Building combined armv7/x86 apk after Crosswalk integration in an Ionic project

I have added Crosswalk to my ionic project. When i do `ionic build android` it outputs two separate APK files, one for armv7 and one for x86.

However i want one single apk file bundling two architectures together.

I tried setting `BUILD_MULTIPLE_APKS = false` but it's still building two separate apks.

There is a post here saying :

> Took me a minute but you can get the single build file by commenting out the cdvBuildMultipleApks=true line in the platforms/android/gradle.properties
>
> Cheers

But i don't have any gradle.properties file in my platforms/android/ folder, i have a project.properties file and there is no

`cdvBuildMultipleApks=true` line in it to comment out.

I tried to manually create a gradle.properties file and putting `cdvBuildMultipleApks=false` in there but still two outputs.

Any ideas how to achieve this?

Update:

My Cordova version is 5.1.1

Solution:

What did work for me as suggested by @mudasserajaz and here, was to :

1. Create a file named `build-extras.gradle` inside my `/platform/android/` directory.

2. put `ext.cdvBuildMultipleApks=false` inside it.

3. `ionic build android --release` then and Voila, it worked.

@mudasserajaz suggested the same thing except variable name is `cdvBuildMultipleApks=false` so in case it didn't work out for you, don't forget the `ext`

cordova   ionic-framework   crosswalk-runtime

<div style="text-align:right">

edited Feb 17 at 3:31   asked Sep 12 '15 at 5:37

Mudasser Ajaz   Sobhan
3,070 7   18   **296** 2   15

</div>

what is your cordova version? – Mudasser Ajaz Sep 12 '15 at 6:23

@mudasserajaz, my Cordova version is 5.1.1 – Sobhan   Sep 12 '15 at 6:44

1 Answer

Here are steps to do that:

1. Make a file `build-extras.gradle` and put it into `platforms/android/` . Add line `cdvBuildMultipleApks=false` into `build-extras.gradle` .

2. Now run `ionic run android`, it will create combined single build `android-debug.apk`.

`build.gradle` will automaticaly import `build-extras.gradle`, if that is defined. You can see code of `build.gradle` doing this :

```
def hasBuildExtras = file('build-extras.gradle').exists()
if (hasBuildExtras) {
    apply from: 'build-extras.gradle'
}
```

Plus these instructions are clearly mentioned in `build.gradle` line 57 :

> Configuration properties. Set these via environment variables, build-extras.gradle, or gradle.properties.

NOTE: Before you follow these steps, remove platform folder and install again.
Observations: Once you have done these steps, and again at some point you want to make separate builds, then setting `cdvBuildMultipleApks=true` or commenting out sometimes wont work, you will simply have to install platform again. (P.S this is my personal observation, otherwise deleting `buid-extras.gradle` should be enough.)

<div style="text-align:right">
edited Sep 13 '15 at 10:55   answered Sep 12 '15 at 12:42

Mudasser Ajaz
3,070   7   18
</div>

Creating `build-extras.gradle` in `platforms/android/` and adding the line `cdvBuildMultipleApks=false` inside it worked like a charm! – Sobhan Sep 13 '15 at 4:41

Oh i realized my variable name inside `build-extras.gradle` was `ext.cdvBuildMultipleApks=false`. So in case the other one didn't work, don't forget the ext. – Sobhan Sep 13 '15 at 7:22

Regarding your update, to make separate builds, I deleted my `build-extras.gradle` file and again it made two separate files. So maybe just doing that or commenting the `cdvBuildMultipleApks=false` is enough. – Sobhan Sep 13 '15 at 10:12

those are just my observations, which i faced on my system. Otherwise you are right, deleting file should be enough. – Mudasser Ajaz Sep 13 '15 at 10:52