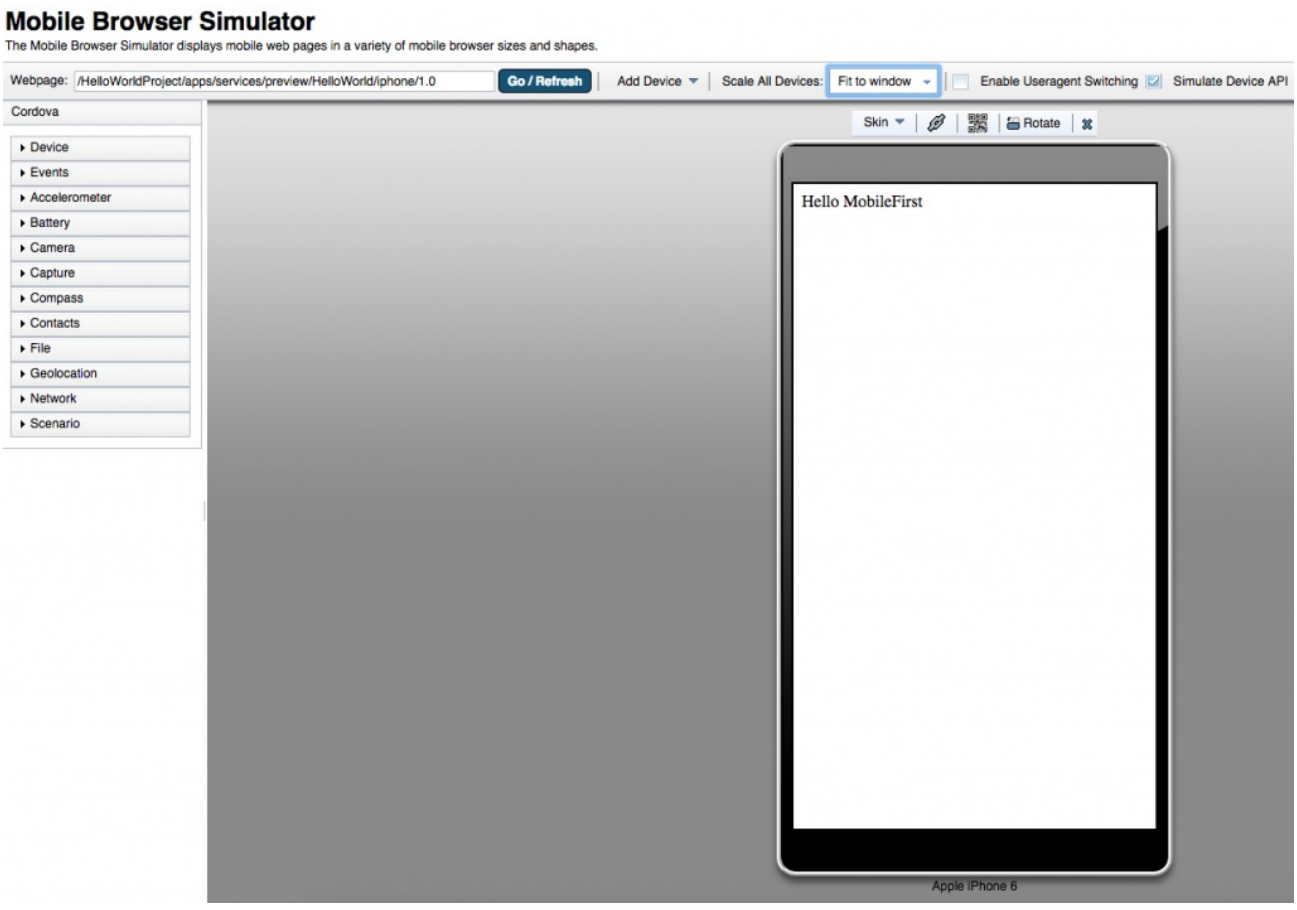
Next, from the console:

1. Click **Applications** to see the list of applications in your project.
2. Click the application name, **HelloWorld**, to see the list of environments.
3. To preview the web resources of an environment, click the **Preview** button that is below the environment name.
4. To preview the common resources, click the **Preview** button that is in **Common Ressources**.



You can use the Mobile Browser Simulator to emulate Cordova or preview your application with different device skins.

Note: Limitation about Mobile Browser Simulator: To preview an application that contains security tests that rely on the device (such as autoprovisioning or authenticity), you must temporarily change the security tests so that they do not use the device identification.



Previewing the application in iOS

Note: The Eclipse IDE **does not support** iOS application development. Therefore, you must transfer the application to Xcode, Apple IDE for iOS development.

