## 概述

### 相关插件

相关插件：

◆Drill\_EventThrough 体积 - 事件穿透关系

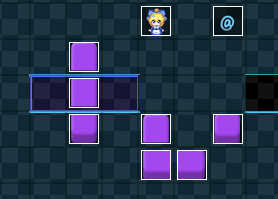
◆Drill\_EventUnificationOfTrigger 体积 - 一体化 & 触发

◆Drill\_EventUnificationOfMove 体积 - 一体化 & 移动

◆Drill\_EventUnificationOfDirection 体积 - 一体化 & 朝向

事件一体化常用于一个同时占领多个图块的大事件。

多个事件能够保持像单个事件那样整体移动、朝向、触发。

**名词索引**

以下你可以按住ctrl键点击下面的词，可以直接定位到想了解的名词：

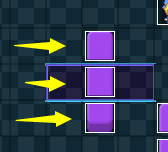
|  |  |
| --- | --- |
| 一体化 | [一体化](#_一体化) [玩家的一体化](#玩家的一体化) [事件组](#事件组)  [移动一体化](#_移动一体化) [动力源](#动力源) [速度统一](#速度统一)  [朝向一体化](#_朝向一体化)  [触发一体化](#_触发一体化) |

## 一体化

### 定义

**一体化：**指多个相同标签的事件绑定在一起，形成的一个整体。

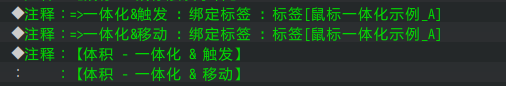
根据绑定类型，分为移动一体化、朝向一体化、触发一体化等。



**玩家的一体化：**指玩家与事件绑定在一起，形成的一个整体。



**事件组：**指多个事件一体化之后，形成的分组集合。



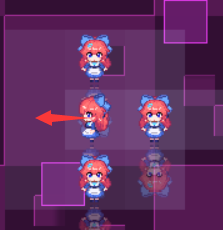
### 移动一体化

#### 1）插件介绍

功能来自插件：

Drill\_EventUnificationOfMove 体积 - 一体化 & 移动

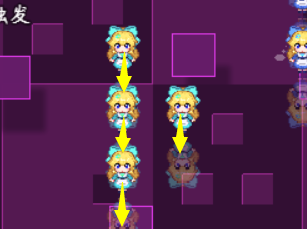
绑定移动一体化后，事件组内的所有事件，能同步执行事件移动。



**移动时相互穿透：**移动一体化的 事件组 自带穿透标签。

常规的多个事件执行移动时，可能会因为前一个事件阻塞后一个而造成移动失败；

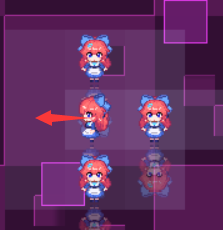
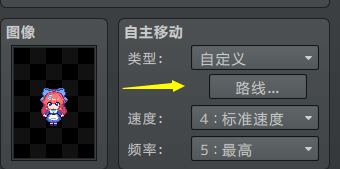
而一体化的事件相互之间会穿透，所以不会出现阻塞掉队的情况。



