MobileFirst Platform {dev}

Using a MobileFirst Hybrid application as a container for server-generated pages

Relevant to:
Native iOS Native Android Hybrid
This tutorial covers the following topics:

Migrating applications to IBM MobileFirst Platform Foundation

- By using mobile web technology, you can deploy applications to the widest variety of devices.
- The existence of public application stores, such as Apple iTunes and Google Play, changes the way applications are hosted and marketed. These changes make traditional methods of distribution less relevant.
- IBM MobileFirst Platform Foundation provides the solution to build cross-platform applications that can be distributed through the application stores by using the hybrid application programming model.
- In the hybrid model, developers typically package the application HTML, CSS, and JavaScript™ code as part of the application that is deployed to the application store.
- This tutorial shows the remote loading of dynamic content capability, where the HTML, CSS, and JavaScript code is hosted externally from the natively packaged hybrid.

Creating MobileFirst applications

CLI

From a terminal window, use the following CLI commands to add a project and application:

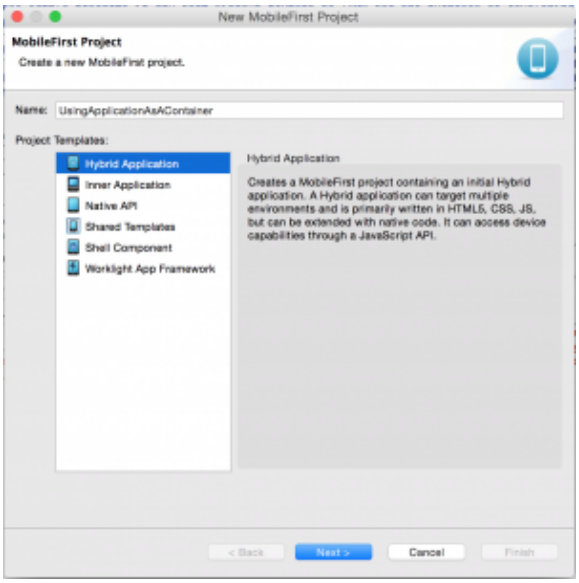
```
mfp create UsingApplicationAsAContainer
cd UsingApplicationAsAContainer
mfp add hybrid UsingApplicationAsAContainer
```

Next, add a required environment, for example Android:

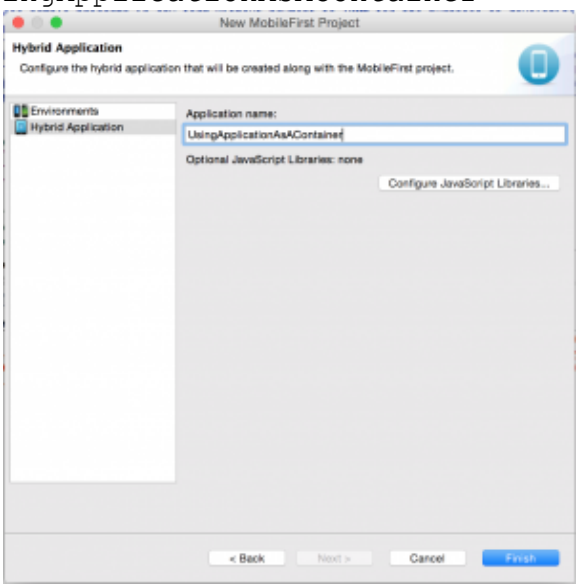
```
cd UsingApplicationAsAContainer
mfp add environment android
```

Studio

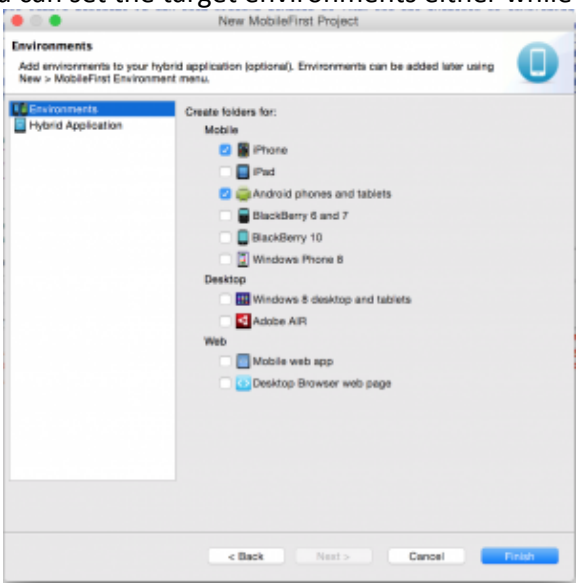
In MobileFirst Studio, create a new project, “UsingApplicationAsAContainer”



A project might host multiple applications. However, in this tutorial, you use only one app: `UsingApplicationAsAContainer`

