TapIt AIR Native Extension Plugin

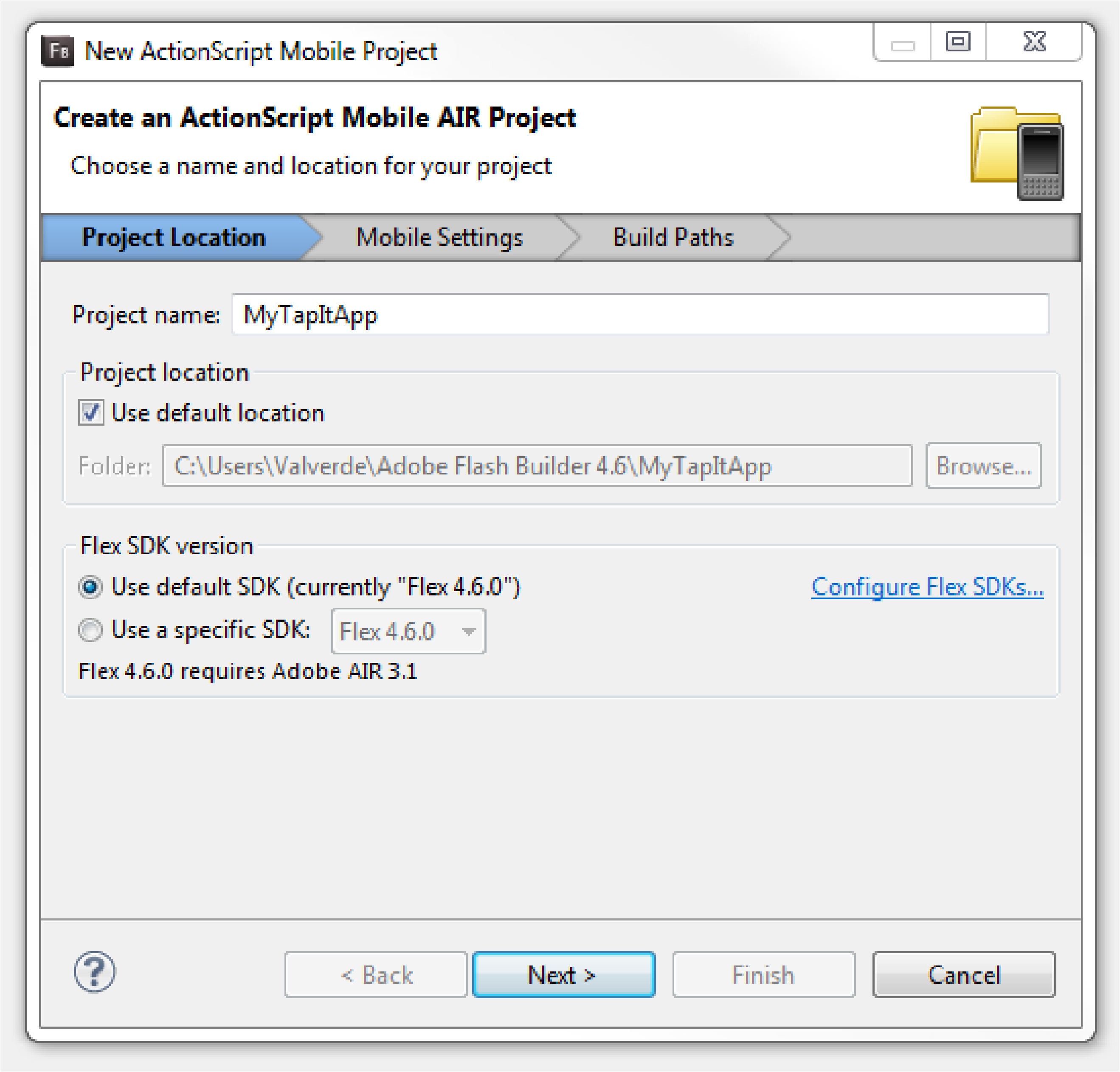
This is an Android AIR Native Extension (ane) built for TapIt Ads and ActionScript.

