## 概述

### 相关插件

核心插件：

◆Drill\_CoreOfEventManager 物体管理 - 事件管理核心

相关子插件：

◆Drill\_EventDuplicator 物体管理 - 事件复制器

◆Drill\_EventDuplicator 物体管理 - 玩家的事件

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

物体管理专门作为一个单独的类型，控制事件的生成、复制、删除。

你可以看看 事件管理的 [定义](#_定义) ，快速了解核心机制。

### 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：

|  |  |
| --- | --- |
| 定义 | [事件管理](#事件管理) [地图与事件机制](#_1）地图与事件机制) [事件管理机制](#_2）事件管理机制) |
| 创建 | [直接生成事件](#_直接生成事件) [复制事件](#_复制事件) |
| 删除 | [事件置空](#_事件置空) [暂时消除](#_暂时消除) [彻底删除](#_彻底删除) |
| 操作事件 | [操作事件](#_操作事件) |

## 事件管理

### 定义

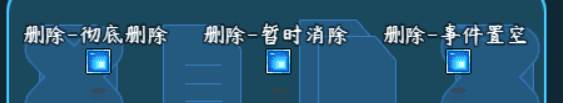
**事件管理：**指专门对事件进行管理的功能。

事件管理包含 创建、删除 两个主要功能。

**创建：**包含 直接生成事件、复制事件 。

**删除：**包含 事件置空、暂时消除、彻底删除 。





你可以去看看示例的 物体管理管理层 。

### 原理介绍

#### 1）地图与事件机制

原游戏设定中，地图与事件直接相关。

地图创建时，事件一并被创建。

地图销毁时，事件一并被销毁。

切换地图，就是先销毁原地图，然后重新载入新地图的过程。

（刷菜单/进入战斗 不会销毁原地图）

其它情况下，你都 **不能** 直接控制事件的创建与销毁。



|  |
| --- |
| 通过开关、独立开关，只能修改这个事件所处的事件页。  因此你使用的所有”删除”指令，都只是让这个事件暂时隐身，并不是彻底删除。 |

如下图，事件只是看不见了，而它还一直在那。



原游戏设定中，

因为设计者无法创建/销毁事件，只能提前把事件摆在那。

所以设计者不需要考虑事件数量管理问题，但代价是高级的功能用不了。

#### 2）事件管理机制

引入事件管理插件后。

你可以复制其他地图中的事件了。

并且复制数量没有限制，功能更加强大。



|  |
| --- |
| 但是，这也带来了一个新问题。  即复制的事件数量不可控，玩家可能会随心所欲地发射1000颗弹丸。  而这些弹丸如果不彻底删除，那么会一直留在后台静默消耗。 |

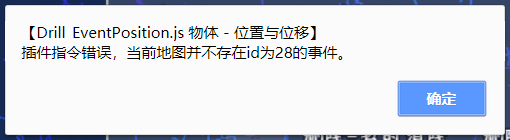
因此，事件管理插件提供了一个指令：彻底删除。

彻底删除与销毁 意思一样，将事件对象从内存上彻底删除，如下图。



但注意，彻底删除后，该事件就不能再被使用了。

如果有插件指令访问这个事件，会报错：找不到对象。



|  |
| --- |
| 并不是所有事件都需要调用 彻底删除 指令。  而是那些可能被复制1000个的弹丸事件，才需要 彻底删除，避免累加性能消耗。  所有事件最终都会在离开地图时一并销毁，进入新地图时再重新创建另一批事件。 |

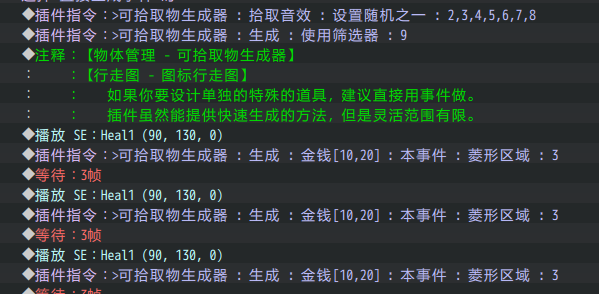
### 创建

#### 创建 - 直接生成事件

部分插件直接提供生成事件的方法，如：

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

插件内置了事件模板数据，你能够根据插件指令，直接生成一个新的事件。





#### 创建 - 复制事件

复制事件一般都来自于事件复制器插件：

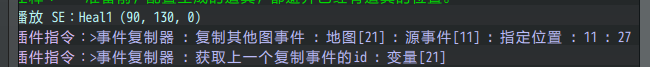
◆Drill\_EventDuplicator 物体管理 - 事件复制器

**1）复制当前地图事件**

复制当前地图的事件，可以在进入地图后立即执行复制。

**2）复制其它地图事件**

复制其他事件时，也可以在进入地图后立即执行复制。





|  |
| --- |
| 旧版本的复制器，进入地图要等90帧左右才可以复制（等待外部地图加载）。  新版解决了此问题，如果外部地图还未加载完成，指令会等待地图加载完成后再执行。 |

