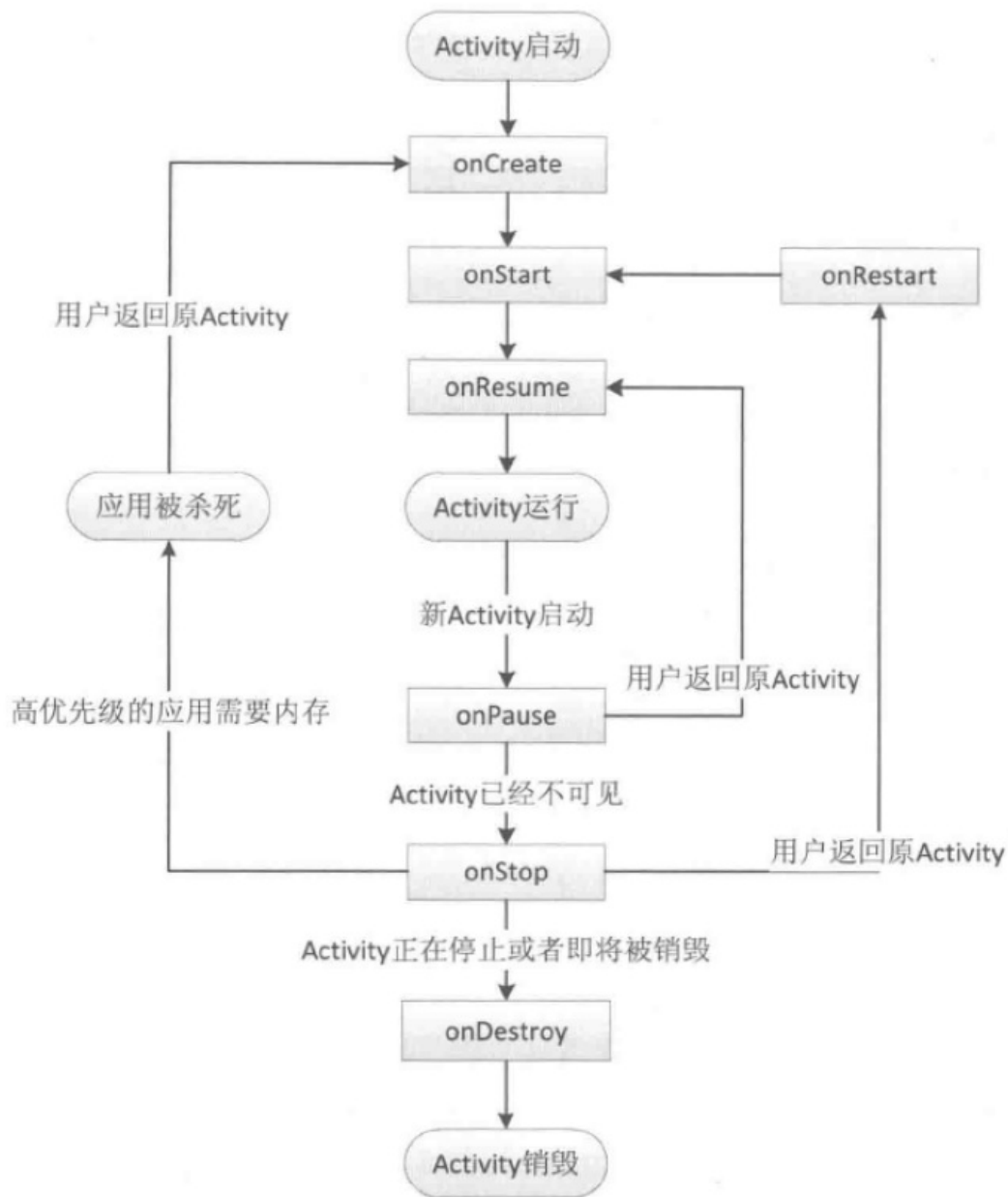


## Activity 生命周期和启动模式

流程图：



关于onRestoreInstanceState方法和onSaveInstanceState方法

系统只会Activity即将被销毁并且有机会重新显示的情况下才会去调用onSaveInstanceState方法，也就是说，如果当前的activity是正常销毁的情况下，onSaveInstanceState方法并不会去调用，因为Activity不可能再次被显示。简单总结，就是系统只会Activity异常终止的情况下，才会调用onSaveInstanceState和onRestoreInstanceState来存储和恢复数据，其他情况不会触发这个过程

Activity启动模式

standard：每次启动一个Activity都会重新创建一个新的实例，并且把这个实例放在当前的task上。

用ApplicationContext去启动standard模式的Activity时会报错，因为standard模式的Activity会默认进入启动它的Activity的任务栈中，由于非Activity的Context并没有对应的任务栈，所以就出问题了。解决方法：在启动Activity的时候，使用FLAG\_ACTIVITY\_NEW\_TASK标记位，这个时候启动的Activity是以singleTask模式启动的

singleTop：每一次启动的时候，判断启动该Activity的Activity任务栈顶有没有对应的Activity，如果有，那么不会创建一个新的Activity，这个时候该Activity的onNewIntent方法会被回调。这个时候oncreate和onStart方法均不会被回调。如果站定没有这个Activity的实例，那么新的Activity会被重建

singleTask：如果启动的Activity不在任意一个栈内，那么系统会新开启一个任务栈并把目标Activity放入栈中，如果目标Activity已经在

栈中，那么会把改Activity上的所有其他Activity清空，并调用onNewIntent方法。

singleInstance：加强版的singleTask。这种模式的Activity只能单独位于一个任务栈中，也就是说，如果ActivityA是singleInstance模式，那么当A启动后，系统会为A单独创建一个任务栈，然后因为栈的复用原理，后续的请求均不会创建新的任务栈了，除非它已经被销毁。

每一个Activity都有一个标识，叫做taskAffinity，主要和singleTask一起使用。