



河北师范大学软件学院
Software College of Hebei Normal University



Android 基础开发

第四章 第二节 Activity的生命周期



Java与移动智能设备开发



教学目标

- 掌握在Activity的生命周期
- 掌握Fragment的生命周期



目录



1 Activity的生命周期

2 Activity的应用

3 Fragment的生命周期



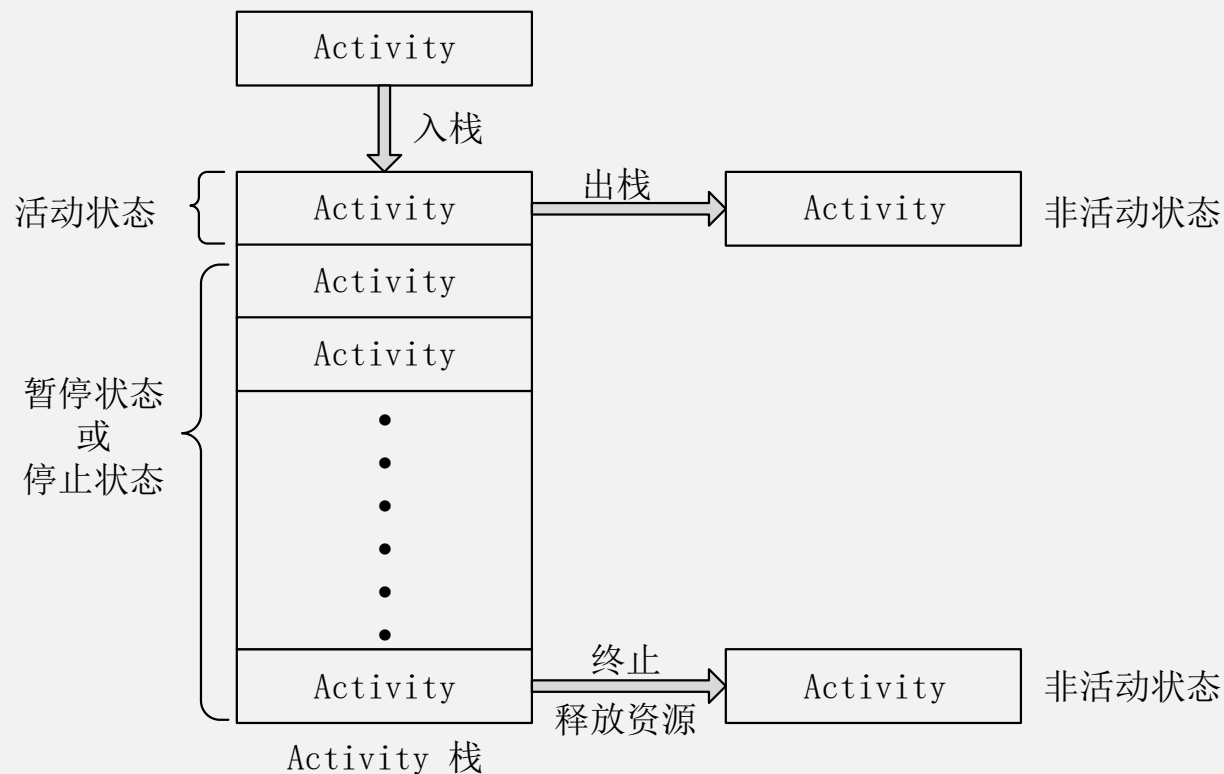
Activity的生命周期

- Activity在运行时会受到一些突发事件的影响（例如在一个Activity中，突然打入一个电话），Android应用需要具备处理这些突发事件的能力，要处理这些事件需要掌握Activity的生命周期。
- 在讨论Activity生命周期之前，补充两点基本内容：
 - Activity活动栈
 - Activity的活动状态



Activity活动栈

- Android应用可能含有多个Activity，如何管理这些Activity之间的先后次序关系，需要借助**Activity活动栈机制**实现。





Activity的活动状态

- 随着Activity的创建、销毁，Activity在内存中有4种状态表现形式：
 - **活动状态**：当前Activity在Activity活动栈中处于最上层，完全能被用户看到，并能够与用户进行交互。
 - 正在运行的屏幕即为此种状态。



