## 概述

### 相关插件

核心插件：

◆Drill\_CoreOfFixedArea 物体触发 - 固定区域核心

可扩展插件：

◆Drill\_EventBufferTags 物体管理 - 事件的缓存标签

**该文档只介绍单独的 固定区域 的定义。**

|  |
| --- |
| 固定区域核心 单独使用没有效果，它需要子插件来实现对区域的操作。  可以看看章节：[从零开始设计](#_从零开始设计) |

### 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：

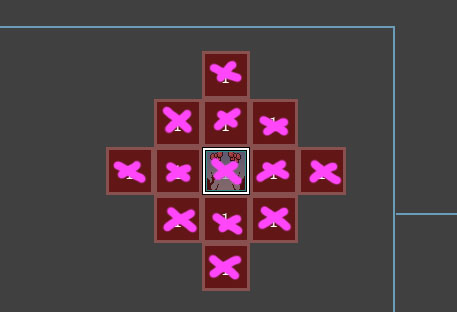
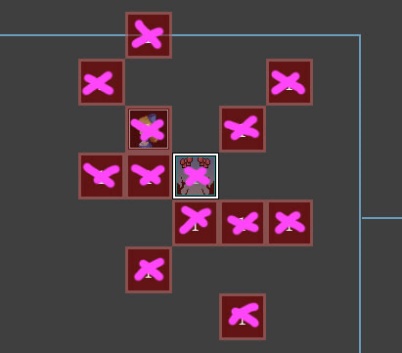
|  |  |
| --- | --- |
| 固定区域 | [固定区域](#_固定区域) [区域主体](#_区域主体) [筛选器](#_筛选器) |
| 形状区域 | [形状区域](#形状区域)  [方形区域](#方形区域) [菱形区域](#菱形区域) [圆形区域](#圆形区域) [十字区域](#十字区域) [横条区域](#横条区域) [竖条区域](#竖条区域) |
| 自定义区域 | [自定义区域](#_自定义区域) [配置自定义区域](#_配置自定义区域) |

### 快速区分

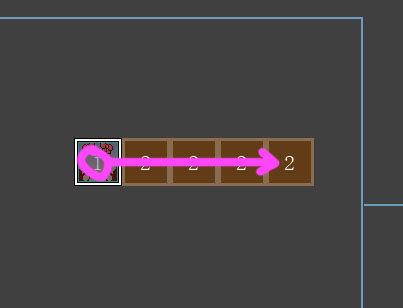
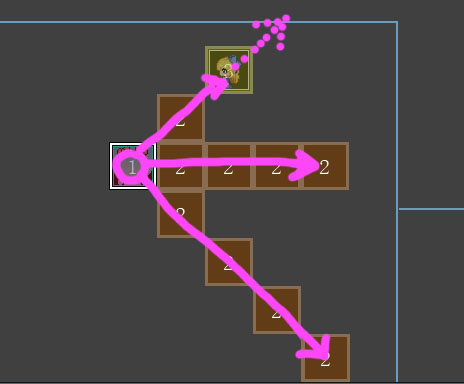
**区域：**指多个图块组成的集合，可以是一个范围的点，也可以是零散的点。

目前区域分为 固定区域 和 可变激光区域。

**固定区域**：多种形状区域、自定义区域、筛选器、必然穿透

**可变激光区域**：激光被阻挡多变、方向固定、穿透属性、含起始点/终止点

### 插件关系

固定区域核心之间的关系如下：



## 固定区域

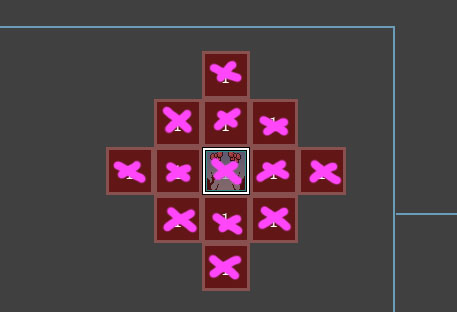
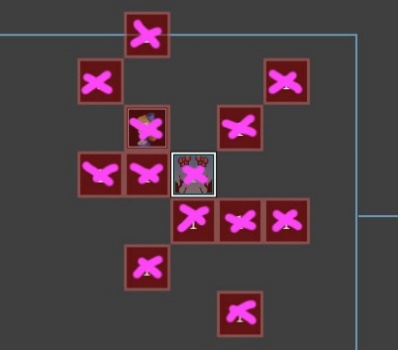
### 定义

