Open Measurement SDK Native-App

Onboarding Guide for Integration Partners

Version 1.4 | September 2022

Native-App Onboarding Guide

Executive Summary

The Open Measurement Software Development Kit (OM SDK) is designed to facilitate third party viewability and verification measurement for ads served to native app and web video environments without requiring multiple Ad Verification Service Providers (Measurement Provider) Software Development Kits (SDKs). This guide will focus on native-app integrations, please see the Web Video Integration Guide for those onboarding details.

The OM SDK consists of a native library for iOS & Android operating systems (OS), as well as a JavaScript API, named Open Measurement Interface Definition (OMID), to communicate with tags from measurement providers in a standard fashion.

The native library collects various signals, such as ad container geometry and playback events in the case of video, via direct communication with the mobile device's native OS and video players, and publishes these signals via the OMID JavaScript API. The OM SDK includes support for native and HTML display banner and interstitial ad types as well as native and HTML5 video players.

OM SDK is developed and managed by the Open Measurement Commit Group (OMCG)

Audience

Though this specification offers guidance for Ad SDK developers & App publishers (Integration Partners), anyone working in advertising operations for nativeapp ad campaigns should be familiar with this document e.g. DSP, Ad Operations and Ad delivery systems developers.

About IAB Tech Lab

The IAB Technology Laboratory is a nonprofit research and development consortium charged with producing and helping companies implement global industry technical standards and solutions. The goal of the Tech Lab is to reduce friction associated with the digital advertising and marketing supply chain while contributing to the safe growth of an industry.

The IAB Tech Lab spearheads the development of technical standards, creates and maintains a code library to assist in rapid, cost-effective implementation of IAB standards, and establishes a test platform for companies to evaluate the compatibility of their technology solutions with IAB standards, which for 18 years have been the foundation for interoperability and profitable growth in the digital advertising supply chain. Further details about the IAB Technology Lab can be found at https://iabtechlab.com.

Open Measurement Working Group

Commit Group Members

DoubleVerify	HUMAN
Google	Nielsen
Integral Ad Science	Oracle Advertising and Customer Experience
IAB Tech Lab	Pandora

Working Group Members

A full list of working group members may be found here:

https://iabtechlab.com/working-groups/open-measurement-working-group/

Change Log

Revision	Description	Author	Date
Version 1.0	Document created, First draft for Om SDK GA Release	Shailley Singh (shailley@iabtechlab.com)	4/10/2018
Version 1.1	Modified Campaign execution guidance for Programmatic and Native ads	Shailley Singh (shailley@iabtechlab.com)	8/24/2018
Version 1.2	Modified Campaign execution for OpenRTB guidance	Shailley Singh (shailley@iabtechlab.com)	10/4/2018
Version 1.3	Added Audio campaigns for version 1.3 release	Shailley Singh (shailley@iabtechlab.com)	12/18/2019
Version 1.4	Updated to include all native-app integrations, including first release of CTV	Jill Wittkopp (jill@iabtechlab.com)	06/27/2022

Table of Contents

Introduction

Scope

This document explains the setup, initiation, functions, properties, events, and expected behaviors for OM SDK native-app integration. However, some operational details are excluded. Examples of features that are out-of-scope for this document include:

- Retrieving the ad from Ad Server, Ad Network, or local resources
- Reporting
- Logging
- Billing and payments
- Ad dimensions and ad behavior

The OM SDK is designed to facilitate third party viewability and verification measurement for ads served to native-app environments without requiring multiple verification vendor SDKs. The collection, processing, analysis, and reporting of information surfaced by the OM SDK is the responsibility of the vendor and requires the vendor to collect additional information with their respective JavaScript tag that is served with the ad creative. Logic for other types of measurement, such as the assessment of brand safety, IVT, audience, etc. may not be explicitly included within the OM SDK native code and may be implemented by measurement vendors using JavaScript alone when it is possible and applicable to do so.

The long-term goal of the Open Measurement Working Group (OMWG) is to create a standardized technology framework that facilitates measurement of 3MS Solutions without requiring multiple Measurement Provider SDKs. As new measurement methods evolve, the Commit Group will evaluate, prioritize, and schedule support for them in the OM SDK accordingly.