Activity的活动状态

- **暂停状态**：当前Activity在界面上被部分遮挡，不再处于用户界面的最上层，不能够与用户进行交互。
 - 若启动一个新的Activity（以对话框形式展示），则原来的Activity就处于暂停状态。
 - 处于暂停状态的Activity仍然保留用户的状态信息，但在系统内存不足时，可能会被系统杀死。



Activity的活动状态

- **停止状态**：Activity在界面上完全不能被用户看到，也就是说这个Activity被其他Activity全部遮挡。
 - 例如：在Activity中，用户按下“Home”键时，原来的Activity就处于停止状态。
 - 处于停止状态的Activity，仍然保留用户状态信息，但当系统内存不足时，会优先杀死该类Activity。



Activity的活动状态

- **非活动状态**：不在以上三种状态中的Activity，处于非活动状态。
 - 被销毁的Activity即处于该类状态。



Activity活动状态之间的切换

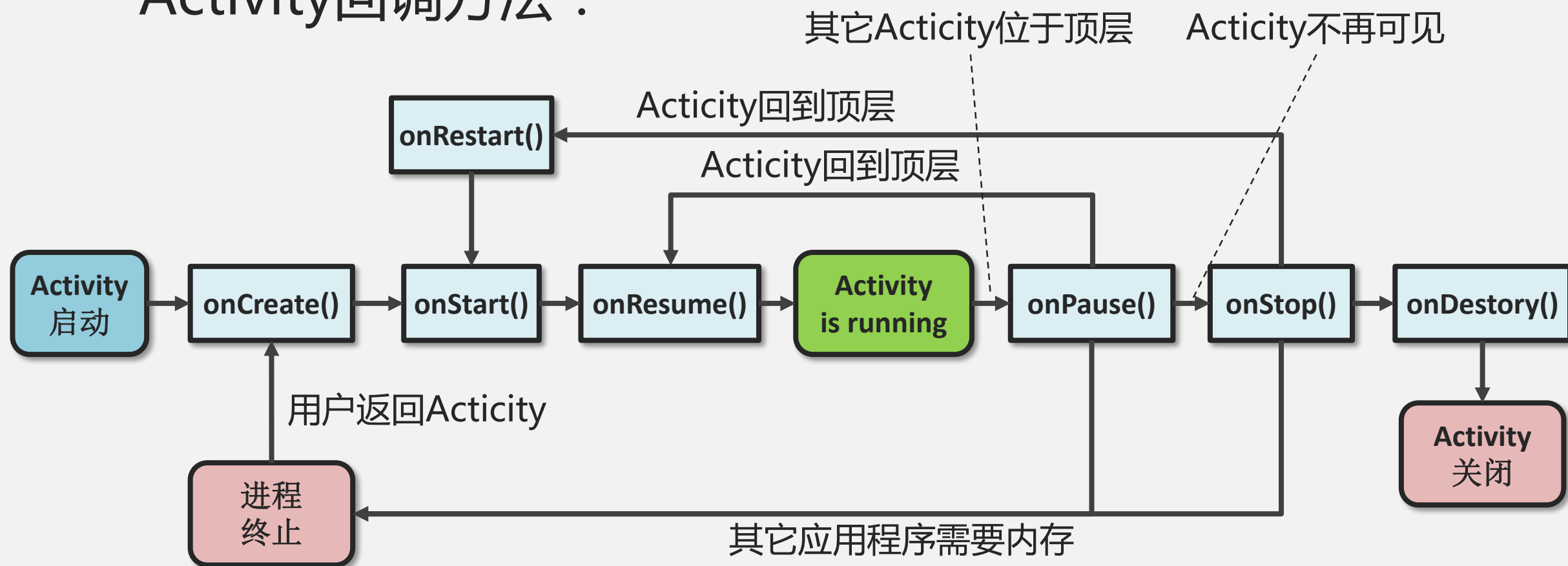
- 当Activity各个活动状态之间发生切换时，会触发以下Activity回调方法：