被复制的事件最好都统一放在模板管理层，方便管理。



你也可以复制其他地图中的事件，

比如在 物体管理管理层 召唤出 地图管理层 中的小爱丽丝。



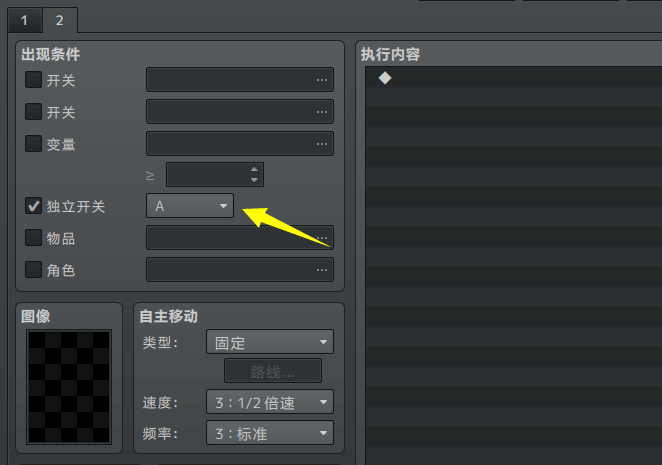
### 删除

#### 删除 - 事件置空

将事件置空，是一种删除事件的方法。

置空后，事件能以低消耗的方式存在于地图中。

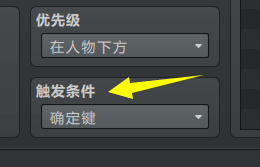
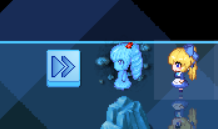
你可以关闭独立开关，让事件重新恢复原样。



但是注意，空事件状态时，如果玩家站在这个事件上，

事件会吸收玩家触发的“确定键”的判定。

比如，玩家站在瓦砾事件上，无法举起前面的 转向砖块。

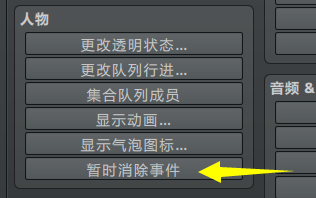
要避免此情况，修改为其他触发条件即可。

具体可以去看看文档：“9.物体触发 > 物体设计-简易弹丸制作.docx”中的常见问题。

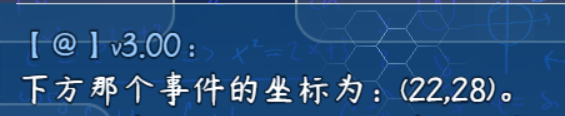
#### 删除 - 暂时消除

编辑器中提供了暂时消除事件的指令。

能够让事件暂时消除。



但是注意，此删除并不是彻底删除，你仍然可以获取到这个事件的属性和坐标。



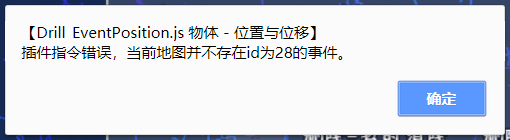
|  |
| --- |
| 如果事件是**编辑器设置**的，离开地图后，再回来，此事件会重新出现。  因为事件数据在地图加载时会重新载入，从而事件被重新创建。  但如果事件是**临时创建**的，离开地图后，再回来，此事件就没了。  因为地图重新载入时，没有临时创建事件的数据。 |