How OMID Works

Included below is a high-level overview of the technical operations for an OMID measurement session. For more detailed documentation, please see the https://iabtechlab.com/standards/open-measurement-sdk/
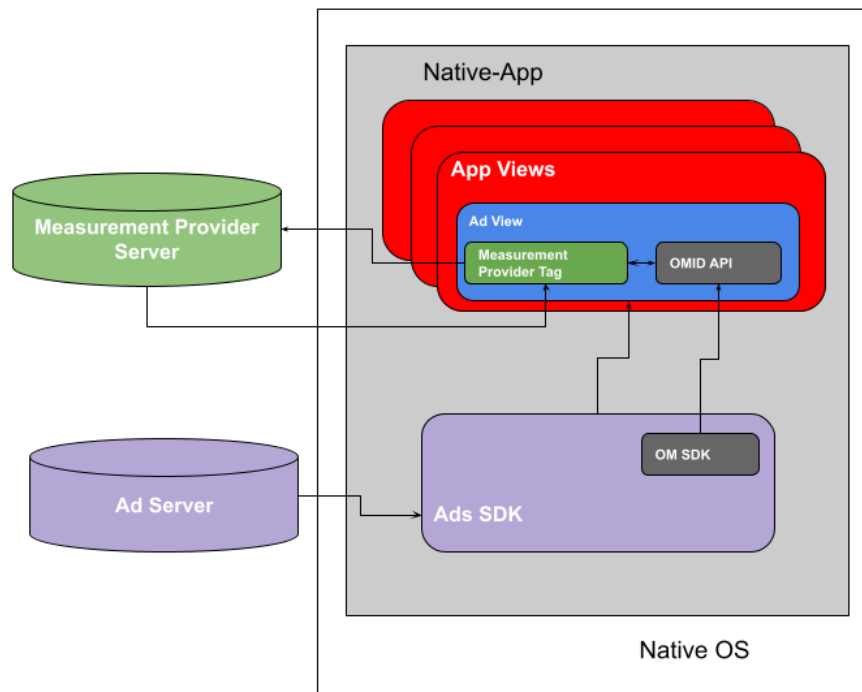
Key Points:

- Measurement Provider's tag is trafficked with the ad creative.
- Ad SDK initiates the OM SDK which triggers the OM SDK Javascript or OMID API.
- Measurement Vendor tag listens to events in OMID API and sends the data to its servers.

Governance Principles

To ensure that this primary purpose is met, the following governance has been established:

- **Object form Integration**: Integration Partners will integrate and maintain OMWG approved OM SDK versions released through the IAB Tech Lab portal. This will ensure a valid and tested version is being used.

- **Integration Validation Compliance Certification:** The IAB Tech Lab has designed integration validation compliance guidance for Integration Partners which is available here https://iabtechlab.com/standards/open-measurement-sdk/ . The certification is voluntary and serves to ensure that your integration is compatible with Measurement Provider measurement tags. More details are available at compliance@iabtechlab.com.

- **Distribution**: The OMWG will manage the release and distribution of OM SDK versions to ensure Measurement Provider tags and Integration Partner implementations are compatible.

- **Measurement Provider Standardization**: Measurement Providers will migrate Integration Partners from their proprietary SDKs to the OM SDK to eliminate technical redundancies and bolster scale of measurement.

Timeline Guidance

The OM SDK is an industry effort through the IAB Tech Lab to standardize the collection of viewability and measurement data in native-apps. Voluntary adoption of the OM SDK will help streamline collection and reporting of measurement data for buyers and provide publishers with the opportunity to integrate one SDK to support multiple buyers. Wide adoption will also help build buyer confidence in measurement data, make Measurement Provider services more efficient and help industry reap the benefits of universal SDK.

With this in mind, if you are using or planning to use an SDK from a Measurement Provider or any other method or API for viewability measurement, we suggest you should consider transitioning to Open Measurement SDK in supported environments.

We recommend that you integrate the Open Measurement SDK at the earliest opportunity with the most recent releases being AndroidTV and tvOS by July 2023.

Source Code

Members of the OMWG are able to review the OM SDK source code in private GitHub repositories. At the discretion of the OMWG Commit Group, certain components of the OM SDK source code may remain private to protect the integrity of the project, validate in-market integrations, and prevent leakage of certain collection methodologies. Further, while the repositories are available to the OMWG members for development and testing purpose, Integration Partners should integrate the OM SDK using the object form of the code made available through the Tech Lab portal. This will ensure the testing and certification process is able to execute and help ensure consistency.

Versions

- OM SDK is designed to ensure interoperability and backwards compatibility
- Semantic versioning will be followed to align JS and native code library releases
- All minor versions of a major OM SDK version will be backward compatible
- Major version change may introduce breaking changes

Onboarding Process

Overview

This section outlines the process for Integration Partners to register for downloading the OM SDK and ongoing releases as well as the process to follow for completing the integration and testing. Please review this mapping guide(s) to understand which SDK to use for which target environment.

- https://iabtechlab.com/standards/open-measurement-sdk/native-app-compatibility-mapping
- https://iabtechlab.com/standards/open-measurement-sdk/web-video-compatibility-mapping

Downloading the SDK

The OM SDK can be downloaded from the IAB Tech Lab tools portal. Please follow the below steps to download the SDK:

1. Register on IAB Tech Lab tools portal here: https://tools.iabtechlab.com/ using your company email domain. *Emails from free email services like google, yahoo, hotmail or others will not be accepted*

