Mocean Android SDK Developer Guide

For Android SDK Version 2.12

136 Baxter St, New York, NY 10013

Table of Contents

Overview

This document provides a description of the following:

- What's new in 2.12
- How to upgrade from previous versions
- System requirements
- Installation Instructions (revised)
- SDK contents
- SDK API reference
- Sample of usage

What's new in 2.12:

- Removed deprecated *isContentAligned* property; alignment is now the default, and can be changed by customizing the ad view injection code with the *setInjectionHeaderCode*() method.
- Replaced Metro property with Dma property (see the *setDma*() method); Metro methods still exist but are now deprecated and will be removed in the future.
- Removed the confusing *injectionBodyCode* property, and moved the body style definition CSS to the header field (see *setInjectionHeaderCode*() method).
- Updated Sample app to demonstrate usage of revised *injectionHeaderCode*() method.
- Corrections to visibility of many methods in MASTAdView, MASTAdViewCore and MASTAdLog clases.
- All public methods intended for SDK users now have javadoc documentation.
- Change to the *MASTOnAdClickListener* interface *event*() method so that implementations can return a Boolean false if they want the standard SDK behavior, or true if the click event has been fully processes and the SDK should ignore it (which was the past behavior.)
- Add correctly spelled string STR_EMPTY_SERVER_RESPONSE in Constants.java and aliases old misspelled (and now deprecated) string to this name.

How to upgrade from previous versions:

1. Replace usage of setInjectionBodyCode() method with revised setInjectionHeaderCode() method.
2. Remove any use of deprecated setContentAlignment() method.
3. Replace usage of the setMetro() property with setDma().
4. Examine ad update any usage of the *MASTOnAdClickListener* interface.

System requirements:

- Android SDK (platform version 2.0 or later)
- Eclipse 3.5 or later
- 10 Mb free disk space

SDK contents:

- Lib - SDK library files
- Sample - sample usage/test app

Installation instructions:

Installing Android SDK
Download and install Android SDK (http://developer.android.com/sdk/index.html).
Follow installation instructions provided by Android SDK. Once SDK has been installed follow to the next step to install Android mOcean SDK.

Installing Android mOcean SDK
The SDK is distributed as a library source code project, but still includes a pre-compiled jar library. To add the SDK to a project, the developer must configure the project properties to indicate the location of SDK files, as well as the names of library dependencies.

1. Unpack the SDK zip file into a convenient location in your source code working area.
2. Open or create a new Android project in the Eclipse development environment.
3. Import the SDK project into your workspace as an existing Android project.
    a. Choose Import from the File menu, then Existing Projects into Workspace as show in Figure 1 below.

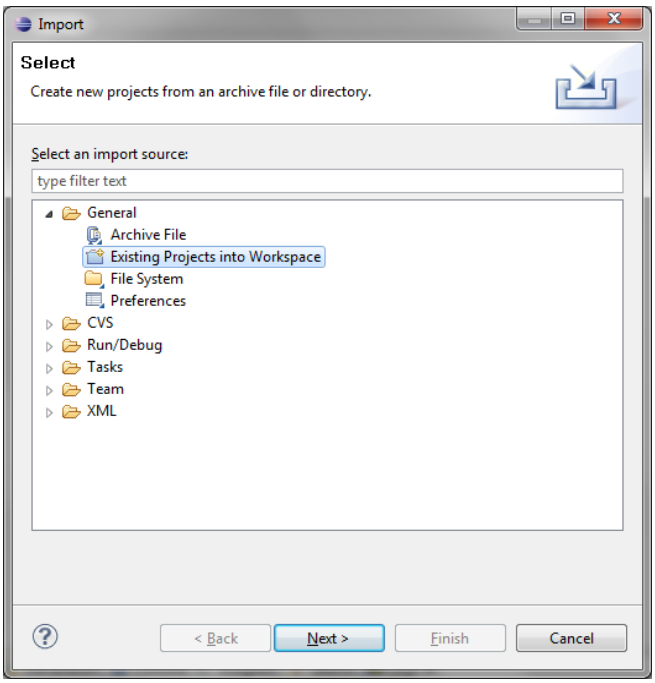

Figure 1- Import Existing Project

b. Browse to the location where you unpacked the SDK file and import the AdserverView project; you can also optionally import the Samples project if you want to work with the SDK sample application. See Figure 2 below for an example.

