Update to DM 5, IK1058, v14.1

Android Maps API v2

The Google Maps API in Android have evolved to v2 (december 2012). The old Maps tutorials are therefore deprecated. It is possible to use the old API (get map keys) until march 3rd, 2013, after that only already issued keys will work.

For an introduction to the new maps API read the information here: https://developers.google.com/maps/documentation/android/. The Google Maps Android API reference is available on developer.android.com, but for more information about adding maps to your app, visit: developers.google.com/maps.

Basically Google have put Maps and some other API libraries in the Google Play Services App which is a automatic download on all Android units from Android 2.2 and newer. Google Play services gives you the freedom to use the newest APIs for popular Google services without worrying about device support.

Read more here: http://developer.android.com/google/play-services/index.html

Note!
Google Play services is since a while back supported on the emulator if you in the SDK Manager install a system image that includes the Google APIs (I recommend API-19 or newer). If you got a compatible Intel CPU I recommend the following article to read: https://software.intel.com/en-us/blogs/2014/03/06/now-available-android-sdk-x86-system-image-with-google-apis.

My tutorial below is for reference, I recommend Googles own tutorial now: https://developers.google.com/maps/documentation/android/start

My Android Maps API v2 tutorial

You can use Lars Vogels maps v2 tutorial as a complement guide at:
http://www.vogella.com/articles/AndroidGoogleMaps/article.html

Google Maps v2 uses fragments which one in short can say is a sub-Activity in an Activity. Lars Vogel have a fragments tutorial as well here:
http://www.vogella.com/articles/AndroidFragments/article.html

Step 1

To use Google Maps, set up the Google Play services SDK. Follow the steps here: http://developer.android.com/google/play-services/setup.html

- **To install the Google Play services SDK for development** (3 steps). Remember to check the "Copy projects ..." check box in Eclipse.
- **To set up a project to use the Google Play services SDK** (2 steps). You need to return to these instructions later on.
- **Ensuring Devices Have the Google Play services APK** (several steps). Ensure that it works for you following/reading the instructions. You may need to return to these instructions later on as well.

Step 2

Follow the **Getting Started** guide here: https://developers.google.com/maps/documentation/android/start. This guide with its links are actually everything you need to get started.

Be warned though, it is significantly harder and takes some time and reading to get it to work compared to the old maps API! When you have set up everything correct it is however a much easier API to use.

The first step (step 1) Google Play services SDK should already have been reviewed first.

Obtain a v2 Maps API key from the API console: https://code.google.com/apis/console. Note that old map keys from API v1 will not work.

To apply for a maps key (you need a Google Account):

- Locate the SDK debug certificate in the default folder. In Vista/7 the folder is C:\Users\<username>\.android. The filename of the debug keystore is debug.keystore.
- Copy the debug.keystore file to a folder named C:_local\Android\.
- Open a command window (cmd.exe) and navigate to C:\Program Files\Java\<JDK_version_number to locate the keytool.exe.
- Execute the following to extract your certificate information:

keytool.exe -list -v -alias androiddebugkey -keystore "C:_local\Android\debug.keystore" -storepass android -keypass android > C:_local\Android\my_cert.txt

- In the file my_cert.txt you should now have information about your certificate.
- Create an API project on: https://code.google.com/apis/console/. The project name can be changed later at any time.
- Select the API Access page and press the “Create new Android key...” button.
- Copy and paste the SHA1 certificate fingerprint from your certificate (my_cert.txt) and an additional domain name.

Step 3

- Create an empty new Android Application in Eclipse.
- Copy your API key and put it in the AndroidManifest.xml file.
- Then specify the other settings in the Application Manifest, see: https://developers.google.com/maps/documentation/android/start > Specify app settings in the application manifest
- Now create your very first simple example maps application, see: https://developers.google.com/maps/documentation/android/start > Add a map
- Before you try the application make sure to review the two tasks in step 1 again (To set up a project to use the Google Play services SDK). Add the google-play-services_lib as a library reference, Also note that you may have to build the library first.

Problems

- When running first time check with LogCat for errors. If everything works but no map is showing you may have: Google Maps Android API(#number here#): Authorization failure. In that case ensure that you used the correct key at: https://code.google.com/apis/console/.
- If using the emulator the resources may get exhausted in the emulator (you get errors you cannot explain or find the cure for). In this case just restart the emulator. I think it may depend on the OpenGL ES subsystem.

Sample code is also bundled with the SDK, check here for more info: https://developers.google.com/maps/documentation/android/intro#sample_code

Running maps on the emulator

It is possible to run maps v2 on the emulator if you install a system image that includes the Google APIs, ie. Google Play services etc.

My example project can be found in: http://users.du.se/~hjo/cs/common/androidexamples/ and the file Mapsproject.7z. Note that I have removed my API key.