**区域：**指多个图块组成的集合，可以是一个范围的点，也可以是零散的点。

目前区域分为 固定区域 和 可变激光区域。

**固定区域**：指通过硬性设定的点集合，而确定的区域。

设置方式有两种：形状区域和自定义区域。还包含筛选器的功能。

**形状区域：**和自定义区域相对，都属于固定区域。是一个统称的名词。

具体介绍可以看后面章节：[形状区域](#_形状区域) 和 [自定义区域](#_自定义区域) 。



### 区域差别

固定区域与可变激光区域中的模块关系如下图：



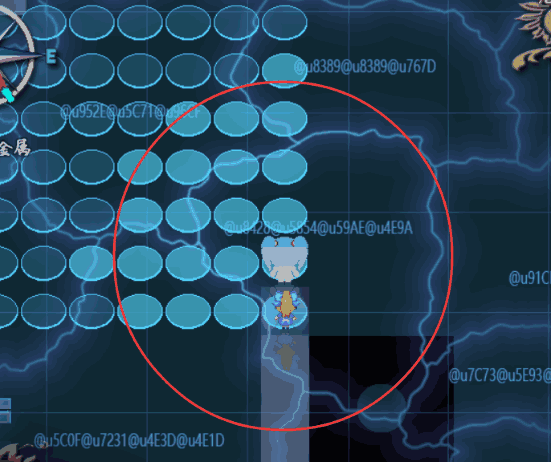
### 形状区域

形状区域如下表：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **大小** | **方形区域** | **菱形区域** | **圆形区域** | **十字区域** | **横条区域** | **竖条区域** |
| 0 |  |  |  |  |  |  |
| 1 | F:\rpg mv箱\aaaa (7).jpg | F:\rpg mv箱\aaaa (2).jpg | F:\rpg mv箱\aaaa (2).jpg | F:\rpg mv箱\aaaa (2).jpg | F:\rpg mv箱\aaaa (3).jpg | F:\rpg mv箱\aaaa (1).jpg |
| 2 | F:\rpg mv箱\aaaa (8).jpg | F:\rpg mv箱\aaaa (6).jpg | F:\rpg mv箱\aaaa (6).jpg | F:\rpg mv箱\aaaa (9).jpg | F:\rpg mv箱\aaaa (5).jpg | F:\rpg mv箱\aaaa (4).jpg |
| … | …… | …… | …… | …… | …… | …… |