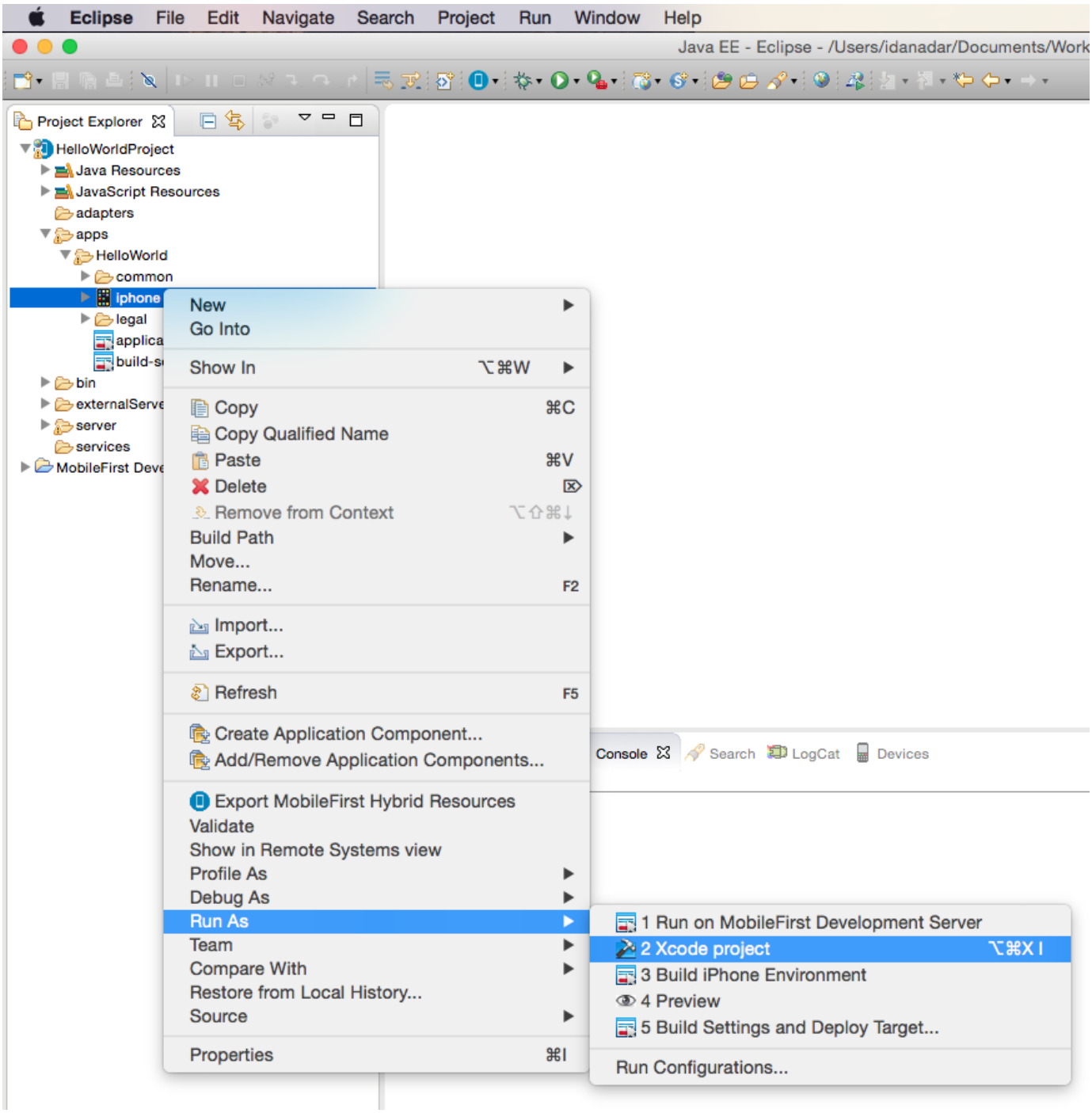
Opening the Xcode project in Xcode

CLI

You cannot open an external IDE from the command line.
If you are not using MobileFirst Studio, navigate to the application folder and open the `.xcodeproj` file.

Studio

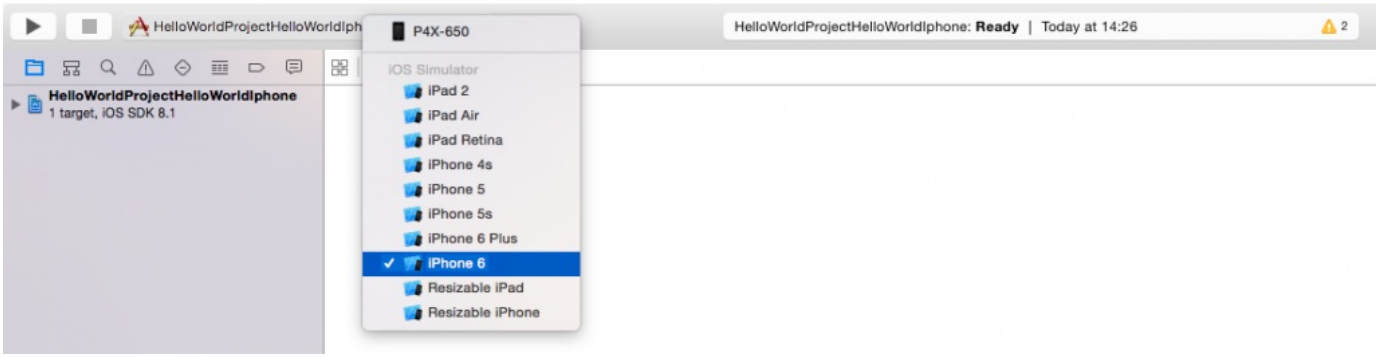
If you are running a Mac version of Eclipse, right-click the iPhone or iPad environment folder and select **Run As > Xcode project**. The MobileFirst Studio plug-in builds and deploys the application and opens the Xcode project in Xcode.



As an alternative, you can manually open an Xcode project from the native folder.
If you are running a Windows™ version of Eclipse, manually compress the native folder and copy it to your Mac computer. The native folder represents a fully compatible native Xcode project.

Previewing the application in the iOS Simulator

After opening the generated Xcode project in Xcode, select a simulator type and click **Play**.



Running the application in a physical device

To deploy an iOS application to a real device, you must enroll to the Apple iOS Developer Program and install a provisioning profile onto Xcode and your iOS device.

For more information, see Apples iOS Developer Center website.
After you have enrolled as an iOS developer and installed a profile, select the device and click **Play**.

> When using Xcode 7 and iOS 9, read notes about ATS and Bitcode (file:////home/travis/build/MFPSamples/DevCenter/_site/blog/2015/09/09/ats-and-bitcode-in-ios9/)

