

Android 应用程序主要组件

知识解析

一般情况 Android 应用程序是由以下四种组件所组成的：

- 活动(Activity)
- 服务(Service)
- 广播接收器(Broadcast Receiver)
- 内容提供者(Content Provider)

活动(Activity)

一般所指的活动是用户界面。一个应用程序可能有一个或以上的活动存在，每个活动也都会有自己的 View。

所有的活动在系统里由活动堆栈（activity stack）所管理，当一个新的活动被执行后，它将会被放置到堆栈的最顶端，并且变成“running activity”，而先前的活动原则上还是会存在于堆栈中，但它此时不会是在前景的情况，除非新加入的活动离开。

服务(Service)

服务是在背景长时间运行的应用组件，不和用户直接进行互动。

例如：某服务可能在后台播放音乐，而用于在执行其他的操作，或者它通过网络抓取资料或者执行某些计算，将结果提供给活动（Activity）。

广播接收器(BroadcastReceiver)

广播接收器负责接受和响应通知，很多通知源自于系统所发送，例如：时区变换，电池电量不足，或用户改变语言设置。

应用程序也可以发出广播通知，例如：通知其它应用程序，数据已下载完毕，可供使用。 拍照已经完成，可以使用照片等。

应用程序可以拥有任意数量的广播接收器来接收任何的通知。另外也可以启动活动(Activity)去响应接收到的通知,或利用通知管理器(Notification Manager)来通知使用者。

内容提供者(Content Provider)

内容提供者将应用程序数据组合成特定的集合供其它应用程序使用。数据可以是储存在文件、SQLite 数据库,或是其它任何用户可以存取资料的地方。

内容提供者继承於内容提供者基础类别,并实现一组标准的方法,使应用程序可以检索和储存它控制的数据。

应用程序不是直接调用这些实现方法。而是通过内容解析器(Content Resolver)对象调用方法。内容解析器能够通知任何的内容提供者,并可以参与这些内容提供者进程间的管理。

后续的内容中将对这些组件进行详细的介绍。

Activity 生命周期

一个 Activity 基本上有 4 个生命状态:

- Active 或 Running
- Paused
- Stop
- Killed

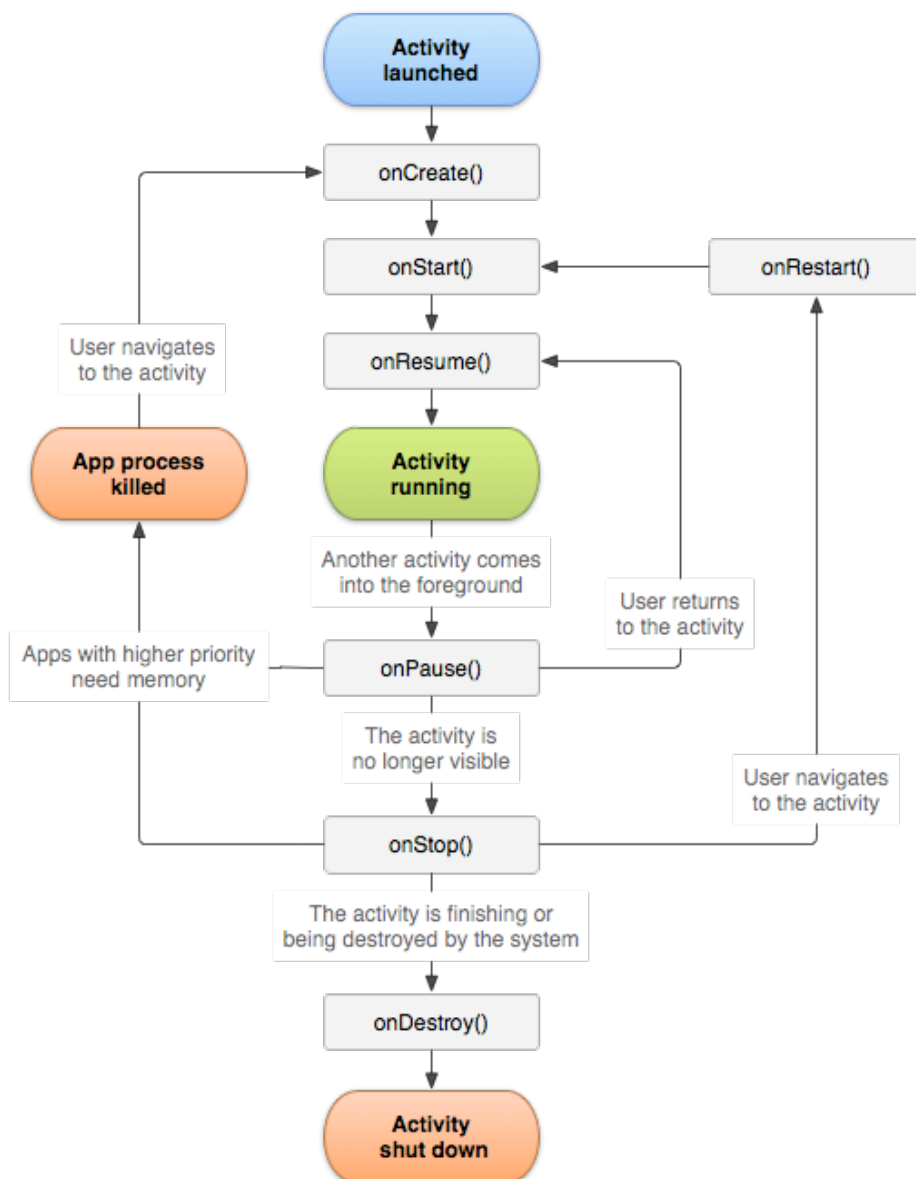
当一个 Activity 处于 Pause 或 Stop 的状态时,系统可以要求 Activity 结束或删除它,当它再度呈现在使用者面前时,要能完整的重新启动及回复先前的状态。

