

## Building AGSforAndroidProxy Wrapper

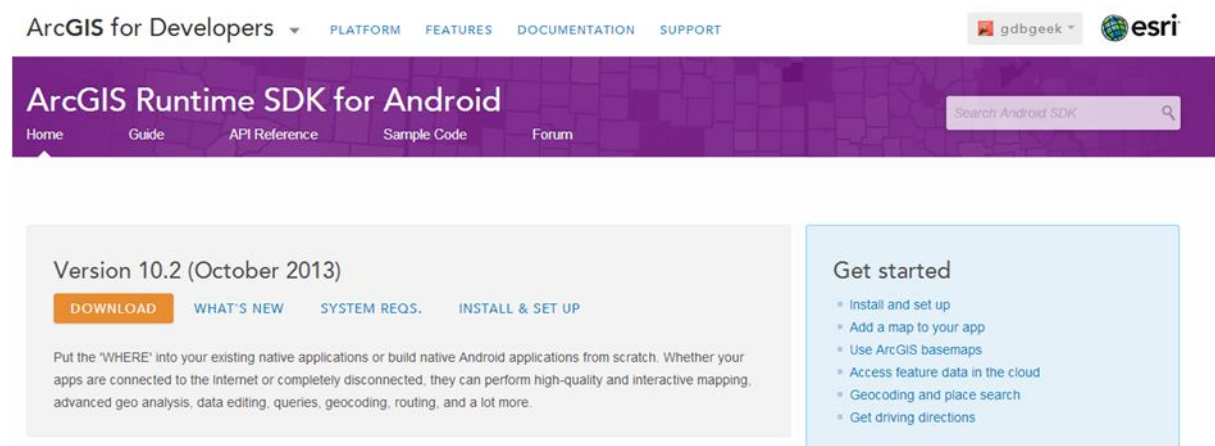
Due to the licensing restrictions on the SDK I cannot distribute the Esri binaries with the wrapper. These instructions will step you through building the wrapper.

You need three things;

- 1) The Esri ArcGIS Runtime SDK for Android
- 2) The wrapper source code
- 3) The Simple Library Compiler (SLC) from B4A

## Download the Esri ArcGIS Runtime SDK for Android

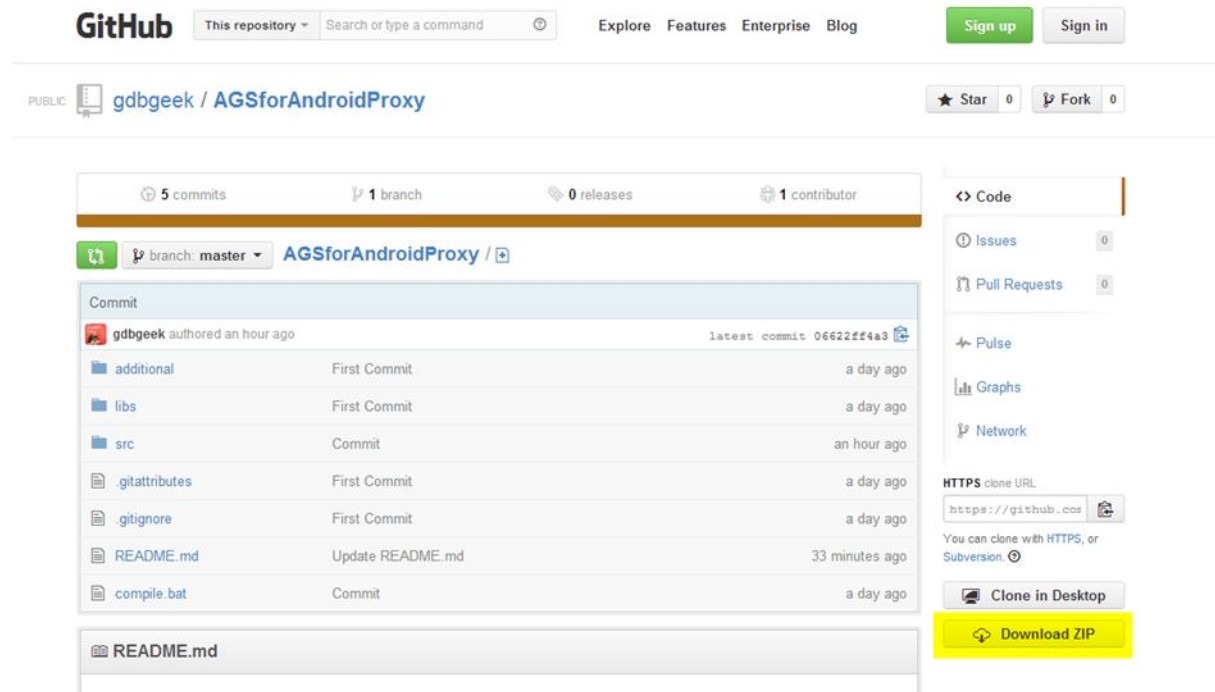
Visit <https://developers.arcgis.com/en/android/> to download the SDK. You will have to sign in or create a new developer account.