Previewing the application in Android

Previewing the application in the Android Emulator

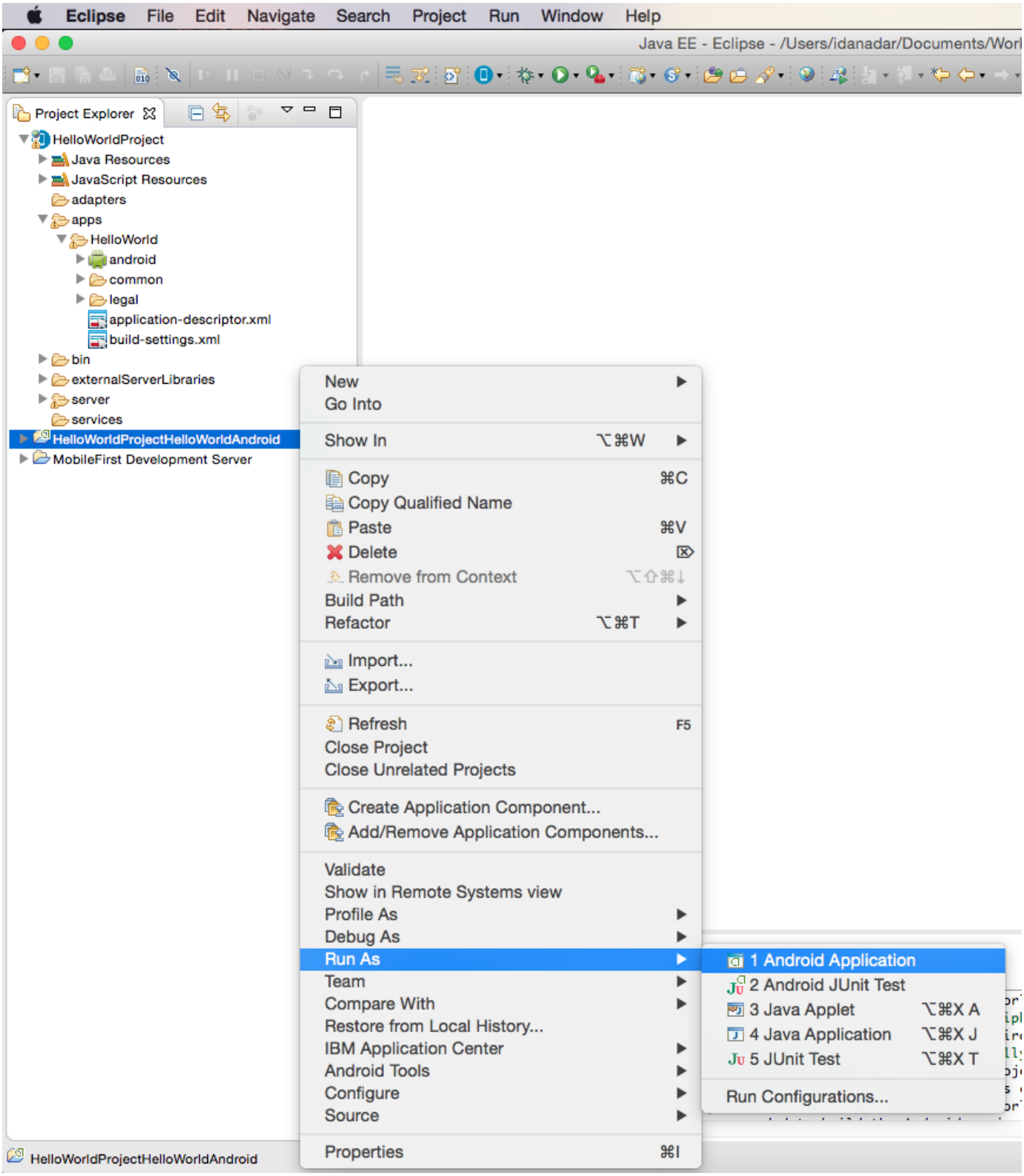
CLI

You cannot open an external IDE from the command line.
If you are not using the MobileFirst Studio Eclipse plug-in, use Android Studio to navigate to the application folder in order to open it.

MobileFirst Studio

Before you can run the Android application in the Android Emulator (or AVD, Android Virtual Device), you must create a virtual device by following the Android documentation. (http://developer.android.com/tools/devices/index.html)

When an AVD is available, right-click the automatically generated Android project and select **Run As > Android Application**.



Running the application in a physical device

When an Android device is connected to the computer with a USB cable, the Eclipse ADT plug-in automatically recognizes the Android device and attempts to deploy applications onto it when you right-click the automatically generated Android Project and select **Run As > Android Application**.

Previewing the application in Windows Phone 8

Opening the Visual Studio project in Visual Studio Express 2013

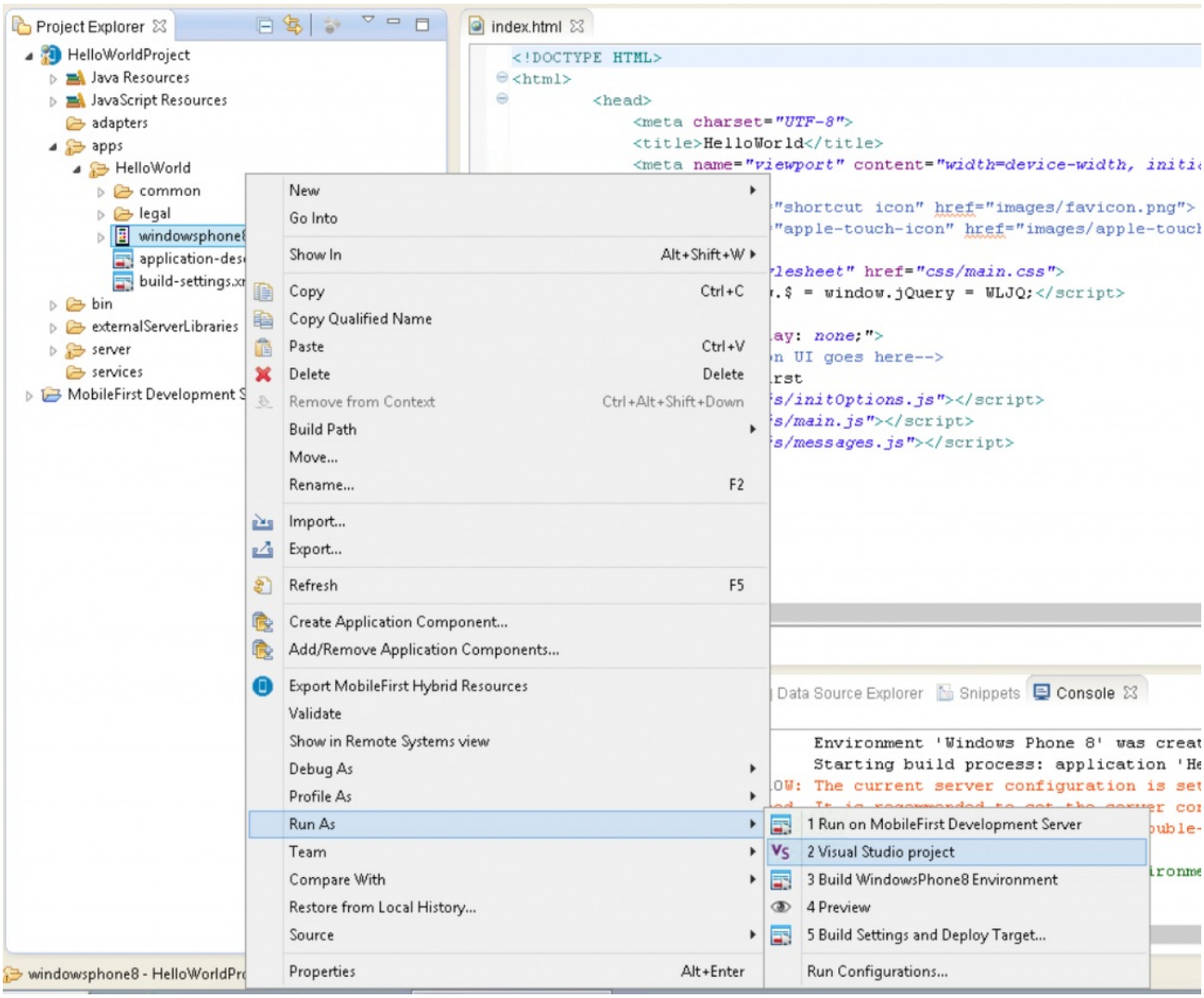
CLI

You cannot open an external IDE from the command line.
If you are not using the MobileFirst Studio Eclipse plug-in, use Visual Studio and navigate to the application folder in order to open it.

