## 概述

### 相关插件

核心插件：

◆Drill\_CoreOfBallistics 数学模型 - 弹道核心

◆Drill\_CoreOfShatterEffect 系统 - 方块粉碎核心

相关子插件：

◆Drill\_BattleShatterEffect 战斗 - 方块粉碎效果

◆Drill\_LayerShatterEffect 地图 - 方块粉碎效果

◆Drill\_EventShatterEffect 行走图 - 方块粉碎效果

◆Drill\_PictureShatterEffect 图片 - 方块粉碎效果

◆Drill\_DialogShatterEffect 对话框 - 方块粉碎效果

能够播放贴图的方块粉碎效果，并且还能支持倒放。

### 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：

|  |  |
| --- | --- |
| 方块粉碎 | [方块粉碎](#方块粉碎) [切割方式](#切割方式) [粉碎时长计算](#_粉碎时长计算) |
| 子插件效果 | [粉碎倒放](#_粉碎倒放) [碎片滞留设置](#_碎片滞留设置) [指令时差问题](#指令时差问题) |

### 插件关系

插件之间的关系如下：

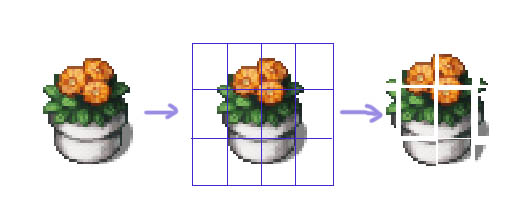


## 方块粉碎

### 定义

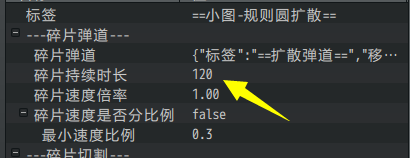
**方块粉碎：**指将目标贴图切割成数个碎片，并播放这些碎片四处分散的动画效果。

由于底层切割方式只能为横竖，因此碎片只能被切成方形碎片，所以称为“方块粉碎”。



常规情况下，方块粉碎为暂时的动画效果，碎片消失后即动画播放结束。

但是你可以在配置中自定义碎片的持续时长，或者直接控制暂停播放，使得碎片长时间滞留在界面中。详细可以去看看后面章节：[碎片滞留设置](#_碎片滞留设置) 。



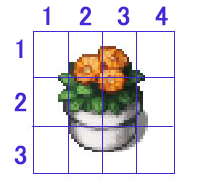
### 切割方式

**切割方式：**即方块粉碎的切割方式，分为两种：切割矩阵 和 固定大小。

切割矩阵时，

切割按照行数和列数来进行切分。

下图中为3行4列的切割情况，一共分成了12等分。



你还可以设置行数12，列数1，切割出12个横条的效果。

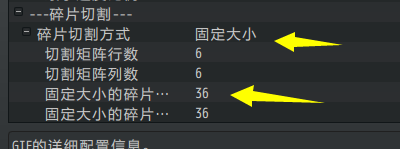
 

图中第9枚（3行1列位置）为全透明的图片，虽然表面上看不见，但第9枚碎片却是真实存在的贴图，也会播放动画效果。

固定大小时，

方块按照固定的像素值来切割。

比如固定36像素，那么每个碎片都会以36像素的方式切割。



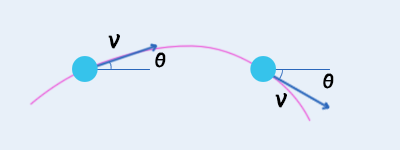
如果资源图片不够36像素整除，比如48大小，那么边缘部分会切割成36x12、12x36、12x12三种边缘大小的碎片。

