

# HenCoder Plus 第 11 课 讲义

---

## 触摸反馈基础

---

### 自定义单 View 的触摸反馈

- 重写 onTouchEvent(), 在方法内部定制触摸反馈算法
  - 是否消费事件取决于 ACTION\_DOWN 事件是否返回 true
  - MotionEvent
    - getActionMasked() 和 getAction()
    - POINTER\_DOWN / POINTER\_UP 和 getActionIndex()

### View.onTouchEvent()

- 当用户按下 (ACTION\_DOWN) :
  - 如果不在滑动控件中, 切换至按下状态, 并注册长按计时器
  - 如果在滑动控件中, 切换至预按下状态, 并注册按下计时器
- 当进入按下状态并移动 (ACTION\_MOVE) :
  - 重绘 Ripple Effect
  - 如果移动出自己的范围, 自我标记本次事件失效, 忽略后续事件
- 当用户抬起 (ACTION\_UP) :
  - 如果是按下状态并且未触发长按, 切换至抬起状态并触发点击事件, 并清除一切状态
  - 如果已经触发长按, 切换至抬起状态并清除一切状态
- 当事件意外结束 (ACTION\_CANCEL) :
  - 切换至抬起状态, 并清除一切状态

### 自定义 ViewGroup 的触摸反馈

- 除了重写 onTouchEvent(), 还需要重写 onInterceptTouchEvent()
- onInterceptTouchEvent() 不用在第一时间返回 true, 而是在任意事件, 需要拦截的时候返回 true 就行

### 触摸反馈的流程

- Activity.dispatchTouchEvent()
  - 递归: ViewGroup(View).dispatchTouchEvent()
    - ViewGroup.onInterceptTouchEvent()
    - child.dispatchTouchEvent()
    - super.dispatchTouchEvent()
      - View.onTouchEvent()

- Activity.onTouchEvent()

## View.dispatchTouchEvent()

- 如果设置了 OnTouchListener，调用 OnTouchListener.onTouch()
  - 如果 OnTouchListener 消费了事件，返回 true
  - 如果 OnTouchListener 没有消费事件，继续调用自己的 onTouchEvent()，并返回和 onTouchEvent() 相同的结果
- 如果没有设置 OnTouchListener，同上

## ViewGroup.dispatchTouchEvent()

- 如果是用户初次按下 (ACTION\_DOWN)，清空 TouchTargets 和 DISALLOW\_INTERCEPT 标记
- 拦截处理
- 如果不拦截并且不是 CANCEL 事件，并且是 DOWN 或者 POINTER\_DOWN，尝试把 pointer（手指）通过 TouchTarget 分配给子 View；并且如果分配给了新的子 View，调用 child.dispatchTouchEvent() 把事件传给子 View
- 看有没有 TouchTarget
  - 如果没有，调用自己的 super.dispatchTouchEvent()
  - 如果有，调用 child.dispatchTouchEvent() 把事件传给对应的子 View（如果有的话）
- 如果是 POINTER\_UP，从 TouchTargets 中清除 POINTER 信息；如果是 UP 或 CANCEL，重置状态

## TouchTarget

- 作用：记录每个子 View 是被哪些 pointer（手指）按下的
- 结构：单向链表

## 拦截处理

- 如果不是初次按下，并且没有 TouchTarget，直接拦截
- 如果是初次按下，或者有 TouchTarget
  - 如果设置了 disallow intercept，不拦截
  - 否则，调用 onInterceptTouchEvent()，如果返回 true 则拦截，返回 false 则不拦截

## 问题和建议？

课上技术相关的问题，都可以在学员群里和大家讨论，我一旦有时间也都会来解答。如果我没来就 @我一下吧！

具体技术之外的问题和建议，都可以找丢物线（微信：diuwuxian），丢丢会为你解答技术以外的一切。



## 更多内容：

- 网站：<https://hencoder.com>
- 微信公众号：HenCoder

# HenCoder

给高级 Android 工程师的进阶手册

微信公众号：HenCoder  
微博：扔物线  
知乎专栏：HenCoder  
稀土掘金：扔物线  
<http://hencoder.com>

