EUROPE'S LEADING AEM DEVELOPER CONFERENCE

27th – 29th SEPTEMBER 2021

Collect Data for the Adobe Experience Platform with the Adobe Mobile SDK

Lukas Greiter, diva-e

Your host

- Lukas Greiter
- Adobe Solutions Developer
- Since 2021 at diva-e
- Profession: Data Collection, Analytics & AEP

Agenda

1. Overview of Adobe Experience Platform & Mobile SDK
2. Implementation process of the Mobile SDK (Android)
3. Testing & Debugging with Project Griffon
4. Data collection in the AEP
5. Prospects

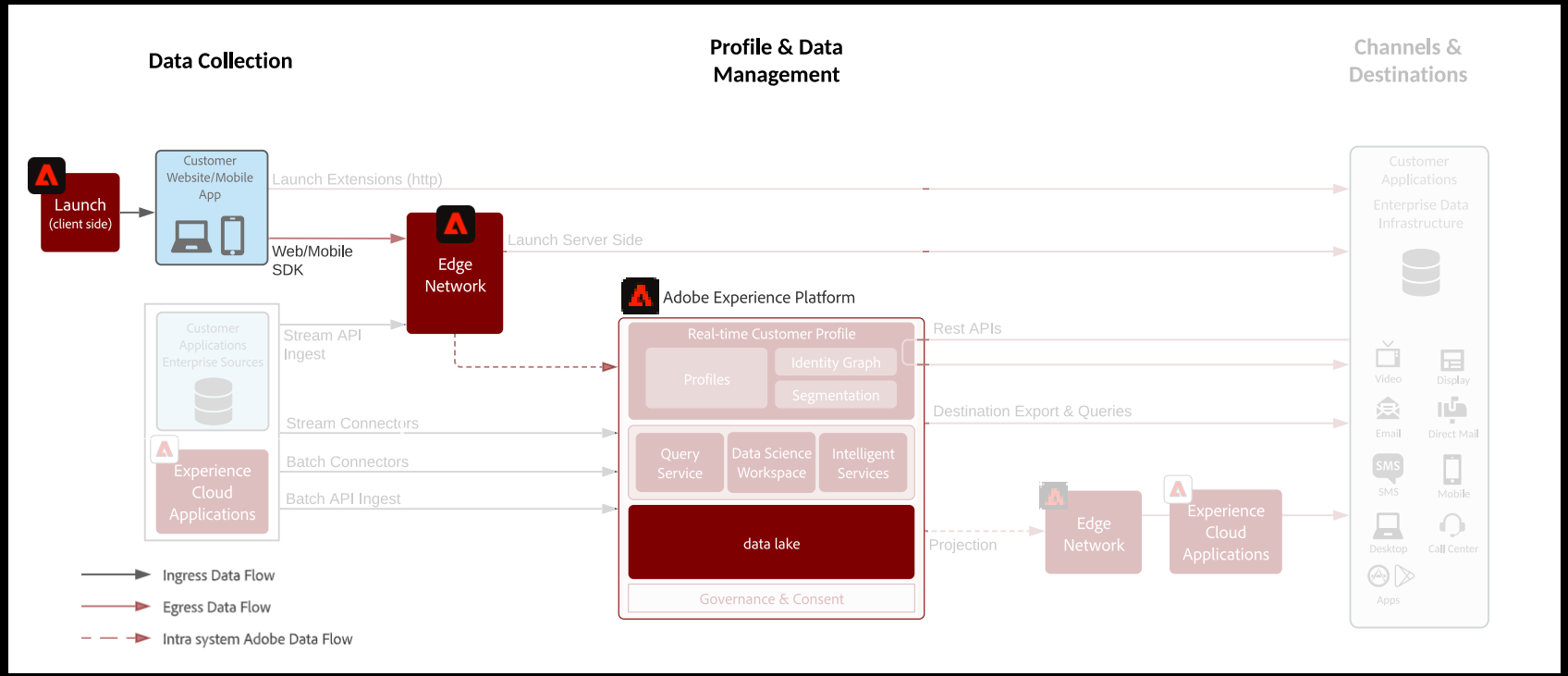
adaptTo()

