



## 主题

- Intent详解
- Activity堆栈与生命周期

主讲人：Colin

# 什么是Intent？

Intent是一种运行时绑定（run-time binding）机制，它能在程序运行过程中连接两个不同的组件。

尽管Intent可以使用多种方式来让组件进行通信，但通常来讲Intent一般用于：

- 启动Activity
- 启动Service
- 传递Broadcast

# 启动Activity

- 通过将 Intent 传递给 `Context.startActivity()`，可以启动新的 Activity 实例。Intent 描述了要启动的 Activity，并携带了相关必要的的数据
- 如果希望在 Activity 完成后收到结果，可以通过调用 `Context.startActivityForResult()`。然后在 Activity 的 `onActivityResult()` 回调中，通过Intent获取结果

# 构建Intent

## ● 特征

- Component - 要启动的组件名称，如果指定了该属性就是显式Intent。
- Action - 指定要执行的操作(可以使用系统或是自定义)的字符串
- Category - 指定Intent组件的类型的字符串
- Type - 指定数据类型（其 MIME 类型）
- Data - 引用待操作数据和/或该数据 MIME 类型的 URI（Uri 对象）

## ● 额外的附带信息

- Extra - 携带完成请求操作所需的附加信息的键值对
- Flag - 标志可以指示 Android 系统如何启动 Activity（例如，Activity 应属于哪个任务），以及启动之后如何处理（例如，它是否属于最近的 Activity 列表）。

# Intent类型

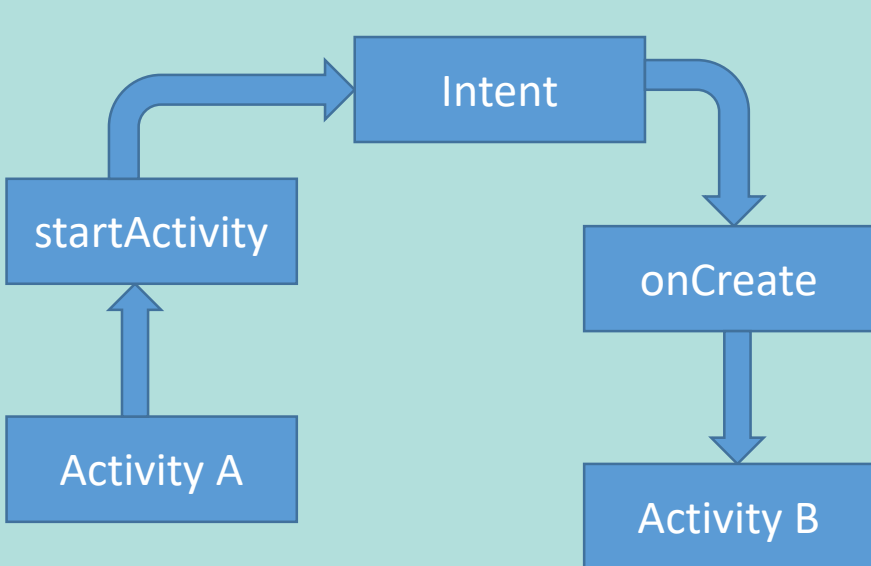
- 显式Intent

按名称（完全限定类名）指定要启动的组件。通常，我们会在自己的应用中使用显式 Intent 来启动组件，例如，启动新 Activity 以响应用户操作，或者启动服务以在后台下载文件。