There are three types of ads:

Banner
Alert
Full Screen

Installation:

1. Create an AIR mobile project for Android.



New ActionScript Mobile Project

Create an ActionScript Mobile AIR Project

Choose a name and location for your project

Project name: MyTapItApp

Project location

☑ Use default location

Folder: C:\Users\Valverde\Adobe Flash Builder 4.6\MyTapItApp

Flex SDK version

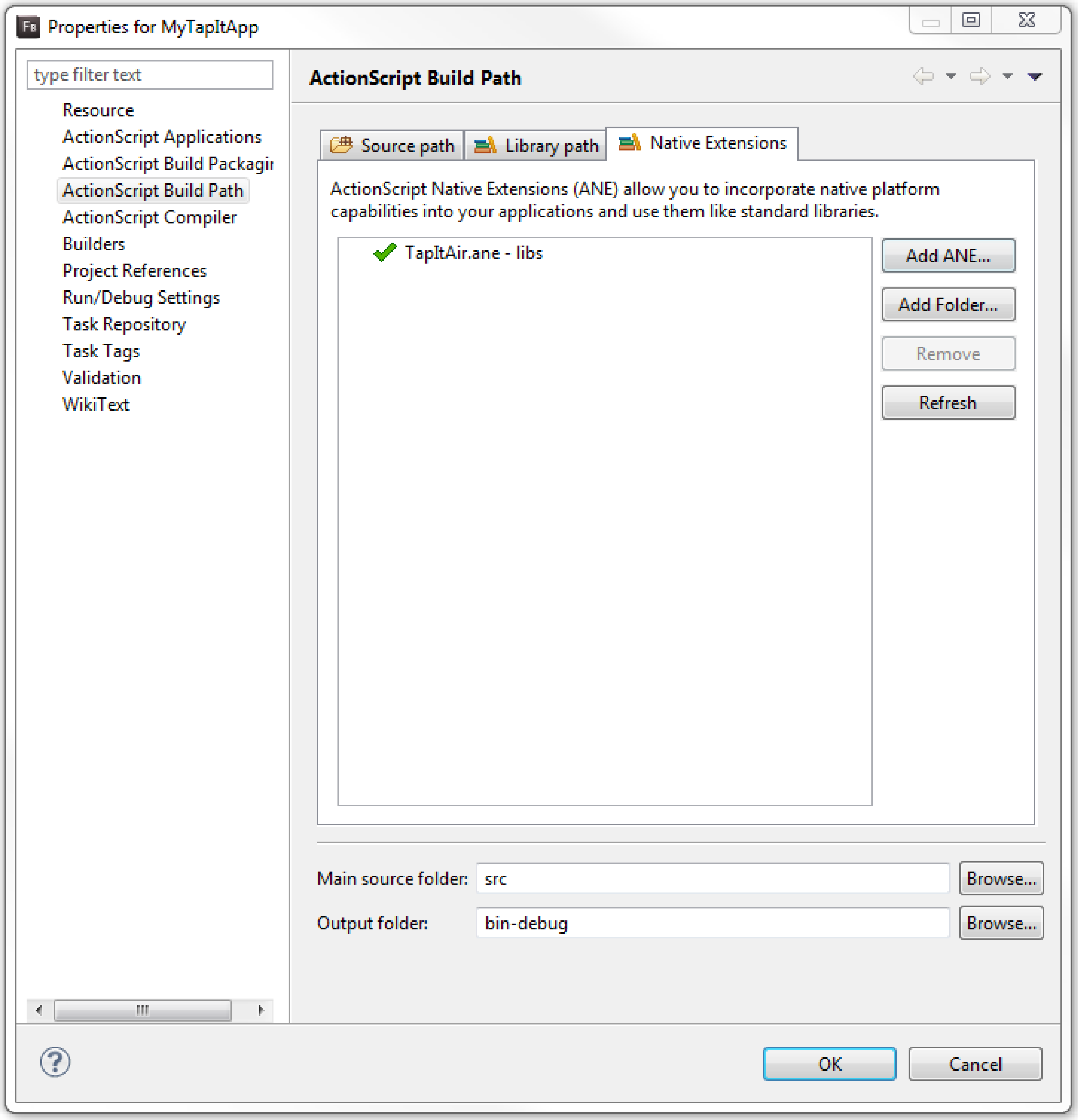
● Use default SDK (currently "Flex 4.6.0")

