

Zipalign: an Easy Optimization

The Android SDK includes a tool called [zipalign](#) that optimizes the way an application is packaged. Running zipalign against your application enables Android to interact it more efficiently at run time and thus has the potential to make it and the overall system run faster. We strongly encourage you to use [zipalign](#) on both new and already published applications and to make the optimized version available — even if your application targets a previous version of Android. This article describes how [zipalign](#) helps performance and how to use it to optimize your app.

In Android, data files stored in each application's apk are accessed by multiple processes: the installer reads the manifest to handle the permissions associated with that application; the Home application reads resources to get the application's name and icon; the system server reads resources for a variety of reasons (e.g. to display that application's notifications); and last but not least, the resource files are obviously used by the application itself.

The resource-handling code in Android can efficiently access resources when they're aligned on 4-byte boundaries by memory-mapping them. But for resources that are not aligned (that is, when [zipalign](#) hasn't been run on an apk), it has to fall back to explicitly reading them — which is slower and consumes additional memory.

For an application developer, this fallback mechanism is very convenient. It provides a lot of flexibility by allowing for several different development methods, including those that don't include aligning resources as part of their normal flow.

Unfortunately, for users the situation is reversed — reading resources from unaligned apks is slow and takes a lot of memory. In the best case, the only visible result is that both the Home application and the unaligned application launch slower than they otherwise should. In the worst case, installing several applications with unaligned resources increases memory pressure, thus causing the system to thrash around by having to constantly start and kill processes. The user ends up with a slow device with a poor battery life.

Luckily, it's very easy for you to align the resources in your application:

- Using ADT:
 - The ADT plugin for Eclipse (starting from version 0.9.3) will automatically align release application packages if the export wizard is used to create them. To use the wizard, right click the project and choose "Android Tools" > "Export Signed Application Package..." It can also be accessed from the first page of the [AndroidManifest.xml](#) editor.
- Using Ant:
 - The *Ant* build script (starting from Android 1.6) can align application packages. Targets for older versions of the Android platform are not aligned by the *Ant* build script and need to be manually aligned.
 - Starting from the Android 1.6 SDK, Ant aligns and signs packages automatically, when building in debug mode.
 - In release mode, Ant aligns packages only if it has enough information to sign the packages, since aligning has to happen after signing. In order to be able to sign packages, and therefore to align them, *Ant* needs to know the location of the keystore and the name of the key in [build.properties](#). The name of the properties are [key.store](#) and [key.alias](#) respectively. If those properties are present, the signing tool will prompt to enter the store/key passwords during the build, and the script will sign and then align the apk file. If the properties are missing, the release package will not be signed, and therefore will not get aligned either.
- Manually:
 - In order to manually align a package, [zipalign](#) is in the [tools/](#) folder of Android 1.6 and later SDKs. You can use it to align application packages targeting any version of Android. You should run it only after signing the apk file, using the following command:

```
zipalign -v 4 source.apk destination.apk
```

- Verifying alignment:
 - The following command verifies that a package is aligned:

```
zipalign -c -v 4 application.apk
```

We encourage you manually run `zipalign` on your currently published applications and to make the newly aligned versions available to users. Also, don't forget to align any new applications going forward!

[← Back to Articles](#)

[↑ Go to top](#)

Except as noted, this content is licensed under [Creative Commons Attribution 2.5](#). For details and restrictions, see the [Content License](#).

[Site Terms of Service](#) - [Privacy Policy](#) - [Brand Guidelines](#)