Creating your first native Windows Phone 8 MobileFirst application

Overview

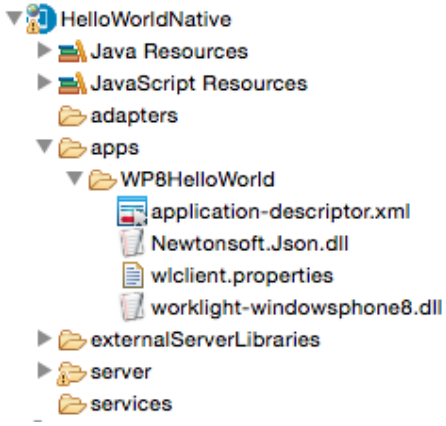
To serve a native Windows Phone 8 application, MobileFirst Server must be aware of it. For this purpose, IBM MobileFirst Platform Foundation provides a Native API library, which contains a set of APIs and configuration files.

This tutorial explains how to generate the Windows Phone 8 Native API and how to integrate it with a native Windows Phone 8 application. These steps are necessary for you to be able to use it later on for tasks such as connecting to MobileFirst Server, invoking adapter procedures, implementing authentication methods, and so on.

Prerequisite: Developers are expected to be proficient with Google's developer tools.

Creating a MobileFirst Native API

- In MobileFirst Studio, create a MobileFirst project and add a MobileFirst Native API.
- In the **New MobileFirst Native API** dialog, enter your application name and select **Windows Phone 8** for the **Environment** field.
- Right-click the generated NativeAPI folder (located in `your-projects/apps/your-nativeapi-app-name`) and select **Run As > Deploy Native API**.
  This action is required in order for MobileFirst Server to recognize the application if it attempts to connect.



The MobileFirst native API contains several components:

- *worklight-windowsphone8.dll* is a MobileFirst API library that you must copy to your native WP8 project.
- *Newtonsoft.Json.dll* is a library that provides JSON support.
- *application-descriptor.xml* defines application metadata and security settings that MobileFirst Server enforces.
- *wlclient.properties* contains connectivity settings that a native Windows Phone 8 application uses. You must copy this file to your native Windows Phone 8 project.
- As with any MobileFirst project, you create the server configuration by modifying the files that are in the *server\conf* folder.

wlclient.properties

You can edit the *wlclient.properties* file to set connectivity information.

- *wlServerProtocol* – The communication protocol to MobileFirst Server, which is either http or https.
- *wlServerHost* – The host name of the MobileFirst Server instance.
- *wlServerPort* – The port of the MobileFirst Server instance.
- *wlServerContext* – The context root path of the application on MobileFirst Server.
- *wlAppId* – The application ID as defined in the *application-descriptor.xml* file.
- *wlAppVersion* – The application version.
- *wlEnvironment* – The target environment of the native application.

