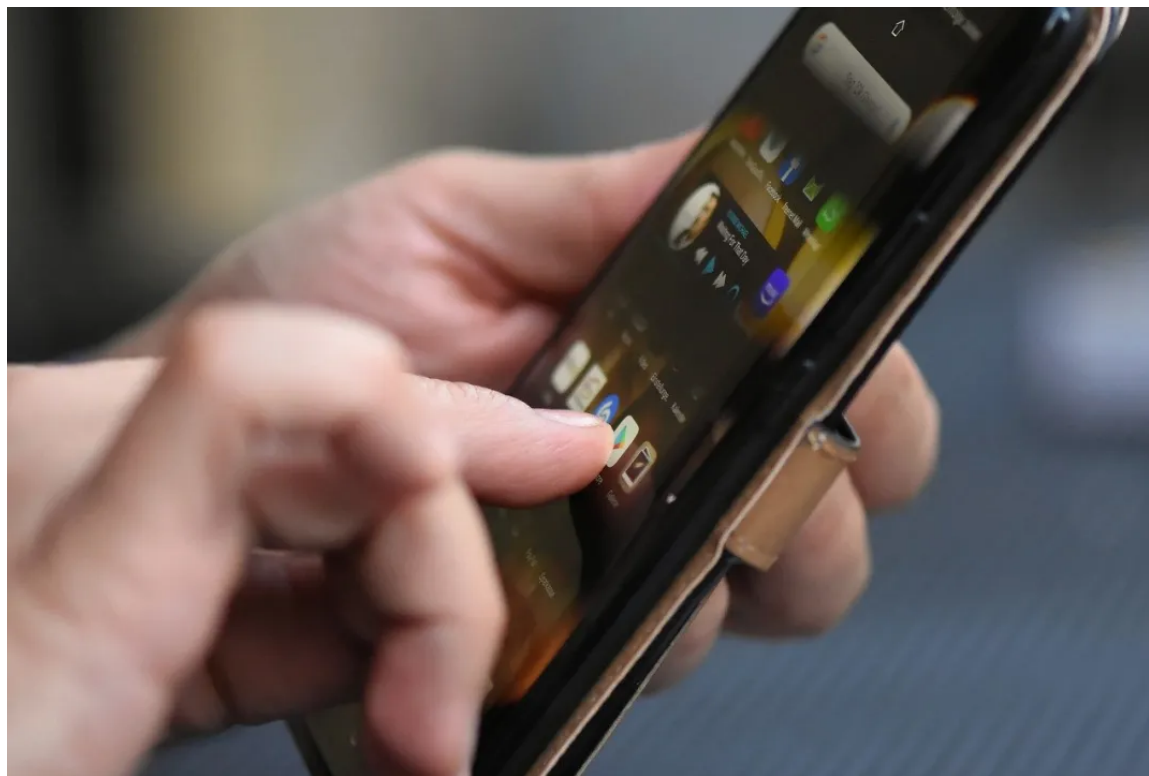


一個時代即將終結！ 安卓應用告別APK格式

機器學習算法與Python學習 前天

點擊[機器學習算法與Python學習](#)，選擇加星標

精彩內容不迷路



出處：電腦報（ID：CQCPCW）

編輯：張毅

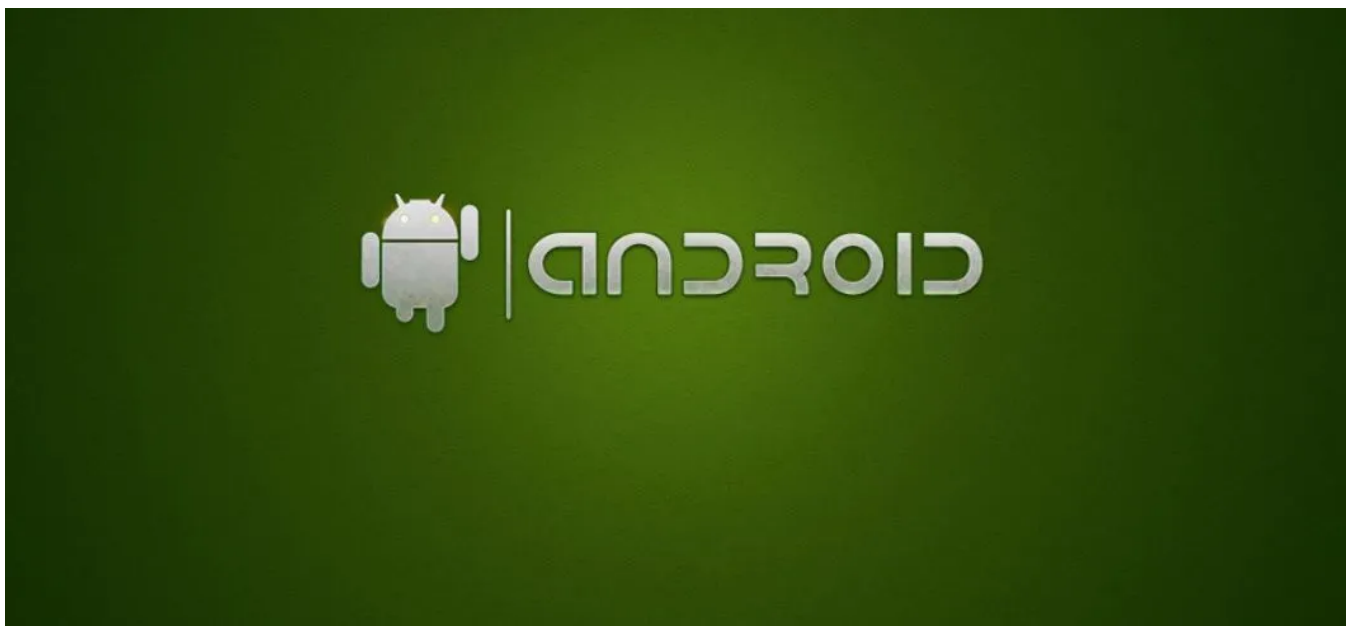
我們知道，Android應用是以APK格式呈現。從接觸安卓系統開始，APK就一直陪伴著我們，可現在，屬於APK的時代恐怕真得要過去了.....



因安卓而被熟知的APK格式

APK全稱Android application package，意為“Android應用程序包”，是Android操作系統使用的一種應用程序包文件格式，用於分發和安裝移動應用及中間件。

一個Android應用程序的代碼想要在Android設備上運行，必須先進行編譯，然後被打包成為一個被Android系統所能識別的文件才可以被運行，而這種能被Android系統識別並運行的文件格式便是“APK”。



APK文件其实是zip格式，但后缀名被修改为apk，通过UnZip解压后，可以看到Dex文件，Dex是DalvikVM executes的简称，即Android Dalvik执行程序，并非Java ME的字节码而是Dalvik字节码。Android在运行一个程序时首先需要UnZip，然后类似Symbian那样直接，但不同于Windows mobile中的PE文件，程序的保密性和可靠性不是很高，通过dexdump命令可以反编译它，但这种架构符合发展规律，微软的WindowsGadgets(WPF)也采用了这种架构方式。

在Android平台中，dalvikvm的执行文件被打包为apk格式，最终运行时加载器会先解压，然后获取编译后的androidmanifest.xml文件中的permission声明对安全访问的限制，要知道仍然存在很多安全限制，但将apk文件传到/system/app文件夹下会发现执行是不受限制的。也许我们平时安装都不会选用这个文件夹，但在androidrom中，系统的apk文件默认会放入这个文件夹，它们拥有root权限。

