

Configuring Eclipse, Android Development Plugin and SDK To Work Properly When Directory Pathing Is Wrong

Today, I started delving into programming in Android and I downloaded all the tools I needed. I went through a part of the tutorial on building a Hello World app as every beginner does when starting something new in programming. I went through the tutorial up to the point where it required me to run the app. I got an error saying that it couldn't find the ini file for the Android Virtual Device (AVD) I created. It turns out that these files are supposed to be in my C:\Users\

Problem 1: Android SDK and AVD Manager is using the wrong folder to store it's AVD's

The Android Virtual Device is an emulator for the developer to test their program. It should be stored in C:\Users\

To fix it, I had to create a new environment variable called ANDROID_SDK_HOME and pointed it to C:\Users\

Problem 2: The Android Development Plugin on Eclipse is also looking at the wrong folder

The ADT plugin on Eclipse allows the developer to test apps with the virtual device. The developer can create/modify/delete new AVD's using this plugin. After I fixed the first problem and created a new AVD, the plugin on Eclipse did not recognize this change because it too was looking in the wrong directory. I found that the directory information was stored in a variable that Eclipse users called *user.home*.

To fix it, I had to go into the Eclipse options > Run/Debug > String Substitution and add a new entry with variable *user.home* with a value of C:\Users\

After fixing problems 1 and 2, there should be two folders inside the folder for that particular user. One says *.android* and the other says *.m2*. Before, I was seeing these two folders in my Dropbox folder - not where I wanted them.

Problem 3: Eclipse complains about an invalid command-line parameter

Thinking that my problems were resolved after fixing the directories, Eclipse gave me this error message when I tried to run my code:

```
[2011-08-16 23:24:00 - Emulator] invalid command-line parameter: Files.
```

```
[2011-08-16 23:24:00 - Emulator] Hint: use '@foo' to launch a virtual device named 'foo'.
```

```
[2011-08-16 23:24:00 - Emulator] please use -help for more information
```

It turns out that the issue was caused by the SDK not knowing how to handle directories with spaces. I installed the Android SDK in C:\Program Files (x86), which is clearly the culprit here.

To fix it, move the whole SDK installation to a directory without spaces. I just used my C drive. After changing this, I had to tell Eclipse the new directory that the Android SDK was now in. This also means changing the registry key at the following places:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Android SDK Tools

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\Android SDK Tools

Anywhere that has the old Android directory must have its value changed to reflect the new directory. For mine, I changed everywhere that said C:\Program Files (x86)\Android...blahblaht to C:\Android...blahblah.

Annoyance: The .m2 folder for Maven doesn't appear in my user folder either

I don't know what this Maven thing is or what it is used for, but I do know that it is part of my Eclipse installation. For some reason, it's local repository didn't appear in my users folder (which is supposed to be the default place), but rather in my Dropbox folder. I ended up getting rid of this too.

I wrote a new *settings.xml* file for Maven and put it in *C:\Users\<username>\.m2* with the following content (but with the *<username>* part changed):

```
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
    http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <localRepository>C:\Users\<username>\.m2</localRepository>
</settings>
```

Afterward, I went into the Eclipse Preferences and pointed to this settings file and restarted the program. Result: no more .m2 folder in my Dropbox stuff.

Final Result: A working setup for Android development!

My code now runs and I actually see something! Hooray!

Source: <http://henrypoon.wordpress.com/2011/08/17/configuring-eclipse-android-development-plugin-and-sdk-to-work-properly-when-directory-pathing-is-wrong/>