



Guidelines to use Mocean SDK Adapter for MoPub Android SDK

Android SDK Version 4.3.3

Dec 9, 2015

© 2015 PubMatic Inc. All rights reserved. Copyright herein is expressly protected at common law, statute, and under various International and Multi-National Treatises (including, but by no means limited to, the Berne Convention for the Protection of Literary and Artistic Works).

The following documentation, the content therein and/or the presentation of its information is proprietary to and embodies the confidential processes, designs, and technologies of PubMatic Inc. All copyrights, trademarks, trade names, patents, industrial designs, and other intellectual property rights contained herein are, unless otherwise specified, the exclusive property of PubMatic Inc. The ideas, concepts, and/or their application, embodied within this documentation remain and constitute items of intellectual property which nevertheless belong to PubMatic Inc.

The information (including, but by no means limited to, data, drawings, specification, documentation, software listings, source and/or object code) shall not be disclosed, manipulated, and/or disseminated in any manner inconsistent with the nature and/or conditions under which this documentation has been issued.

The information contained herein is believed to be accurate and reliable. PubMatic Inc. accepts no responsibility for its use in any way whatsoever. PubMatic Inc. shall not be liable for any expenses, damages, and/or related costs, which may result from the use of any information, contained hereafter.

PubMatic Inc. reserves the right to make any modification to this manual or the information contained herein at any time without notice.

CORPORATE HEADQUARTERS

PubMatic, Inc.
305 Main Street Suite 100
Redwood City, CA 94063
USA

www.pubmatic.com

Change History

Version	Date	Changes
1.0	Oct 7, 2015	Initial Release
2.0	Dec 9, 2015	Changes for PubMatic SDK v4.3.3

Table of Contents

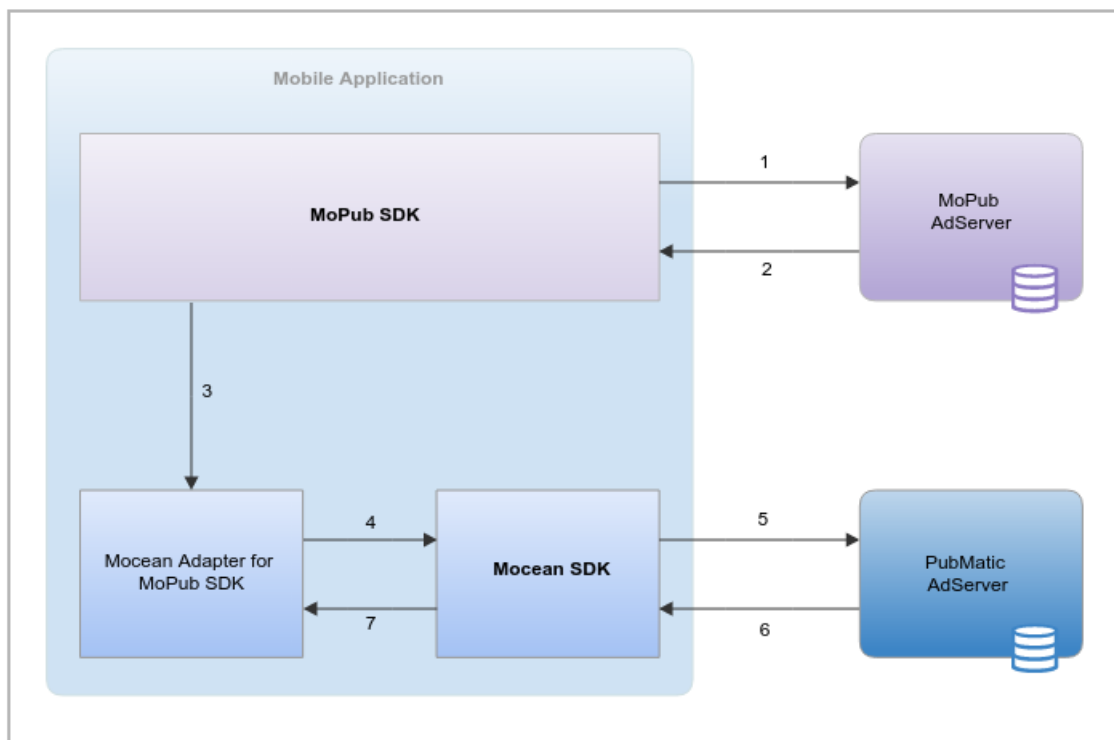
Introduction	5
Mediation Ad Serving call flow	6
Steps to integrate Mocean Adapter and Native SDK to application	7
Configuring Mocean Network on MoPub Portal	10
References	13