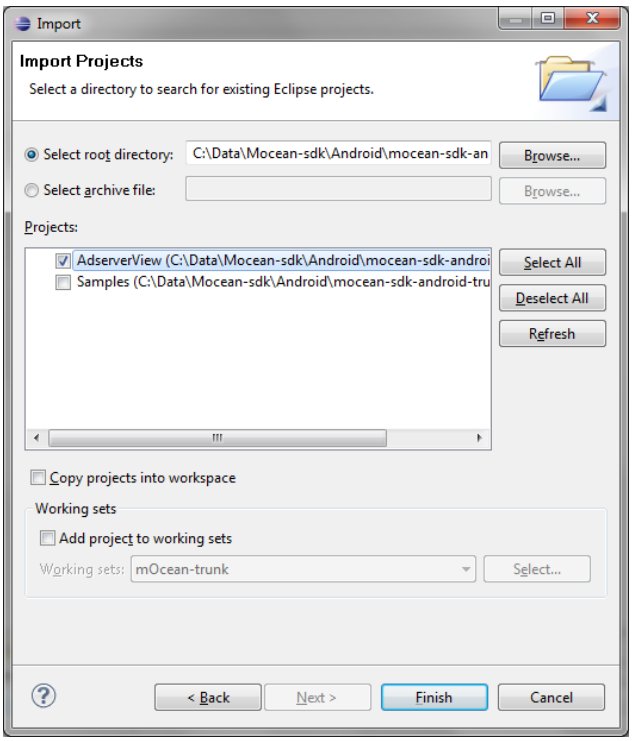

Figure 2 - Import Projects into SDK

c. Choose Properties from the Project menu, and then select the Java Build Path category followed by the Libraries tab, as shown in Figure 3 below.

