



← ↻ ☆ ProviderBasedDirectoryEntryImpl in AGP 7.3 is too strict

+1²

Hotlists

Mark as Duplicate



Comments (1) Dependencies Duplicates (0) Blocking (0) Resources (1)

Assigned

Bug

P3

+ Add Hotlist



STATUS UPDATE No update yet.

Edit



DESCRIPTION tr...@squareup.com created issue #1

DESCRIBE THE ISSUE IN DETAIL:

STEPS TO REPRODUCE:

1. See the error message and code snippets below.

```
# build error message
* What went wrong:
Execution failed for task ':sdk:mergeReleaseJniLibFolders'.
> There are more than one element in jniLibs
   [/Users/trobalik/.gradle/caches/transforms-3/23d55ef38f0704fc99239b6afd91a657/transformed/native-resources-20221117_211916-efa360dfd6/jni, /Users/trobalik/
```

```
// build.gradle
android
    .sourceSets
    .getByName("main")
    .jniLibs
    .srcDirs(jniLibs /* a file collection with two top-level entries */)


```

```
// an internal plugin
// jniLibs is a ConfigurableFileCollection
jniLibs.setFrom(getArtifactTypeFiles(configurations.getByName("releaseRuntimeClasspath").incoming, ArtifactType.JNI.type))

internal fun getArtifactTypeFiles(
    configuration: ResolvableDependencies,
    artifactType: String
): FileCollection = configuration.artifactView { view ->
    view.attributes { container ->
        container.attribute(ARTIFACT_TYPE, artifactType)
    }
}.artifacts.artifactFiles


```

```
# the two top-level entries in the file collection
0 = {DefaultFileSystemLocation@28972} "/Users/trobalik/.gradle/caches/transforms-3/23d55ef38f0704fc99239b6afd91a657/transformed/native-resources-20221117_211
1 = {DefaultFileSystemLocation@28973} "/Users/trobalik/.gradle/caches/transforms-3/bc6f5eaf694eec75d59795b423dc53ff/transformed/bugsnag-android-core-5.24.0/j
```

```
// AGP 7.3.1
class ProviderBasedDirectoryEntryImpl(
    override val name: String,
    val elements: Provider<Set<FileSystemLocation>>,
    override val filter: PatternFilterable? = null
): DirectoryEntry {

    override fun asFiles(directoryPropertyCreator: () -> DirectoryProperty): Provider<Directory> {
        return elements.flatMap {
            if (it.size > 1) {
                throw RuntimeException("There are more than one element in $name\n" +
                    "${it.map { it.asFile.absolutePath }}")
            }
            directoryPropertyCreator().also { directoryProperty ->
                directoryProperty.set(it.single().asFile)
            }
        }
    }
}


```

```
// Workaround that resolves the issue
/*
 * Workaround for issue with AGP 7.3.x. Here we "flatten" the hierarchy of native libs into a
 * single directory and pass that as the input to 'embedJni'. See AGP 7.3.1's
 * ProviderBasedDirectoryEntryImpl's 'asFiles()' method, which can throw a RuntimeException
 * exception.
 */


```

```
val copy = tasks.register<Copy>("copyJni") {
    from(getArtifactTypeFiles(incomingDependencies, ArtifactType.JNI.type))
    into(layout.buildDirectory.dir("intermediates/jni_copy"))
}

/* Adds the jni/ directory for each dependency as a source directory for the final AAR. */
val embedJni = tasks.register<EmbedJni>("embedJniLibs$variantSlug") {
    dependsOn(preBuild)
    jniLibs.setFrom(copy)
    android = project.android!!
}
```

The Copy task flattens the hierarchy into a single directory.

IMPORTANT: Please read <https://developer.android.com/studio/report-bugs.html> carefully and supply all required information.


Studio Build: n/a Version of Gradle Plugin: 7.3.1 Version of Gradle: 7.5 Version of Java: 11 OS: macOS

COMMENTS



hu...@google.com <hu...@google.com>

Assigned to an...@google.com.



je...@google.com <je...@google.com>

Reassigned to cm...@google.com.