Introduction

In order to use PubMatic as a client side demand source, publishers can use third party mediation support in MoPub SDK to integrate PubMatic Mocean Android SDK for Native ads. PubMatic provides client side mediation adapter to integrate Mocean SDK with MoPub SDK. Publishers can easily integrate Mocean adapter and Mocean SDK into mobile application with minimal code changes.

This document provides guidelines for integrating Mocean Native SDK into MoPub SDK using Mocean adapter.

Following diagram illustrates the call flow in case of mediation ad serving.

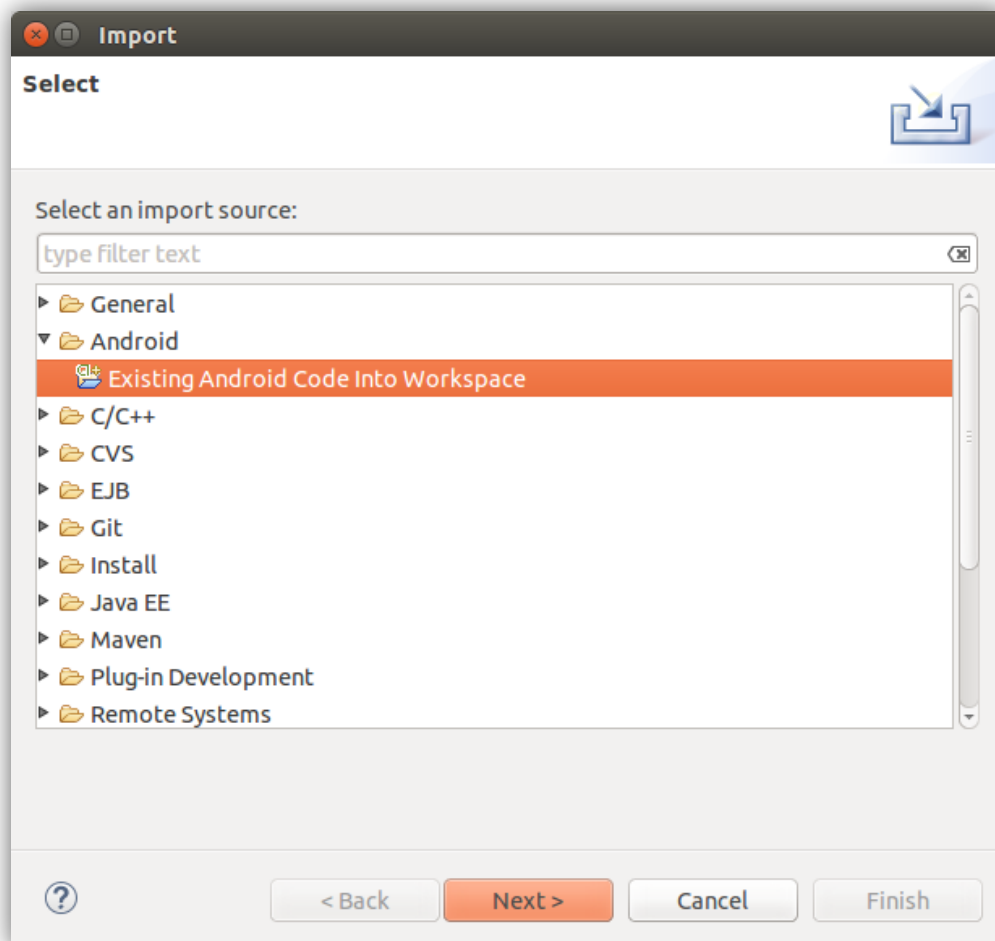


Mediation Ad Serving call flow

1. MoPub SDK which is integrated into application, requests ad from MoPub ad server.
2. MoPub responds back with mediation response containing Custom Event Class name (`com.mopub.mobileads.MoceanNativeAdapter`) and server extra parameters containing Mocean placement zoneid.