回调方法↵	描述↵
<u>onCreate(Bundle savedInstanceState)</u> ↵	创建 Activity 时被回调；进行 Activity 的初始化，例如创建 View、绑定数据或恢复状态信息等。↵
<u>onStart()</u> ↵	启动 Activity 时（Activity 显示在屏幕上时）被回调。↵
<u>onRestart()</u> ↵	当 Activity 从停止状态进入活动状态时被回调；此时可能需要恢复用户保存的数据。↵
<u>onResume()</u> ↵	当 Activity 能够与用户交互时（或 Activity 从暂停状态恢复时）被回调；此时当前屏幕所对应的 Activity 处于 Activity 活动栈的栈顶。↵
<u>onPause()</u> ↵	当 Activity 进入暂停状态时被回调；此时常需要保存持久性数据或释放占用的资源。↵
<u>onStop()</u> ↵	当 Activity 进入停止状态时被回调。↵
<u>onDestroy()</u> ↵	在 Activity 被终止前（即进入非活动状态前）被回调。↵



Activity活动状态之间的切换

- 当Activity各个活动状态之间发生切换时，会触发以下Activity回调方法：





Activity状态切换

- 查看Activity状态切换之间触发的回调方法。
 - 使用Log.v()方法在Logcat窗口查看提示信息。



Activity状态切换时数据保持

- 实现状态切换之间用户信息的维持。
 - 首先用户在输入框中输入一段文本；
 - 用户点击“Home”键，返回桌面（Activity状态由活动状态切换为暂停状态）；
 - 用户再次打开应用，期望输入框中信息保持不变（Activity状态由暂停状态切换为活动状态）。



Activity中的数据保持

- Activity活动状态切换时，Android提供了两种机制实现数据的保持：
 - 借助onPause()和onRestart()回调方法实现。
 - 当Activity状态由活动态切换到暂停状态时，可以在onPause()回调方法，保持持久化数据（可以把数据保存在内存中，或把数据保存在文本、数据库中）。
 - 当Activity由暂停状态或停止状态恢复到活动状态时，可以在onRestart()回调文中，恢复用户所保存的状态数据。



目录



- 1 Activity的生命周期
- 2 Activity的应用
- 3 Fragment的生命周期



Activity的应用

- Activity是Android应用中最核心的组件，每一个Android应用必须涉及到Activity的使用。
- 实际应用：
 - 切换Activity时的状态保存。
 - Activity栈的基础应用。



