Workspace ONE Gradle Plugin (BETA)

The Workspace ONE plugin for Gradle facilitates integration of Android applications with the VMware Workspace ONE® platform. The code changes that are required in order to integrate an application can be made automatically by the Gradle plugin.

The Gradle plugin is included with the Workspace ONE software development kit (SDK) for Android.

Table of Contents

Introduction

The Workspace ONE SDK Gradle plugin can automate the following work.

- Build configuration changes.

  For example, adding the required packaging and compile options, and adding library dependencies.

- Android Manifest changes.

  For example, declaring the required permissions, intent receiver, and service class.

- Code changes.

  For example, changing the inheritance of the Application and Activity classes.

  The plugin transforms the generated class files, not the original source code.

- Resource file changes.

  For example, changing View subclasses in a layout XML file.

The plugin is configurable. The app developer can select which of the above changes are to be made automatically by the plugin.

Use of the Gradle plugin is optional. The alternative is to make the required changes manually. For a guide to manual integration, see the Workspace ONE Integration Guide for Android set of documents. An overview that includes links to all the guides is available

- in Markdown format, in the repository that also holds sample applications: https://github.com/vmware-samples/…IntegrationOverview.md

- in Portable Document Format (PDF), on the VMware website: https://code.vmware.com/…IntegrationOverview.pdf

Agreement

Before downloading, installing or using the VMware Workspace ONE SDK you must:

- Review the VMware Workspace ONE Software Development Kit License Agreement. By downloading, installing, or using the VMware Workspace ONE SDK for Android you agree to these license terms. If you disagree with any of the terms, then do not use the software.

- Review the VMware Privacy Notice and the Workspace ONE UEM Privacy Disclosure, for information on applicable privacy policies.

Integration

To integrate your application using the Workspace ONE Gradle plugin, follow the instructions below.

Compatibility

Before you begin, ensure you have access to compatible versions of software. The following table shows the minimum required versions.

Software	Required
Workspace ONE software development kit for Android	21.6.1

Table 1: Software Compatibility Version Numbers

Also, note the following.

- The plugin is compatible with the AndroidX libraries. It isn't compatible with the Android support libraries.

- The plugin supports Framework integration and networking integration. It doesn't support Client SDK integration.

- The plugin will implement Framework integration by extension, not by delegation.

- The plugin can be configured to make code changes.

  - By default, the code in all Application and Activity classes in the app will be modified by the plugin. Exclusions can be configured.

  - Any life cycle method overrides in the modified classes must call the corresponding method in the base class, by using the `super` keyword. This means methods such as onCreate, onResume, onNewIntent, and onActivityResult for example.

  - Code that is in scope of modification can't be observed with a debugger.

- Changes to resource files are made by a standalone Gradle task.

  When the task is run, XML files that define screen layouts will be changed. The indentation and formatting of the XML won't be preserved.

Add Plugin

First, add the Workspace ONE Gradle plugin to your application. Proceed as follows.

1. Download the plugin file.

    The plugin isn't yet available from the plugins.gradle.org repository.

    You can download the plugin file from here:
    vmwaresaas.jfrog.io/ … /ws1-sdk-plugin.gradle

2. Copy the plugin file to the application module sub-directory.

    The typical project structure is as follows.

    - *Project* files in the root directory.
    - *Application* module in a sub-directory.
    - Separate `build.gradle` files for the project and application.

    The destination for the plugin file would be the same directory as the `build.gradle` file for the application.

3. Add the required repository and dependencies to the application build configuration.

    The required additions are illustrated in the following code snippet.

    ```
    buildscript {
        repositories {
            google()
            mavenCentral()
            maven {
                url 'https://vmwaresaas.jfrog.io/artifactory/Workspace-ONE-Android-SDK/'
            }
        }
        dependencies {
            classpath "org.javassist:javassist:3.27.0-GA"
            classpath "com.vmware.ws1.sdk:ws1-sdk-gradle-plugin:1.0-beta"
        }
    }
    ```

4. Apply the plugin to the application build configuration.

    Add a line like this:

    ```
    apply from: 'ws1-sdk-plugin.gradle'
    ```

5. Synchronize the project with the Gradle files.

This completes the addition of the plugin. You are now ready to continue to the next task, to change the app resources.