如今这一格式要被取代了，**据Android Authority报道，安卓宣布AAB格式将取代Android APK。**



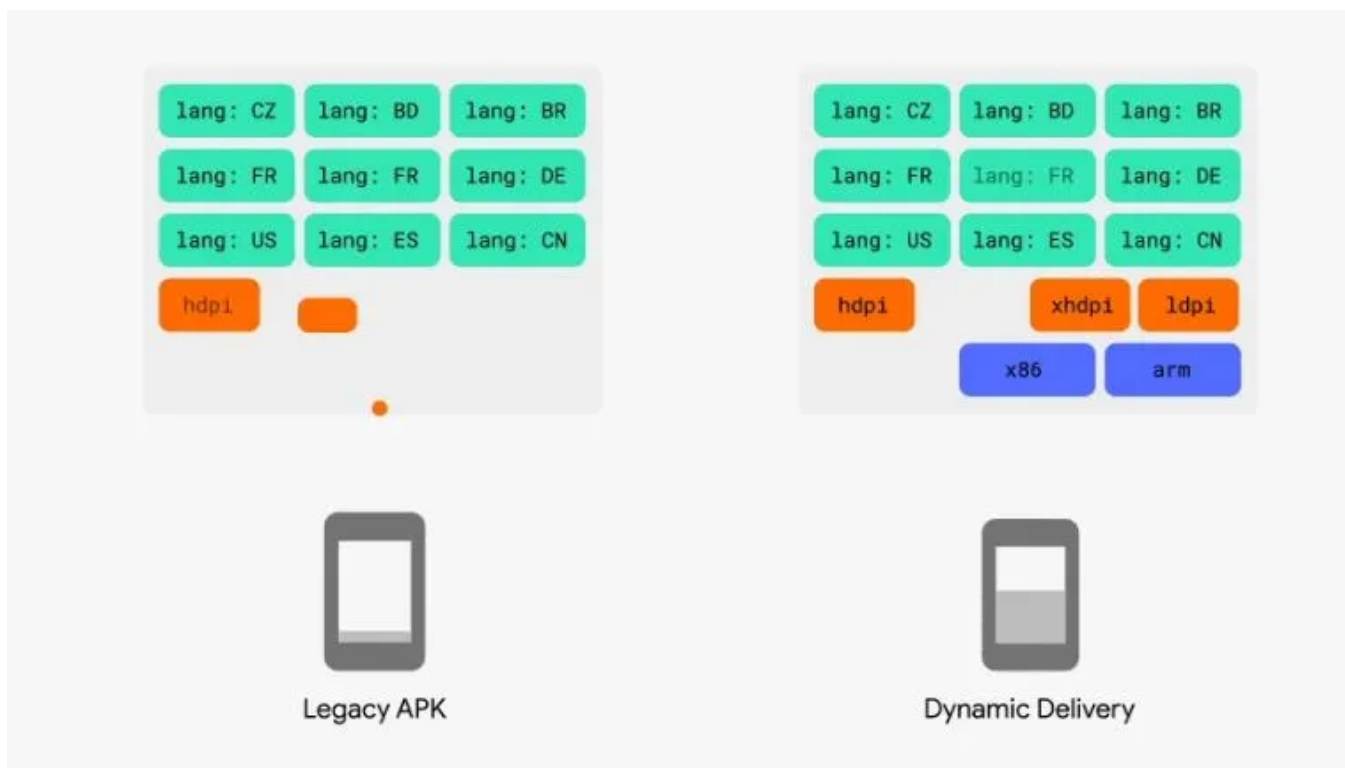
据悉，安卓早在2018年推出了AAB新格式（AAB全称为“Android App Bundles”），安卓声称这种新格式将使应用程序文件更小。

目前在Google Play数百万个应用程序中，已经有数千个应用程序率先跟进了AAB格式。

现在安卓宣布AAB正式取代Android APK，从今年8月份开始，所有提交到Google Play商店的新应用必须采用AAB格式。



Android App Bundle 是一种发布格式 —— 精确地说，是一个带有 .aab 扩展名的 zip 文件。它包含应用支持的所有设备的代码和资源，例如 DEX 文件、本地代码库、清单文件、各种资源文件等。一旦上传用于发布，Google Play 就会处理 APK 的签名和生成，这个过程称为动态交付 (Dynamic Delivery)。动态交付的用途是，根据用户的设备配置为用户生成优化的 APK。那么这究竟是怎么做到的？

