

Titanium AAR based Modules

prerequisites

- Bring SDK directory under git control and commit original state
Create and switch to a branch optionally
- Install appc-aar-transform npm in Titanium SDK node_modules
- Install base-builder.js in SDK/android/cli/
- Exchange _build.js and _buildModule.js in SDK/android/cli/
commands with provided ones, which support AARs in modules

Module Setup

- Create a folder aar in the android folder
- Create a folder aar-jars in the android folder
 - this will be used for AAR based jars, if required
 - separate folder for those, to make house keeping easier
- Put the required .aar files in the aar folder
- Extract the jar file from the TM SDK and put it into the aar-jars folder
- Configure the Build Path to have the aar-jars on it as well

Pitfalls

- Beware of additional jar files in aar modules.
 - Those will be handled automatically in the build process of app
- Beware of transitive dependencies of the aar libs and their jars
- Beware of conflicting jars, contained in aar and provided in some other version by Titanium platform itself (e.g. org.apache.http.legacy)
 - currently, this will be handled by a blacklist of jars to be filtered