Change App Resources

The Workspace ONE Gradle plugin can facilitate changes to layout XML files in the app resources. This is implemented as a standalone Gradle task. When you run the task, the files will be modified in place.

After the plugin has been added to the project, see above, the Gradle task will be present in the `ws1sdk` category, named `replaceResources`. The task can be run from the Gradle command line or from the Android Studio integrated development environment, for example. See the Appendix: Screen Captures for the location.

The replaceResources task replaces native View subclasses with equivalent subclasses from the Workspace ONE SDK for Android. For example, TextView instances will be replaced with AWTextView instances. The replacement classes will enforce copy-paste restrictions and other data loss prevention (DLP) policy settings in the app user interface.

This task could involve extensive file reading and writing, depending on the size of the app user interface. Only run it after changes have been made to the resource files. Under continuous integration or continuous deployment (CI-CD), put this task before the Android Package (APK) build in the plan.

Use of the plugin to change app resources is optional. If it isn't used then the changes must be made manually or handled another way.

After the app resources have been changed, you are ready to continue to the next task, to configure the plugin.

Configure Plugin

After the app resources have been changed, see above, the next step is to configure the plugin.

Default Configuration

The Workspace ONE Gradle plugin makes a number of configurable modifications, to the manifest file and to the code. The plugin has a default configuration that is intended to be sufficient for the typical application. If you don't specify any configuration, the default will be applied.

Try building and running your app with the default configuration first.

Custom Configuration

In case the default configuration makes problematic changes to your application or you prefer to make some changes manually, you can create a custom configuration for the Workspace ONE Gradle plugin.

Create a `ws1configuration` block in your application build file and set the required options as block attributes.

The following code snippet lists the configuration options and describes what they do in comments.

```
ws1configuration {
    modifyManifest = true
    // Activate modification of the Android Manifest file, default: true.
    // Modifications include declaring the required permissions, intent receiver, and service
    // class. If activated then launcherActivity, below, must be configured.
    // If deactivated, the manifest won't be changed.

    launcherActivity = "com.example.myapplication.MainActivity"
    // Specify the app Activity to launch after SDK initialization is complete. The SDK has
    // its own launch and loading user interface. The SDK user interface includes screens on
    // which, for example, the end user authenticates themselves with a passcode if required by
    // enterprise policy. Set the fully qualified class name of the application's main
    // activity, including package.

    modifyCode = true
    // Activate modification of the app code, default: true.
    // Modifications include changing the inheritance of Application and Activity classes.
    // Kotlin and Java classes will be modified.
    // If deactivated, no code will be modified.

    excludeClasses = [
        'com.example.myapplication.ExcludedActivity',
        'com.example.myapplication.ExcludedApplication'
    ]
    // Specify classes to exclude from code modification.
    // The classes in this list won't be modified by the plugin. Specify the fully qualified
    // name of each class.
    // In an Application subclass, the onCreate method will be changed to onPostCreate. If the
    // app has an inheritance hierarchy of Application subclasses, then all or none must be
    // excluded.
    // Default is to modify all Activity and Application classes, if modifyCode:true is set.

    enableWatermark = false
    // Activate user interface watermark, default: false.
    //
    // Workspace ONE supports overlaying a watermark text on the user interface of a mobile
    // application. The watermark is generated at run time and can include identifying details
    // of the application end user. Watermark overlay is configured in the management console.
    // If the feature isn't configured for the current end user at run time, then no watermark
    // will be overlaid regardless of the plugin configuration.
    // For details, see the technical feature guide:
    // https://code.vmware.com/docs/12968/UserInterfaceWatermark.pdf
    //
    // The plugin modifies the code of the app Activity classes to support the watermark.
    // Activity classes that are excluded from code modification are also excluded from
    // watermark support.
    // If code modification is deactivated then no Activity classes will be modified to support
    // the watermark.
}
```

You can paste the above block into your Gradle then modify the settings to configure the plugin to meet the requirements for your app.