2. Request access to OM SDK by clicking on "OM SDK" option on the tools dashboard

Dashboard → Get OM SDK
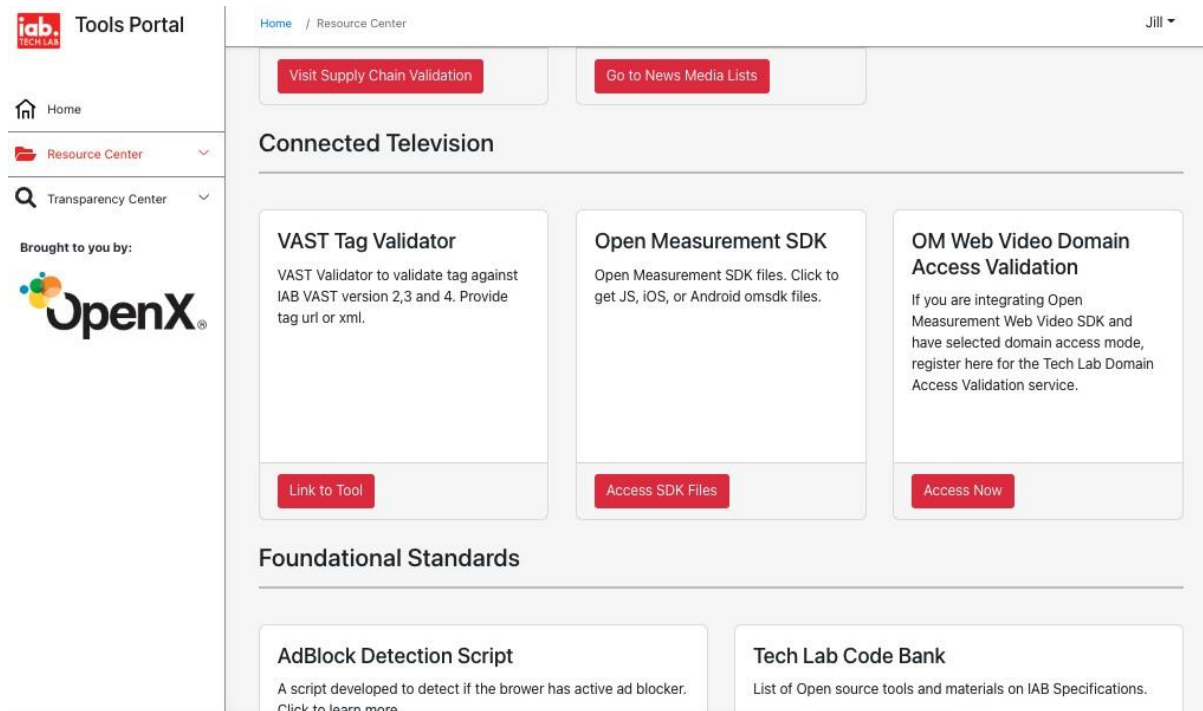
Dashboard→ Get OM SDK → OM SDK Home
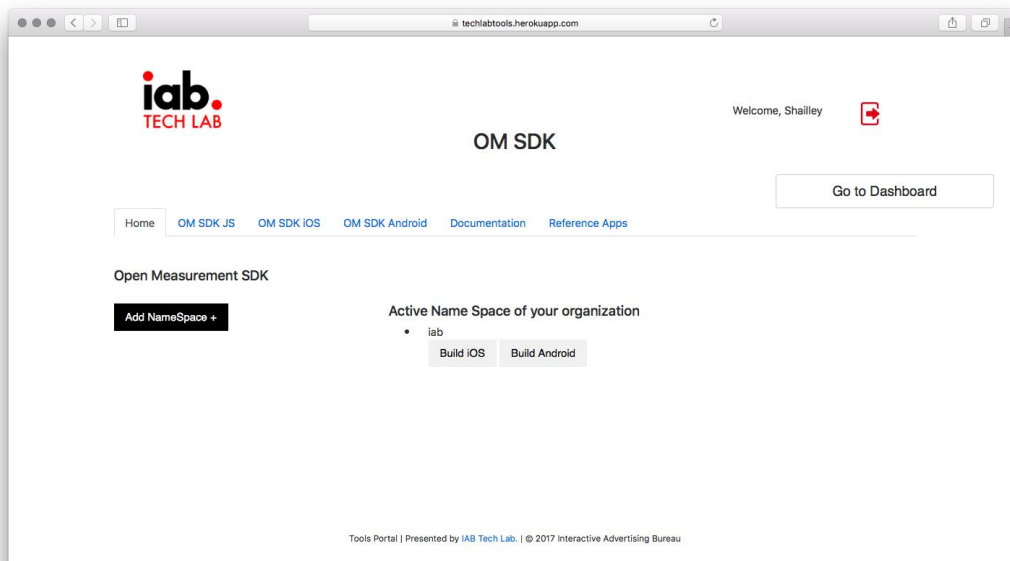
3. Generate a build by clicking on "build" button for iOS and/ or Android versions listed under

the specific namespace. CTV builds for tvOS and AndroidTV are included in 1.4 builds and after. As default an initial namespace with your company domain will be created.