表中的菱形区域和圆形区域区分度不高，下图为圆型区域半径为4的情况。

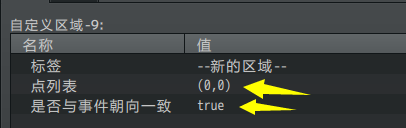
（由于图块的是一块块固定的，所以圆一般都比较丑）



### 自定义区域

自定义区域与 玩家/事件 朝向的方向有关系。

你需要考虑下区域是否根据朝向变化。



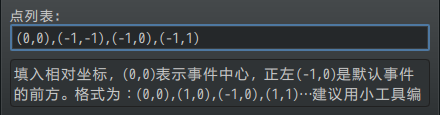
点列表的排布如下图。

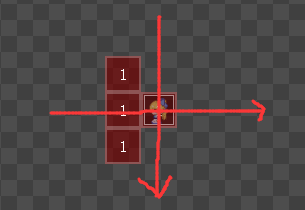
相对坐标。(0,0)是物体的中心，正左方(-1,0)为默认物体的前方。

X轴方向，向左为负，向右为正。

Y轴方向，向上为负，向下为正。

详细配置看看后面章节：[配置自定义区域](#_配置自定义区域) 。





### 区域主体

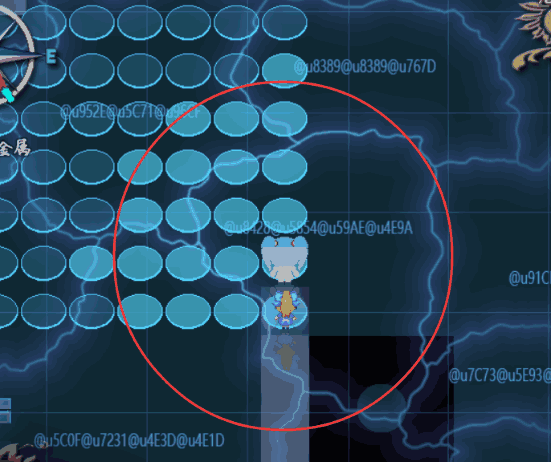
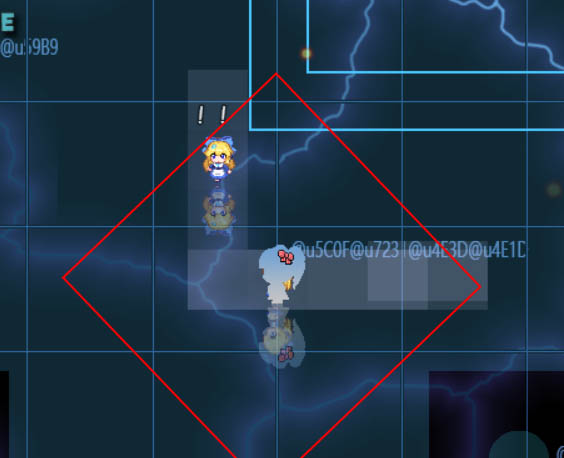
**区域主体：**指区域绑定的对象，区域会以这个对象进行功能展开。