The current version is 10.2 and the downloaded file should be called "arcgis-android-sdk-v10.2.zip"

## Download the wrapper source

Visit <https://github.com/gdbgeek/AGSforAndroidProxy> to download the wrapper source. You shouldn't need an account to download the source code. Click the "Download Zip" link on the bottom right.



The screenshot shows the GitHub interface for the repository `gdbgeek / AGSforAndroidProxy`. At the top, there's a search bar and navigation links like 'Explore', 'Features', 'Enterprise', and 'Blog'. Below the repository name, it shows '5 commits', '1 branch', '0 releases', and '1 contributor'. The main content area displays a list of commits, with the latest commit by 'gdbgeek' at the top. The commit list includes files like 'additional', 'libs', 'src', '.gitattributes', '.gitignore', 'README.md', and 'compile.bat'. On the right sidebar, there are links for 'Code', 'Issues', 'Pull Requests', 'Pulse', 'Graphs', and 'Network'. At the bottom right, there are buttons for 'Clone in Desktop' and 'Download ZIP' (highlighted in yellow).

The direct link is <https://github.com/gdbgeek/AGSforAndroidProxy/archive/master.zip>

The downloaded file should be named "AGSforAndroidProxy-master.zip"

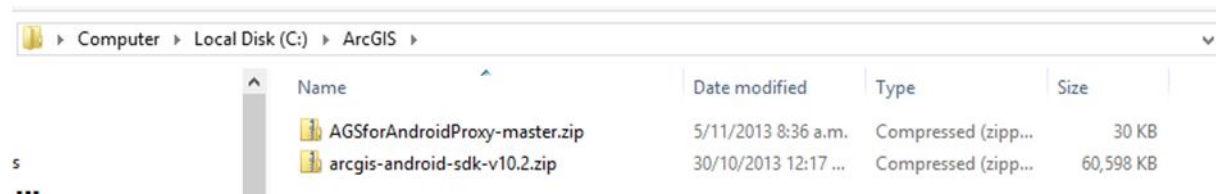
## Download the SLC

Visit <http://www.basic4ppc.com/android/forum/threads/tool-simple-library-compiler-build-libraries-without-eclipse.29918/> and follow the instructions there to get the tool. You need to be a licensed B4A user and a registered member of the forum to acquire the tool.