Generating the build
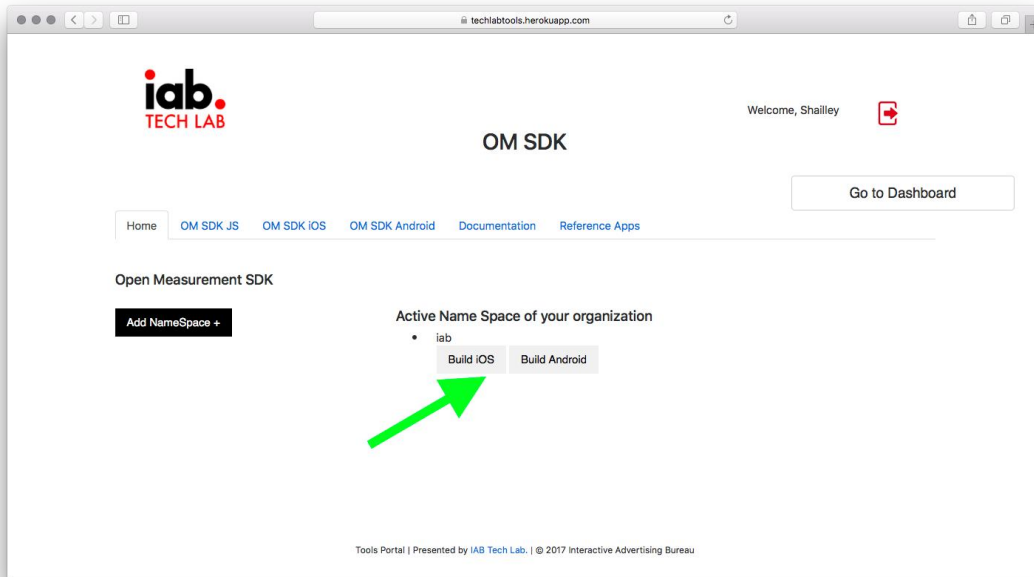
4. Additional namespaces can be created by clicking on "Add Namespace" button. Please review the unique namespace guidance in this guide before adding namespaces.

Adding new namespace
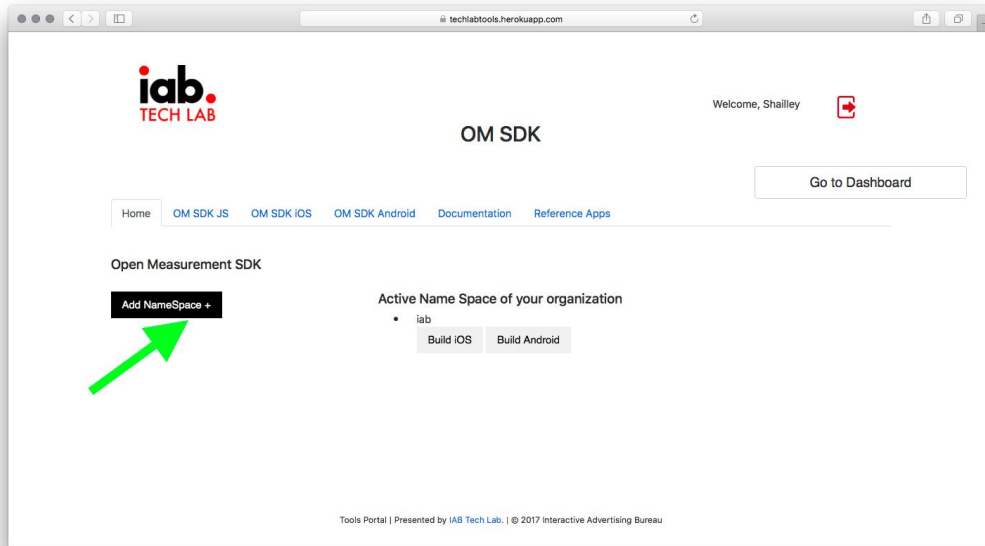
Additional namespace created
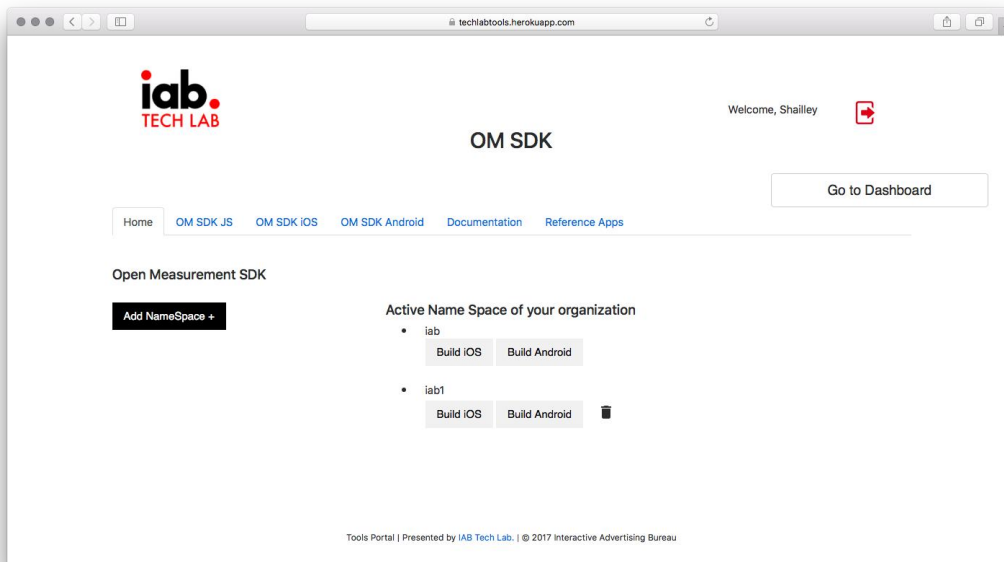
5. User will be notified when the build is complete and available.