在使用子插件时，需要区分一下主体是谁，具体情况具体分析。

1）你需要注意区域是跟随哪个主体。

比如，指定一个点触发，则这个点是主体，以该点为中心进行区域作用。

再比如，玩家接近插件，主体是玩家，以玩家为中心，区域内触发事件。

2）你需要了解区域主体是什么物体。

比如，若主体是一个点，那么"朝向一致"的功能是无效的，因为点没有朝向。

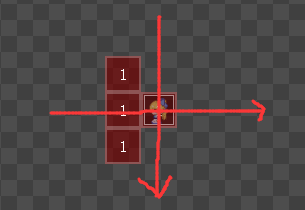
再比如，若主体是玩家，那么"朝向一致"的功能是有效的，和玩家朝向一致。

（简单来说，点没有朝向，事件有朝向。点因为没有朝向，所以朝向一致肯定是无效的。）

### 区域主体的方向

下图为自定义区域不同朝向的效果。

如果自定义区域**设置为方向无关**，则只会出现第二种效果（配置坐标中，以朝左为准）。

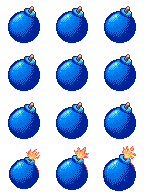


注意，示例中的炸弹行走图，由于炸弹到要切换到点燃状态，而点燃状态是朝向上方。

因此炸弹行走图和它的朝向不吻合，会存在触发方向的问题。

你可以关闭朝向一致，也可以将炸弹的行走图分成两张，以避免朝向与行走图的问题。

另外，你如果想做举起炸弹后与扔出去的朝向一致，那么你先要修改举起物体朝向一致，然后再修改炸弹的行走图。

## 筛选器

### 1）一般筛选

**筛选器：**指在 固定区域 的基础上，对点集合位置进行条件筛选，留下符合条件的点。



举个例子，勾选了”图块-R图块标志” 的筛选器。

下图为筛选R图块1的部分：



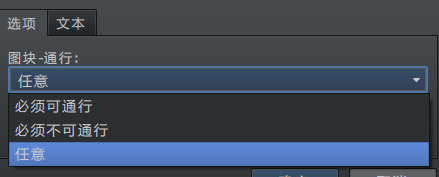


分别为**排除R图块1**，与**只留R图块1**的菱形3区域。

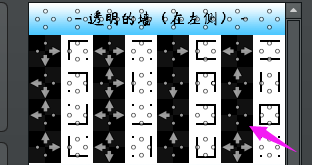
### 2）可通行筛选

图块-通行 不考虑事件，事件的阻塞不是 不可通行，两者要分开考虑。

可通行的判定为： 图块中只要有任一处可通行，则被认作可通行。

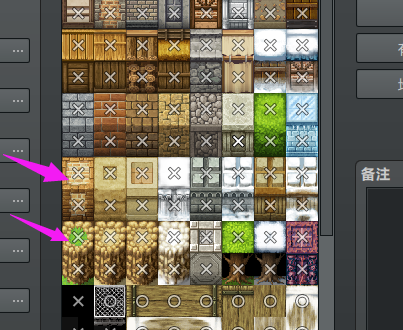


必须全部四个方向都不可通行，才会被判定为不可通行。



但是注意，这里有一类图块非常特殊：墙壁和天花板图块。

天花板即使设置了不可通行，在涂抹后，却仍然可以在天花板上面行走。



也就是说天花板一直会被视作为可通行的。

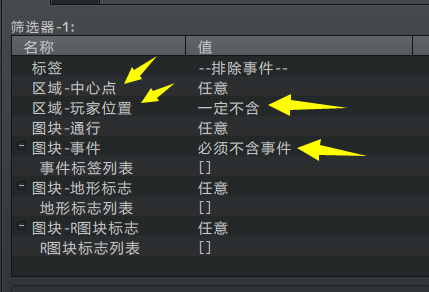
你设计筛选器时，需要特别注意一下天花板情况。

### 3）中心点与玩家位置筛选

中心点、玩家位置属于特殊的情况。

因为玩家不属于事件，所以在触发时，需要单独检查是否需要筛选。

（如果要控制玩家能像事件一样受伤，可以使用“玩家的事件”插件来控制。）



以面板管理层的金币箱为例，去掉玩家位置后，可以确保金币不会直接落到玩家身上。

（可拾取事件落到玩家身上不算接触，需要离开然后返回才能捡起）



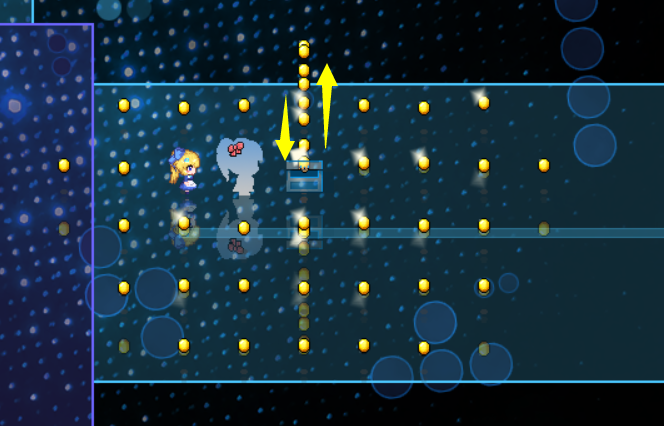
### 4）筛选器占满时情况

当筛选器条件筛选后，没有得到任何符合条件的图块位置时，

那么筛选器则只能无视全部条件出现在原地。

以可拾取物生成器为例，如下图，金币没地方放了，就只能出现在原地。

◆Drill\_EventItemGenerator 物体管理 - 可拾取物生成器



### 5）缓存标签的筛选

筛选器中，能设置“事件标签列表”的条件。

这里的标签，来自插件：

◆Drill\_EventBufferTags 物体管理 - 事件的缓存标签



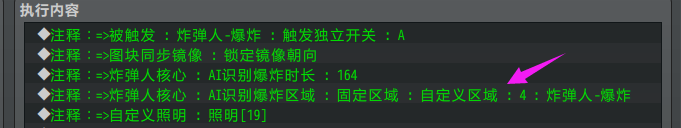
详细去看看 "28.物体管理 > 关于事件的缓存数据.docx"中筛选器的介绍。