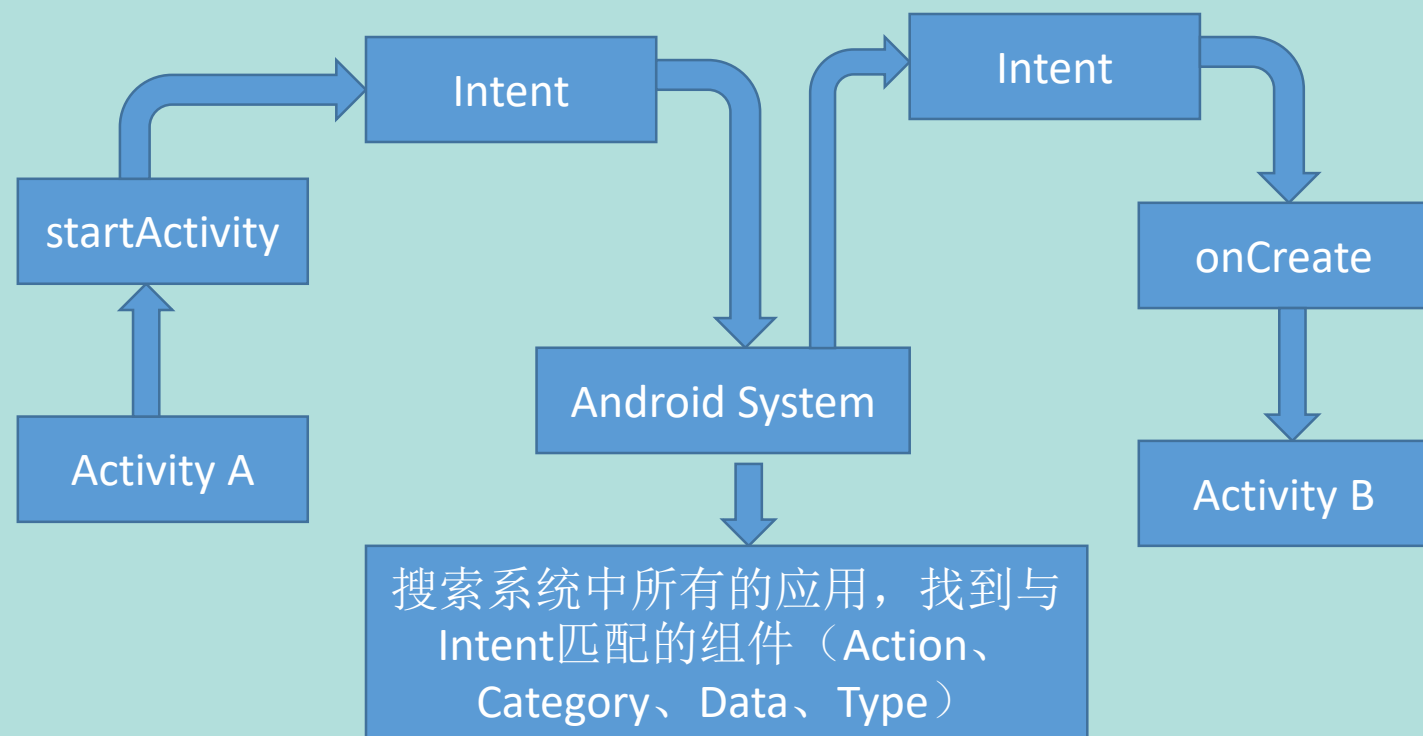
- 隐式Intent

不会指定特定的组件，而是声明要执行的常规操作，从而允许其他应用中的组件来处理它。例如，如需在地图上向用户显示位置，则可以使用隐式 Intent，请求另一具有此功能的应用在地图上显示指定的位置。

# Intent启动Activity图解

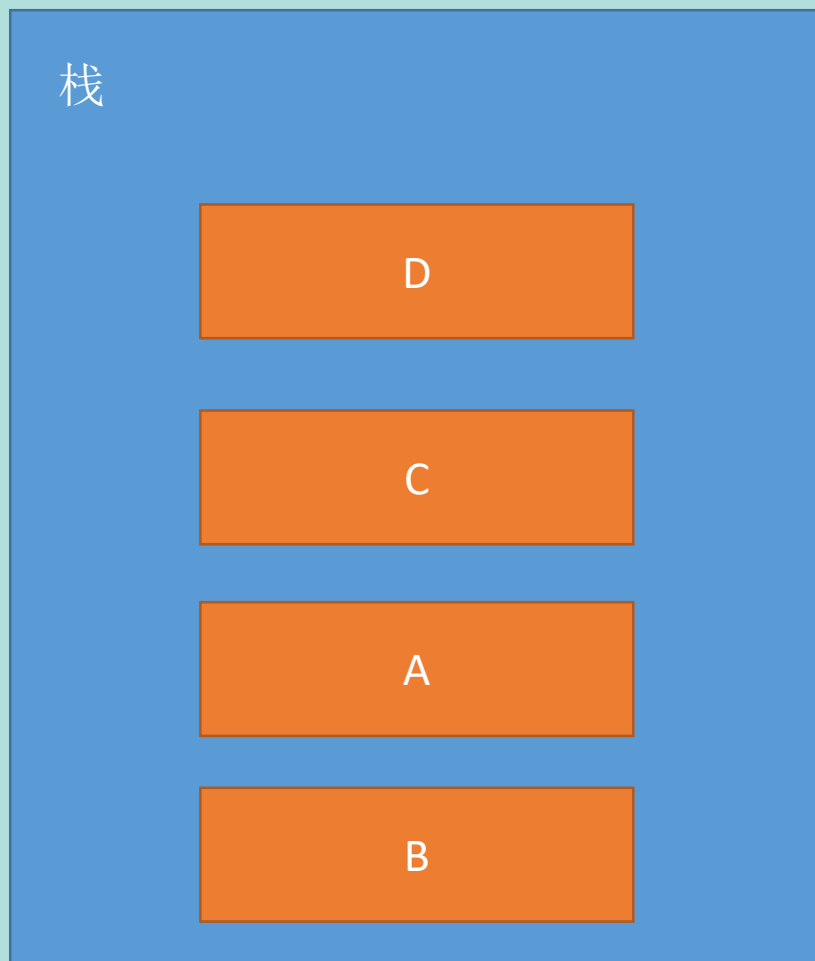


显式启动，设置了Component  
(完全限定了类名)