6. Download the files for integration: Native libraries for iOS or Android and OMSDK JS
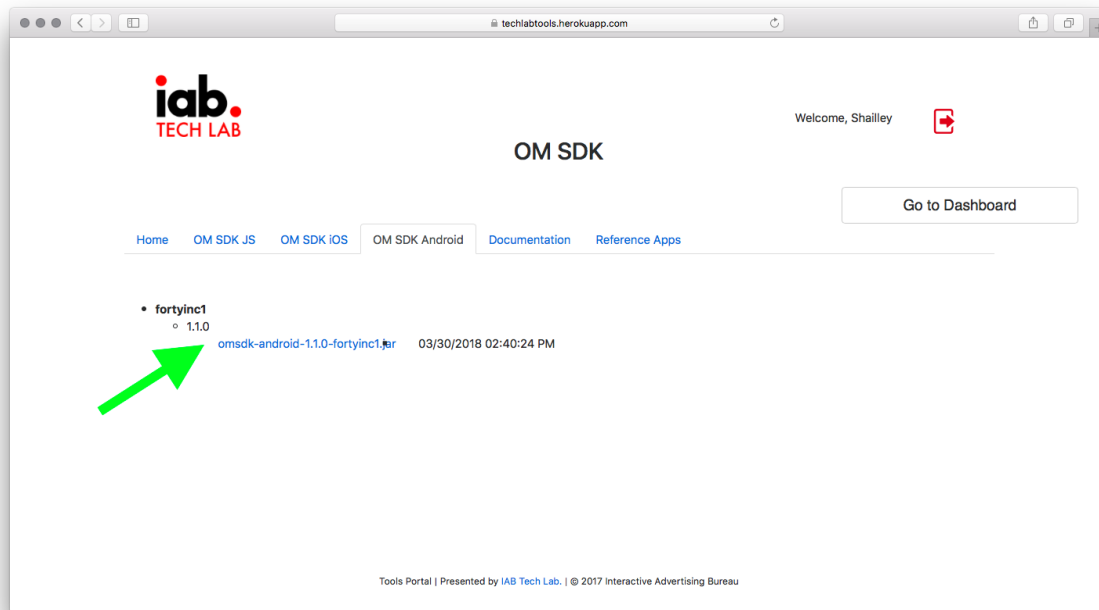
7. Other documentation and guidance for integration is also available through this UI in the Documentation tab.

The build will always be generated from the latest available release of OM SDK. Future releases will be made available through the portal and users will be notified that a new release is available for them to build.

Unique Namespace

The OM SDK is generated for each Integration Partner with their unique namespace. To maintain unique namespaces, Tech Lab generates the namespace based on the domain name of the company. Additional namespace can be created by users, if required for following use cases:

1. An integration Partner has multiple SDKs that require a unique namespace
2. An integration Partner white labels their SDK to other SDKs or app developers and wants to track each integration with unique references

Please avoid generating unique namespaces for testing or development purposes. The Namespace is also used to track compliance certification.

When not to generate a new namespace:

1. For testing purpose

2. Using separate namespace for iOS and Android versions of the same SDK or App

Integrating and testing the SDK

After downloading the SDK, follow the Developer Integration Guide to integrate the SDK:

- iOS Guide : https://iabtechlab.com/omsdk/docs/devguideiOS

- Android Guide: https://iabtechlab.com/omsdk/docs/devguideAndroid

Follow below steps to test:

1. Add Verification Validation Script

    (`omid-validation-verification-script-v1.js`) to creatives to test OMID

    integration. It is part of OM SDK JS libraries

    a. For webview ads, add it as part of the HTML

    b. For Video ads, host it and provide the url where javascript resource or URL is

       expected. Review Video Creatives section in this document for more details

    c. For native ads host the file and provide it in JavascriptResourceURL. Review

       Native delivery section in this document

2. Serve the creatives to test OMID integration

3. Test the integration using Integration Validation Test cases here:

    https://omsdk-demo-files.s3-us-west-2.amazonaws.com/docs/TestCases.pdf

4. The Verification Validation script will send logs of all OMID events that can be captured

    via tools like Charles Proxy

Integration Validation Compliance

Tech Lab provides a service for Integration Partners to certify their OM SDK integrations. This ensures that Measurement Provider tags will work with your OM SDK implementation and assures that the integration is working as expected. More details available here:

https://iabtechlab.com/omsdk/docs/compliance

Measurement Provider Testing

Integration Partners can perform this additional step of testing Measurement Provider tags. Follow below steps to complete this testing:

1. Requesting Measurement Providers for test tags
2. Serve creatives with Measurement Provider tags
3. Perform the Integration Validation Test Cases
4. Compare the data from Measurement Provider portal

OM SDK JS Weekly Updates

OM SDK Js is a critical component that enables communication between OM SDK native libraries and OMID API

1. As the OMCG releases new version, OM SDK JS will be updated on a regular basis
2. It is recommended that OM SDK JS be updated from the Tech Lab portal at least every other week.