#### 2）动力源

**动力源：**指群体事件绑定时，群体中会执行**自主移动**的事件。

群体事件中通常都只留一个动力源，因为存在多个动力源时，会出现谁决定如何移动的问题。

程序中如果整体里一个事件要向上走，另一个事件向下走，则根据id顺序来决定听谁的。假设向上走的事件被阻塞不能向上走，则听从下一个事件的走法。



具体可以去鼠标管理层的右上角区域，

看看单个动力源于多个动力源之间的移动效果。

#### 3）速度统一

**速度统一：**指绑定了移动一体化的事件组中，所有事件的移动速度一致。

事件组整体移动时，速度可以不一样，如果未设置 速度统一，可能会发现有些事件赶不上整体事件，虽然行走图赶不上，但并不会脱节。



具体可以去 体积管理层 看看玩家绑定速度统一后的效果。

### 朝向一体化

#### 1）插件介绍

功能来自插件：

Drill\_EventUnificationOfDirection 体积 - 一体化 & 朝向

绑定朝向一体化后，事件组内的所有事件，朝向始终保持一致。



具体可以去体积管理层，看看绑定朝向后，事件的行为。

### 触发一体化

#### 1）插件介绍

功能来自插件：

Drill\_EventUnificationOfTrigger 体积 - 一体化 & 触发

绑定触发一体化后，事件组内的所有事件，能同步触发独立开关。



注意，一体化触发只同步 开启/关闭独立开关。

而开关的触发条件本身，不会因为绑定了一体化触发，而被扩充条件。

比如，下图中，蓝色的重力开关与灰色开关相互绑定，但是灰色开关没有加触发。

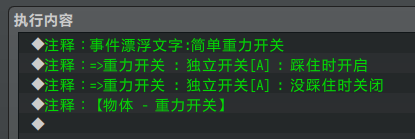
玩家踩住灰色开关不会触发，只有踩了蓝色开关，才会全部触发。



#### 2）一体化与触发条件的冲突

根据”8.物体 > 大家族-开关.docx”的介绍。

我们知道重力开关的功能，为两个触发的组合结构：踩住时开启+没踩住时关闭。



问题就在于”没踩住时关闭”的触发。

即使 一体化触发 在任何一个开关被踩住时，同步开启了独立开关，

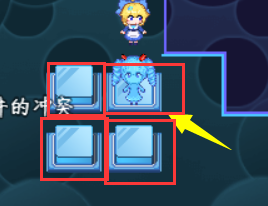
由于这个开关没有被踩住，所以开关会被立即恢复到关闭状态。

以下图为例，一个开关被踩住，而其他三个开关没被踩住。

由于四个重力开关都为持续触发，

那么踩住的开关会保持 开启独立开关；

而没踩住的开关会保持 关闭独立开关。



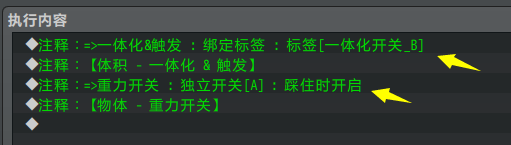
因此，绑定触发一体化 与 ”没踩住时关闭”的触发 相互冲突，在设计时要小心。

我们常规理解的触发一体化，是“踩住任意一个按钮，其他按钮全部同步开启”，这也就造成了理解上的偏差。



因此，设计时只留 ”踩住时开启”触发+一体化触发，

能实现简单的大体积开关按下的功能，但是回弹需要单独用其他事件来控制。



#### 3）鼠标悬停响应开关

功能来自插件：

Drill\_EventMouseHoverSwitch 物体 - 鼠标悬停响应开关

前面介绍了 一体化 与 ”没踩住时关闭”的触发 **存在设计冲突**。

经过插件兼容，（行走图-行走图与鼠标控制核心 的一体化支持）

该插件能合并 事件组 内的所有事件的触发条件：

悬停、悬停左键、悬停右键、悬停滚轮

鼠标触发任意一个，都同步开启相应的独立开关。

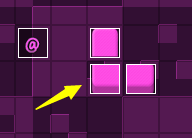
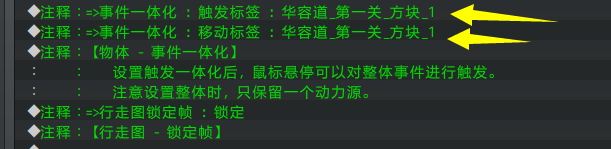
 