## Building the wrapper

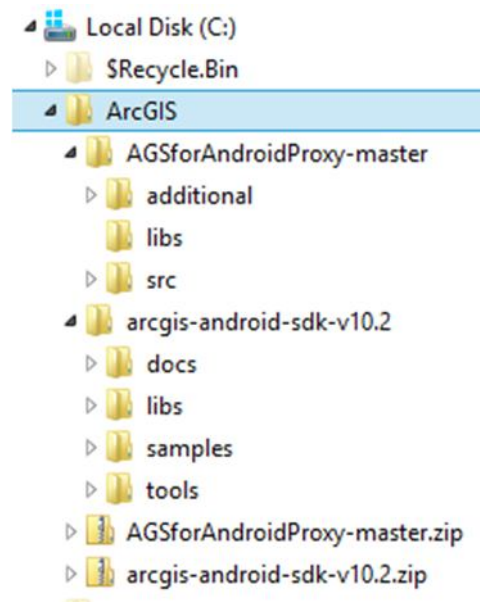
### Step 1

Copy "arcgis-android-sdk-v10.2.zip" and "AGSforAndroidProxy-master.zip" to a suitable folder. In my case I use C:\ArcGIS\



### Step 2

Unzip both files. I use 7-zip and I use the "Unzip Here" right click option. The final directory structure is as follows;

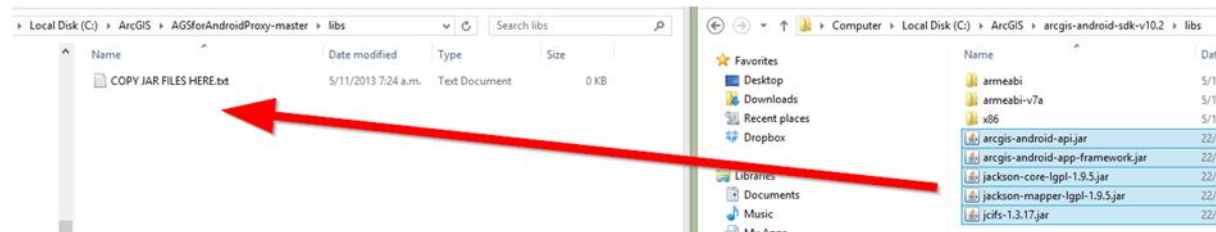


### Step 3

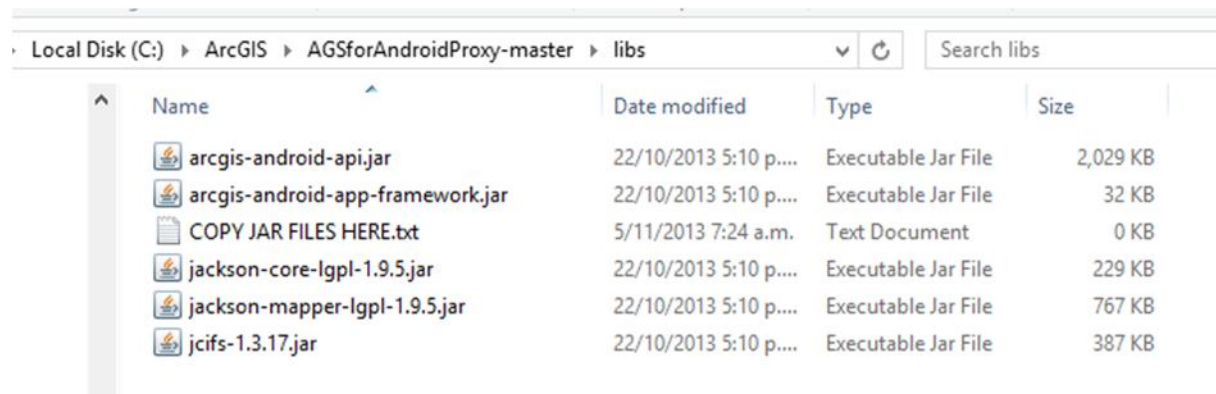
Locate the "libs" folder under the SDK. In my case it is C:\ArcGIS\arcgis-android-sdk-v10.2\libs

Locate the "libs" folder under the wrapper source. In my case it is C:\ArcGIS\AGSforAndroidProxy-master\libs

Copy the JAR files from the SDK libs folder to the wrapper libs folder.