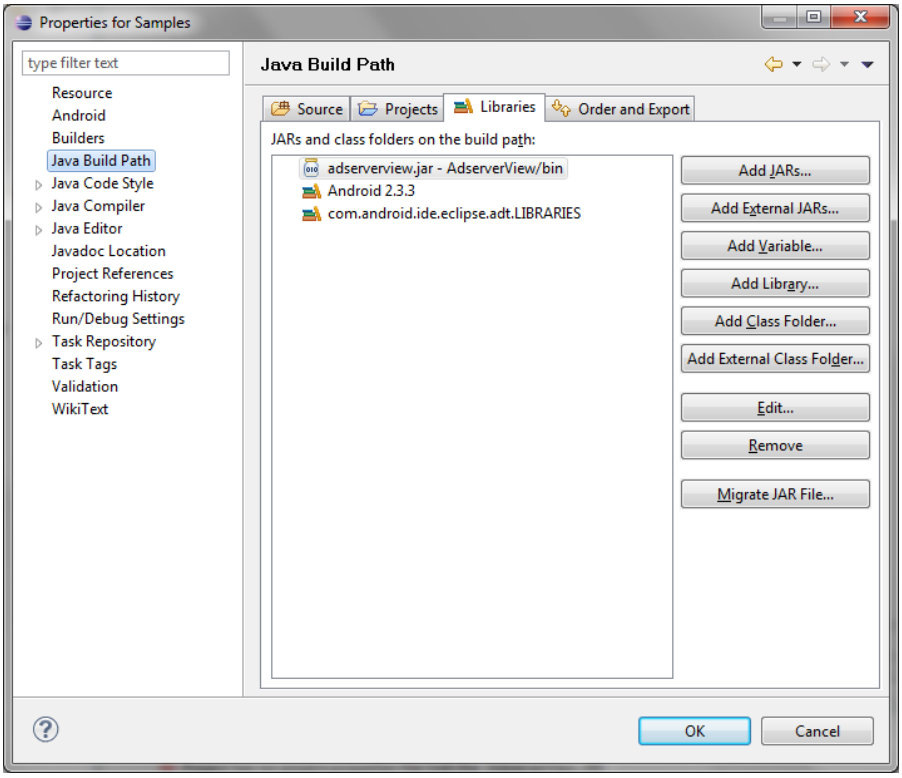

Figure 3 - Add Library Jar

d. Choose the Add Jar button, and then navigate into the bin folder of the AdserverView project and choose the adserverview.jar Jar file as shown in Figure 4 below.

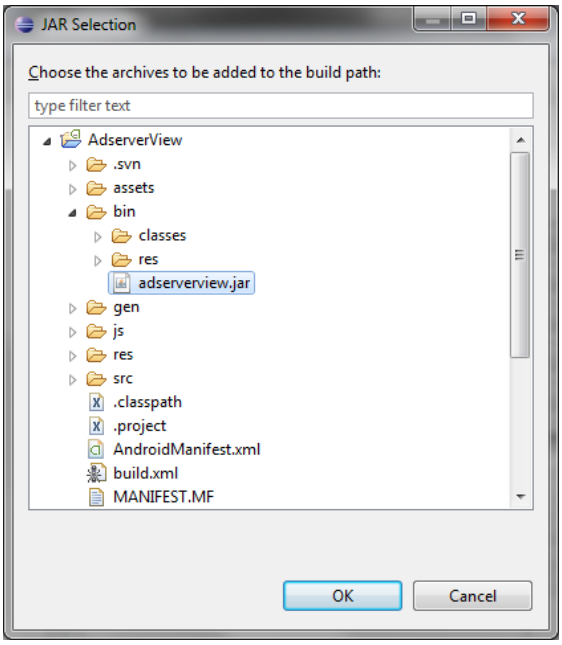

Figure 4 - Ad SDK Jar File

e. **IMPORTANT**: If using release 18 or later of the Android SDK tools, choose the Order and Export tab, and check the box to export the SDK Jar file as shown in Figure 5 below.

Without this, applications will compile but the resulting apk file will not include the required SDK code and the app will crash at runtime due to missing symbols.

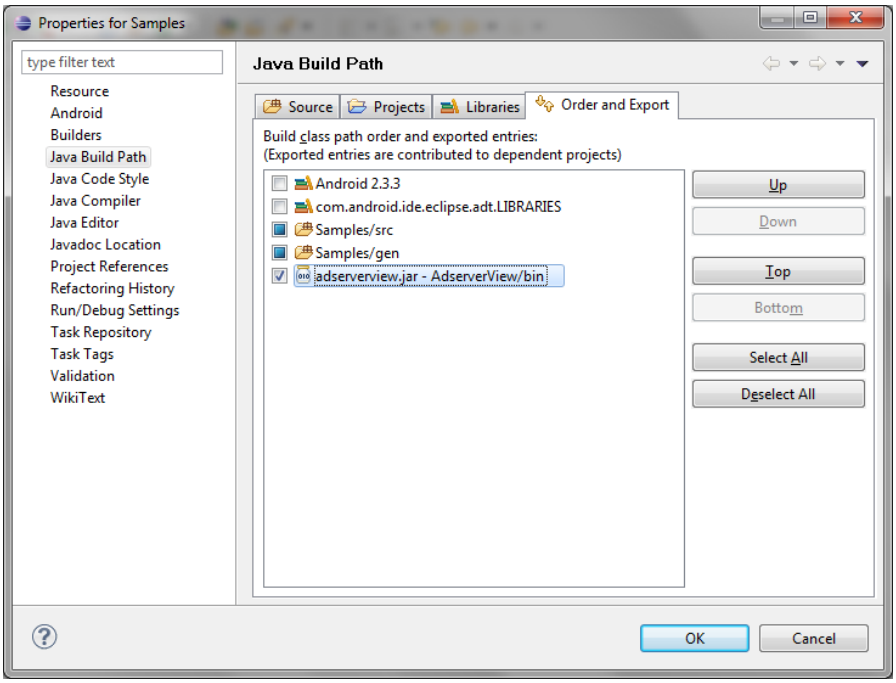

Figure 5 - Export SDK Jar File

Updating the manifest file (AndroidManifest.xml)
Add “minSdkVersion” parameter in project manifest file (AndroidManifest.xml)
Example: <uses-sdk android:minSdkVersion="5" />