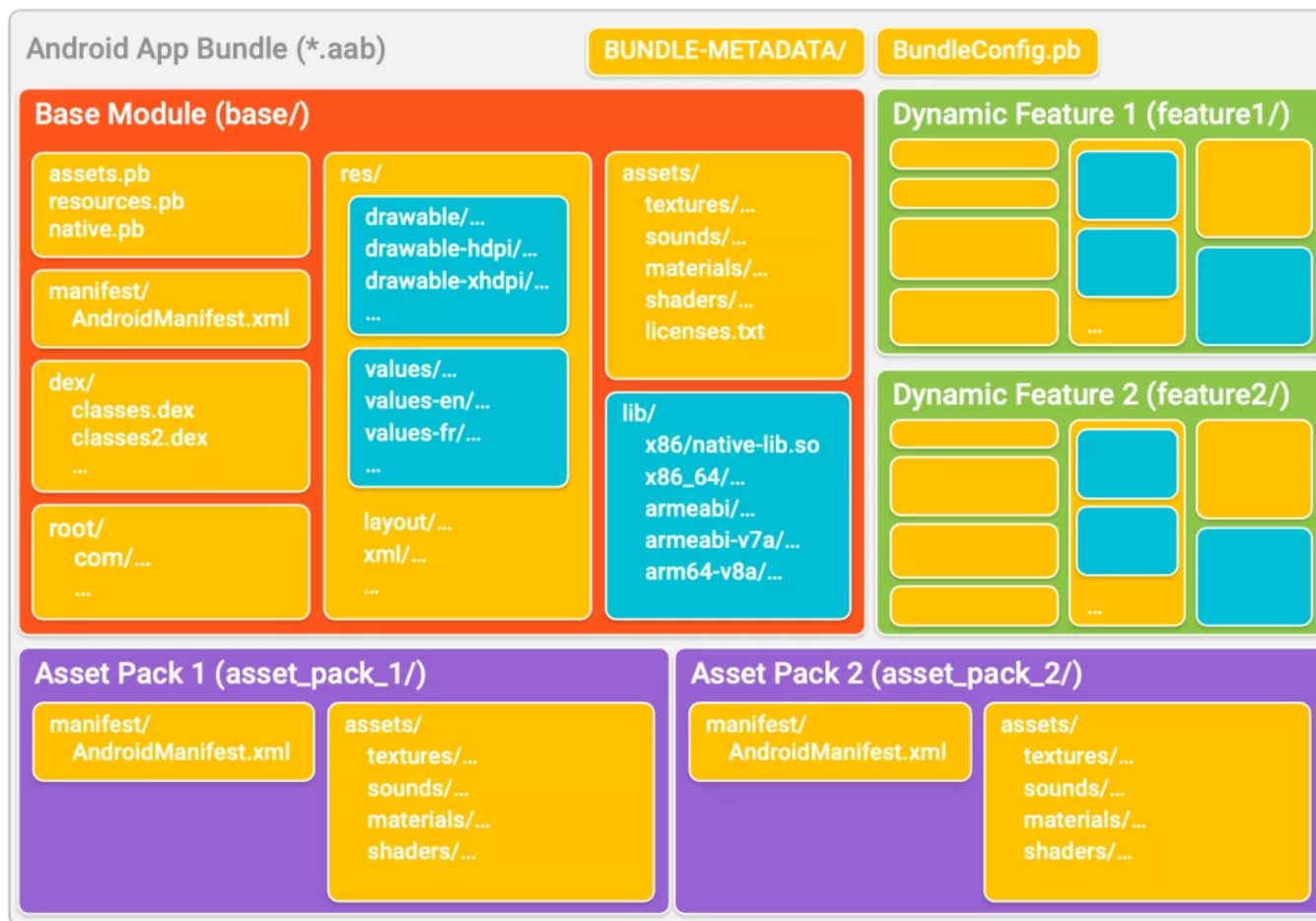


分拆 APK (在 Lollipop 中引入) 是从给定的 Android App Bundle 生成的，其行为与单个 APK 无异。一个典型的应用可以获得一个基础 APK 和多个配置 APK。而且，如果应用具有动态功能，用户也可以获得动态功能 APK 及其配置 APK。基本 APK 包含所有设备配置共有的文件，如清单文件。配置 APK 是为您生成的，每个之中都包含有特定设备配置的相关资源：语言、CPU 架构或屏幕像素密度。因此，用户将获得标准的基本 APK (与所有其他设备一样) 以及仅包含用户设备相关资源的配置 APK。