MobileFirst Studio

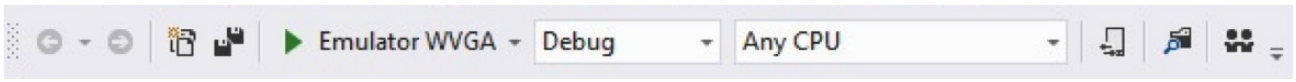
You can open the Visual Studio project from the native folder in one of the following ways:

- Right-click the `windowsphone8\native\.csproj` file and select **Open**.
- As an alternative, you can manually open the `.csproj` file within Microsoft Visual Studio. Click **Open Project** on the left menu and navigate to the project folder.
- If you are running Eclipse in Windows 8, you can also right-click the Windows Phone 8 project file and select **Run As > Visual Studio project**. The MobileFirst Studio plug-in then builds and deploys the application to MobileFirst Server and automatically opens the project in Visual Studio.

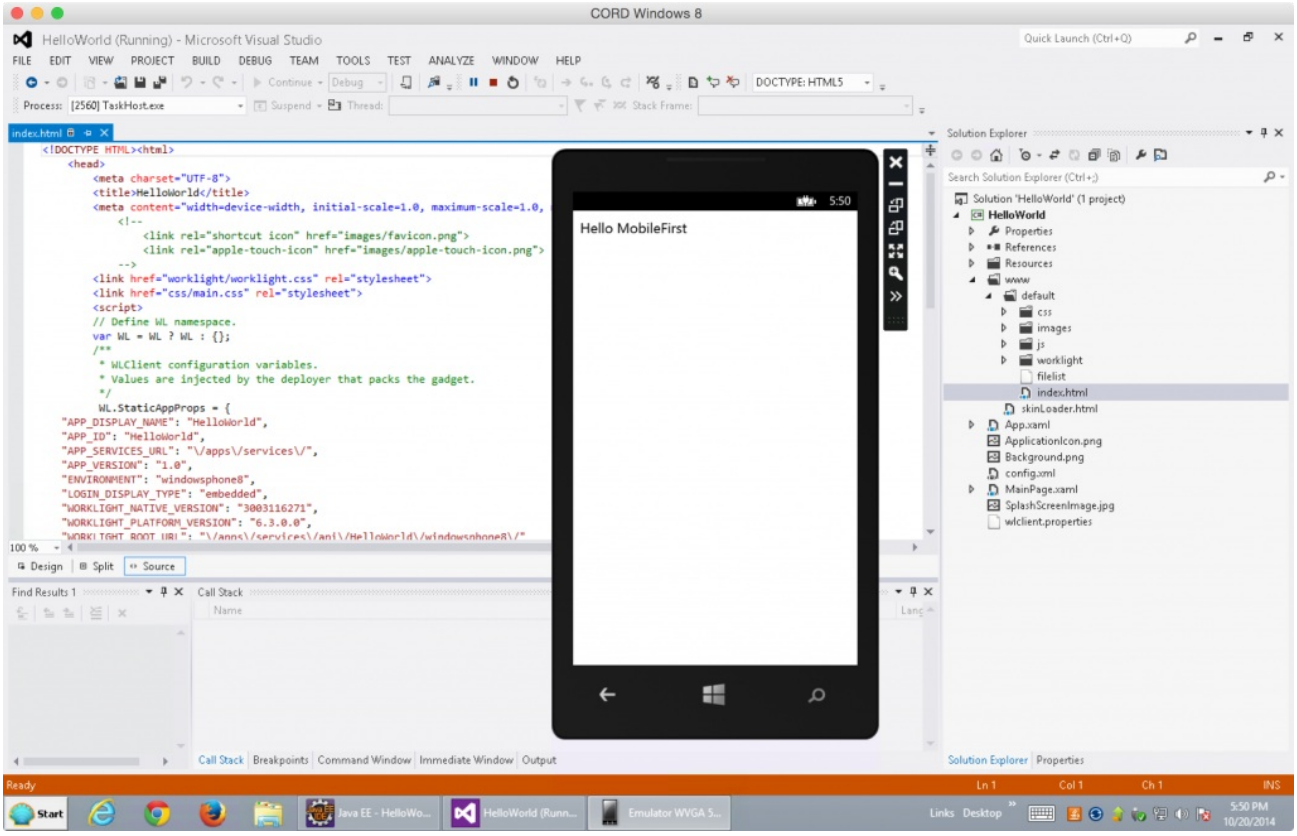


Previewing the application in the Windows Phone 8 Emulator

1. In Visual Studio, make sure that both **Windows Phone 8 Emulator** and the x86 platform are selected.



2. Click the green play button (or press **F5**) to start your application in debug mode.
   To start your application without debug mode, press **Ctrl + F5** or change **Debug** to **Release** in the drop-down list.



Running the application in a physical device

1. Make sure that **Device** is selected in the toolbar instead of the Windows Phone 8 Emulator.
2. Make sure that your device is connected and is not in sleep mode.
3. Click the green play button (or press **F5**) to start your application in debug mode.
4. Click the transparent play button to start your application without debug mode (or press **Ctrl + F5**).