1. Overview of Adobe Experience Platform & Mobile SDK

Overview of AEP Mobile SDK

- Adobe’s up-to-date solution to utilize the Adobe Experience Cloud in connection with mobile applications
- Successor of Mobile Services 4x SDKs
- Sourcing and handling of the SDK via Adobe Launch
- Designed to leverage the possibilities of the new Adobe Experience Platform

Overview of Adobe Experience Platform



adaptTo()

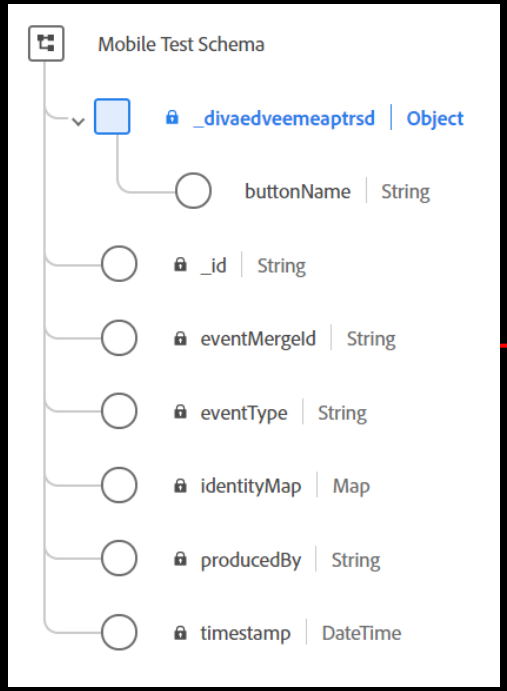
2. Implementation process of the Mobile SDK (Android)

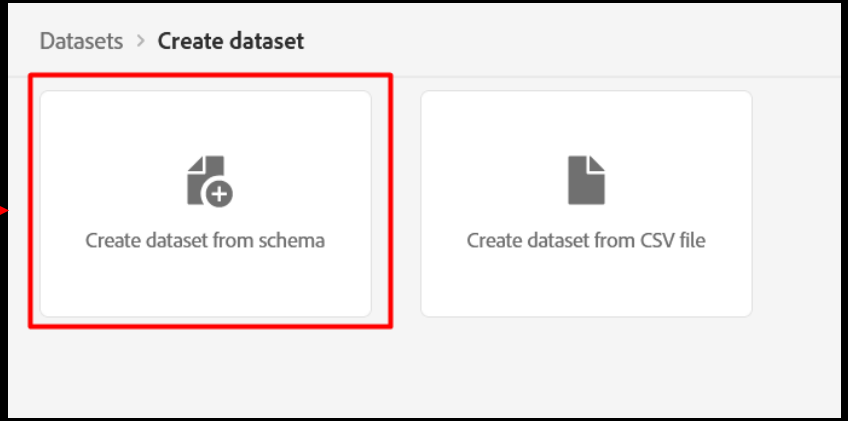
Tools & Requirements

- Android Studio (Gradle as dependency manager)
- Android App (Bus Booking app as example)
- Access to AEP & Adobe Launch
- Access to Adobe‘s Project Griffon

Preparations

Create a Schema and a Dataset in the AEP





Preparations

Set up a new mobile property in Adobe Launch

- MobilePrivacyStatus.OPT_IN
- MobilePrivacyStatus.OPT_OUT
- MobilePrivacyStatus.UNKNOWN

```
MobileCore.setPrivacyStatus(MobilePrivacyStatus.OPT_OUT);
```

Preparations

Add a Datastream (Edge Configuration)

Adobe Experience Platform Data Collection

RODUCTION Prod (NLD2)

DATA COLLECTION
- Tags
- Datastreams
- Event Forwarding

CUSTOMER
- Identities

DATA MANAGEMENT
- Schemas

Datastreams > **Edit Datastream**

Friendly Name (required)

Mobile App Test Edge Server

Description

Enter optional description

Save Cancel

Preparations

Enable AEP & Dataset in Datastream

Preparations

Add and configure Launch Extensions

Installed Catalog

Search

Adobe Experience Platform Edge Network
Adobe Systems · v1.0.6
The Adobe Experience Platform Edge Network mobile extension allows for sending XDM formatted data to Adobe Experience Platform and Adobe Experience Cloud solutions, by leveraging Experience Edge capabilities.
Configure ···