Onboarding Workflow and Checklist

Step	Integration Partner	Comments/ Milestones
1. Tech Lab Registration	Sign up on IAB Tech Lab tools portal here: https://tools.iabtechlab.com/	User should get their namespace based on company email domain and be able to view links to build their Android and iOS SDK
2. Create your SDK build	Click on "Build" button to generate the SDK with the namespace	User is notified when the build generation is completed
3. Create additional namespace	Create additional namespace by clicking on "Add namespace" button	Namespaces should be created
4. Download SDK	Download SDK libraries and developer integration guide	

5. Integration	Integrate SDKs & Host OM SDK JS	Completed SDK integration
6. Testing	Integration testing with validation tools and any internal processes	Integration Partner signs off on SDK integration.
7. Validation of integration	Submit integration build (Ad SDK or Test App) to IAB Tech Lab	Commit Group signs off on SDK integration.
8. Measurement Provider Testing	Release integration, certify with measurement provider(s) of choice if desired	GA release of SDK integration.
9. OM SDK JS updates	Checks and updates for latest OMID JS at least once a week and preferably more frequently. The latest JS files will be available through the portal login for manual download as well as a URL with API key for automated download	New version of OMID JS released

Campaign Execution

Overview

Advertisers contract third party Measurement Providers to provide independent measurement and verification services on their campaigns. Advertisers use this viewability and verification measurement to optimize their campaigns and better ensure ads are delivered to valid environments where consumers have valid opportunities to interact with them.

In order to provide these measurement services, Measurement Providers operate proprietary JavaScript tag solutions that are trafficked with the ad creative. These JavaScript tags interact with native-app environments at the time of ad delivery, enabling the collection of signals surfaced by the OM SDK, as well as other signals available in the ad environment.

HTML Display Creatives

When measuring HTML display ad impressions, Measurement Provider tags are delivered in a script block along with the ad creative.

Native Display Creatives

Ad servers are expected to provide OMID with following parameters for each Measurement Provider tag contracted for a given campaign:

1. Measurement provider vendor key to identify the vendor
2. Measurement Provider verification resource URL for the verification tag
3. Verification parameters string

Webview or Native delivery is not known at trafficking time

Where trafficking workflows are not distinguished between webview and native in native-app delivery at the time of trafficking the creative, it is recommended that integrators work with measurement providers, to determine a suitable parsing method to create the necessary input changes to convert a measurement provider's tag for webview delivery into native-app delivery,

where such measurement is supported with a unified tagging structure by the measurement provider.

Native delivery is known at trafficking time

When native-app delivery is known or OpenRTB Dynamic Native Ads API is being used for request and response, please refer to Native Bid Response in OpenRTB

Also check the following advisory for updates in supporting OMDSK with OpenRTB 3 implementations:

https://github.com/InteractiveAdvertisingBureau/AdCOM/blob/master/OpenRTB%20support%20for%20OMSDK.md

Video Creatives

VAST

For VAST video, OMWG recommends using 'AdVerifications' node as described in VAST 4.1. For the purpose of measuring video ads in OM SDK, two options have been defined in consultation with IAB Tech Lab's "Digital Video Technical Standards Working Group"

Since OM SDK supports methods to deploy Measurement Provider Verification code for measurement, VPAID method for verification is not recommended.

For VAST 4.1 version, verification code should be loaded in AdVerifications node as per VAST version 4.1 guidance

Verification node Example in VAST 4.1

```
<AdVerifications>
     <Verification vendor="company.com-omid">
       <JavaScriptResource apiFramework="omid" browserOptional="true">
                <![CDATA[https://verification.com/omid_verification.js]]>
       </JavaScriptResource>
      <TrackingEvents>
        <Tracking event="verificationNotExecuted">
          <![CDATA[https://verification.com/trackingur/[REASON]l]]>
            </Tracking>
     </TrackingEvents>
          <VerificationParameters
```

```
            <![CDATA[verification params key value pairs]]>
          </VerificationParameters>
       </Verification>
   </AdVerifications>
```

For older (e.g. 2.0, 3.0, 4.0) version VAST documents, verification code should be loaded via Extension node (extension type as 'AdVerifications') where the root is an AdVerifications node with the same schema as the VAST 4.1 element.

Verification node example in VAST pre-Version 4.1

```
...
<Extensions>
  <Extension type="AdVerifications">
    <AdVerifications>
      <Verification vendor="company.com-omid">
        <JavaScriptResource apiFramework="omid" browserOptional="true">
                <![CDATA[https://verification.com/omid_verification.js]]>
        </JavaScriptResource>
          <TrackingEvents>
         <Tracking event="verificationNotExecuted">
            <![CDATA[https://verification.com/trackingurl]]>
             </Tracking>
      </TrackingEvents>
      <VerificationParameters>
             <![CDATA[verification params key value pairs]]>
        </VerificationParameters>
      </Verification>
    </AdVerifications>
  </Extension>
</Extensions>
...
```

For pre VAST 4.1 versions, please follow the same rules as VAST 4.1 for handling the parameters in the 'AdVerifications' node