Previewing the application in Windows 8

Opening the Visual Studio project in Visual Studio Express 2013

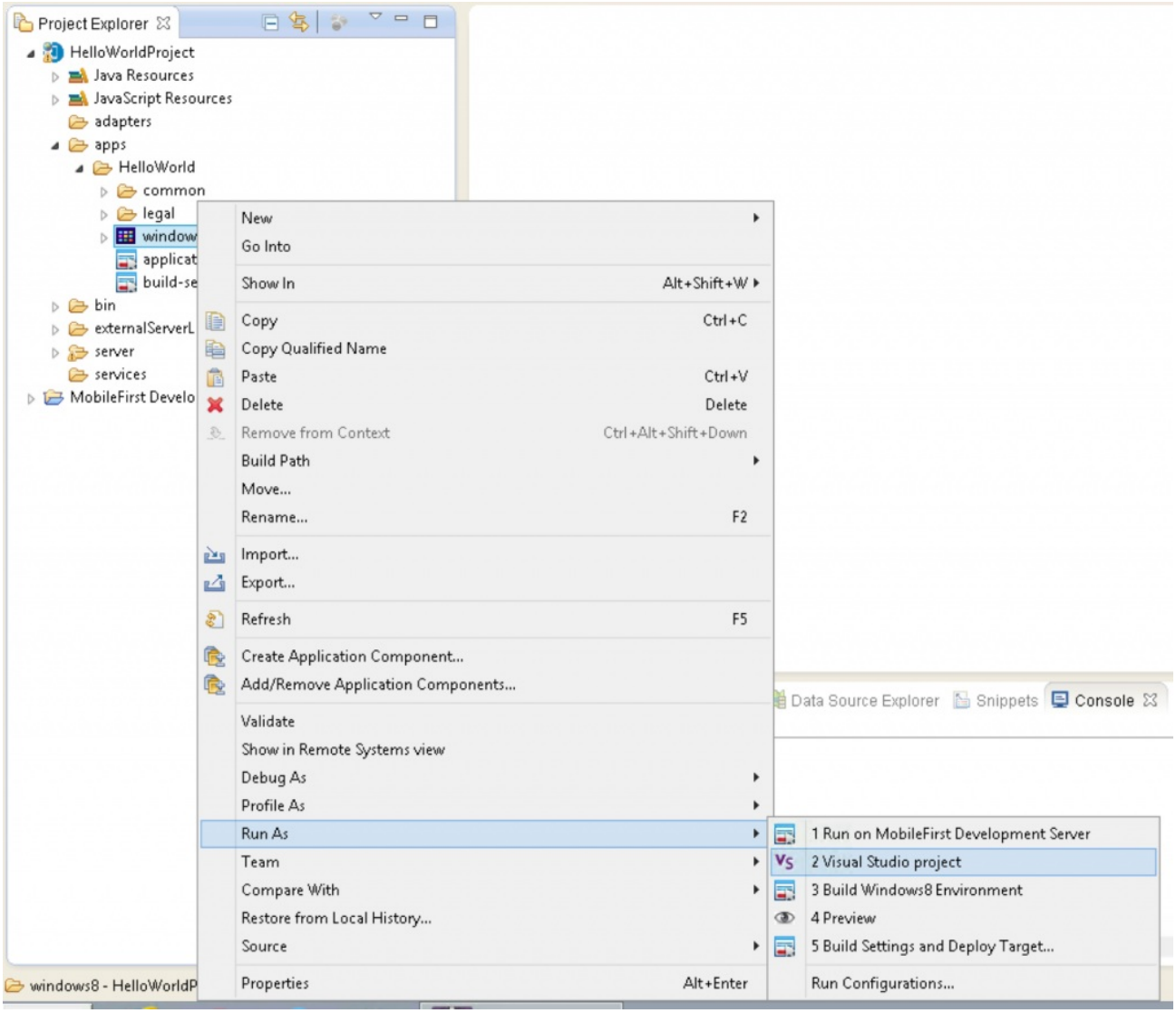
CLI

You cannot open an external IDE from the command line.
If you are not using the MobileFirst Studio Eclipse plug-in, use Visual Studio and navigate to the application folder in order to open it.

MobileFirst Studio

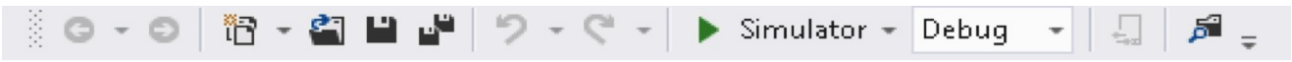
You can open a Visual Studio project from the native folder in one of the following ways:

- Right-click the `.jsproj` file and select **Open**.
- As an alternative, you can manually open the `.jsproj` file within Microsoft Visual Studio. Click **Open Project** on the left menu and navigate to the project folder.
- If you are running Eclipse in Windows 8, you can also right-click the Windows 8 project file and select **Run As > Visual Studio project**. Visual Studio opens.

