

Android 中的进程与线程

一、基本知识

如果某个应用程序组件是第一次被启动，且这时应用程序也没有其他组件在运行，则 Android 系统会为应用程序创建一个包含单个线程的 linux 进程。默认情况下，同一个应用程序的所有组件都运行在同一个进程和线程里（叫做“main”主线程）。如果组件启动时，已经存在应用程序的进程了（因为应用程序的其它组件已经在运行了），则此组件会在已有的进程和线程中启动运行。不过，可以指定组件运行在其他进程里，也可以为任何进程创建额外的线程。

二、进程的分类(根据进程的重要性, 从高到低)和生命周期

1. 前台进程

1. 用户当前操作所必须的进程。满足任一条件时，进程被视作处于前台：
其中运行着正与用户交互的 Activity（Activity 对象的 `onResume()` 方法已被调用）。
 - 其中运行着被正与用户交互的 activity 绑定的服务 Service。
 - 其中运行着“前台”服务 Service——服务以 `startForeground()` 方式被调用。
 - 其中运行着正在执行生命周期回调方法（`onCreate()`、`onStart()` 或 `onDestroy()`）的服务 Service。
 - 其中运行着正在执行 `onReceive()` 方法的 BroadcastReceiver。
 - 绝大多数情况下不会终止此类进程，当内存不足以维持它们同时运行时——才会被终止

2. 可见进程

- 没有前台组件、但仍会影响用户在屏幕上所见内容的进程
- 其中运行着不在前台的 Activity，但用户仍然可见到此 activity（`onPause()` 方法被调用了）
- 其中运行着被可见（或前台）activity 绑定的服务 Service。
- 可见进程被认为是非常重要的进程，除非无法维持所有前台进程同时运行了，它们是不会被终止的。

3. 服务进程

- 此进程运行着由 `startService()` 方法启动的服务
- 除非内存不足以维持所有前台、可见进程同时运行，系统会保持服务进程的运行。

4. 后台进程

- 包含目前用户不可见 activity（Activity 对象的 `onStop()` 方法已被调用）的进程

- 这些进程对用户体验没有直接的影响，系统可能在任意时间终止它们，以回收内存供前台进程、可见进程及服务进程使用
5. 空进程
- 不含任何活动应用程序组件的进程
 - 保留这种进程的唯一目的就是用作缓存，以改善下次在此进程中运行组件的启动时间。
 - 为了在进程缓存和内核缓存间平衡系统整体资源，系统经常会终止这种进程

三、关于线程

1. 应用程序启动时，系统会为它创建一个名为“main”的主线程，也称为“UI Thread”
2. UI Thread 特点重要，组件的回调方法，事件的分发和回调，UI 的更新都是在线程执行的
3. 一些长时间的工作(如联网)不能在 UI Thread 中执行，只能在分线程(worker Thread)中执行