3. On receiving mediation response, MoPub SDK initializes Mocean adapter using class name, and pass all server extra parameters.
4. Mocean adapter initializes Mocean SDK with zoneid.
5. Mocean SDK then makes call to PubMatic ad server requesting ad.
6. PubMatic ad server responds with ad.
7. Mocean SDK sets the received native ad assets to MoPub SDK, which in turn provides the assets to publisher with standard callbacks.

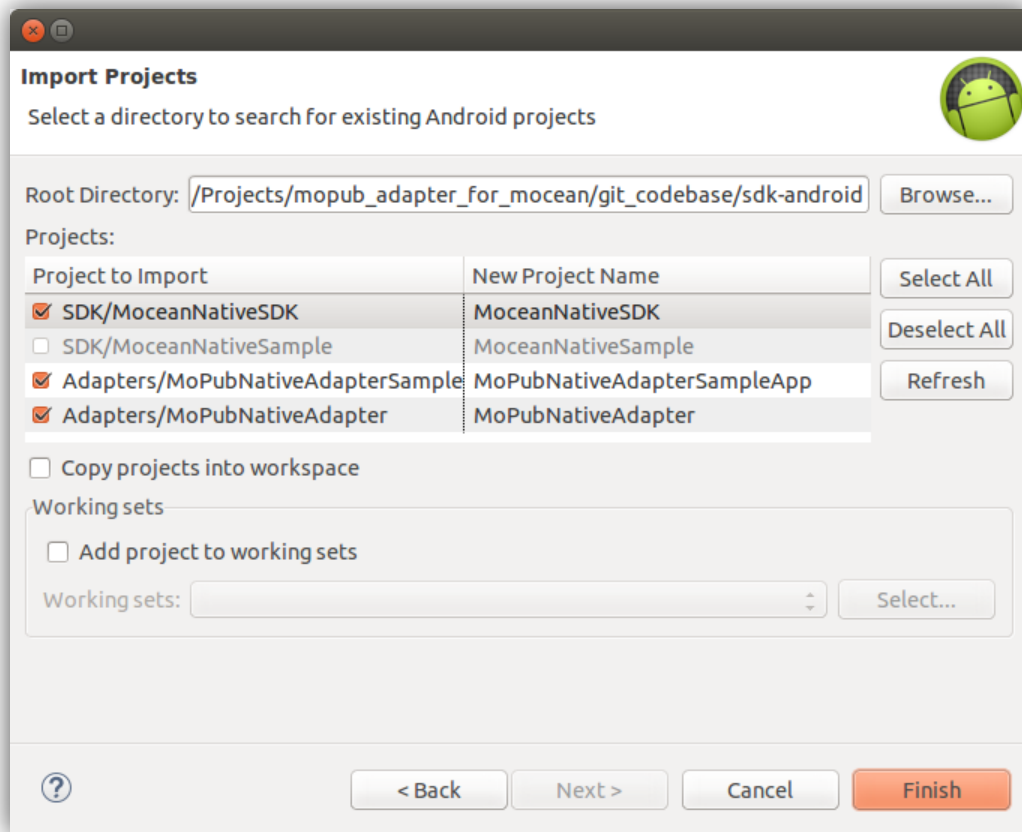
Steps to integrate Mocean Adapter and Native SDK to application

1. Download **Mocean SDK and Adapter**.
2. You can add **Mocean Native Adapter and Native SDK** as jar file dependency OR you can add both these projects as Android Library Projects and link to your application in Eclipse.
Following are the steps to add Mocean Native Adapter and Native SDK as Android library project.
3. Import **Mocean Native SDK** and **Mocean Native Adapter** as Android library project.