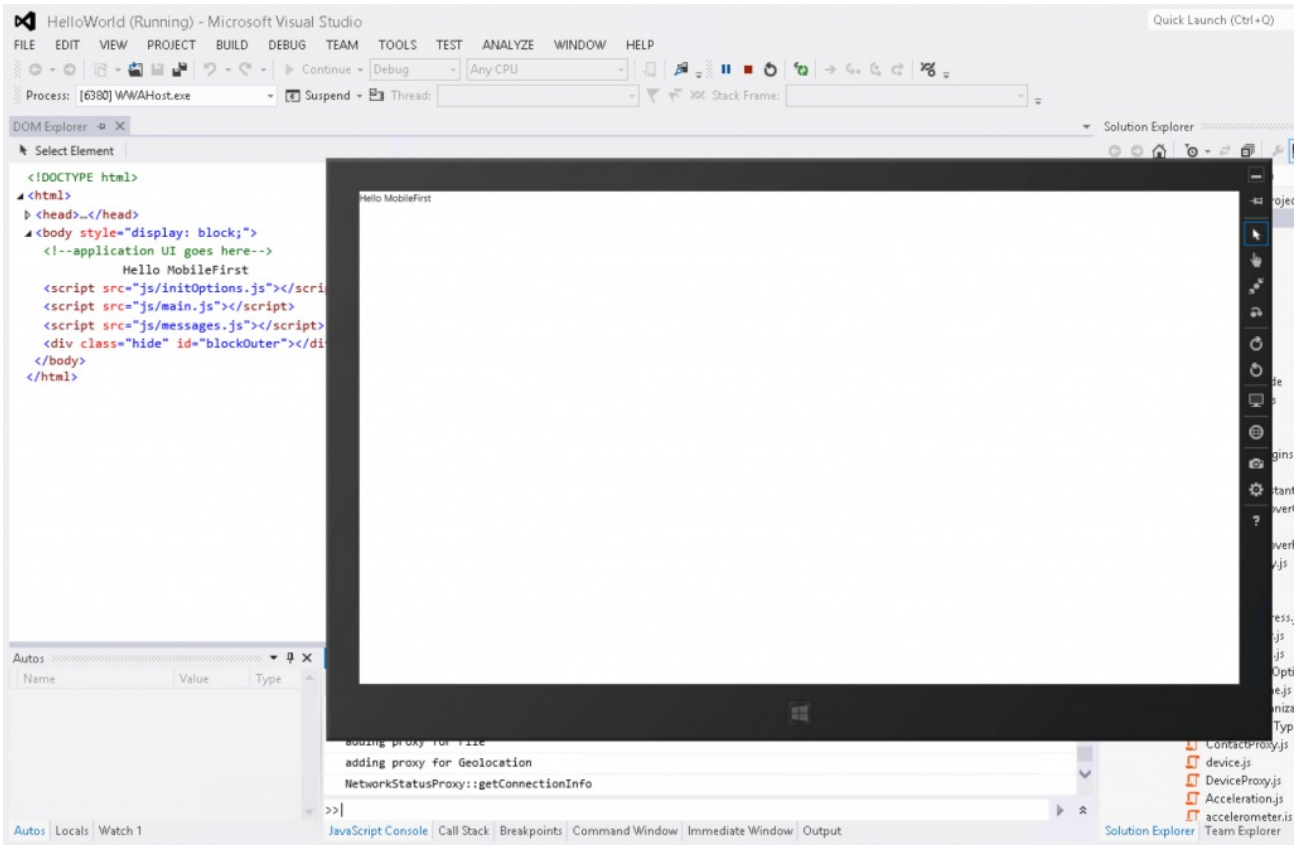


Previewing the application in the Windows 8 Emulator

1. In Visual Studio, make sure that **Windows 8 Simulator** is selected:



2. Click the green play button (or press **F5**) to start your application in debug mode.
   To start your application without debug mode, press **Ctrl + F5** or change **Debug** to **Release** in the drop-down list.

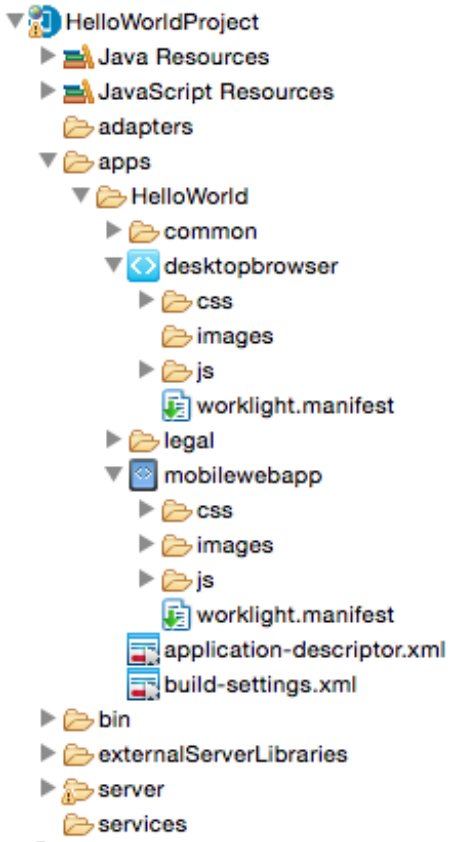


Running the application in a physical device

1. Make sure that **Device** is selected.
2. Make sure that your device is connected and is not in sleep mode.
3. Click the green play button (or press **F5**) to start your application in debug mode.
4. Click the transparent play button to start your application without debug mode (or press **Ctrl + F5**).

Previewing the application in Mobile Web and Desktop Browser

The Mobile Web and Desktop Browser environments have an additional file, `worklight.manifest`, which is a cache manifest file that allows you to manage and edit the contents of the application cache.



Previewing the application in the Mobile Web Simulator

CLI

From a terminal window, use the `mfp console` command to open MobileFirst Operations Console.

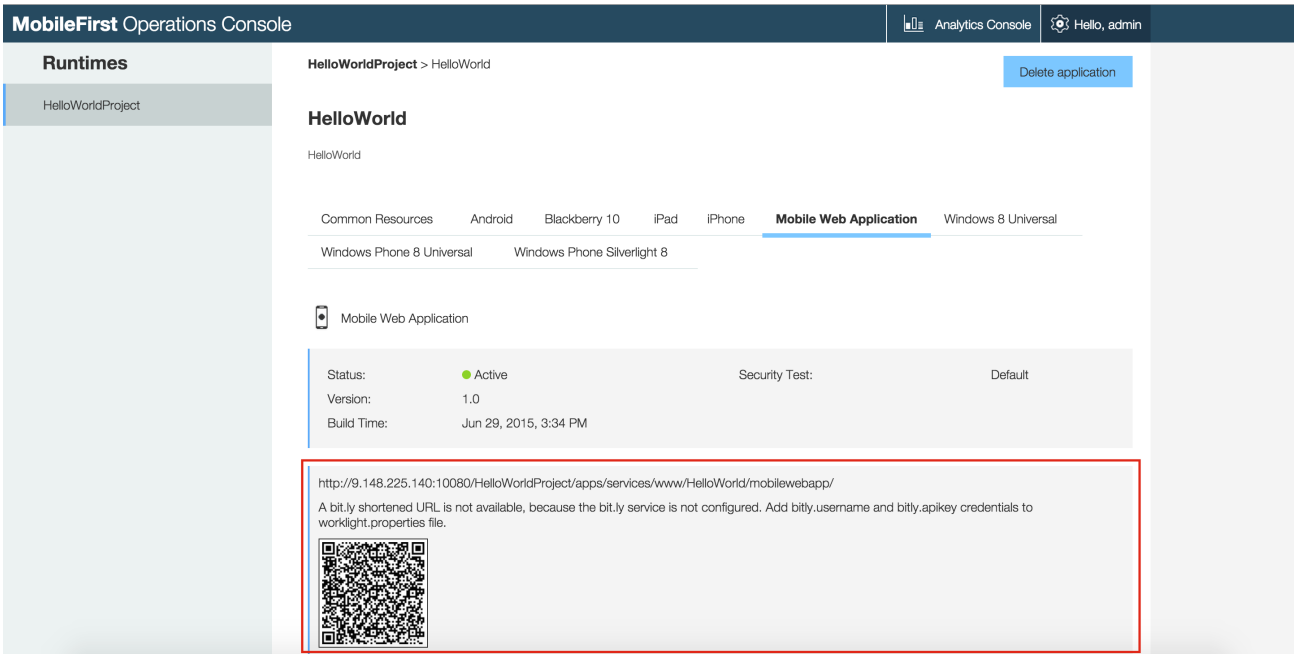
MobileFirst Studio

The desktop browser web page is, in fact, a web page, and thus it is not displayed in the Mobile Browser Simulator. Instead, a new tab opens up and presents the web page.