AEP Assurance
Adobe Systems · v1.0.0
Easily inspect, validate, and debug app data collection and experiences.
Configure ···

Identity
Adobe Systems · v1.0.2
The Identity mobile extension enables handling of user identity data from a mobile app when using the Adobe Experience Platform Mobile SDK and the Edge Network extension.
Preparations
Configure ···

Mobile Core
Adobe Systems · v2.1.16
Allows configuration of the Mobile SDK and provides access to default lifecycle events and conditions.
Configure ···

Profile
Adobe Systems · v2.0.1
Allows the SDK to store data into a client-side profile.
Configure ···

Implementation in the app

Get code snippets to be ingested into the app

PROPERTY
Property Overview
AUTHORING
Rules
Data Elements
Extensions
PUBLISHING
Publishing Flow
Environments
Hosts
MONITORING
Audit Events

Tag Properties > Mobile SDK Configuration (TESTING diva-e)

All | test | Build

Search

Add Environment

NAME	ENVIRONMENT	CREATED	LAST MODIFIED	INSTALL
Development	Development	May 18, 2021, 10:56 AM	May 20, 2021, 2:09 PM	
Staging	Staging	May 18, 2021, 10:56 AM	May 18, 2021, 10:56 AM	
Production	Production	May 18, 2021, 10:56 AM	May 18, 2021, 10:58 AM	

Mobile Install Instructions

Environment	Environment File ID
Development	072780f565a4/167f0783e372/launch-[redacted]development

Android | iOS

We'll use Gradle to manage dependencies.

Add the dependency to build.gradle:

```
implementation 'com.adobe.marketing.mobile:edgeidentity:1.+'
implementation 'com.adobe.marketing.mobile:assurance:1.+'
implementation 'com.adobe.marketing.mobile:edge:1.+'
implementation 'com.adobe.marketing.mobile:sdk-core:1.+'
implementation 'com.adobe.marketing.mobile:userprofile:1.+'
```

Add Initialization Code

```
import com.adobe.marketing.mobile.AdobeCallback;
import com.adobe.marketing.mobile.Assurance;
import com.adobe.marketing.mobile.Edge;
import com.adobe.marketing.mobile.Identity;
import com.adobe.marketing.mobile.InvalidInitException;
import com.adobe.marketing.mobile.Lifecycle;
import com.adobe.marketing.mobile.LoggingMode;
import com.adobe.marketing.mobile.MobileCore;
import com.adobe.marketing.mobile.Signal;
import com.adobe.marketing.mobile.UserProfile;
import com.adobe.marketing.mobile.edge.identity.Identity;
...
import android.app.Application;
...
public class MainApp extends Application {
```

Implementation in the app

Add adobe mobile dependencies to *build.gradle*

Mobile Install Instructions

Environment	Environment File ID
Development	072780f565a4/167f0783e372/launch-[redacted]-development

Android iOS

We'll use Gradle to manage dependencies.

Add the dependency to build.gradle:

```
implementation 'com.adobe.marketing.mobile:edgeidentity:1.+'
implementation 'com.adobe.marketing.mobile:assurance:1.+'
implementation 'com.adobe.marketing.mobile:edge:1.+'
implementation 'com.adobe.marketing.mobile:sdk-core:1.+'
implementation 'com.adobe.marketing.mobile:userprofile:1.+'
```

Add Initialization Code

```
import com.adobe.marketing.mobile.AdobeCallback;
import com.adobe.marketing.mobile.Assurance;
import com.adobe.marketing.mobile.Edge;
import com.adobe.marketing.mobile.Identity;
import com.adobe.marketing.mobile.InvalidInitException;
import com.adobe.marketing.mobile.Lifecycle;
```

build.gradle (:bus) | BusBookingActivity.java | DemoApplication.java | AndroidManifest.xml