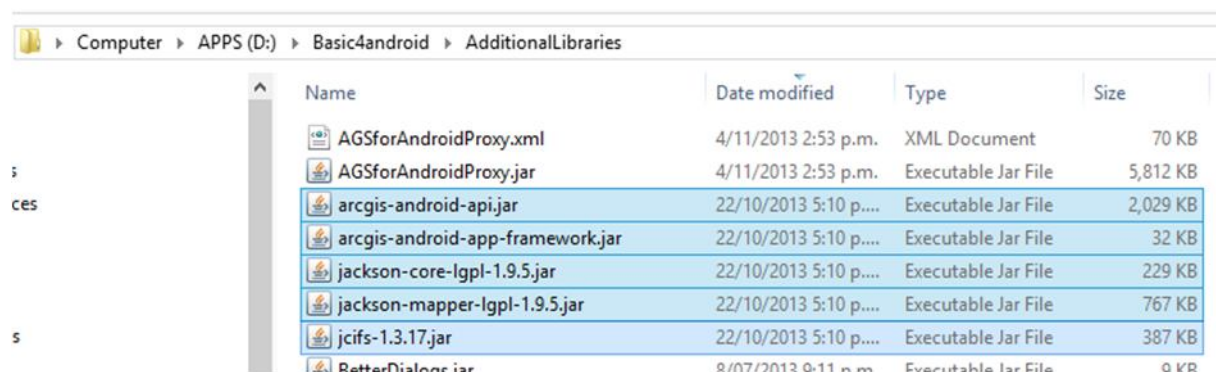


You should now have 5 JAR files in the source libs folder (in my case C:\ArcGIS\AGSforAndroidProxy-master\libs)



### Step 4

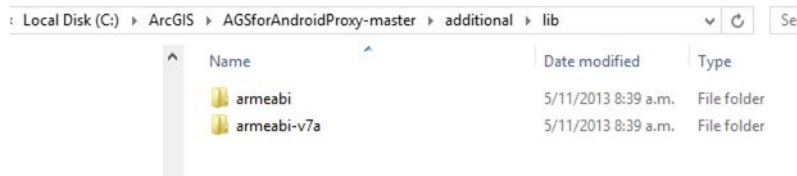
You also need to copy the JAR files to the B4A additional libraries folder otherwise B4A won't find and copy the dependencies across to the device when you deploy.



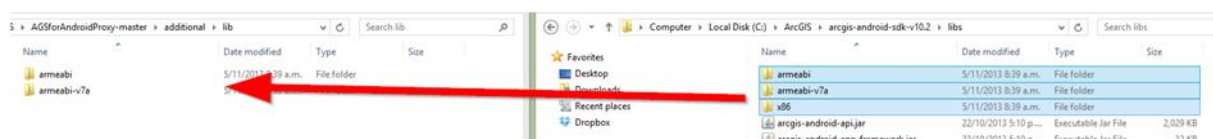
## Step 5

You also need to copy the .SO library files to the “additional” folder under the wrapper source directory. In my case this is C:\ArcGIS\AGSforAndroidProxy-master\additional\lib

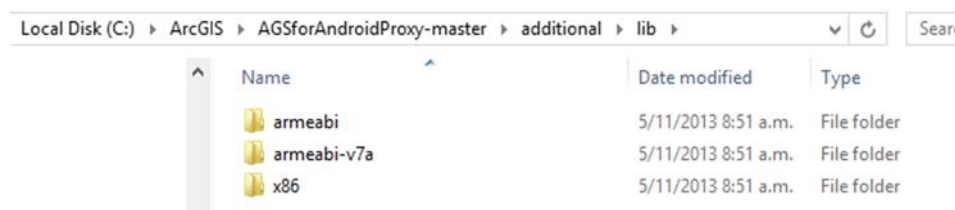
You will notice there are two folders there already;