这意味着，如果我使用的是一台 Android One 手机 (小米 A1) 而且我设置的主要语言是英文，则这台手机将获得基础 APK 以及支持英文、arm64 CPU 架构和 xhdpi 屏幕分辨率的配置 APK。更棒的是，当设备配置 (如语言) 发生变化时，Google Play 会检测到它，并下载该语言的配置 APK。为了进一步降低 APK 大小，我们正计划推出基于纹理压缩格式、图形 API 和新平台功能的分发方案。

动态功能 APK 包含用户首次安装应用时不需要的应用功能代码和资源。开发者可以把这些用途或功能添加到他们的应用中，Google Play 会按需提供这些动态功能模块，而不是在安装时统一添加，从而进一步减少应用下载体积。这也很好理解：我们有必要将那些消耗空间且在安装时根本用

不着的功能，以及那些很少用得着的功能，都打包进动态功能模块中，这将显著减少用户安装时的文件下载量。



安装早于 Android Lollipop 版本的设备也可以享受安装文件体积缩小的福利，但其 APK 中将包含所有语言。

在如今，很显然构建一个统一的臃肿的 APK 的做法已经过时了。Android App Bundle 代表着 Android 应用交付的未来，接下来我们就可以看到如何构建这样的一个安装包。



了解ABB是什么和它的工作原理以后，人们不禁会好奇ABB究竟能让APK程序占用的空间小多少？

目前，国内的开发者将所有资源统一放在单个 APK 中，这样就会导致 APK 特别庞大，而AAB在压缩APK体积方面具有优势。

而为了缩小体积，部分开发者会有意缩减 APK 中的 ABI 目录。例如，将 arm64-v8a 的 SO 从 APK 中去除，只留下 armeabi-v7a 的 SO。但这种做法使得64位 CPU 的手机无法发挥出其64位的运算优势，降低程序运行速度。

Split APKs 是 Android 5.0 开始提供的多 APK 构建机制，借助 Split APKs 可以将一个 APK 基于 ABI、屏幕密度和 CPU 架构拆分成多个 APK，这样可以有效减少单个 APK 体积。当用户下载应用程序安装包时，Google Play 会自动识别用户的语言和 CPU 架构，自动将对应平台 SO 和资源的 APK 下发给用户。



这样说是不是还是有些云里雾里的感觉，其实Android App Bundle 的早期采用者已经发现，动态交付显著减小了他们的应用体积。一些开发者甚至可以将他们的 APK 大小减半，而一些知名 app 使用 App Bundle 减小应用体积的数据也有放出来。