Set the security permissions in your manifest file (AndroidManifest.xml)

Permission	Description & Manifest XML fragment
INTERNET	Access the Internet. Required for ad content download. <uses-permission android:name="android.permission.INTERNET"></ uses-permission>
Network State	Access the network state. Required for ad request parameter setting, and ORMMA support. <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"></ uses-permission>
Fine Location	Use GPS to obtain location information. Needed if SDK or ORMMA ad enables location detection; off by default. <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"></ uses-permission>
Phone State	Read state of phone data connection. Required for ad request parameter setting. <uses-permission android:name="android.permission.READ_PHONE_STATE"></ uses-permission>
Read Calendar	Read calendar events. Required if ORMMA ad makes use of calendar features. <uses-permission android:name="android.permission.READ_CALENDAR"></ uses-permission>
Write Calendar	Write calendar events. Required if ORMMA ad makes use of calendar features. <uses-permission android:name="android.permission.WRITE_CALENDAR"></ uses-permission>
Call Phone	Initiate a phone call. Required if an ad makes use of the ORMMA feature to place a phone call. <uses-permission android:name="android.permission.CALL_PHONE"></ uses-permission>
Send SMS	Send an SMS (text) message. Required if an ad makes use of the ORMMA feature to send a text message. <uses-permission android:name="android.permission.SEND_SMS"></ uses-permission>
Camera	Use camera to take a picture. Required if an ad makes use of the ORMMA feature to capture photos. <uses-permission android:name="android.permission.CAMERA"></ uses-permission>
External Storage	Access the SD card storage area. Required for debug logs, photo, and file access to support SDK logging and ORMMA features. <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"></ uses-permission>
Wake Lock	Allows using PowerManager WakeLocks to keep processor from sleeping or screen from dimming . <uses-permission android:name="android.permission.WAKE_LOCK" / >

SDK API reference:

Global functions:

MASTAdLog.setDefaultLogLevel(int logLevel)

Set default log level to one of the log level values defined in he MASTAdLog class (corresponding to errors, errors + warnings, or everything including server traffic.)

The following values are allowable:

AdLog.LOG_LEVEL_NONE	none
AdLog.LOG_LEVEL_1	only errors
AdLog.LOG_LEVEL_2	+warning
AdLog.LOG_LEVEL_3	+server traffic

AdLog.*setFileLog*(String fileName)

File to output log data to.

MASTAdView(Context context, java.lang.String site, java.lang.String zone)

Creation of view showing ad content.

void **show**()

Show interstitial ad, which appears as a full screen and will popup over top of the current application activity. A close standard close button will be included at the bottom of the view.

void **SetLogLevel**(int logLevel)

Set log level for one ad view instance using the log levels shown above.

All settings can be set by means of methods get and set (for example: getCity() and setCity(String city)).

java.lang.String getAdserverURL()
Optional. Get URL of ad server this ad view will use when communicating with back end.

java.lang.String getArea()
Optional. Get Area code of a user, if any.

java.lang.String getBackgroundColor()
Optional. Get Background color of ad view.

java.lang.String getCarrier()
Optional. Get User carrier name, if any set.

java.lang.String getCity()
Optional. Get City of the device user (with state), if any set.

java.lang.String getCountry()
Optional. Get Country of ad viewer, if any set.

java.util.Hashtable<java.lang.String,java.lang.String> getCustomParameters()
Optional. Get Custom Parameters, to be passed to back-end server to help with ad selection, if any set.

java.lang.Integer getDefaultImage()
Optional. Get image resource identifier which will be shown during ad loading if there is no ad content in cache, if any set.