应用程序存在与否并非由应用程序所自行决定,而是由 Android 系统通过运行机制决定。

活动状态间的切换包含了调用下列几种回调方法:

- void onCreate(Bundle savedInstanceState)

- `void onStart()`
- `void onRestart()`
- `void onResume()`
- `void onPause()`
- `void onStop()`
- `void onDestroy()`



上图是 Activity 生命周期对应的各个方法，根据 Activity 的不同状态，可以细分成不同的生命周期：

Activity 的整个生命周期

一个活动的整个生命周期是由 onCreate(Bundle)开始，直到 onDestroy()结束。

一个活动可以把所有的资源设置写在 onCreate 中，直到 onDestroy()时，再释放出来。

activity 的可视生命周期（Visible Lifetime）

一个 Activity 的 VisibleLifetime 则是指在 onStart()到 onStop()之间，称为“可视生命周期”，在这段时间内，用户可以在屏幕上看见 Activity，但这个 Activity 不见得一定在前景 (foreground)跟使用者直接互动。例如，它的上面有一个半透明或者透明的 activity 遮住了它，此时，用户可看见这个 activity，但无法通过点击屏幕和其互动。

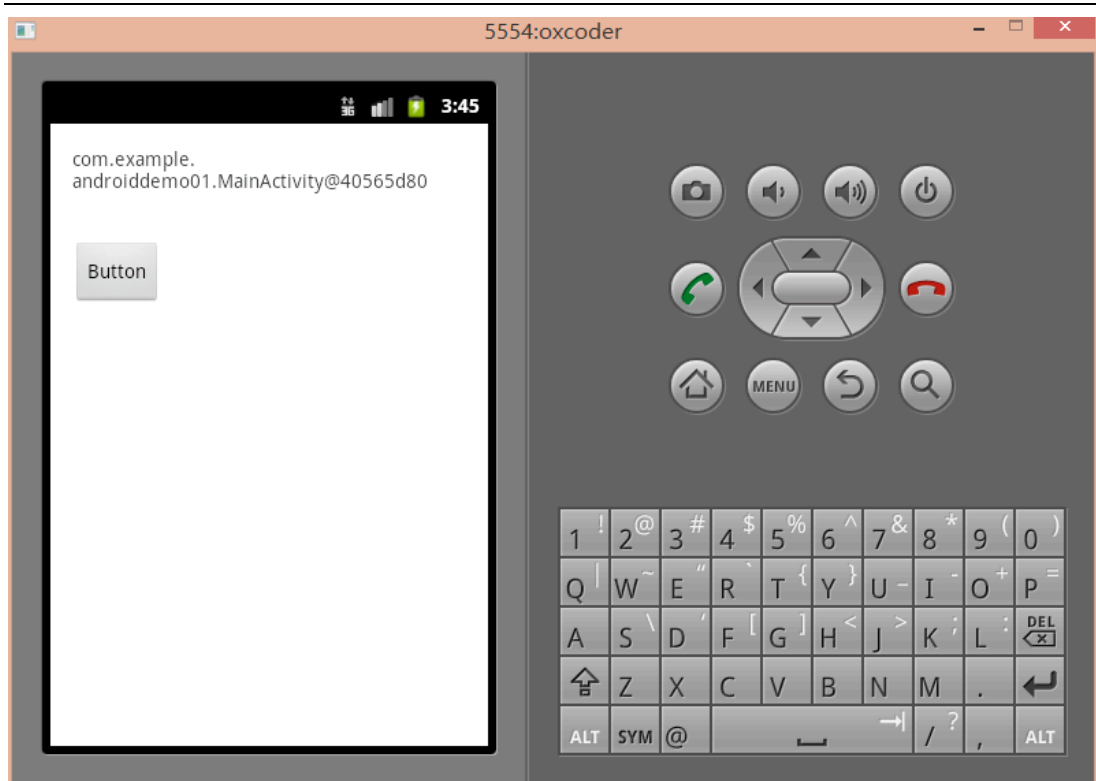
活动的前景生命周期（Foreground Lifetime）：

ForegroundLifetime 则是指 onResume()到 onPause()之间，在这个时期的活动是在所有的活动的前面，并且直接跟使用者进行互动。

一个活动能很频繁的在 Resume 及 Pause 这两个状态切换，所以在 onResume()及 onPause()中实现的程序应尽量精简。

职业素质：四大组件是 Android 的基础，只有把基础打好了，才能建出高楼大厦。任何开发都离不开四大组件。

功能演示



操作实践

```
public class MainActivity extends Activity {  
    TextView tv;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        tv = (TextView) this.findViewById(R.id.textView1);  
        tv.setText(this.toString());  
    }  
  
    public void launch(View view) {  
        Intent intent = new Intent(MainActivity.this, MainActivity.class);  
        this.startActivity(intent);  
    }  
}
```

职业素质

Android 应用通常由一个或多个基本组件组成，所有的应用程序都离不开这里的组件，我们所有的方法都会在这些组件中被调用才得以执行。才会呈现出功能丰富强大的 Android 应用程序。