○ Use a specific SDK: Flex 4.6.0

Flex 4.6.0 requires Adobe AIR 3.1

2. Add the TapItAir.ane to your project build path.

Located under *project - properties - build path - Native Extensio*n



ActionScript Native Extensions (ANE) allow you to incorporate native platform capabilities into your applications and use them like standard libraries.

✓ TapItAir.ane - libs

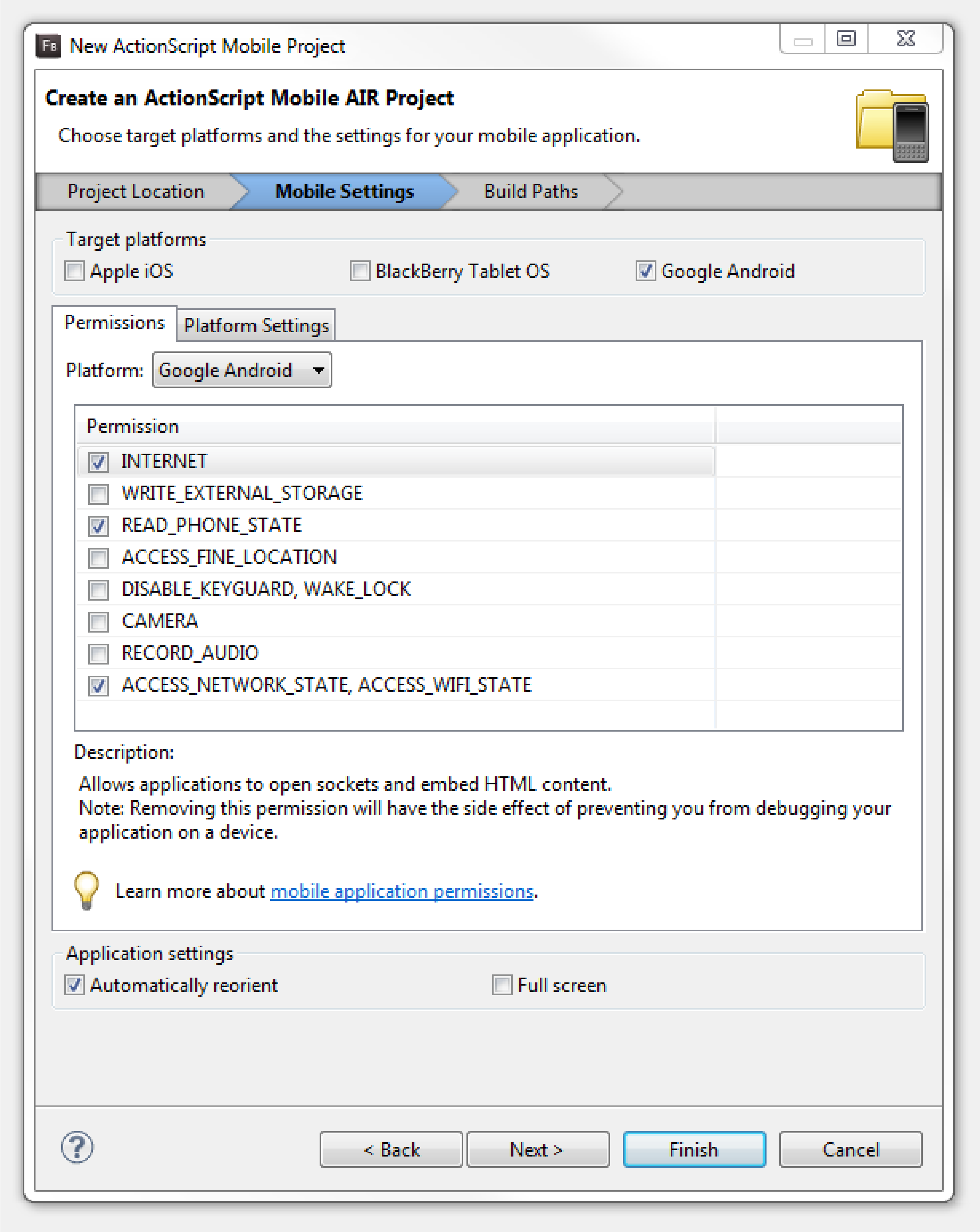
Main source folder: src

Output folder: bin-debug

3. Check the required permissions.

The TapItAir native extension requires specific permissions to run:

- `android.permission.INTERNET`
- `android.permission.ACCESS_NETWORK_STATE`
- `android.permission.READ_PHONE_STATE`



New ActionScript Mobile Project

Create an ActionScript Mobile AIR Project

Choose target platforms and the settings for your mobile application.

Target platforms

□ Apple iOS □ BlackBerry Tablet OS ☑ Google Android

Platform: Google Android

Permission
☑ INTERNET
□ WRITE_EXTERNAL_STORAGE
☑ READ_PHONE_STATE
□ ACCESS_FINE_LOCATION
□ DISABLE_KEYGUARD, WAKE_LOCK
□ CAMERA
□ RECORD_AUDIO
☑ ACCESS_NETWORK_STATE, ACCESS_WIFI_STATE

Description:

Allows applications to open sockets and embed HTML content.
Note: Removing this permission will have the side effect of preventing you from debugging your application on a device.

Learn more about mobile application permissions.

Application settings

☑ Automatically reorient □ Full screen

4. Add Activity to Android manifest:

The TapItAir native extension also requires a TapIt Android activity to be manually written to your project-app.xml:

The activity must be located within the Android manifest tags for the AIR application.

The android manifest tags are located in your projec -app.xml:

```
<android>
    <manifestAdditions><![CDATA[
        <manifest>
         </manifest>
    ]]></manifestAdditions>
</android>
```

within the manifest tags add the activity:

```
<application>
    <activity android:name="com.tapit.adview.AdActivity"
android:configChanges="keyboard|keyboardHidden|orientation"/>
</application>
```

If you choose not to check the permissions during setup-- The permissions must be manually entered into the android manifest for the AIR application.

```
<uses-permission android:name="android.permission.INTERNET"></uses-permission>
<uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"></uses-permission>
<uses-permission
android:name="android.permission.READ_PHONE_STATE"></uses-permission>
```

Within the manifest tags to finally appear as:

```
<android>
    <manifestAdditions><![CDATA[
        <manifest android:installLocation="auto">
            <uses-permission
android:name="android.permission.INTERNET"></uses-permission>
            <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"></uses-permission>
            <uses-permission
android:name="android.permission.READ_PHONE_STATE"></uses-permission>
            <application>
                 <activity android:name="com.tapit.adview.AdActivity"
android:configChanges="keyboard|keyboardHidden|orientation"/>
            </application>
        </manifest>
    ]]></manifestAdditions>
</android>
```

ActionScript usage:

Add Banner to the bottom of your mobile application.

```
package
{
    import flash.display.Sprite;
    import com.tapit.air.TapItAir;

    public class MyTapItApp extends Sprite
    {
        public function MyTapItApp()
        {
            TapItAir.addBanner();
        }
    }
}
```

