

Service

1.1 知识解析

Service 是运行在后台的服务，如听音乐时，可利用 Service 让音乐在后台执行，这样就可以关闭执行音乐的 Activity 来做其他事情，当要关闭音乐时，可以在获取 Service 信息后将其停止或是其他动作。

Service 服务必须在 AndroidManifest.xml 中声明。Service 可由 Activity 来启动，启动之后可利用 BroadcastReceiver 来取得相关 Service 消息。

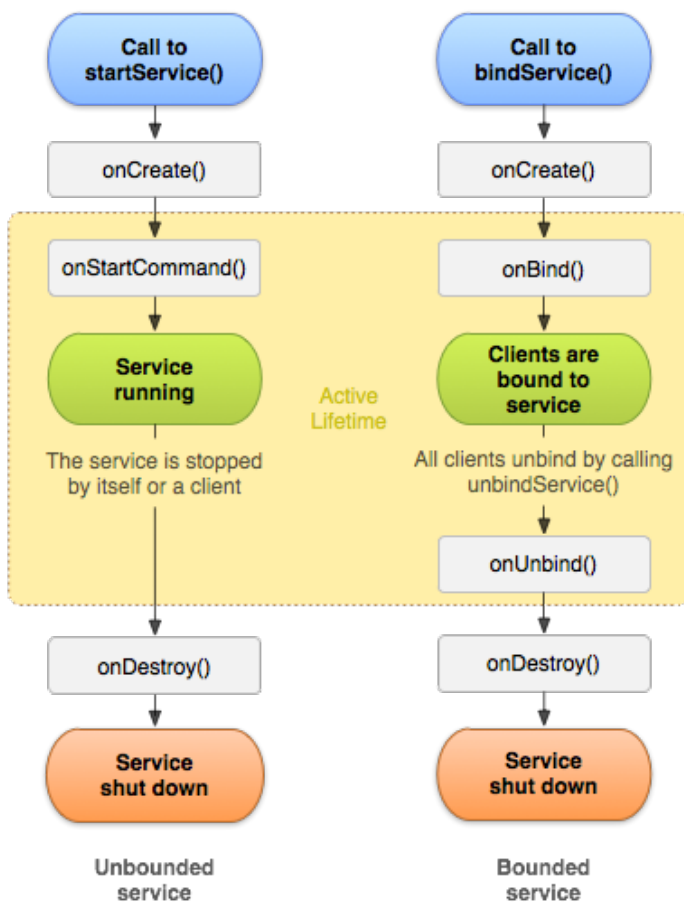
Service 有 2 种类型：

- 本地服务（Local Service）：用于应用程序内部。用于实现应用程序自己的一些耗时任务，比如查询升级信息、下载微博信息、在后台播放音乐等，并不占用应用程序比如 Activity 所属线程，而是单开线程后台执行，这样用户体验比较好。
- 远程服务（Remote Service）：用于 Android 系统内部的应用程序之间。可被其他应用程序复用，比如天气预报服务，其他应用程序不需要再写这样的服务，调用已有的即可。

对于本地服务来说，根据情况又可以分成 2 种：

- 服务（Service）和 Activity 之间没有方法调用交互的情况。这可以通过 startService() 和 stopService() 来控制。
- 服务（Service）和 Activity 之间有方法调用的情况。这可以通过 bindService 和 unbindService() 来控制。

Service 生命周期



服务不能自己运行，需要通过调用 `Context.startService()` 或 `Context.bindService()` 方法启动服务。这两个方法都可以启动 Service，但是它们的使用场合有所不同。使用 `startService()` 方法启用服务，调用者与 Service 之间没有关联，即使调用者退出了，服务仍然运行。使用 `bindService()` 方法启用服务，调用者与 Service 绑定在了一起，调用者一旦退出，服务也就终止。

使用 `bindService`，调用者可以直接调用 Service 中的方法。

如果采用 `Context.startService()` 方法启动服务，在服务未被创建时，系统会先调用服务的 `onCreate()` 方法，接着调用 `onStartCommand()` 方法。如果调用 `startService()` 方法前服务已经被创建，多次调用 `startService()` 方法并不会导致多次创建服务。采用 `startService()` 方法启动的服务，只能调用 `Context.stopService()` 方法

`/Service.stopSelf()` 结束服务，服务结束时会调用 `onDestroy()` 方法。

如果采用 `Context.bindService()` 方法启动服务，在服务未被创建时，系统会先调用服务的 `onCreate()` 方法，接着调用 `onBind()` 方法。这个时候调用者和服务绑定在一起，调用者退出了，系统就会先调用服务的 `onUnbind()` 方法，接着调用 `onDestroy()` 方法。如果调用 `bindService()` 方法前服务已经被绑定，多次调用 `bindService()` 方法并不会导致多次创建服务及绑定（也就是说 `onCreate()` 和 `onBind()` 方法并不会被多次调用）。如果调用者希望与正在绑定的服务解除绑定，可以调用 `unbindService()` 方法，调用该方法也会导致系统调用服务的 `onUnbind()` → `onDestroy()` 方法。

服务常用生命周期回调方法如下：

- `onCreate()` 该方法在服务被创建时调用，该方法只会被调用一次，无论调用多少次 `startService()` 或 `bindService()` 方法，服务也只被创建一次。
- `onDestroy()` 该方法在服务被终止时调用。

与采用 `Context.startService()` 方法启动服务有关的生命周期方法

- `onStartCommand()` 只有采用 `Context.startService()` 方法启动服务时才会回调该方法。该方法在服务开始运行时被调用。多次调用 `startService()` 方法尽管不会多次创建服务，但 `onStartCommand()` 方法会被多次调用。

与采用 `Context.bindService()` 方法启动服务有关的生命周期方法

- `onBind()` 只有采用 `Context.bindService()` 方法启动服务时才会回调该方法。该方法在调用者与服务绑定时被调用，当调用者与服务已经绑定，多次调用 `Context.bindService()` 方法并不会导致该方法被多次调用。
- `onUnbind()` 只有采用 `Context.bindService()` 方法启动服务时才会回调该方法。该方法在调用者与服务解除绑定时（调用 `Context.unbindService()`）被调用。

1.2 功能演示



1.4 职业素质

android中服务是运行在后台的东西，级别与activity差不多。既然说service是运行在后台的服务，那么它就是不可见的，没有界面的东西。你可以启动一个服务Service来播放音乐，或者记录你地理信息位置的改变，或者启动一个服务来运行并一直监听某种动作。

Service 和其他组件一样，都是运行在主线程中，因此不能用它来做耗时的请求或者动作。你可以在服务中开一个线程，在线程中做耗时动作。