String getInjectionHeaderCode()
Optional. Get code fragment to be injected into HTML header to set viewport and body style.

boolean getInternalBrowser()
Get current setting for internal browser usage.

java.lang.String getKeywords()
Optional. Get Keywords sent to server to search for ads; comma-delimited, if any set.

java.lang.String getLatitude()
Optional. Get user location latitude value (given in degrees.decimal degrees), if any.

float setLocationMoveDistance()
Optional. Get distance device must move to trigger a location update.

int setLocationMinWait()
Optional. Get time interval in milliseconds between location updates.

java.lang.String getLongitude()
Optional. Get user location longitude value (given in degrees.decimal degrees), if any.

java.lang.Integer getMaxSizeX()
Optional. Get maximum width of ad to be requested from server.

java.lang.Integer getMaxSizeY()
Optional. Get maximum height of ad to be requested from server.

java.lang.String getDma()
Optional. Get Dma code of a user, if any set.

java.lang.Integer getMinSizeX()
Optional. Get minimum width of ad to request from server.

java.lang.Integer getMinSizeY()
Optional. Get minimum height of ad to request from server.

MASTAdViewCore.OnAdClickListener getOnAdClickListener()
Get interface for ad view with a click() method which will be invoked when loading a URL from an ad.

MASTAdViewCore.OnAdDownload getOnAdDownload()
Get object for handling ad download events.

java.lang.Integer getPremium()
Optional. Get Filter by premium setting, if any.

java.lang.String getRegion()
Optional. Get Region of ad viewer, if any set.

java.lang.String getSite()
Required. Get the id of the publisher site.

java.lang.String getState()
Get name of current view state for the ad view. Will have one of the values: Default, Resized, Expanded or Hidden.

java.lang.Boolean getTest()
Optional. Get test mode setting, if any.

java.lang.String getTextColor()
Optional. Get Text color value for links.

java.lang.Integer getUpdateTime()
Optional. Get banner refresh interval (in seconds).

java.lang.String getZip()
Optional. Get Zip/Postal code of user, if any.

java.lang.String getZone()
Required. Get the id of the zone of publisher site.

void setAdserverURL(java.lang.String adserverURL)
Optional/Advanced. Overrides the URL of ad server SDK will communicate with.

void setArea(java.lang.String area)
Optional. Set Area code of a user.

void setBackgroundColor(java.lang.String backgroundColor)
Optional. Set Background color of ad view.

void setCarrier(java.lang.String carrier)
Optional. Set User carrier name, to be passed to back-end server to help with ad selection.

void setCity(java.lang.String city)
Optional. Set City of the device user (with state), to be passed to back-end server to help with ad selection.

void setCountry(java.lang.String country)
Optional. Set Country of ad viewer, to be passed to back-end server to help with ad selection; use ISO 3166 format.

void setCustomParameters(java.util.Hashtable<java.lang.String,java.lang.String> customParameters)
Optional. Set Custom Parameters send to server with ad request.

void setDefaultImage(java.lang.Integer defaultImage)
Optional. Set image resource which will be shown during ad loading if there is no ad content in cache.

void setInternalBrowser(boolean internalBrowser)
Set flag controlling use of internal browser field when opening ad URLs.

void setKeywords(java.lang.String keywords)
Optional. Set Keywords to search ad, delimited by commas, to be passed to back-end server to help with ad selection.

void setLatitude(java.lang.String latitude)
Optional. Set user location latitude value (given in degrees.decimal degrees).

void setLongitude(java.lang.String longitude)
Optional. Set user location longitude value (given in degrees.decimal degrees).

void setMaxSizeX(java.lang.Integer maxSizeX)
Optional. Set maximum width of ad to request from back-end.

void setMaxSizeY(java.lang.Integer maxSizeY)
Optional. Set maximum height of ad to request from back-end.