VAST example is available here:
https://github.com/InteractiveAdvertisingBureau/VAST_Samples/blob/master/VAST%201-2.0%20Samples/Inline_NonLinear_Verification_VAST2.0.xml

Non VAST Video

Although IAB Tech Lab recommends using VAST for Video ad serving, it is possible that an ad server accepts video files or video ad assets directly and serves them in their apps without the use of VAST. In such cases the ad server using OM SDK for verification purpose must request the advertiser for the following details that need to be passed to the OM SDK on initiation:

4. Measurement provider vendor key to identify the vendor
5. Measurement Provider verification resource URL for the verification tag
6. Verification parameters

The Integration Partner must then provide the above details to OMID after starting the ad session.

These can be communicated to Integration Partner in standard JSON format as below or a predetermined method between integration partner and advertiser.

Suggested JSON format for communicating OM SDK verification details

```
[
  {
   "vendorKey": "company.com-omid",
   "javascriptResourceUrl": "https://verification.com/omid_verification.js",
   "verificationParameters": "[parameters string]"
  },
]
```

Audio Creatives

VAST

For VAST audio, OMWG recommends using 'AdVerifications' node as described in VAST 4.x.

For VAST 4.1 and higher versions, verification code should be loaded in AdVerifications node as per VAST version guidance

Verification node Example in VAST 4.1 or higher

```
<AdVerifications>
      <Verification vendor="company.com-omid">
       <JavaScriptResource apiFramework="omid" browserOptional="true">
                <![CDATA[https://verification.com/omid_verification.js]]>
       </JavaScriptResource>
      <TrackingEvents>
        <Tracking event="verificationNotExecuted">
          <![CDATA[https://verification.com/trackingur/[REASON]l]]>
            </Tracking>
      </TrackingEvents>
          <VerificationParameters>
          <![CDATA[verification params key value pairs]]>
         </VerificationParameters>
      </Verification>
  </AdVerifications>
```

DAAST

OM SDK does NOT support DAAST. Please use VAST 4.1 or higher for enabling audio ads with OM SDK.

Non VAST Audio

Although IAB Tech Lab recommends using VAST for audio ad serving, it is possible that an ad server accepts audio files or audio ad assets directly and serves them in their apps without the use of VAST. In such cases the ad server using OM SDK for verification purpose must request the advertiser for the following details that need to be passed to the OM SDK on initiation:

1. Measurement provider vendor key to identify the vendor
2. Measurement Provider verification resource URL for the verification tag
3. Verification parameters

The Integration Partner must then provide the above details to OMID after starting the ad session.

These can be communicated to Integration Partner in standard JSON format as below or a predetermined method between integration partner and advertiser.

Suggested JSON format for communicating OM SDK verification details

```
[
  {
   "vendorKey": "company.com-omid",
   "javascriptResourceUrl": "https://verification.com/omid_verification.js",
   "verificationParameters": "[parameters string]"
  },
]
```

OpenRTB Guidance

The following guidance is the OpenRTB working group advisory on supporting OM SDK to indicate availability of OM SDK for ad verification.

Also check the following advisory for updates

https://github.com/InteractiveAdvertisingBureau/AdCOM/blob/master/OpenRTB%20support%20for%20OMSDK.md

OpenRTB 2.6 allows for communication of OMSDK support:

1. If OMID is available for a given impression in a bid request
2. If the ad being used in the bid response requires OMID
3. The "partner name" assigned to Publishers who have installed the OMSDK

Please use the API Frameworks list from AdCOM 1.0: https://github.com/InteractiveAdvertisingBureau/AdCOM/blob/master/AdCOM%20v1.0%20FINAL.md#list_apiframeworks

OpenRTB 2.0 through 2.5 - OpenRTB's list of API Frameworks should be extended with a value for OMID. Extend List 5.6 API Frameworks as follows:

List 5.6 API Frameworks

Value	Description
1	VPAID 1.0
2	VPAID 2.0
3	MRAID-1
4	ORMMA

5	MRAID-2
6	MRAID-3
7	OMID-1

In OpenRTB 2.0 and 2.1, value 6 is not present. If extending an existing OpenRTB 2.0 or 2.1 implementation, it is suggested that value 6 is skipped and 7 is used to indicate OMID support to preserve an upgrade path.

Bid Request

The "api/apis" field of the Banner, Video, or Native Object should be used to indicate availability of OMID. The value is apis in OpenRTB 2.6 and api in prior versions.

Banner/Video/Native/Audio Object

In OpenRTB 2.6, use:

Attribute	Type	Description
apis	integer array	List of supported APIs for the markup. If an API is not explicitly listed, it is assumed to be unsupported. Refer to List: API Frameworks in AdCOM 1.0.

Prior to OpenRTB 2.6 use:

Attribute	Type	Description
api	integer	List of supported API Frameworks for this impression. Refer to List 5.6. If an API is not explicitly listed, it is assumed not to be supported.