Running the application in a physical device

To get the URL of a mobile web application:

1. Browse to the **Mobile Web Application** environment in the MobileFirst Operations Console (../../hello-world/introduction-to-mobilefirst-platform-operations-console/).
2. Copy the regular or shortened URL from the pop-up window, or use the QR code.
   To provide a shortened URL, provide your `bit.ly` account username and password in the `worklight.properties` file.
3. Enter the shortened URL in the mobile browser of your device.



Previewing the application in BlackBerry 10

CLI

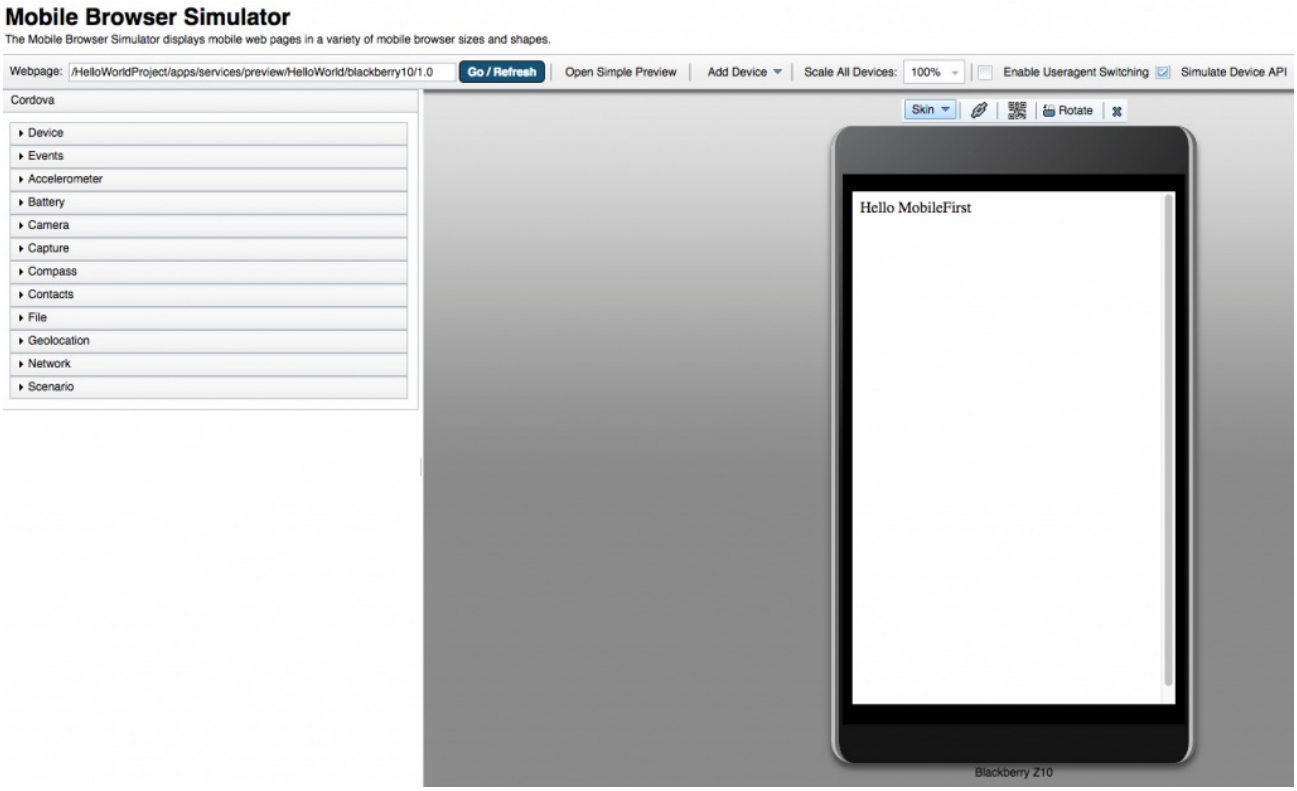
From a terminal window, use the `mfp console` command to open the MobileFirst Operations Console.

MobileFirst Studio

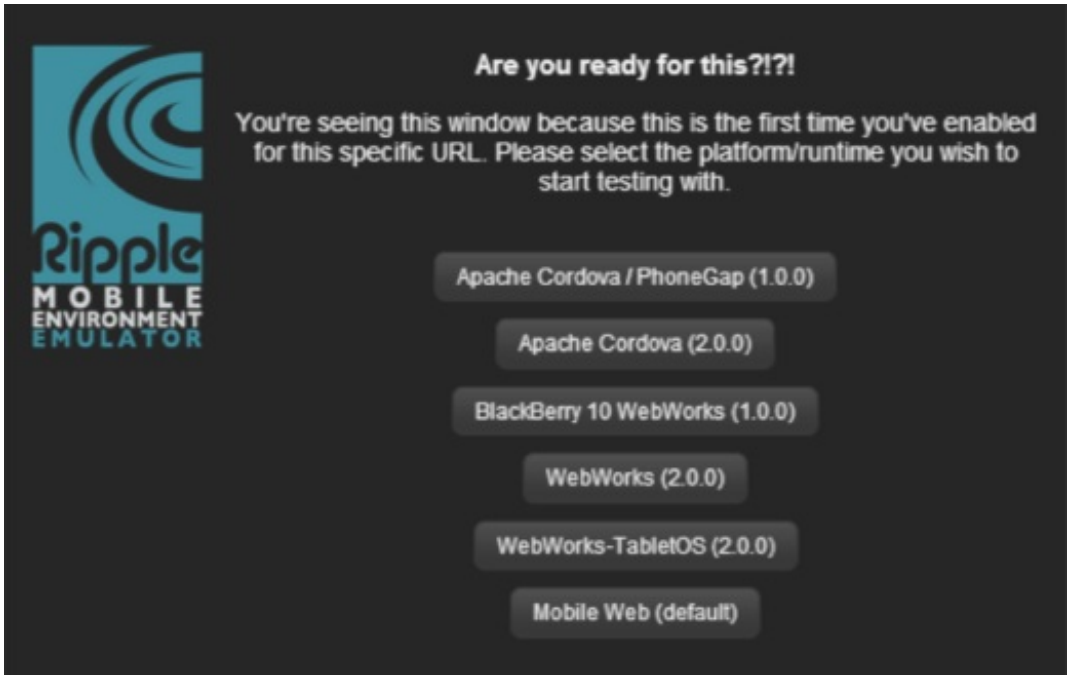
Previewing the application in the BlackBerry Ripple Emulator

Prerequisite: To be able to preview the application in the Ripple Emulator, you must first install Ripple. Ripple is a Chrome extension that can be downloaded and installed from the Chrome Web Store. After Ripple is installed, follow these steps.