You can use the Project Structure dialog to view and edit your project configuration

```
        multiDexEnabled true
        versionCode 1
        versionName "1.0"
    }

}

dependencies {
    implementation fileTree(include: ['*.jar'], dir: 'libs')
    testImplementation 'junit:junit:4.12'
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'com.android.support:multidex:1.0.3'
    implementation 'com.android.support:design:28.0.0'
    implementation 'com.android.support:cardview-v7:28.0.0'
    implementation('commons-beanutils:commons-beanutils-core:1.8.3') {
        exclude group: 'commons-logging', module: 'commons-logging'
    }
    implementation 'com.adobe.marketing.mobile:edgeidentity:1.+'
    implementation 'com.adobe.marketing.mobile:assurance:1.+'
    implementation 'com.adobe.marketing.mobile:edge:1.+'
    implementation 'com.adobe.marketing.mobile:sdk-core:1.+'
    implementation 'com.adobe.marketing.mobile:userprofile:1.+'
}
```

Implementation in the app

Add imports to main app (*DemoApplication.java*)

Mobile Install Instructions

Add Initialization Code

```
import com.adobe.marketing.mobile.AdobeCallback;
import com.adobe.marketing.mobile.Assurance;
import com.adobe.marketing.mobile.Edge;
import com.adobe.marketing.mobile.Identity;
import com.adobe.marketing.mobile.InvalidInitException;
import com.adobe.marketing.mobile.Lifecycle;
import com.adobe.marketing.mobile.LoggingMode;
import com.adobe.marketing.mobile.MobileCore;
import com.adobe.marketing.mobile.Signal;
import com.adobe.marketing.mobile.UserProfile;
import com.adobe.marketing.mobile.edge.identity.Identity;
...
import android.app.Application;
...
public class MainApp extends Application {
```

build.gradle (:bus) | DemoApplication.java | BusBookingActivity.java | And

```
1   /.../
18  package com.adobe.busbooking;
19  import com.adobe.marketing.mobile.*;
20
21  import android.app.Application;
22  import android.content.Context;
23  import android.support.multidex.MultiDex;
24
25  import com.adobe.marketing.mobile.AdobeCallback;
26  import com.adobe.marketing.mobile.Assurance;
27  import com.adobe.marketing.mobile.Edge;
28  import com.adobe.marketing.mobile.Identity;
29  import com.adobe.marketing.mobile.InvalidInitException;
30  import com.adobe.marketing.mobile.Lifecycle;
31  import com.adobe.marketing.mobile.LoggingMode;
32  import com.adobe.marketing.mobile.MobileCore;
33  import com.adobe.marketing.mobile.Signal;
34  import com.adobe.marketing.mobile.UserProfile;
```

Implementation in the app

Add initialization code to main app (*DemoApplication.java*)

Mobile Install Instructions

```
public class MainApp extends Application {
  ...
  @Override
  public void onCreate() {
    super.onCreate();
    MobileCore.setApplication(this);
    MobileCore.setLogLevel(LoggingMode.DEBUG);
    ...
    try {
      com.adobe.marketing.mobile.edge.identity.Identity.registerExtension();
      com.adobe.marketing.mobile.Identity.registerExtension();
      Assurance.registerExtension();
      Edge.registerExtension();
      Lifecycle.registerExtension();
      Signal.registerExtension();
      UserProfile.registerExtension();
      MobileCore.start(new AdobeCallback () {
        @Override
        public void call(Object o) {
          MobileCore.configureWithAppID("[redacted]
        }
      });
    } catch (InvalidInitException e) {
      ...
    }
  }
}
```

build.gradle (:bus) | DemoApplication.java | BusBookingActivity.java | AndroidManifest.xml

```
38
39      @Override
40      public void onCreate() {
41          super.onCreate();
42          MobileCore.setApplication(this);
43          MobileCore.setLogLevel(LoggingMode.DEBUG);
44
45          try{
46              Edge.registerExtension();
47              com.adobe.marketing.mobile.edge.identity.Identity.registerExtension();
48              com.adobe.marketing.mobile.Identity.registerExtension();
49              Assurance.registerExtension();
50              Lifecycle.registerExtension();
51              Signal.registerExtension();
52              UserProfile.registerExtension();
53              MobileCore.start(new AdobeCallback () {
54                  @Override
55                  public void call(Object o) {
56                      MobileCore.configureWithAppID("d9b757a7e654/52710d2a8f19/launch-4
57                  }
58              });
59          } catch (InvalidInitException e){
60              e.printStackTrace();
61          }
62      }
```