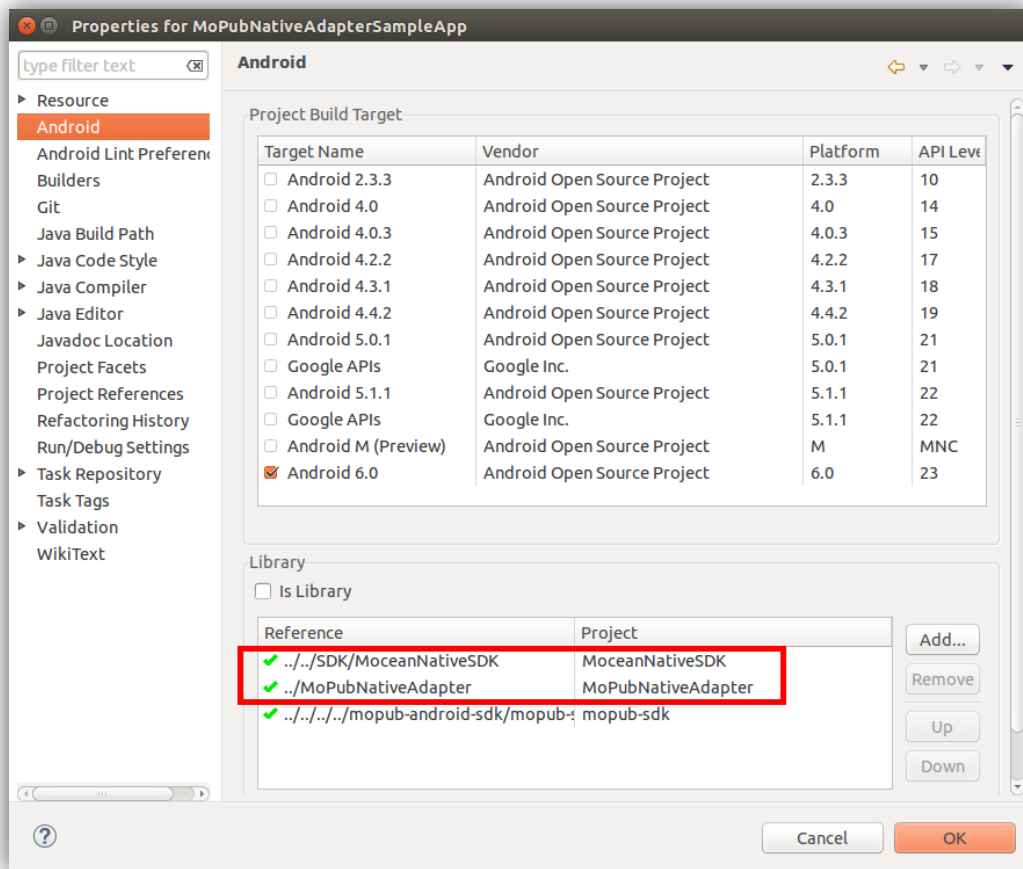


4. Import **MoceanNativeSDK** and **MoPubNativeAdapter** library projects into your eclipse workspace.

5. Optionally, you can also import **MoPubNativeAdapterSampleApp** which contains a sample implementation of Native adapter.



6. Link both these projects as Android library project to your application in Eclipse.



OR,

7. Instead of adding **Mocean Native adapter** and **Native SDK** as Android library project, you can add them as jar file dependencies to your project.

