

Native Extension for App Rating dialog popup from Action Script.



Prerequisites:

- ✓ Flash Builder 4.6 or higher
- ✓ AIRSDK 3.4 or higher
- ✓ Eclipse with Android SDK

The following document shows you how to create your own AIR native extension for Android.

Here are the following steps to follow for creating “.ane” file and including that in user defined application. Just mentioned in a brief what steps are involved in making native extensions.

The challenge: App Rating dialog from Action Script android project.

The native extension has the following parts:

- 1) Creating native java library.
- 2) Create Action Script library part.
- 3) Create an extension.xml file
- 4) Building the .ane file (AIR native Extension).
- 5) Embedding the .ane file in an application

Creating Native JAVA library

- ✓ Create an android project. Don't create an activity.
- ✓ When Project is created, open the project properties and add the AIR extension library **FlashRuntimeExtensions.jar** from AIRSDK\lib\android\FlashRuntimeExtensions.jar (Properties--> Java Build Path-->Add External Jars).
- ✓ Create a class Extension class. From the example you can get the class **AppRatingExtension.java**
- ✓ Create a Context Class. **AppRatingExtensionContext.java**
- ✓ After creating all particular files clean the project from Project tab and create jar file by Exporting the Project. (Right click on AndroidProject -->Export).

Create Action Script Library

- ✓ Create new Action script library project. (New->Flex Library Project). Include the AIRSDK library from project properties
- ✓ Fundamentally Action Script Class has two parts.
 - I. Create and Initialize the Context.
 - II. Call functions in Java Code and return the results if required

You can get the Class from the example with the name **“AppRatingExtensionScript.as”**

- ✓ Flash Builder will generate the action script **SWC** file. Now build the project. It will generate the .swc file. Now change the extension of .swc file with .zip file. Now extract the zip file. You will find the library file from extracted file.

Creating extension.xml File

Create an extension.xml file which contains **platform**, **nativeLibrary** and **initializer** tags. You can get the sample from the above example file “extension.xml”

Platform: which platform you want to create the .ane file. For this example it’s “Android-ARM”

Native Library: Specifies the jar file generated from the native java code (in this example it is AppRatingApp.jar)

Initializer: The class that implements FREEExtension (In this example it is “com.mangogames.apprating”)

Building the .ane file

Follow the below structure for creating ane file

Command for creating .ane file is:

```
adt -package -target ane Output.ane extension.xml -swc VolumeLib.swc -platform iPhone-ARM -C ios
. -platformoptions platformoptions.xml -platform Android-ARM -C android . -platform default -
C default .
```

Example:

```
adt -package -target ane AppRating.ane extension.xml -swc AppRatingExtensionScript.swc -platform
Android-ARM -C android .
```

Embedding the .ane file in an application

- ✓ Create an Action Script Mobile Project (File->New->ActionScriptMobileProject)
- ✓ Include the ane file in the actionscriptmobileproject as right click on your mobile project->properties->ActionScript Build Path->Native Extensions tab and then click on AddAne. After adding Click OK.
- ✓ Then right click on your mobile project ->properties->ActionScript Build Packaging -> Choose Google Android. In that go to Native Extensions tab and check the ANE Details under Package. Click Apply and OK.
- ✓ Now Clean the Project and Connect the Device and debug the application.

The AIR native extension will be included.

NOTE: While including the ANE in ActionScript mobile project one thing we need to keep in mind. The xxx-app.xml will be generated in the project if we include the native extension. In that file for this project, if any permission required for the native extension then we need to assign like Internet permission.