Introduction

Sometimes at the end of developing a mobile application, the APK file size is so large (over 50 MB) that it becomes a problem. Here are some methods for reducing the APK file size so that your app can be download more quickly, and require less of your customers' data allotment and memory.

Remove Debug Statements

Debug statement are useful during the development process, but serve no function once the app is completed. Remove all debus statements before deploying your app.

In order to make removing the debug statements easier, enclose all debug functionality into a conditional block, as in the following example:

```
 public static final boolean isDebug = false;
public final String TAG= "MyLogger";
...
public void MyLogger(String statement){
         if (isDebug) {
                    Log.v(TAG, Statement);
         }


}
...
```

Code 1: Condition Debug Block

Media Formats

Most applications use images, audio, and/or video. Choosing a compact format for each of these can reduce the APK file size significantly. The following formats are good choices:

- Images: PNG or JPEG.
- PNG is a compact but lossless format, meaning there is no loss of image quality. It is ideal for textures and artwork.
- JPEG images are best for photographs or similar graphics.
- Audio: AAC (Advanced Audio Coding) for all audio resources. AAC provides better quality than MP3 or OGG Voribs.
- Video: H264 AVC. It provides high quality video in relatively low bit rates than other video formats.

Open Source Utilities to Optimize PNG Images

There are free tools such as OptiPNG or PNGCrush for optimizing PNG image files. These tools are open-source and use command line utilities for optimizing PNG images. They can compress the image by using various combinations of algorithms, such as changing the bit depth, replacing unwanted chucks of data with text, delta filters, and so forth to provide the smallest compressed output.

9patch A 9patch PNG is a special format PNG image for specific circumstances. It defines nine segments of the image, four corner segments, four edges, and the center.

It is especially useful when designing buttons instead of regular bitmap images.

You can find this utility as "draw9patch.bat" in the Android SDK under the \tools directory. For more information on how to run the batch file, see:

http://developer.android.com/guide/developing/tools/draw9patch.html

Remove Unused Resources

During development you typically add a lot of resources such as files, layouts, or drawables. As you go back and made changes and improvements, some of these resources are no longer used, and get left in your code. To detect such unused resources and remove them from your APK file, use the android-unused-resources tool. It scans your project and identifies unused resources. Removing them minimizes the build time and reduces the APK file size.

Avoid Duplication

Check for duplicate functionality and assets that seem to increase the file size of your app. Applying an inheritance or bride pattern into the project can reduce the file size significantly, and makes the application code reusable.

Obfuscation

Applying an obfuscation tool such as Proguard has the following benefits:

- Protects your intellectual property.
- Reduces the APK file size.
- Improves the run time performance.

To configure obfuscation process for APK creation visit here.

Ref:http://developer.samsung.com/android/technical-docs/How-To-Reduce-APK-File-Size-in-Android