void setDma(java.lang.String metro)
Optional. Set Dma code of a user, to be passed to back-end server to help with ad selection.

void setMinSizeX(java.lang.Integer minSizeX)
Optional. Set minimum width of ad to request from back-end.

void setMinSizeY(java.lang.Integer minSizeY)
Optional. Set minimum height of ad to request from back-end.

void setOnAdClickListener(MASTAdViewCore.OnAdClickListener adClickListener)
Set interface for ad view with a click() method which will be invoked when loading a URL.

void setOnAdDownload(MASTAdViewCore.OnAdDownload adDownload)
Set handler for ad download events.

void setPremium(java.lang.Integer premium)
Optional. Set premium ad flag used to help select ads for display.

void setRegion(java.lang.String region)
Optional. Set Region of viewer, to be passed to back-end server to help with ad selection.

void setSite(java.lang.String site)
Required. Set the id of the publisher site, used when sending request for ads to back-end.

void setTest(java.lang.Boolean enabled)
Optional. Set test mode flag, for requesting test ads from server.

void setTextColor(java.lang.String textColor)
Optional. Set Text color of links.

void setUpdateTime(java.lang.Integer updateTime)
Optional. Set banner refresh interval (in seconds).

void setZip(java.lang.String zip)

Optional. Set Zip/Postal code of user, to be passed to back-end server to help with ad selection.

void setZone(java.lang.String zone)

Required. Set the id of the zone to send to back-end when retrieving ads.

void update()

Immediately update ad view contents.

void setInjectionHeaderCode(String value)

Optional. Customize code added to HTML header when creating ad content web view, to setup the viewport and body CSS style. Default value is:

```
<meta name="viewport" content="target-densitydpi=device-dpi"/>
<style>body{margin: 0px; padding: 0px; display:-webkit-box;-webkit-box-
orient:horizontal;-webkit-box-pack:center;-webkit-box-
align:center;}</style>
```

void setLocationMoveDistance(float meters)

Optional. Set distance (in meters) device must move to trigger a location update. Default value is 1000 meters. Only relevant if the setLocationDetection property has been set to true.

void setLocationMinWait(int millis)

Optional. Set the delay (in milliseconds) between periodic location updates from the OS. Default value is 5 minutes. Only relevant if the setLocationDetection property has been set to true.

void setUseSystemDeviceId(boolean value)

Optional. Set flag indicating permission to use the system device ID, or not.

Sample usage

See the Sample project source code included with the SDK for a more complete example, and also consult the new Getting Started Guide available for download along with this SDK package.

To create view of advertising and to add it to the form, use one of the following two variations:

1. Dynamic creation, is created in a form **class**; for example:

```
MASTAdView adserverView = new MASTAdView(this, "5441", "9312");
adserverView.setLayoutParams(new
ViewGroup.LayoutParams(ViewGroup.LayoutParams.FILL_PARENT, 250));
adserverView.setTest(false);
adserverView.setPremium(MASTAdView.PREMIUM_STATUS_PREMIUM);
adserverView.setKeywords("test");
adserverView.setMinSizeX(200);
adserverView.setMinSizeY(40);
adserverView.setMaxSizeX(320);
adserverView.setMaxSizeY(50);
adserverView.setBackgroundColor("33CCFF");
adserverView.setTextColor("11CC22");
adserverView.setInternalBrowser(false);
adserverView.update();                    // fetch initial ad content
linearLayout.addView(adserverView);       // insert ad view into layout
```

2. Template creation, is created in XML layout; for example:

```
<com.adserver.adview.MASTAdView
      android:id="@+id/adViewer1"
      android:layout_width="fill_parent"
      android:layout_height="50dip"
      site="5441"
      zone="6365"
      isTestModeEnabled="false"
      defaultImage="@drawable/test_banner"
/>
```

To show View of interstitial advertising, create view as above and invoke the show() method after calling update():

```
// create or obtain reference to view object…
adserverView.show();
```

-- END OF DOCUMENT --