目录



- 1 Activity的生命周期
- 2 Activity的应用
- 3 **Fragment的生命周期**



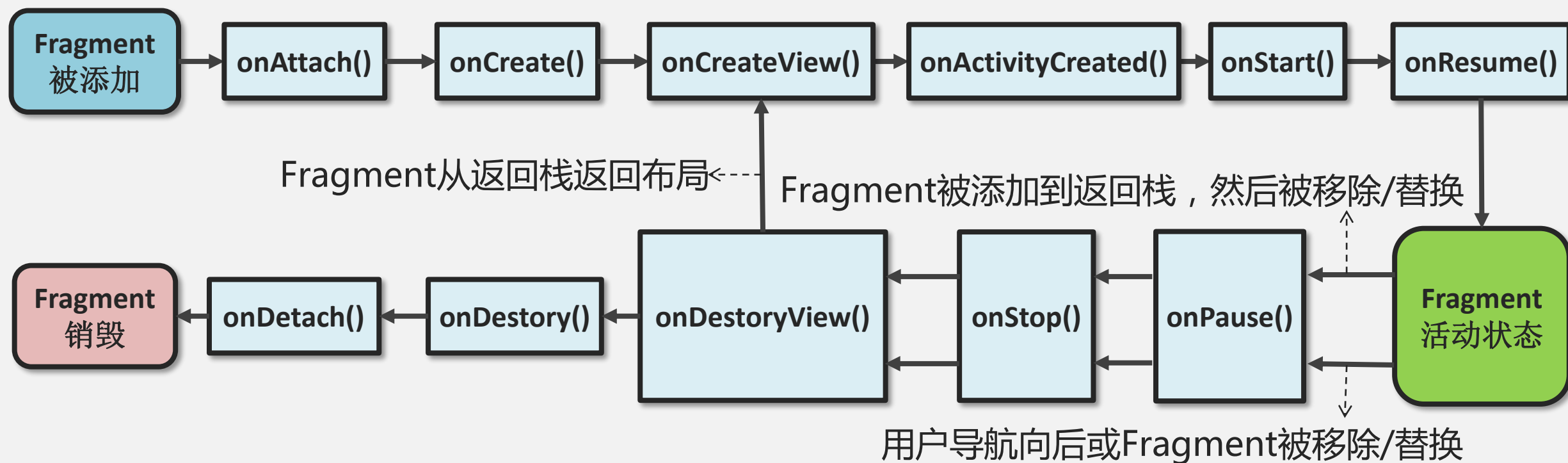
Fragment简介

- Fragment从Android v3.0版本开始引入。
- Fragment不具有在屏幕上显示视图的能力。
- Fragment依赖Activity的存在而存在。
- 在创建Activity时生命周期中的方法先于Fragment生命周期中的方法执行。
- 在销毁Activity时，Fragment生命周期中的方法先于Activity生命周期中的方法执行。



Fragment活动状态之间的切换

- 当Fragment各个活动状态之间发生切换时，会触发以下Fragment回调方法：





Fragment使用回顾

- 主布局文件中添加FrameLayout控件（略）。
- 自定义Fragment页面对应的布局文件（略）。
- 自定义Fragment类文件并加载对应的布局文件（略）。
- 在Activity中添加Fragment页面。

```
FragmentManager manager = getFragmentManager();  
FragmentTransaction tr = manager.beginTransaction();  
FragmentTest fragment = new FragmentTest();  
tr.add(R.id.frl_fragment, fragment);  
tr.commit();
```



Fragment生命周期

- Demo中测试Fragment与Activity生命周期方法的关系。

```
V/lifeCircle: Mainactivity: onCreate()  
V/lifeCircle: Fragment: onAttach(Context context)  
V/lifeCircle: Fragment: onAttach(Context cActivity activity)  
V/lifeCircle: Fragment: onCreate()  
V/lifeCircle: Fragment: onCreateView()  
V/lifeCircle: Fragment: onActivityCreated()  
V/lifeCircle: Mainactivity: onStart()  
V/lifeCircle: Fragment: onStart()  
V/lifeCircle: Mainactivity: onResume()  
V/lifeCircle: Fragment: onResume()
```

MainActivity创建时



```
V/lifeCircle: Fragment: onPause()  
V/lifeCircle: MainActivity: onPause()  
V/lifeCircle: AnotherActivity: onCreate()  
V/lifeCircle: AnotherActivity: onStart()  
V/lifeCircle: AnotherActivity: onResume()
```

MainActivity跳转到
AnotherActivity时



返回MainActivity
并退出



```
V/lifeCircle: Fragment: onStop()  
V/lifeCircle: Mainactivity: onStop()  
V/lifeCircle: AnotherActivity: onPause()  
V/lifeCircle: Mainactivity: onRestart()  
V/lifeCircle: Mainactivity: onStart()  
V/lifeCircle: Fragment: onStart()  
V/lifeCircle: Mainactivity: onResume()  
V/lifeCircle: Fragment: onResume()  
V/lifeCircle: AnotherActivity: onStop()  
V/lifeCircle: AnotherActivity: onDestroy()  
V/lifeCircle: Fragment: onPause()  
V/lifeCircle: MainActivity: onPause()  
V/lifeCircle: Fragment: onStop()  
V/lifeCircle: Mainactivity: onStop()  
V/lifeCircle: Fragment: onDestroyView()  
V/lifeCircle: Fragment: onDestroy()  
V/lifeCircle: Fragment: onDetach()  
V/lifeCircle: Mainactivity: onDestroy()
```



内容回顾

- 1 Activity的生命周期
- 2 Activity的应用
- 3 Fragment的生命周期



Thank you!

