# 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

# AARs required for TM

- crashlytics-2.5.5.aar
- hologram-1.0.2.aar
- stickylistheaders-2.7.0.aar
- util-1.0.0.aar
- design-23.4.0.aar
- sacklist-1.0.3.aar
- ticketmaster_SDK-release-2.6.3.aar
- util-otto-1.0.0.aar

# Jars required for TM

- android-async-http-1.4.5.jar
- gson-2.4.jar
- okhttp-urlconnection-2.4.0.jar
- otto-1.3.6.jar
- volley_library-1.0.18.jar
- cwac-merge-1.1.1.jar
- okhttp-2.4.0.jar
- okio-1.4.0.jar
- picasso-2.5.2.jar
- zxing_core-3.2.1.jar