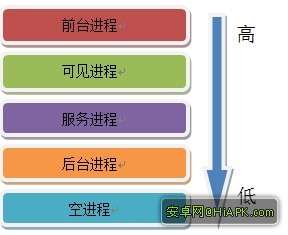
**安卓内存回收机制**

**退出但不关闭：**  
　　这是Android对于Linux的优化。当 Android 应用程序退出时，并不清理其所占用的内存，Linux 内核进程也相应的继续存在，所谓“退出但不关闭”。从而使得用户调用程序时能够在第一时间得到响应。  
　　应用切换到后台是暂停的，完全不耗cpu和电量，只保留了运行状态。如果app需要后台处理事务(放音乐)，那么app进程会开启相应的服务，所以在后台有消耗的只有服务。  
　　占用CPU的应用才耗电，后台的只占内存，所以是不耗电的。  
**默认回收机制：**  
　　当系统内存不足时，系统将激活内存回收过程。为了不因内存回收影响用户体验（如杀死当前的活动进程），  
**回收优先级：**  
　　Android 基于进程中运行的组件及其状态规定了默认的五个回收优先级：  
　　IMPORTANCE\_FOREGROUND:前台进程，目前正在屏幕上显示的进程和一些系统进程。  
　　IMPORTANCE\_VISIBLE:可见进程，可见进程是一些不再前台，但用户依然可见的进程，比如输入法、天气、时钟等。  
　　IMPORTANCE\_SERVICE:服务进程，拨号、邮件存储之类的。  
　　IMPORTANCE\_BACKGROUND:后台进程，启动后被切换到后台的进程。  
　　IMPORTANCE\_EMPTY:没有任何东西在内运行的进程，有些程序，比如BTE，在程序退出后，依然会在进程中驻留一个空进程，这个进程里没有任何数据在运行，作用往往是　　提高该程序下次的启动速度或者记录程序的一些历史信息。  
　　home app:桌面进程，即launcher，保证在多任务切换之后，可以快速返回到home界面而不需重新加载launcher。  
　　系统会对进程的重要性进行评估，并将重要性以“oom\_adj”这个数值表示出来。前台程序的“oom\_adj”值为0，这意味着它不会被系统终止。  
　　可根据“oom\_adj”值的范围来决定进程管理策略，比如可以设定“当内存小于X时，结束“oom\_adj”大于Y的进程”。

  
**触发内存回收的阈值：**  
　　进行内存调度有个阀值，只有低于这个值系统才会按一个列表来关闭用户不需要的东西。当然这个值默认设置得很小，所以你会看到内存老在很少的数值徘徊。  
　　为什么内存少的时候运行大型程序会慢呢，原因是：在内存剩余不多时打开大型程序时会触发系统自身的调进程调度策略，这是十分消耗系统资源的操作，特别是在一个程序频　　繁向系统申请内存的时候。这种情况下系统并不会关闭所有打开的进程，而是选择性关闭，频繁的调度自然会拖慢系统。  
**APP虚拟机：**  
　　安卓Android上的应用是java,当然需要虚拟机,而安卓Android上的应用是带有独立虚拟机的,也就是每开一个应用就会打开一个独立的虚拟机.这样设计的原因是可以避免虚拟机崩溃导致整个系统崩溃,但代价就是需要更多内存。  
**Home键和返回键：**  
　　Home键用来多任务切换，返回键相当于退出应用程序。  
　　按home退出，程序保留状态为后台进程；按返回键退出，程序保留状态为空进程。