Next Steps

When the above tasks are complete, the app should be tested. Install the app on a device via Workspace ONE Intelligent Hub and enroll with a Workspace ONE Unified Endpoint Manager (UEM) console in the usual way.

This completes integration of the Workspace ONE plugin for Gradle.

Appendix: Screen Captures

The following screen capture shows the location of the Gradle task for replace resources. For step-by-step instructions, see Change App Resources.
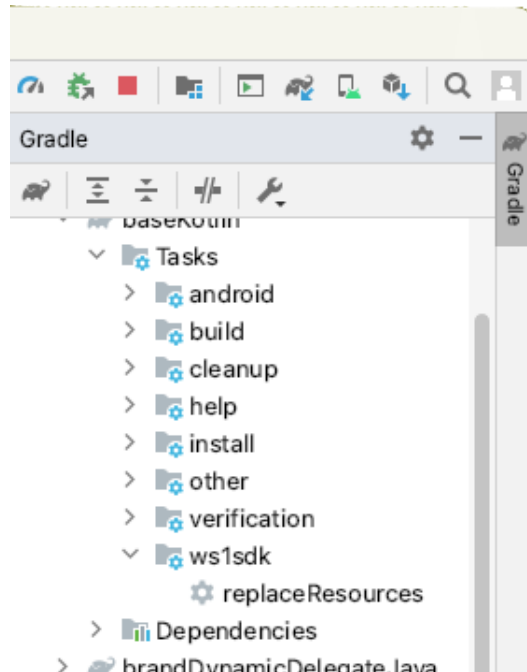
Screen Capture 1: Replace resources task in the integrated development environment

If the Gradle tasks don't appear in the Android Studio Gradle tool window, try the experimental option to deactivate task list building. The following screen capture shows its location in the Android Studio user interface.
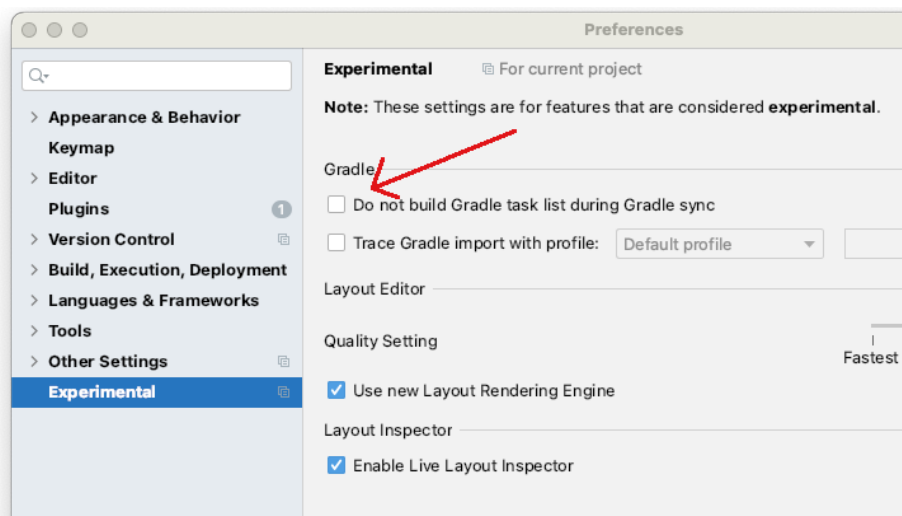
Screen Capture 2: Gradle task list experimental deactivation

(Thanks are due to this StackOverflow answer: stackoverflow.com/a/67636646.)

Document Information

Revision History

01aug2021 Initial Publication.

Legal