#### 4）华容道方块设计

华容道关卡中的事件，

多个事件绑定了”华容道\_第一关\_方块\_1” 的移动标签；

并且还绑定了”华容道\_第一关\_方块\_1” 的触发标签。

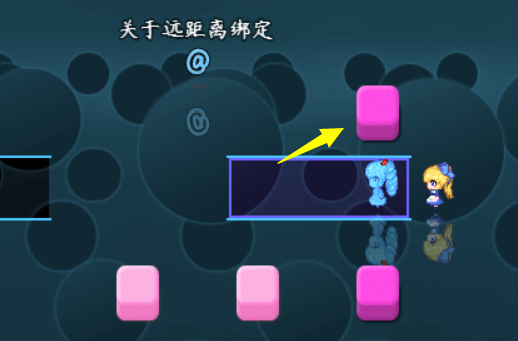
 

移动一体化用于 鼠标拖拽时，能被整体拖拽移动。

触发一体化用于 鼠标接近方块后，可以点亮方块。

顺带一提，事件进行一体化绑定后，与它所在的位置没有关系。

也就是说，即使两个事件距离很远，也能同步移动、朝向、触发一体化。



你可以去看看示例中 设计-华容道 的方块事件。

## 从零开始设计（DIY）

### 设计一个大体积事件

#### 1. 设置一个目标

2019年12月8日，作者我完成了一体化的插件。

于是这里作者我想设计一个体积为三的大方块。



#### 2. 结构规划/流程梳理

大方块需要三个事件，并且事件之间需要添加一体化标签。

所以用到下面两个插件：

◆Drill\_EventUnificationOfTrigger 体积 - 一体化 & 触发

◆Drill\_EventUnificationOfMove 体积 - 一体化 & 移动



#### 3. 事件设置

**a）行走图设置**

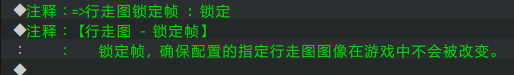
先绘制方块，画出 头部、中部、尾部 三个部分。

绘制3d的方块行走图，

可以去看看文档“7.行走图 > 关于行走图与图块.docx”中的一体化行走图的3d画法。



由于方块是固定的静态物体，所以需要添加锁定帧的注释，确保行走图不会变化。



**b）自主移动设置**

多个事件时，移动一体化一般要考虑 [动力源](#动力源) 的情况。

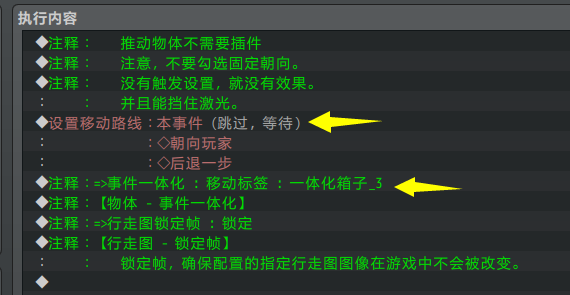
由于 这里的设计 是被玩家推动的（事件指令执行移动路线），

不需要依靠自主移动，所以不需要考虑动力源问题。

**c）事件指令设置**

事件指令如下，普通的推动物体功能即可。

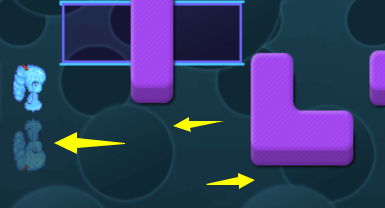
需要添加“移动标签”，确保三个事件能够同步移动。



#### 4. 细节设置

需要注意的是，由于一体化的事件，是单独的事件，如果反射镜面，会非常糟糕，需要注意关掉倒影镜像。



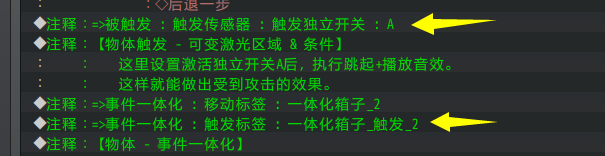


#### 5. 高级设置

一体化事件，可以与 物体触发 相结合使用。

三个事件中，任何一个事件被触发开关A（比如闪电击中），三个事件会同步反应执行 受伤事件页的流程。



需要注意的是，三个事件都执行事件页时，要设为并行执行；

并且，播放声音、伤害计算，只让其中一个事件执行即可，注意不要让三个事件都执行。