### 6）炸弹人AI不能识别筛选器

使用炸弹人插件时，你如果放置了固定区域的炸弹，炸弹人的AI可以识别并躲避。

但是如果炸弹的爆炸区域具有筛选器，AI无法识别 筛选器炸不到的地方。

因为目前炸弹的属性，只能配置识别固定区域，不能添加筛选器。



## 从零开始设计（DIY）

### 子插件功能

固定区域核心只对 固定区域 进行了定义。

单独无法工作，它需要依赖于 子插件 才能生效。

若你设计的是 插件指令条件触发，

可以去看文档：” 9.物体触发 > 关于物体触发-固定区域触发.docx”。

若你设计的是 弹丸事件，

可以去看文档：” 9.物体触发 > 物体设计-简易弹丸制作.docx”。

若你设计的是 玩家接近触发，

可以去看文档：” 9.物体触发 > 关于玩家接近触发.docx”。

若你设计的是 事件接近触发，

可以去看文档：” 9.物体触发 > 关于事件接近触发.docx”。

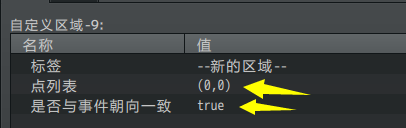
### 配置自定义区域

#### 1. 自定义区域属性

首先，新建一个自定义区域，可以看到两个属性：

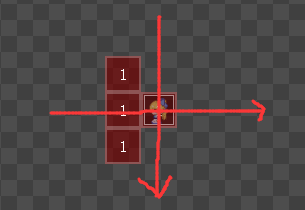
点列表、朝向一致

一般都默认 朝向一致 为开启，如果具体场景中有特殊情况，可以回头再关闭朝向一致。

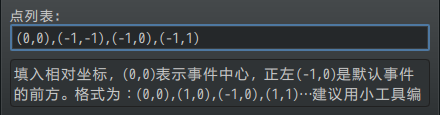


#### 2. 点坐标关系

坐标关系如下图。注意 **事件前方** 是以 **朝左** 为准的。

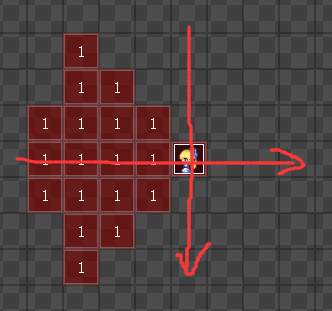


对应的点列表如下：



#### 3. 自定义点列表

举个例子，假设我要做一个下图的大扇形范围：



那么对应的xy坐标点如下：

(0,0),

(-1,0), (-1,-1), (-1,1),

(-2,0), (-2,-1), (-2,-2), (-2,1), (-2,2),

(-3,0), (-3,-1), (-3,-2), (-3,-3), (-3,1), (-3,2), (-3,3),

(-4,0), (-4,-1), (-4,1),

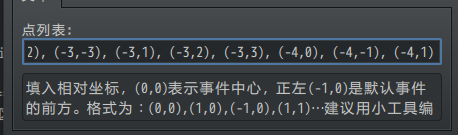
根据事件朝左的 相对坐标 进行一个一个点描述。

将上述的点合并在一起填入点列表即可。

#### 4. 点列表格式

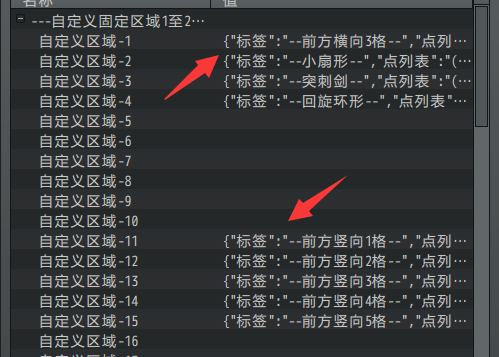
这里填写的格式必须为括号和逗号的格式 ”(x,y)”，多个点用逗号隔开。

可以用 中文逗号 或 中文括号。



#### 5. 配置参考

插件中也有默认的一些 自定义区域 的配置，你可以参考那些点来设计区域，比如：



## 常见问题（FAQ）

### 可拾取物生成器设置筛选器无效

|  |  |
| --- | --- |
| **问题名称** | 可拾取物生成器设置筛选器无效 |
| **问题图示** |  |
| **原理解析** | 插件 Drill\_EventItemGenerator 可拾取物生成器 可以单独使用。  与固定区域核心的关系并不紧密，即使你没有开 固定区域核心 插件，也是能生成事件的。 |
| **解决方案** | 1. 看看固定区域核心是不是忘了开，或者直接忘了加。  关核心的情况下，修改核心里的筛选器肯定是无效的。    2. 查看筛选器中的条件。  如果条件中全是”任意”，则表示筛选器 没有任何 约束条件。 |