#### 删除 - 彻底删除

事件管理核心中，提供了最终删除方法：彻底删除。

彻底删除是将事件对象完全置空，其他插件尝试获取此事件时，直接会报错。

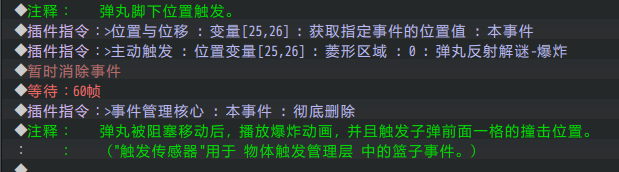




设置彻底删除，主要是考虑到如果存在大量事件创建的情况（比如1000个弹丸事件），如果删除不彻底，会造成过量的事件挤压，严重影响性能消耗。

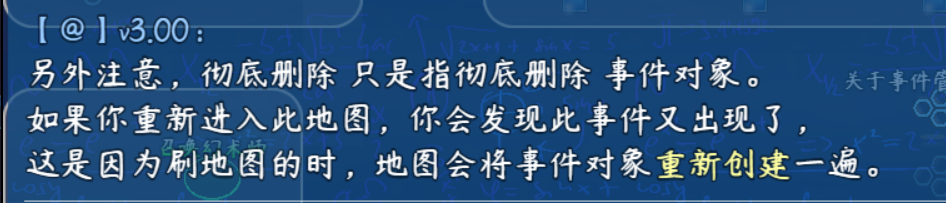
比如弹丸发射关卡，每个子弹爆炸后，需要彻底删除。

前面两种删除方法都不彻底，事件对象和属性仍然存在。





|  |
| --- |
| 如果事件是**编辑器设置**的，离开地图后，再回来，此事件会重新出现。  因为事件数据在地图加载时会重新载入，从而事件被重新创建。  但如果事件是**临时创建**的，离开地图后，再回来，此事件就没了。  因为地图重新载入时，没有临时创建事件的数据。 |



### 操作事件

#### 1）获取上一次创建的事件的id

所有新创建的事件，都具有不一样的id编号。

如果你需要对新事件进行插件指令操作，那么你必须在复制它之后，存储它的id编号。

比如下图，将上一个事件的id，给了临时变量21。

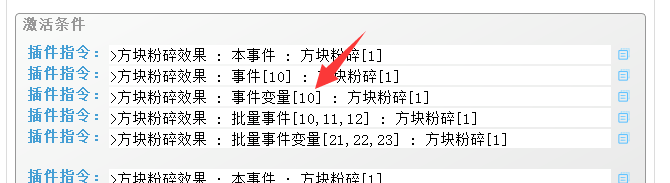


由于创建的新事件，都是临时分配的id，你无法确定下一个事件将会持有什么样的id。

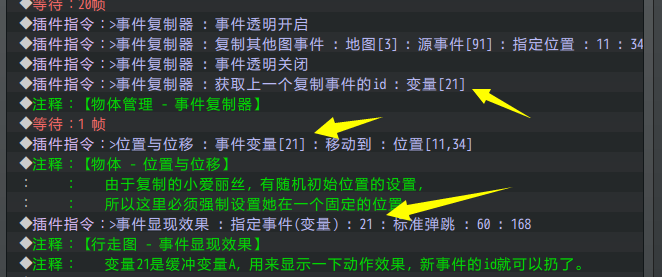
因此你需要在事件创建后，将其新id存储到变量，或者变量数组中，管理起来。

#### 2）插件指令中的“事件变量”

大多数插件指令都提供”事件变量”的功能，此功能是专门给新生成的事件用的。



比如，事件复制后。通过临时变量21来存储，然后你通过变量21记录的新事件id，传递给其他插件让其对这个新事件进行操作。



## 子插件介绍

### 【物体管理 - 玩家的事件】

#### 1）原理介绍

相关插件如下：

◆Drill\_EventDuplicator 物体管理 - 玩家的事件

插件可以创建一个事件并时刻与玩家位置重合，还够与玩家一起跨地图。

因为玩家没有 事件页 的功能，

所以使用玩家的事件，可以代替执行玩家受伤，也可以作为装扮效果。



从前面介绍中 [地图与事件机制](#_1）地图与事件机制) 可知，

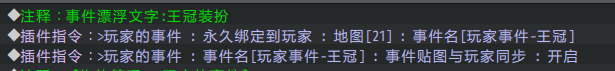
玩家离开地图后，所有事件会被销毁，并根据新地图重新创建。

因此，玩家的事件，进入不同地图后会被重新创建，因此玩家的事件id会变。

#### 2）使用方法

直接指定 地图id 和 事件名 ，即可将事件永久绑定到玩家。

绑定后是永久性的，无法解除，如果你要消除该事件，切换事件页为空事件页即可。



|  |
| --- |
| 由于玩家的事件可以跨地图，事件id会变。  因此事件id不能作为标识，而使用事件名作为标识，所以事件名不能重复。 |