

## Public Maven Repository Integration Note

### Workspace ONE for Android

The Omnissa Workspace ONE® software development kit for Android can now be utilized directly from a public Maven repository. Direct utilization from the repository is an alternative to downloading the software development kit from the My Workspace ONE website.

### Downloads

Omnissa provides this Software Development Kit (the “Software”) to you subject to the following terms and conditions. By downloading, installing, or using the Software, you agree to be bound by the terms of [SDK License Agreement](#). If you disagree with any of the terms, then do not use the Software.

For additional information, please visit the [Omnissa Legal Center](#).

### License

This software is licensed under the [Omnissa Software Development Kit \(SDK\) License Agreement](#); you may not use this software except in compliance with the License.

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

That applies however you obtain or integrate the software.

## Table of Contents

Tasks and Documentation.....	2
Integration Preparation Guide Changes.....	2
Downloads	
Base Integration Guide Changes.....	3
Open Source Sample Code.....	5
Document Information.....	6

## Tasks and Documentation

Integration of an Android application with the Workspace ONE platform includes a number of tasks for the application developer. Instructions for these tasks are given in a set of documents: the Workspace ONE Integration Guide for Android.

The integration guide documents are available in the [Workspace ONE Android SDK Overview](#).

Most of the instructions apply the same whether the SDK is utilized directly from the public Maven repository or by downloading from My Workspace ONE. However, some instructions apply only if downloading and must be changed for direct utilization.

Changed instructions are given below for each document in the integration guide set.

There are no changes for the public Maven repository in the

- Integration Overview.
- Branding Integration Guide.

## Integration Preparation Guide Changes

Follow these changed instructions instead of those in the Integration Preparation Guide to utilize the public Maven repository.

Skip the **Task: Obtain software development kit** instructions.

## Downloads

Omnissa provides this Software Development Kit (the “Software”) to you subject to the following terms and conditions. By downloading, installing, or using the Software, you agree to be bound by the terms of [SDK License Agreement](#). If you disagree with any of the terms, then do not use the Software.

For additional information, please visit the [Omnissa Legal Center](#).

## Base Integration Guide Changes

Follow these changed instructions instead of those in the Base Integration Guide to utilize the public Maven repository.

In the **Task: Add Client SDK** section, the **Build Configuration and Files** instructions are changed as follows.

- In the step **Add the required libraries to the build**:

Instead of adding all the separate dependencies given in the guide, add only the following.

```
repositories {
    maven {
        url = uri("https://maven.pkg.github.com/euc-releases/Android-WorkspaceONE-SDK/")
        credentials {
            /** In gradle.properties file of root project folder, add github.user=GITHUB_USERNAME & github.token=GITHUB_ACCESS_TOKEN **/
            username = project.findProperty("github.user") ?: System.getenv("USERNAME")
            password = project.findProperty("github.token") ?: System.getenv("TOKEN")
        }
    }
    maven {
        url = uri("https://maven.pkg.github.com/euc-releases/ws1-intelligencesdk-sdk-android/")
        credentials {
            /** In gradle.properties file of root project folder, add github.user=GITHUB_USERNAME & github.token=GITHUB_ACCESS_TOKEN **/
            username = project.findProperty("github.user") ?: System.getenv("USERNAME")
            password = project.findProperty("github.token") ?: System.getenv("TOKEN")
        }
    }
}

dependencies {
    // Integrate Omnissa Workspace ONE at the Client level.
    //
    // - Omnissa provides this Software Development Kit (the "Software") to
    // you subject to the following terms and conditions. By downloading,
    // installing, or using the Software, you agree to be bound by the terms
    // of https://static.omnissa.com/sites/default/files/omnissa-sdk-agreement.pdf
    // If you disagree with any of the terms, then do not use the Software.
    //
    // For additional information, please visit the https://www.omnissa.com/legal-center/.
    //
    // - Review the Omnissa Privacy Notice and the Workspace ONE UEM Privacy
    // Disclosure for information on applicable privacy policies, and
    // for additional information, please visit the
    // https://www.omnissa.com/legal-center/
    implementation "com.airwatch.android:airwatchsdk:24.10"
}
```

- Skip the step **Copy the required library files**.
- Skip the step **Add the library files' location to the application build configuration**.

In the **Task: Add Framework** section, the **Build Configuration and Files** instructions are changed as follows.

- In the step Add the required libraries to the build:

Instead of adding all the separate dependencies given in the guide, add only the following.

```
repositories {
    maven {
        url = uri("https://maven.pkg.github.com/euc-releases/Android-WorkspaceONE-SDK/")
        credentials {
            /** Create github.properties in root project folder file with github.user=GITHUB_USERNAME & github.token=GITHUB_ACCESS_TOKEN */
            username = project.findProperty("github.user") ?: System.getenv("USERNAME")
            password = project.findProperty("github.token") ?: System.getenv("TOKEN")
        }
    }
    maven {
        url = uri("https://maven.pkg.github.com/euc-releases/wsl-intelligencesdk-sdk-android/")
        credentials {
            /** Create github.properties in root project folder file with github.user=GITHUB_USERNAME & github.token=GITHUB_ACCESS_TOKEN */
            username = project.findProperty("github.user") ?: System.getenv("USERNAME")
            password = project.findProperty("github.token") ?: System.getenv("TOKEN")
        }
    }
}

dependencies {
    // Integrate Omnissa Workspace ONE at the Client level.
    //
    // - Omnissa provides this Software Development Kit (the "Software") to
    // you subject to the following terms and conditions. By downloading,
    // installing, or using the Software, you agree to be bound by the terms
    // of https://static.omnissa.com/sites/default/files/omnissa-sdk-agreement.pdf
    // If you disagree with any of the terms, then do not use the Software.
    //
    // For additional information, please visit the https://www.omnissa.com/legal-center/.
    //
    // - Review the Omnissa Privacy Notice and the Workspace ONE UEM Privacy
    // Disclosure for information on applicable privacy policies, and
    // for additional information, please visit the
    // https://www.omnissa.com/legal-center/
    implementation "com.airwatch.android:awframework:24.07"
}
```

- Skip the step Copy the required library files.

## Open Source Sample Code

Open Source sample code for integration of Android applications with Workspace ONE is published to a GitHub repository. The build files of one set of the applications in the repository have been updated to utilize the public Maven repository.

The sample code repository is here:

[github.com/euc-releases/workspace-ONE-SDK-integration-samples](https://github.com/euc-releases/workspace-ONE-SDK-integration-samples)

The following sample apps have updated build files.

- [brandStaticDelegateJava](#)
- [brandStaticDelegateKotlin](#)
- [brandStaticExtendJava](#)
- [brandStaticExtendKotlin](#)

Each app directory has the following build files.

- `publicMavenClient.gradle` to integrate at the Client level.
- `publicMavenFramework.gradle` to add integration at the Framework level.

## Document Information

### Revision History

21apr2021 First publication, for 21.3 SDK for Android.  
05jul2024 Second publication, for 24.06 SDK for Android.