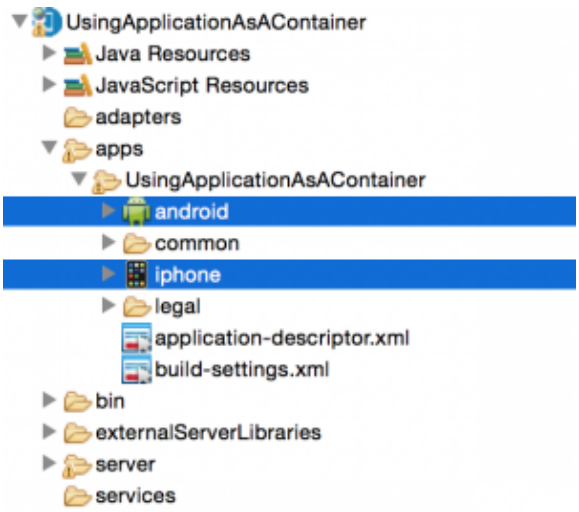


You can set the target environments either while you are working in the **MobileFirst Project** wizard, or later.



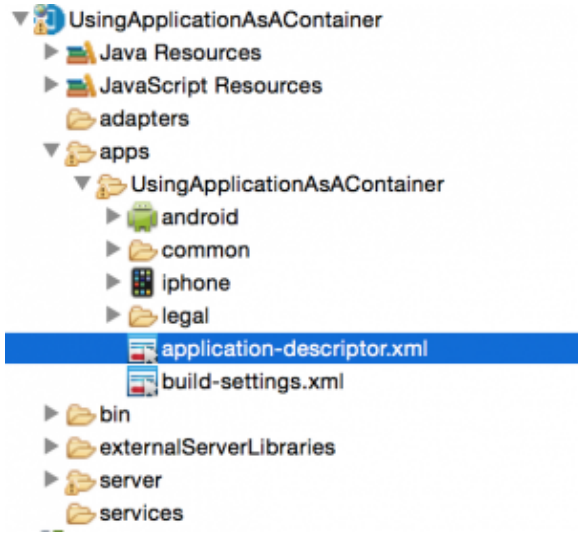
IBM MobileFirst environments

- IBM MobileFirst Foundation provides a model for organizing the application project structure for each target environment (for example, Android, iPhone, iPad).
- You select your target environment through the **MobileFirst Environment** wizard.



IBM MobileFirst Common environment

- The simplest way to use MobileFirst apps as containers for server-generated pages is through the Common environment.
- Open the `application-descriptor.xml` file and edit the `mainFile` tag to point to `http://m.ibm.com`.

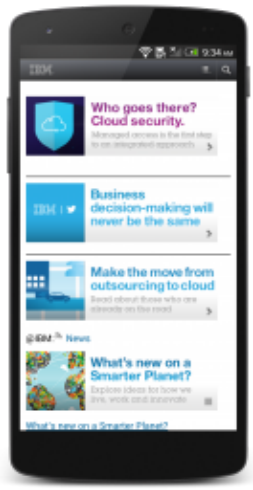



```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<application xmlns="http://www.worklight.com/application-descriptor"
 id="UsingApplicationAsAContainer" platformVersion="6.3.0.00.20141003-1438">
    <displayName>UsingApplicationAsAContainer</displayName>
    <description>UsingApplicationAsAContainer</description>
    <author>
        <name>application's author</name>
        <email>application author's e-mail</email>
        <homepage>http://mycompany.com</homepage>
        <copyright>Copyright My Company</copyright>
    </author>
    <mainFile>http://m.ibm.com</mainFile>
    <features/>
    <thumbnailImage>common/images/thumbnail.png</thumbnailImage>
    <iphone bundleId="com.UsingApplicationAsAContainer" version="1.0">
        <worklightSettings include="false"/>
        <security>
            <encryptWebResources enabled="false"/>
            <testWebResourcesChecksum enabled="false" ignoreFileExtensions="png,
jpg, jpeg, gif, mp4, mp3"/>
        </security>
    </iphone>
    <android version="1.0">
        <worklightSettings include="false"/>
        <security>
            <encryptWebResources enabled="false"/>
            <testWebResourcesChecksum enabled="false" ignoreFileExtensions="png,
jpg, jpeg, gif, mp4, mp3"/>
            <publicSigningKey/>
            <packageName/>
        </security>
    </android>
</application>
```

Running your application on the Android emulator

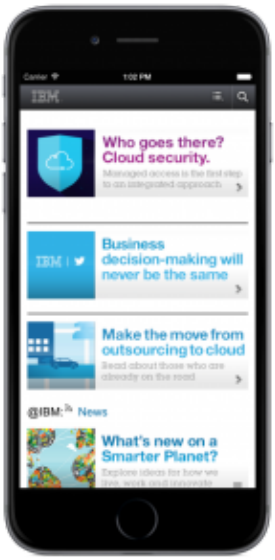
1. Build the Android environment.
2. Deploy the application.
3. Right-click the generated Android project and click **Run As** > **Android Application**.

You can see that the `http://m.ibm.com` URL is displayed in your Android emulator.



Running your application on the iOS emulator

1. Deploy the application to your iOS emulator.
2. Right-click the IOS environment and click **Run As** > **Xcode project**.



Sample application

Click to download the sample.