To add the adapter and SDK as jar file dependency, copy

SDK/MoceanNativeSDK/bin/**moceannativesdk.jar** and

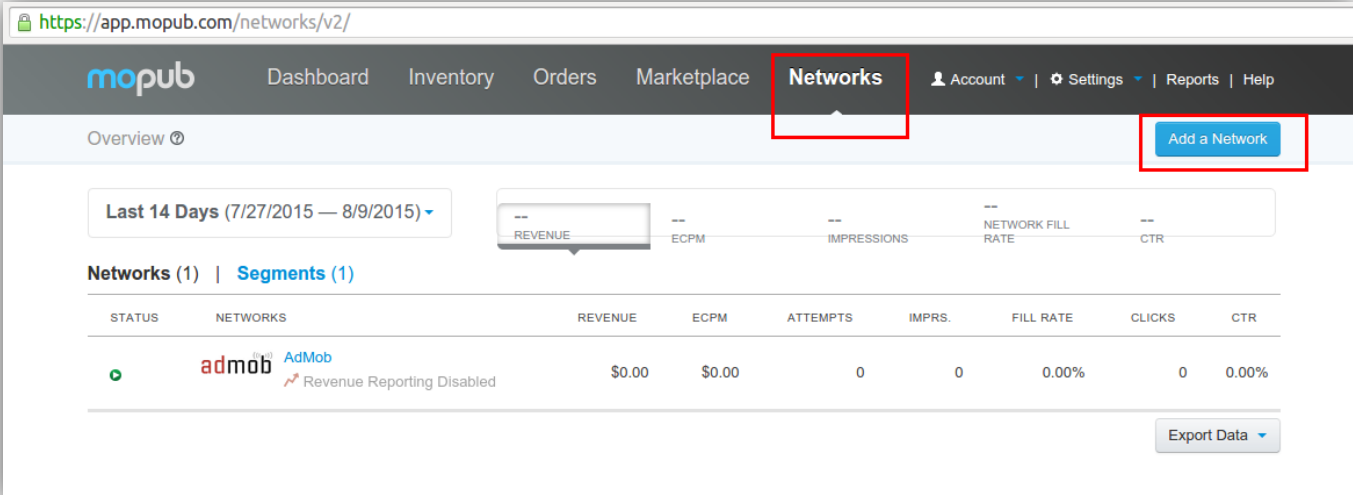
Adapters/MoPubNativeAdapter/bin/**mopubnativeadapter.jar**

to your projects **/libs** folder. Make sure that both these files are added to your projects classpath.


Configuring Mocean Network on MoPub Portal

In order to configure Custom Event on MoPub portal, follow the below steps.

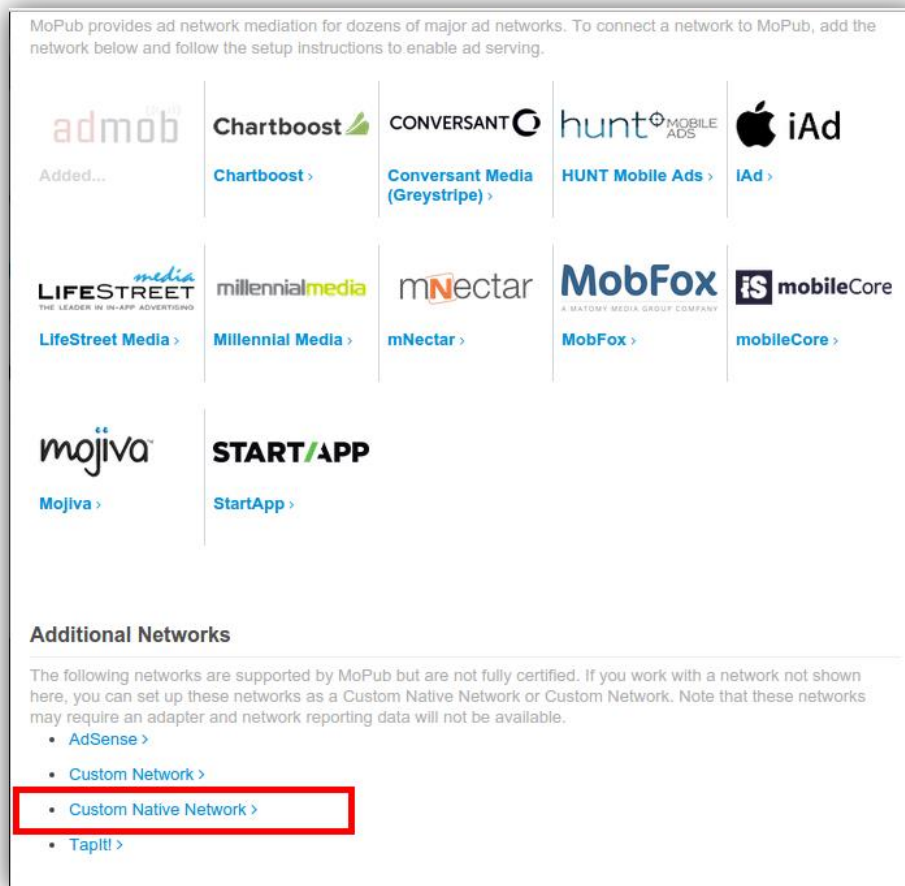
1. Login into the **MoPub** account ([link](#)).
2. Navigate to the **Networks** tab and click **Add a Network**.