1. To preview the BlackBerry environment from the MobileFirst Operations Console, click the **eye** icon and then click **Open Simple Preview**.

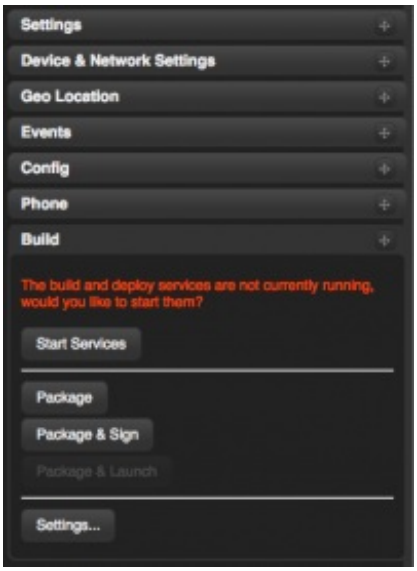


2. Enable Ripple by clicking **Enable** from the extension menu in Chrome.
3. If you get the following screen, select **BlackBerry 10 WebWorks (1.0.0)** to proceed with previewing the application.



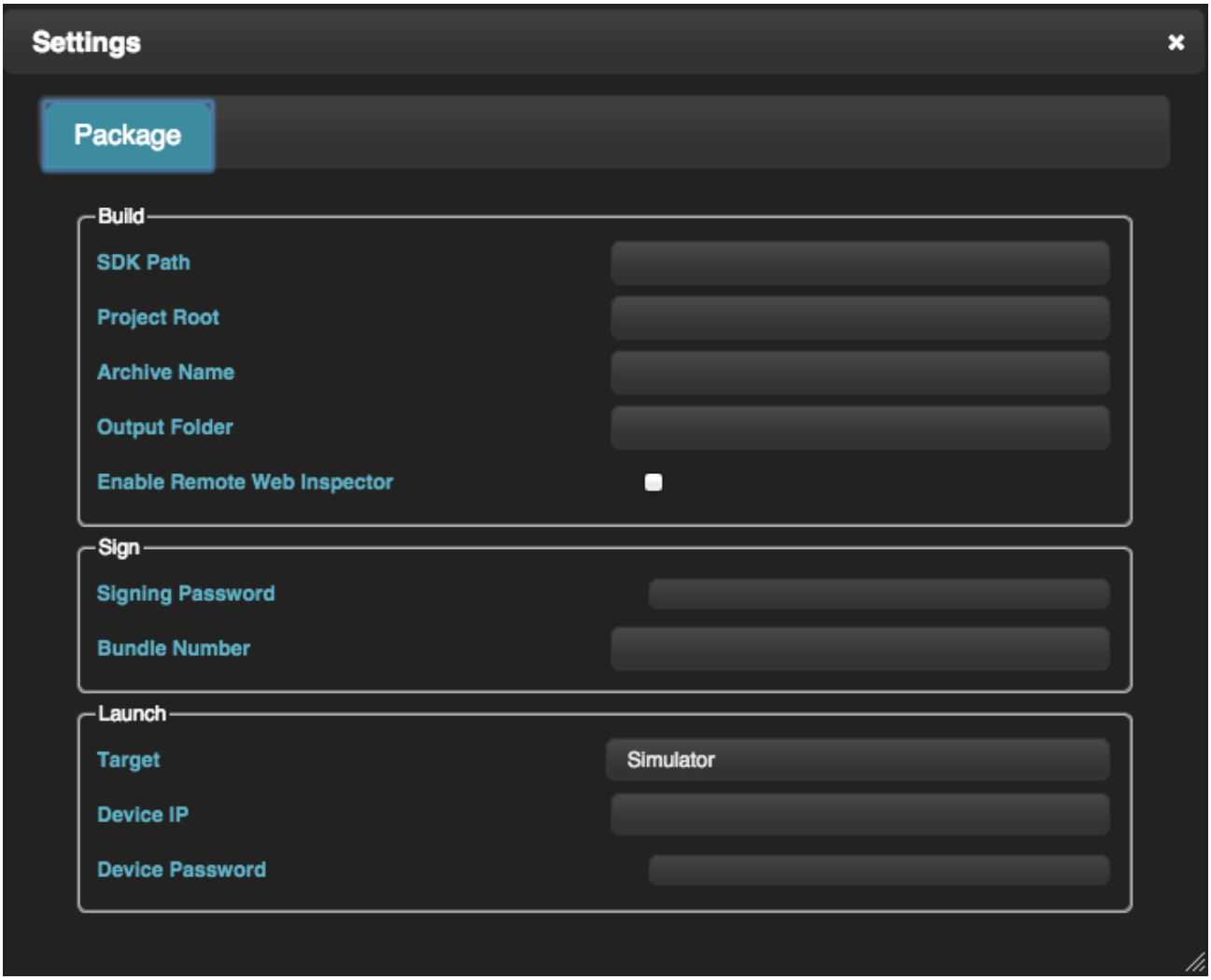
The page autorefreshes and the environment web resources are displayed in the Ripple Emulator.

Running the application in a physical device or simulator



Prerequisite: Before proceeding, you must have the BlackBerry WebWorks SDK installed.
For more information, see Setting up your BlackBerry 10 development environment (../../setting-up-your-development-environment/setting-up-the-blackberry-10-development-environment/).

1. In Ripple, on the right menu, click **Build**, and then click the **Settings** button at the bottom.
2. Enter the following package settings.



SDK Path: The path where BlackBerry WebWorks SDK is installed.
Project root: The root folder of your project. To find the root folder, right-click the native folder in your Eclipse blackberry environment and select **Properties**.
Archive Name: The name for your archive.
Output Folder: The folder where to output the application. Specify a folder outside your project.
Signing Password: Sign your app with a BlackBerry 10 developer certificate to run it on a BlackBerry 10 handset. Specify your certificate password here. Signing is NOT required to run the application in the BlackBerry 10 Simulator.
Launch: You can find the Device IP address from the BlackBerry Desktop Manager or in the device settings.

3. After the package settings are defined, open the Ripple menu in Chrome and click **start services**.
4. Select one of the Packaging options.