Example for Banner sizing, position and zone:

```
package
{
    import flash.display.Sprite;
    import com.tapit.air.TapItAir;
    import com.tapit.air.BannerSizes;

    public class MyTapItApp extends Sprite
    {
        public function MyTapItApp()
        {
            TapItAir.addBanner(BannerSizes.IPHONE_BANNER, "top", "7979");
            // size 320 x 50;   position on top; publisher zone = 7979.
            //TapItAir.addBanner(BannerSizes.AUTOSIZE_AD, "bottom", "7979"); // default
        }
    }
}
```

Removing the Banner from mobile application:

```
package
{
    import flash.display.Sprite;
    import com.tapit.air.TapItAir;

    public class MyTapItApp extends Sprite
    {
        public function MyTapItApp()
        {
            TapItAir.removeBanner();
        }
    }
}
```

Banner size options:

- `BannerSizes.AUTOSIZE_AD = auto`
- `BannerSizes.IPHONE_BANNER = 320 x 50`
- `BannerSizes.XL_BANNER = 300x50`
- `BannerSizes.LARGE_BANNER = 216x36`
- `BannerSizes.MEDIUM_BANNER = 168x28`
- `BannerSizes.SMALL_BANNER = 120x20`

Add an Ad Alert overlay to you mobile application:

```
package
{
    import flash.display.Sprite;
    import com.tapit.air.TapItAir;

    public class MyTapItApp extends Sprite
    {
        public function MyTapItApp()
        {
            TapItAir.addAlert();

            // To change publisher zone -- TapItAir.addAlert("7984");
        }
    }
}
```

Add a FullScreen Ad to your mobile application:

```
package
{
    import flash.display.Sprite;
    import com.tapit.air.TapItAir;
    import com.tapit.air.BannerSizes;

    public class MyTapItApp extends Sprite
    {
        public function MyTapItApp()
        {
            TapItAir.addFullScreen();

            // To change publisher zone -- TapItAir.addFullScreen("7979");
        }
    }
}
```

How to receive a call back from the TapIt AIR Library:

```
package
{
    import flash.display.Sprite;
    import flash.events.StatusEvent;
    import com.tapit.air.TapItAir;

    public class MyTapItApp extends Sprite
    {
        public function MyTapItApp()
        {
            TapItAir.addEventListener(StatusEvent.STATUS, statusUpdate);

            TapItAir.addBanner();
        }

        private function statusUpdate(event:StatusEvent):void
        {
            if(event.code=="BANNER_ADDED")
            {
                // do something;
            }

            if(event.code=="BANNER_CLOSED")
            {
                // do something;
            }
        }
    }
}
```

Call back available codes:

- BANNER_ADDED
- BANNER_CLOSED
- BANNER_ERROR
- BANNER_CLICKED
- BANNER_START_FULLSCREEN
- BANNER_ADDED_FULLSCREEN
- BANNER_DISMISS_FULLSCREEN
- ALERT_ADDED
- ALERT_CLOSED
- ALERT_ERROR
- FULLSCREEN_START
- FULLSCREEN_LOADING
- FULLSCREEN_READY
- FULLSCREEN_ADDED
- FULLSCREEN_ERROR
- FULLSCREEN_CLICKED
- FULLSCREEN_DISMISSED
- FULLSCREEN_CLOSED

Note about multiple ads:

There can only be one instance of any of the available ad options.
You can add 1 banner, 1 ad alert and 1 full screen add-- at the same time, but you are unable to add multiple instances of any ad option.
i.e.-- not 2+ banners, 2+ ad alerts, or 2+ fullscreen ads.

Special note regarding debugging and testing via your IDE.

While this Native Extension is for Android, you must debug or run to a connected Android device or emulator. If you attempt to debug or run via your IDE, you will receive an error message like so:

```
ArgumentError: Error #3500: The extension context does not have a method with the name addBanner
    at flash.external::ExtensionContext/_call()
    at flash.external::ExtensionContext/call()
```