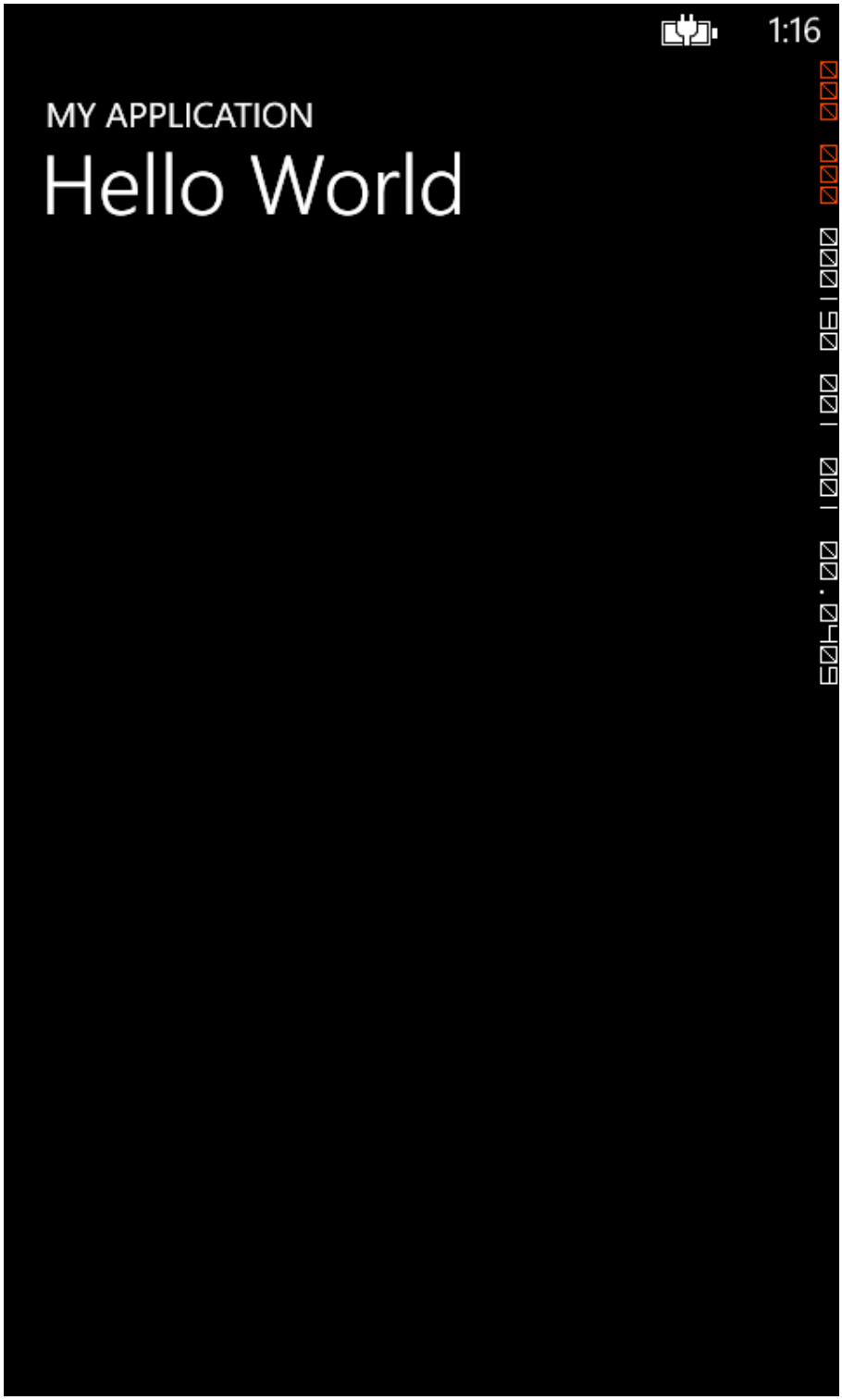
- *wlPlatformVersion* – The MobileFirst Studio version.
- *languagePreferences* – The list of preferred locales.

Creating and configuring a Windows Phone 8 native application

1. Create a Windows Phone App project or use an existing one.
2. Add as a *reference* the files *worklight-windowsphone8.dll* and *Newtonsoft.Json.dll*.
3. Copy the *wlclient.properties* file to the root of the native project.
4. In Visual Studio, open the *wlclient.properties* **Properties** window and set the **Copy to Output Directory** option to **Copy always**.
5. Add the following capabilities to the *WMAppManifest.xml* file:
   ID_CAP_NETWORKING
   ID_CAP_IDENTITY_DEVICE

> For more information, review the "developing native C# applications for Windows Phone 8" user documentation topic

After you run the application in Visual Studio, the final result is a native application that contains the MobileFirst API library.



Sample application

Click to download (http://public.dhe.ibm.com/software/products/en/MobileFirstPlatform/docs/v630/HelloWorldNativeProject.zip) the Studio project.
Click to download (http://public.dhe.ibm.com/software/products/en/MobileFirstPlatform/docs/v630/WP8HelloWorldNativeProject.zip) the Native project.