The screenshot shows the MoPub portal interface. The 'Networks' tab is selected in the top navigation bar. The 'Add a Network' button is highlighted with a red box. Below the navigation bar, there is a section for 'Last 14 Days (7/27/2015 — 8/9/2015)' with a dropdown menu. To the right of this section are five summary cards: REVENUE, ECPM, IMPRESSIONS, NETWORK FILL RATE, and CTR. Below these cards, there is a table with the following columns: STATUS, NETWORKS, REVENUE, ECPM, ATTEMPTS, IMPRS., FILL RATE, CLICKS, and CTR. The table contains one entry for 'admob AdMob' with a status of 'Revenue Reporting Disabled'. The values for the other columns are \$0.00, \$0.00, 0, 0, 0.00%, 0, and 0.00%. An 'Export Data' button is located at the bottom right of the table.

STATUS	NETWORKS	REVENUE	ECPM	ATTEMPTS	IMPRS.	FILL RATE	CLICKS	CTR
	admob AdMob Revenue Reporting Disabled	\$0.00	\$0.00	0	0	0.00%	0	0.00%

3. On *Add a network* pop-up, click **Custom Native Network**.



4. Give a suitable name to Mocean custom network.
5. In set up inventory section, add the custom event class and data only for the native ad unit for which you want to enable Mocean network.
6. In the custom event class field add following class name for Mocean Native Adapter.

```
com.mopub.mobileads.MoceanNativeAdapter
```

7. The Custom Event Class Data field takes a JSON containing Mocean placement zoneId. Enter zoneId provided by your PubMatic account manager, in the following format.

```
{"zoneId": "<your_zone_id>"}
```

Example:

```
{"zoneId": "123456"}
```

Please enter Custom Event Class name and data exactly in the same format mentioned above. Any changes in these may cause failure in initializing Mocean adapter.

8. Optionally, if required, you can also set custom base URL for Mocean ad server using **adServerUrl** JSON parameter.

```
{"zoneId": "<your_zone_id>", "adServerUrl": "<custom_base_url>"}
```

Example:

```
{"zoneId": "123456", "adServerUrl": "http://customUrl.com/ad?"}
```

Custom Native

Set up Custom Native Network


Special Instructions

- Custom Native Networks require the development of a Custom Event Class
- You must enter a Custom Class to enable ad serving for this type of network. Custom Event Class Data can be sent down in JSON format. Note: Custom Native Networks using a Custom Method implementation will not be supported in future MoPub SDKs.

Title

Set Up Your Inventory

Filters: App: TestMoceanNativeAdapter Clear All | X

APP AND ADUNITS	PLATFORM	AD FORMAT	CUSTOM EVENT METHOD	CUSTOM EVENT CLASS	CUSTOM EVENT CLASS DATA
 TestMoceanNativeAdapter	Android		No Custom Content Entered	<input type="text" value="com.mopub.mobileads.MoceanNativeAdapter"/>	<input data-bbox="1177 1123 1421 1165" type="text" value='{"zoneId": "123456"}'/>
 TestMoceanNativeAdapterAdUnit		Native			

9. Save the network. Please note that it may take from 15 to 30 minutes for MoPub ad server to reflect this change.
10. Done!

References

1. [MoPub Custom Events Wiki](#)
2. [MoPub Custom Native Network Setup](#)