WS1 Intelligence SDK for Android 24.1.0 Release Notes

- Minimum Requirements
- New Features
- Resolved Issues
- Known Issues

Minimum Requirements

- Android 5.0 or later
- API Level 21 or later
- Workspace ONE UEM Console 2109 or later
- Android Studio with the Gradle Android Build System (Gradle) 4.1.3 or later
- Workspace ONE Intelligent Hub for Android version 21.9 or later.

New Features

- SDK now targets up to latest SDK version 33
- Compatibility for Android devices back to minimum SDK version level 21 has been added.
- Network Insights Refactor, Instrumentation of OkHttp, Http(s)UrlConnection. Guide found here: Android Intelligence SDK Network Insights Clients User Guide.
  - Alternate link: https://developer.vmware.com/web/sdk/24.1.0/ws1-intelligence-sdk/ under 'Documentation and Reference"
- Intelligence SDK Configuration Attributes updated documentation on the following page: Crittercism Configuration Attributes
- Digital Employee Experience (DEX) Telemetry Collection: New feature introduced to collect device telemetry around device metrics, battery statistics, network information, and more.
- A new API has been introduced to set `Application` or `DEX` Telemetry Opt-In status.
  The default Opt-In state for the `Application` is set to true, however the default state for the `DEX` feature is set to false and will need to be enabled to make use of the feature. Status settings must be performed after the initialization of the SDK through the `Crittercism.initialize` API.
  - Example:

Set Opt In Status Example

```
// Must Initialize the Crittercism object before setting status
Crittercism.initialize(context, MY_APP_ID, crittercismConfig)

// Set and Get Application Opt Status
Crittercism.setOptInStatus(Crittercism.TelemetryType.APPLICATION, false) //Default for this
Telemetry Type is true
Crittercism.setOptInStatus(Crittercism.TelemetryType.APPLICATION, true)
val sdkOptInStatus = Crittercism.getOptInStatus(Crittercism.TelemetryType.APPLICATION)

// Set and Get DEX Feature Opt Status
Crittercism.setOptInStatus(Crittercism.TelemetryType.DEX, true) //Default for this Telemetry Type
is false
Crittercism.setOptInStatus(Crittercism.TelemetryType.DEX, false)
val dexOptInStatus = Crittercism.getOptInStatus(Crittercism.TelemetryType.DEX)
```

- Developers should now pull the Maven Artifact for the Intelligence SDK using the following:

Installing via Gradle

In the `build.gradle` file living in the `projectname/app` directory, add the Workspace ONE Intelligence repository information in `repositories{...}`

- Download the .aar from HERE
  - Alternate Link under "Downloads"
- Add it to the libs directory. NOTE - libs directory should ideally be under the app directory.
- Pull the IntelligenceSDK dependency like below

  ```
  implementation files('libs/WS1IntelligenceSDK-24.1.0-20240213.231417-5.aar')
  ```

Be sure to sync your project with your gradle files after you add this!

- Event Caching mode has been added for apps targeting **Tenant Region reporting**: The use of the Workspace One Android SDK (WS1 SDK) with the Intelligence SDK (Intel SDK).

In the Caching mode, the Intel SDK will cache events until the application is able to authenticate with WS1 SDK in a 5 minute grace period. After 5 minutes, if we still haven't authenticated, the Intel SDK will drop into the unauthenticated state and send event to the developer endpoint.

- To allow for Tenant Reporting in this case; please enable the following flags in the `CrittercismConfig` passed into the Intelligence SDK initializer method `initialize`.
  - `isTenantRegionEnabled`
  - `isCachingModeEnabled`
- If your application does not utilize the WS1 SDK, do not set the above flags in order to not cause event reporting delays.

- App Name will now default to the app name specified in the application's Android Manifest. Previously, it defaulted to an empty string.
- Users can now specify a reason for failing a Userflow using the following API:

Android Userflow Fail Reason Example

```
Crittercism.failUserFlow(String name, String failureReason)
```

This Failure Reason will be sent to the Intelligence Console, allowing users to provide information on why a Userflow failed.
NOTE: Like Userflow names, Failure Reasons are limited to a max of 255 characters.
- New Public API added to provide UEM attributes to the IntelligenceSDK's DEX feature:

  - This API takes an information provider interface which allows users to create getter methods for each available data attribute: **UEM Device UDID, UEM Serial Number, UEM Username.**

Android UEMInfo Interface

```
------------------------------------------------------------
/**
 * Interface for providing UEM Information to our DEX
 */
interface UemInfo {
    /**
     * Returns the UEM Device UDID as a String
     */
    fun getUemDeviceUDID(): String?
    /**
     * Returns the UEM Serial Number as a String
     */
    fun getUemSerialNumber(): String?
    /**
     * Returns the UEM Username as a String
     */
    fun getUemUsername(): String?
}
```

IntelligenceSDK UEMInfo Sample Implementation

```
import android.content.Context
import android.content.RestrictionsManager
import com.crittercism.uemprovider.UemInfo
import com.crittercism.uemprovider.UemInfoKeys

internal class IntelSDKUemInfo(context: Context): UemInfo {

    private val restrictionsMgr = context.getSystemService(Context.RESTRICTIONS_SERVICE) as
RestrictionsManager

    override fun getUemDeviceUDID(): String = restrictionsMgr.applicationRestrictions.getString
(UemInfoKeys.UEM_KEY_DEVICE_UDID,
            UemInfoKeys.UEM_KEY_UNAVAILABLE)

    override fun getUemSerialNumber(): String = restrictionsMgr.applicationRestrictions.getString
(UemInfoKeys.UEM_KEY_SERIAL_NUMBER,
            UemInfoKeys.UEM_KEY_UNAVAILABLE)

    override fun getUemUsername(): String = restrictionsMgr.applicationRestrictions.getString
(UemInfoKeys.UEM_KEY_USERNAME,
            UemInfoKeys.UEM_KEY_UNAVAILABLE)
}
```

Registering IntelligenceSDK UEM Info

```
// Create instance of UemInfo
uemInfo = IntelSDKUemInfo(context)
Crittercism.setUemInfo(uemInfo)
Crittercism.initialize(context, APTELIGENT_APP_ID, config)
```

- **NOTE - To get UEM Information to DEX Feature, set UemInfo before enabling DEX through the Crittercism singleton. API to set UemInfo is Crittercism.setUemInfo(UemInfo info).**

Resolved Issues

- WebView Instrumentation fixed for devices targeting above SDK 23.
- Network Insights are now supported on devices up to SDK 33.
- Fix syncing of breadcrumb's creation.

Known Issues

- Android Devices Below SDK version level 23 instrumenting WebViews will not be able to report HTTP errors in their Network Insights, this is due to the following APIs not being supported previous to Android Marshmallow (23):
    - onReceivedHttpError(WebView, WebResourceRequest, WebResourceResponse)
    - onReceivedError(WebView, WebResourceRequest, WebResourceError)