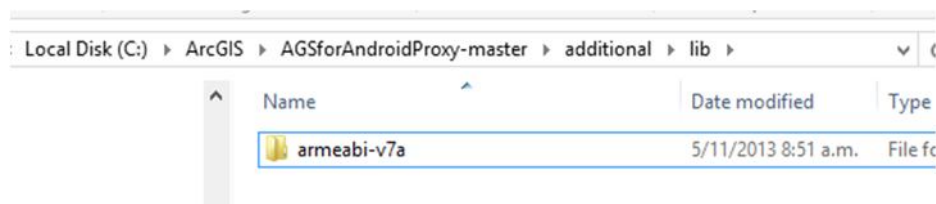
Copy the three folders to the wrapper “additional\libs” folder (in my case C:\ArcGIS\arcgis-android-sdk-v10.2\libs)



You will end up with 3 folders under the “additional\libs” folder



Now it is important to note that as we have 3 folders here Esri support 3 CPU architectures ie Arm, Arm v7 and x86. The .SO files that are required to support each architecture are large (>15mb). This will make the compiled wrapper library very large (>40mb). You should delete the architectures you aren't going to support in your app (eg x86). Normally during development I only leave Arm v7 to keep the wrapper size down.

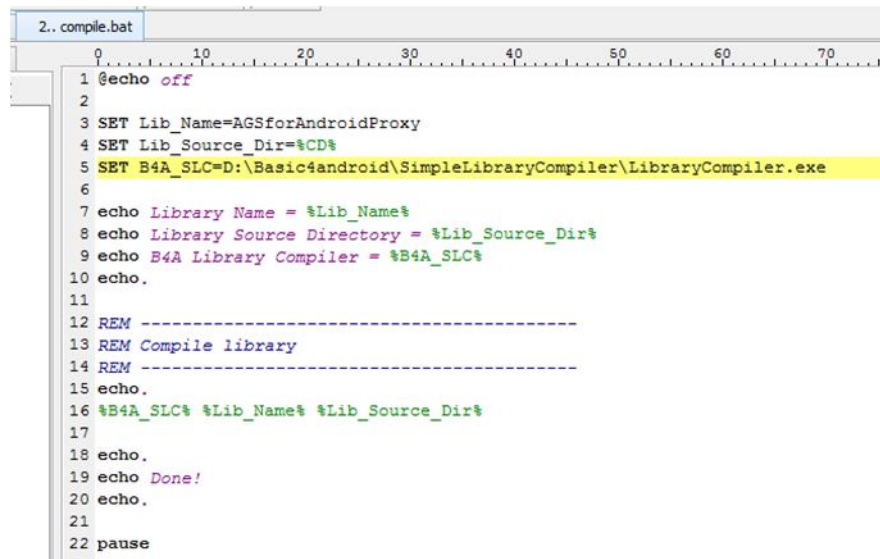


## Step 6

We are almost ready to compile the library. Browse to the root of the wrapper source – in my case that's C:\ArcGIS\AGSforAndroidProxy-master.

There is a batch file called "compile.bat". Open this file in a suitable text editor (don't double click it) as Github seems to mess up the formatting of the file (so Notepad won't work effectively).

The only change you need to make is the path to the SLC tool. As per the screenshot below mine is located in D:\Basic4android\SimpleLibraryCompiler\LibraryCompiler.exe – update this line to reflect your setup.