Implementation in the app

Add intent-filter to *AndroidManifest.xml*

build.gradle (:bus) | DemoApplication.java | BusBookingActivity.java | AndroidManifest.xml

```
<application
    android:name=".DemoApplication"
    android:allowBackup="true"
    android:icon="@drawable/bus_ic"
    android:label="Bus Demo"
    android:supportsRtl="true"
    android:theme="@style/MaterialTheme"
    tools:replace="android:label">
    <activity android:name=".BusBookingActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>

        <intent-filter>
            <action android:name="android.intent.action.VIEW" />

            <category android:name="android.intent.category.DEFAULT" />
            <category android:name="android.intent.category.BROWSABLE" />
            <data
                android:scheme="http"
                android:host="test.my.app" />
        </intent-filter>
```

Implementation in the app

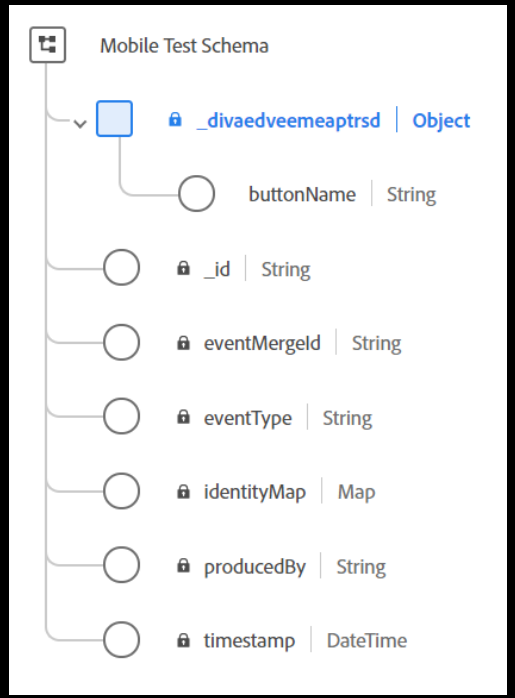
Add xdmData data code to *BusBookingActivity.java* (example: click on *Get Buses*)

```
Map<String, Object> xdmData = new HashMap<String, Object>();
xdmData.put("_tenantId",new HashMap<String, Object>() {{
        put("buttonName", "Get Buses");
}});
xdmData.put("eventType", "click");

ExperienceEvent event = new ExperienceEvent.Builder()
    .setXdmSchema(xdmData, "xdmSchemaId")
    .build();
Edge.sendEvent(event, null);
```

Implementation in the app

Add xdmData data code to *BusBookingActivity.java* (example: click on *Get Buses*)



DemoApplication.java | BusBookingActivity.java | AndroidManifest.xml

0 results

```java
findViewById(R.id.btn_find_buses).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        showConfirmationDialog();

        Map<String, Object> xdmData = new HashMap<String, Object>();
        xdmData.put("_divaedveemeaptrsd",new HashMap<String, Object>() {{
            put("buttonName", "Get Buses");
        }});
        xdmData.put("eventType","click");

        ExperienceEvent event = new ExperienceEvent.Builder()
                .setXdmSchema(xdmData, datasetIdentifier: "614e0645c372511948b81095")
                .build();
        Edge.sendEvent(event, callback: null);

    }
});
```

adaptTo()

3. Testing & Debugging with Project Griffon (Hands-on)

Testing & Debugging with Project Griffon (Hands-on)

- What is Project Griffon and why is it useful
- Connecting to Project Griffon
- Explaining & showing where to find the debugging information