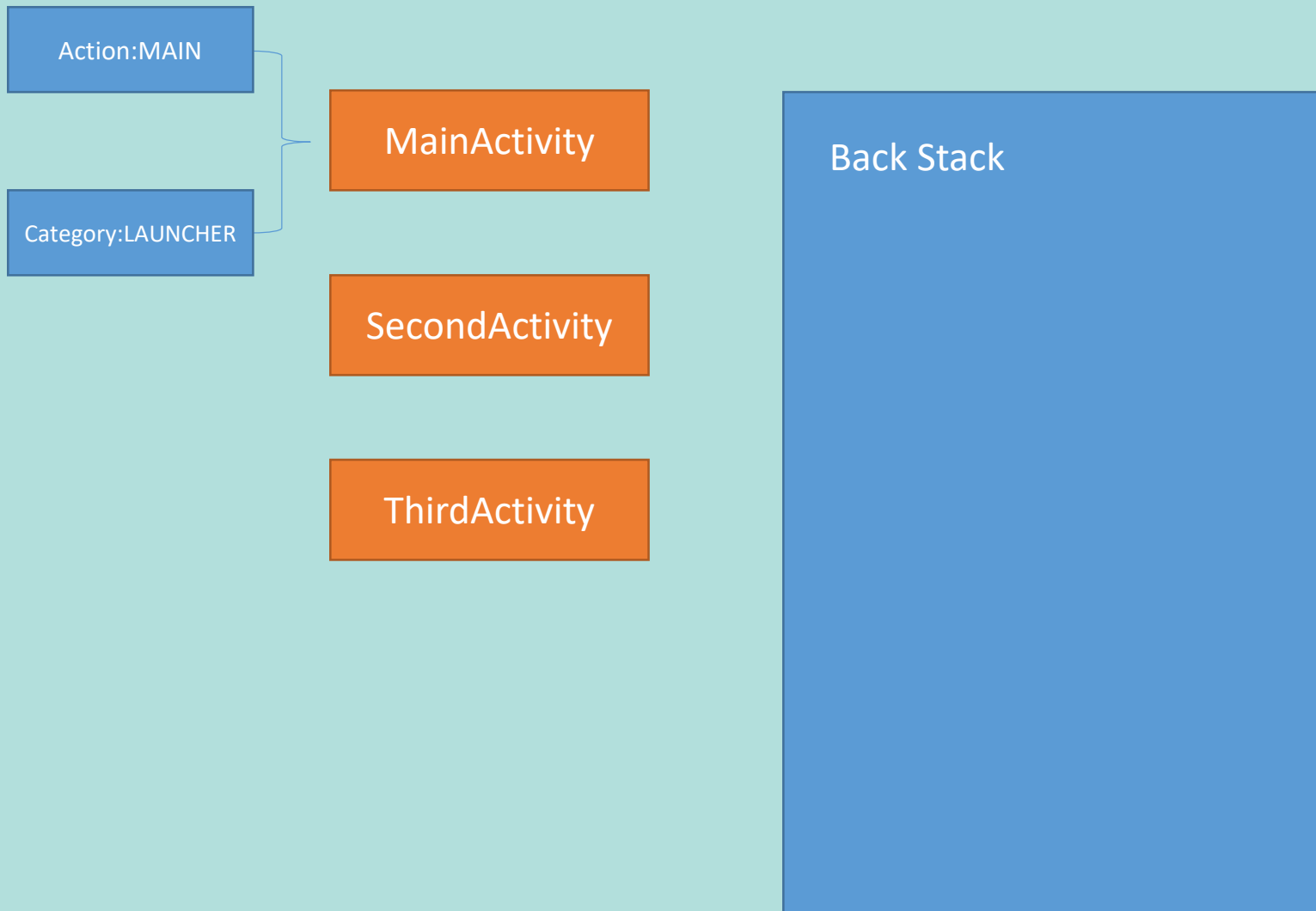


隐式启动，没有设置Component

# 栈的数据结构 后入先出



# Activity的堆栈





# 启动Activity的4种模式

- **Standard** 每次启动的时候都会在（当前）Task创建一个新的实例
- **SingleTop** 当Task的栈顶不为目标Activity的时候，才创建新的实例
- **SingleTask** 当Task中没有目标Activity的时候，才创建新的实例。否则的话就会将旧的Activity上面的Activity全部销毁（出栈），从而达到在栈顶的操作
- **SingleInstance** 独立存在一个Task中，并且该Task不能存储其他的Activity