Launcher部分拖拽逻辑

类图如下

说明

DropTarget.DragObject

|  |  |
| --- | --- |
| 类名/变量名 | 说明 |
| DragObject | 拖拽的对象 |
| DragView dragView | 拖拽时的view |
| Object dragInfo | 拖拽的iteminfo对象 |
| DragSource dragSource | 拖拽的源头区域, |
| Runnable postAnimationRunnable | 拖拽完成的动画回调 |

DragSource

|  |  |
| --- | --- |
| 类名/方法名 | 说明 |
| DragSource | 拖拽的源头,可以理解从哪个区域开始拖拽的,包括workspace,Folder,AllApp |
| onDropCompleted(GLView target, DragObject d, boolean isFlingToDelete, boolean success) | 拖拽完成,可以认为item落下时的回调,但是如果落下时需要对话框确定,那么会等到对话消失之后才回调该方法,可以简单的认为,对话框阻塞了item的落下动作 |

DropTarget接口

|  |  |
| --- | --- |
| 类名/方法名 | 说明 |
| DropTarget | item落下的区域(target),包括WorkSpace, Folder,ButtonDropTarget等等 |
| boolean isDropEnabled() | 该区域是被disable |
| onDrop(DragObject dragObject) | item落下的回调,但是如果落下时需要对话框确定,那么只有点击确定之后才会回调该方法,如果点击取消或者返回按键,不回调此方法,可以简单的认为该事件的回调时依赖于操作有没有发生, |
| onDragEnter(DragObject dragObject) | item拖拽进入目标区域的回调 |
| onDragOver(DragObject dragObject) | item在目标区域拖动的回调,如果一直拖动,一直回调,可以理解为move事件 |
| onDragExit(DragObject dragObject) | item拖拽出目标区域的回调,当item落下时也会回调该函数,可以理解为落下时,事件结束,回到最初状态 |
| boolean acceptDrop(DragObject dragObject) | 该目标区域是否接受item的落下动作,比如说有些item不支持delete操作,如果该item拖拽到DeleteDropTarget时,该函数返回false |
| getHitRectRelativeToDragLayer(Rect outRect) | 获取target的区域坐标,便于findDropTarget根据坐标位置找到正确的target |

DragController.DragListener接口

|  |  |
| --- | --- |
| 类名/方法名 | 说明 |
| DragController.DragListener | 拖拽动作的监听 |
| onDragStart(DragSource source, Object info, int dragAction); | 从 DragSource开始拖动时回调, info表示iteminfo |
| onDragEnd(); | 拖拽完成的回调,如果有确认对话框出现,必须等待确认对话框消失后,此时如果有动画,必须等待动画结束后回调,没有动画则直接回调,同理,没有对话框出现,也要等待拖拽完成的动画的结束后回调 |

DragController

|  |  |
| --- | --- |
| 类名/方法名/变量名 | 说明 |
| DragController | 拖拽的控制中心,包括拖拽动作的接收,分析,寻找target,分发各种事件 |
| ArrayList<DragListener> mListeners | DragListener的列表,用于分发 |
| ArrayList<DropTarget> mDropTargets | item落下区域的Target列表,DragController主要根据move坐标从该列表寻找合适的target |
| DropTarget.DragObject mDragObject | 拖拽的对象,每次拖拽只会针对一个对象 |
| startDrag(DragSource source,Object dragInfo,...) | 开始拖拽 |
| boolean onTouchEvent(MotionEvent ev) | touch事件,由DragLayer的touch事件分发至此 |
| findDropTarget(int x, int y, int[] dropCoordinates | 根据X,Y,dropCoordinates需要合适的target |
| drop(float x, float y) | item的落下动作,实际是手指up的时候触发 |
| onDeferredEndDrag | 延迟结束拖动,在动画结束的时候才会调用onDeferredEndDrag |
| cancelDrag() | 取消拖拽,一般有确认对话框出现,点击取消或者返回按键的调用 |
| endDrag() | 结束拖拽 |
| handleMoveEvent(int x, int y) | 处理touch事件中的move,这里面主要是move的过程中寻找target, 分发给target的一些事件,比如enter,exit,over |