4. Data Collection in the AEP (Hands-on)

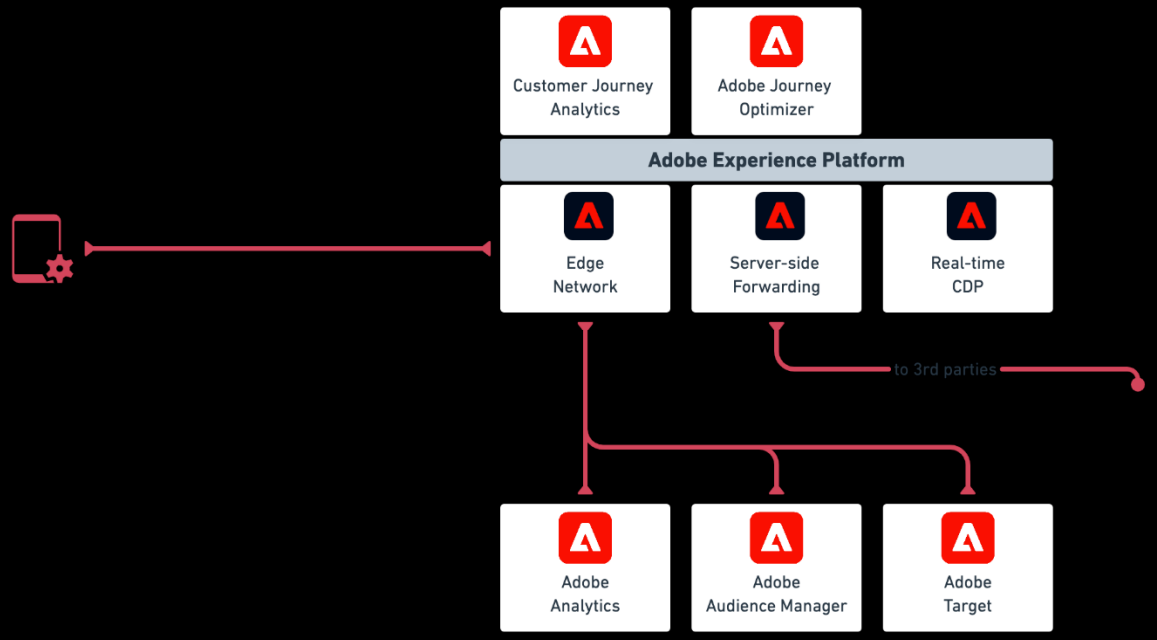
Data Collection in the AEP (Hands-on)

- Where to find the data in the AEP
- Check for successful ingestion
- Preview Data

adaptTo()

5. Prospects

Connecting further Adobe tools



Using Launch as Tag Manager for the Mobile SDK?

• Create data elements & rules • Add static data to payload • Add a given set of global data to payload, e.g. lifecycle metrics, ECID,… • Modify data of payload (static and global)	• Adding, removing or updating Launch extensions on the fly • Add dynamic event data to the payload • Modify dynamic event data

adaptTo()

Thank you!

Any questions?

References (1/2)

- Official Adobe Experience Platform Mobile SDK Documentation:
  - https://aep-sdks.gitbook.io/docs/
  - https://aep-sdks.gitbook.io/docs/getting-started/overview
  - https://aep-sdks.gitbook.io/docs/getting-started/setup-schemas-and-datasets
  - https://aep-sdks.gitbook.io/docs/getting-started/configure-datastreams
  - https://aep-sdks.gitbook.io/docs/getting-started/create-a-mobile-property
  - https://aep-sdks.gitbook.io/docs/getting-started/get-the-sdk
  - https://aep-sdks.gitbook.io/docs/getting-started/initialize-the-sdk
  - https://aep-sdks.gitbook.io/docs/beta/project-griffon
  - https://aep-sdks.gitbook.io/docs/resources/privacy-and-gdpr#set-and-get-privacy-status

- Git Repository of Bus Booking demo app:
  - https://github.com/Adobe-Marketing-Cloud/busbooking-mobileapps

References (2/2)

- Jan Exner on Adobe Experience Platform Mobile SDKs:
  - https://webanalyticsfordevelopers.com/2020/01/21/adobe-experience-platform-mobile-sdks-overview/
  - https://webanalyticsfordevelopers.com/2020/02/11/adobe-experience-platform-mobile-sdks-deep-links-more-setup/
  - https://webanalyticsfordevelopers.com/2020/02/25/adobe-experience-platform-mobile-sdks-debugging/
  - https://webanalyticsfordevelopers.com/2020/03/17/adobe-experience-platform-mobile-sdks-using-launch-as-tag-manager-almost-like-on-the-web/