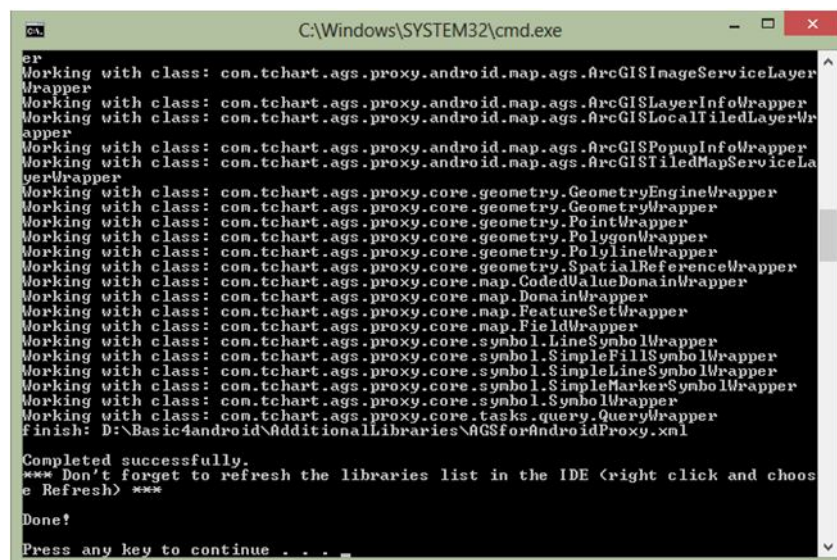
```
2.. compile.bat
1 @echo off
2
3 SET Lib_Name=AGSforAndroidProxy
4 SET Lib_Source_Dir=%CD%
5 SET B4A_SLC=D:\Basic4android\SimpleLibraryCompiler\LibraryCompiler.exe
6
7 echo Library Name = %Lib_Name%
8 echo Library Source Directory = %Lib_Source_Dir%
9 echo B4A Library Compiler = %B4A_SLC%
10 echo.
11
12 REM -----
13 REM Compile library
14 REM -----
15 echo.
16 %B4A_SLC% %Lib_Name% %Lib_Source_Dir%
17
18 echo.
19 echo Done!
20 echo.
21
22 pause
```

Save the file once updated. Close the file.

## Step 7

Now double click the "compile.bat" file to compile the library.

The library should compile successfully.



```
C:\Windows\SYSTEM32\cmd.exe
er
Working with class: con.tchart.ags.proxy.android.map.ags.ArcGISImageServiceLayer
Wrapper
Working with class: con.tchart.ags.proxy.android.map.ags.ArcGISLayerInfoWrapper
Working with class: con.tchart.ags.proxy.android.map.ags.ArcGISLocalFileLayerWr
apper
Working with class: con.tchart.ags.proxy.android.map.ags.ArcGISPopupInfoWrapper
Working with class: con.tchart.ags.proxy.android.map.ags.ArcGISStyledMapServiceLa
yerWrapper
Working with class: con.tchart.ags.proxy.core.geometry.GeometryEngineWrapper
Working with class: con.tchart.ags.proxy.core.geometry.GeometryWrapper
Working with class: con.tchart.ags.proxy.core.geometry.PointWrapper
Working with class: con.tchart.ags.proxy.core.geometry.PolygonWrapper
Working with class: con.tchart.ags.proxy.core.geometry.PolylineWrapper
Working with class: con.tchart.ags.proxy.core.geometry.SpatialReferenceWrapper
Working with class: con.tchart.ags.proxy.core.map.CodedValueDomainWrapper
Working with class: con.tchart.ags.proxy.core.map.DomainWrapper
Working with class: con.tchart.ags.proxy.core.map.FeatureSetWrapper
Working with class: con.tchart.ags.proxy.core.map.FieldWrapper
Working with class: con.tchart.ags.proxy.core.symbol.LineSymbolWrapper
Working with class: con.tchart.ags.proxy.core.symbol.SimpleFillSymbolWrapper
Working with class: con.tchart.ags.proxy.core.symbol.SimpleLineSymbolWrapper
Working with class: con.tchart.ags.proxy.core.symbol.SimpleMarkerSymbolWrapper
Working with class: con.tchart.ags.proxy.core.symbol.SymbolWrapper
Working with class: con.tchart.ags.proxy.core.tasks.query.QueryWrapper
finish: D:\Basic4android\AdditionalLibraries\AGSforAndroidProxy.xml

Completed successfully.
*** Don't forget to refresh the libraries list in the IDE (right click and choos
e Refresh) ***

Done!

Press any key to continue . . .
```

## Step 8

Open up B4A and make sure the library has loaded.