Bid Response

The "api/apis" field of the Bid Object should be used to indicate that the ad being used in the bid requires use of OMID. This field is supported in OpenRTB 2.4+, though consistent with the design principles of OpenRTB, it should be possible to retrofit this field into OpenRTB 2.3 and earlier versions as SSPs and DSPs should gracefully ignore fields they do not understand.

Since pre-OpenRTB 2.6 versions only support an integer (not an array) for the api field in the response, an extension field should be used to convey if more than one API type is to be used (e.g. MRAID 2.0 and OMID-1). Recommendation is that SSP and DSPs who support that should be indifferent to the seatbid.bid.api field vs. seatbid.bid.ext.apis, listening for OMID-1 listed in either field.

Possible ext field in the Bid Object of the response in OpenRTB 2.x: seatbid.bid.ext.apis

Attribute	Type	Description
apis	integer array	(optional) APIs required by the markup, if applicable. Refer to List 5.6. Should be used if the markup requires more than one API, in place of current "api" field.

OpenRTB and AdCOM

Bid requests will include two new fields to identify the partner integration. OpenRTB 2.x and 3.0 will carry these fields in different objects, but the field names and types are the same in both protocols.

Attribute	Type	Description

omidpn	string	Identifier of the OM SDK integration. This is the same as the "name" parameter of the OMID Partner object.
omidpv	string	(optional) Version of the OM SDK integration. This is the same as the "versionString" parameter of the OMID Partner object.

Note that these fields are in addition to the API frameworks array, which is carried in an "api" field in OpenRTB 2.5 and earlier, and an 'apis' field in OpenRTB 2.6 & AdCOM. The API frameworks array must include the value 7 to indicate OMID 1.0 support.

OpenRTB 2.x

The two fields will be added to an extension object within the bid request. There are several options for the extension object to use. This section lists three possibilities.

BidRequest.Source.Ext

The object BidRequest.Source describes the source of the OpenRTB 2.x bid request. Adding "omidpn" and 'omidpv' to Source.Ext would be appropriate if certification applies to all placements in the bid request.

BidRequest.Source.Ext example

```
"source" {
  "ext": {
    "omidpn": "MyIntegrationPartner",
    "omidpv": "7.1"
  }
},
"imp" [{
  "banner": {
    "api": [7]
  }
}]
```

OpenRTB 3.0 and AdCOM 1.0 (in Beta)

The new AdCOM spec represents ad request data in OpenRTB 3.0. Unlike OpenRTB 2.x, the "omidpn" and 'omidpv' fields will be extensions in the 'events' object defined in the protocol.

Event Trackers

The OMID capability of a bid request is encoded in an "events" object, making it a logical place for the partner information.

BidRequest.Source.Ext example

<pre>"events" {
  "apis": [7]
  "ext": {
          "omidpn": "MyIntegrationPartner",
          "omidpv": "7.1"
}</pre>

Native Bid Response in OpenRTB

Native ads that use OMID in OpenRTB 2.x require the treatment below:

This should be used in "Event Trackers Request Object" (section 4.7 of OpenRTB Dynamic Native Ads API Specification Version 1.2:

https://www.iab.com/wp-content/uploads/2018/03/OpenRTB-Native-Ads-Specification-Final-1.2.pdf) in "ext" field as custom JSON.

- *Event* should be set to a custom value of "555" to signify that this is OMID and the SSP and publisher will determine when to execute OMID script.
- *Method* should always be "2" for JS. Previous guidance to use '555' is not necessary unless the script is non-JS, non-img.
- The javascript *URL* should be inserted into the standard eventrackers.url field

- OMID event trackers can include two extension values, which SSP/publisher must pass to OMID when provided:
  - "vendorKey"
  - "verification_parameters"

Suggested values to use for "Event Trackers Request Object" are:

Object	Value	Name
event	555	OMID- new custom value
methods	2	JS- existing value

Sample Bid Response

```
"eventtrackers" {
  "event": 555  // New, custom value to signify OMID
   "method": 2   // Previous guidance recommended a value of '555' but '2' is
preferred for Javascript
  "url": "https://verification.com/omid_verification.js"  // This is where DSPs add
their actual Javascript URL for measurement
  "ext": {     // Custom extensions that any exchange can support; these extensions
are strongly recommended for SSPs and DSPs to support as part of their OMID support

    "vendorKey":  // The vendor key for the company providing the OMID javascript,
e.g. "company.com-omid"
    "verification_parameters":  //Additional verification parameters for the OMSDK,
populated by e.g,. "[parameters string]"
      }

}
```

Resources

- [OMID v1 API Specification](#)

- [OMID v1.2 API Specification](#) (includes OMID for WEB enhancements)

- [Implementation Guide for OMID 1.2](#) for web browsers

- [IAB Tech Lab portal](#)

- [Validation of Integration Test cases](#)

- [Integration Validation Compliance Guide](#)

- [iOS Developer Integration Guide](#)

- [Android Developer Integration Guide](#)

- [VAST example](#)

- [Expected implementation for OpenRTB 3.0](#)

Support

If you need support to integrate OM SDK in your app or Ad SDK or have any questions about how to use OM SDK, please send an email to

omsdksupport@iabtechlab.com.

We will respond within 24-72 hours of receiving the request.