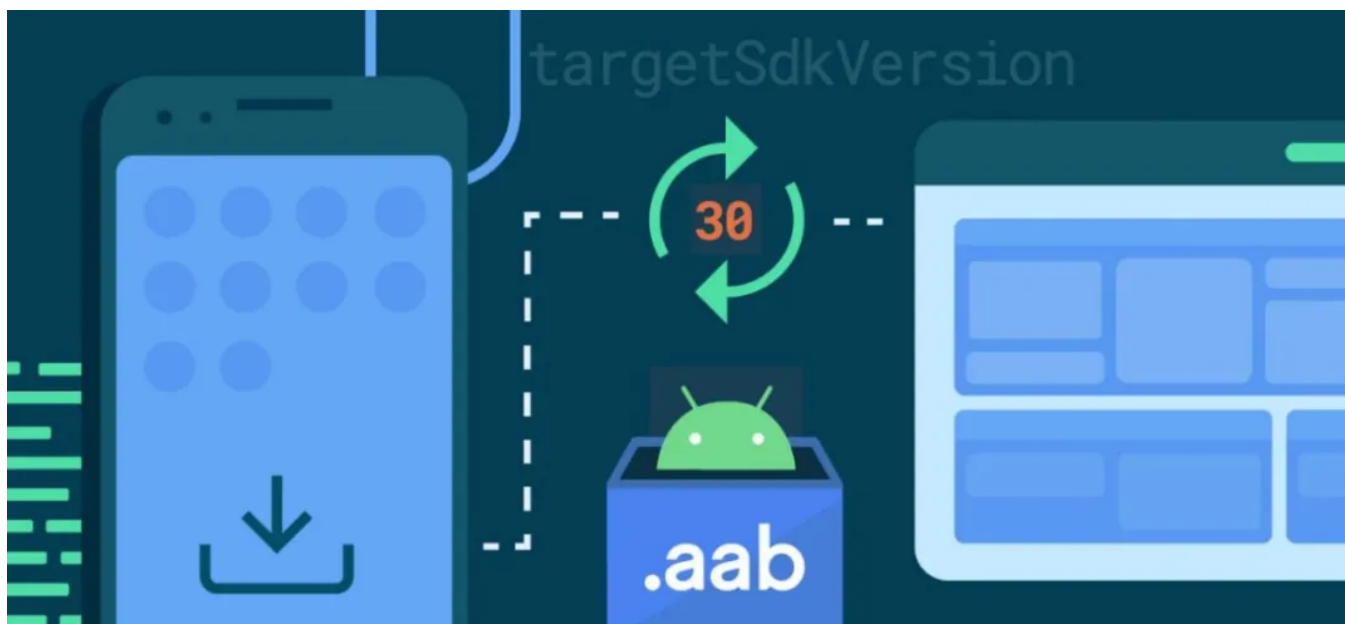
除了压缩体积外，ABB在“防二次打包”一类安全性上也有所表现，可安卓这一次改动真是技术更迭的推动又或者为了给用户更好的使用体验吗？

想要收拢权限的安卓

.aab 模块引入了 Split APK 概念。简单的来讲，就是在安装前，会自动检测用户的硬件配置，然后以多个 .apk 的形式安装应用。可目前，使用 Split APK 的应用程序，用户是无法直接提取安装的，都需要借助第三方工具来备份安装。这意味着，未来用户在非谷歌应用商店的第三方平台，下载安装应用会越来越困难。

虽然 .aab 模块化特性，极大的提升了开发者的更新维护的便捷性，节省了用户在安装应用的时间和存储空间。就因为 .aab 的存在，随着用户使用设备、所在环境的不同，所安装的应用可能也不尽相同，应用也就是“不完整的”。

并且，通过官方文档，我们发现了：使用 app bundle，开发者就必须加入 Google Play 应用签名计划。签名相当于打上唯一的电子标签，因此，如果应用被以非正常方式提取分享，就可能导致签名改变，最终影响应用运行。除非，开发者自行在第三方平台提供完整的应用安装包。



只能通过指定应用商店下载、应用“不完整”、分享限制。这些重重限制，Android 用户都有受到影响。

再往深处想一下，如果说 .aab 应用格式落实，对 Android 用户而言，只是增加了第三方下载应用的难度。那么对于鸿蒙 OS 而言，这可能是一个巨大的挑战。



目前鸿蒙 OS 的软件大多还是以安卓应用为主。所以如果谷歌全面使用 .aab ，肯定会对鸿蒙产生不利条件。

当然，这样的想法或许是我们多心了，究竟安卓应用这一次改变剑指何处，恐怕还需要时间来验证。



☑ 每日推送精彩干货

☑ 免费赠送各类学习教程

☑ 机器学习、深度学习、算法集中营

☑ 后台回复关键词,近百G学习资料

关键词: 推荐系统、机器学习、Python、爬虫教程

ID: MLPython



长按上方二维码
学习更多AI知识

如果对你有帮助。

請不吝點贊，點在看，謝謝

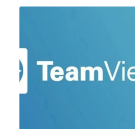
喜歡此內容的人還喜歡

缺貨下，瑞薩MCU開發環境搭建踩坑記

嵌入式ARM

天下苦Teamviewer久矣！GitHub大佬搞定遠程問題

碼個蛋



FunTester分享會第二期視頻回顧

FunTester