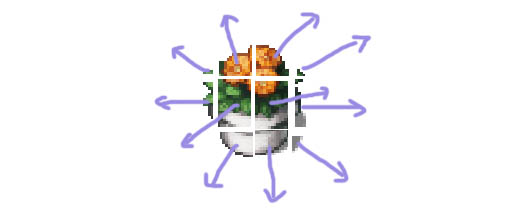


### 自定义弹道

**弹道：**是只描述一群粒子/碎片/子弹运动的轨迹。

具体弹道的内容，可以去了解下 ”32.数学模型 > 关于弹道.docx”。





方块粉碎后，由于被切成了许多片，每块碎片的弹道轨迹都不同，从而形成了碎片四处散落的效果。

**1）弹道全配置**

所有碎片按照设置的自定义弹道来进行移动。

如下图弹道设置的四周扩散效果：

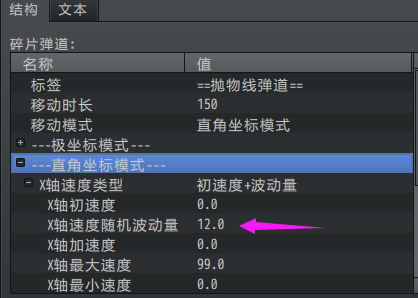


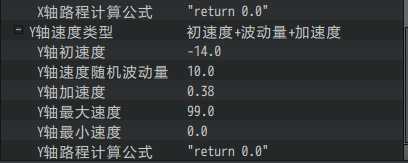
碎片的弹道具有回溯反转的功能，你可以使用插件指令倒放粉碎效果。

具体可以看后面章节：[粉碎倒放](#_粉碎倒放) 。



设计自定义粉碎时，你需要随时调整**随机波动量**，使得方块粉碎能够比较均匀地散开，不然所有碎片挤在一起，看不出粉碎的效果。



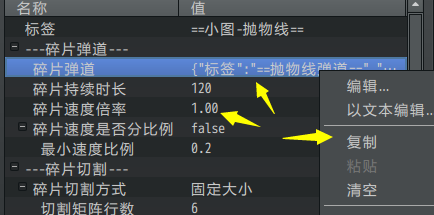


**2）复制弹道，简单配置**

你也可以照搬现成的粉碎效果，然后在其配置的基础上，简单自定义一些属性。

比如在一个已有的弹道基础上，复制，然后粘贴到新的配置中。

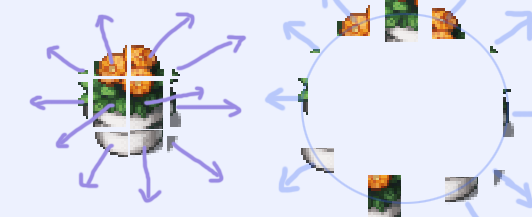
也可以调整速度倍率，让碎片散开速度的慢一点或快一点。



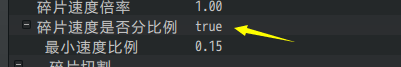
**3）比例粉碎**

通常的粉碎中，如果每块碎片的速度都一样，那么在散开时，会显得不那么均匀。

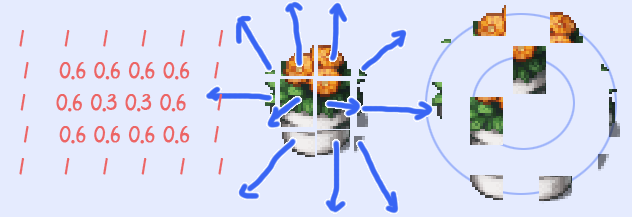
比如线性粉碎中，你能够清晰地看到一个圆向外扩散的样子。



于是，你可以考虑勾选比例粉碎，使得**最外面碎片的速度最快，里面的速度较慢**。通过设置 "碎片速度是否分比例" 可以使得碎片里外的速度不一样。



（波动量也可以让速度不一样，但是每块碎片的速度都会随机快慢。而比例速度，是里面的碎片一定慢，外面的碎片一定快。）



### 性能影响

方块粉碎是**性能消耗大户**，因为他能够将1个贴图变成大量的新贴图碎片，并且每个贴图都有自己的弹道与运动。碎片的数量 = 切割矩阵列数 x 切割矩阵行数。

注意不要设置太多贴图对象去同时播放方块粉碎效果即可。

### 粉碎时长计算

由于粉碎为并行执行，因此在设计时，还需考虑粉碎动画在特定时间内播放完的情况。

**1）未设置间隔**

如果延迟间隔为0，那么 持续时长 就是粉碎动画的实际时长。

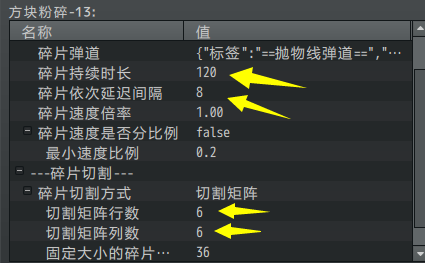


**2）设置了碎片依次间隔**

如果你设置了碎片依次延迟，那么粉碎的时间会变得很长，因为：

动画播放的时间 = 持续时长 + 碎片数量x间隔

比如下图，延迟间隔为8，切割的碎片为 6x6 块。



即得出粉碎时长：120 +（6x6）x8 = 408帧。

## 子插件效果

### 粉碎倒放

大部分子插件都支持粉碎正放与倒放：



可以去 图片管理层 看看图片粉碎然后复原的设置。

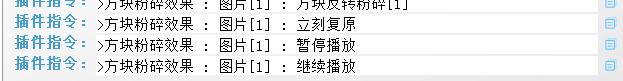


### 粉碎暂停/继续播放

**1）暂停、继续设置**

你还可以设置粉碎动画暂停、继续。

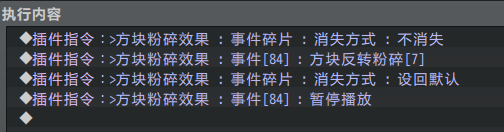
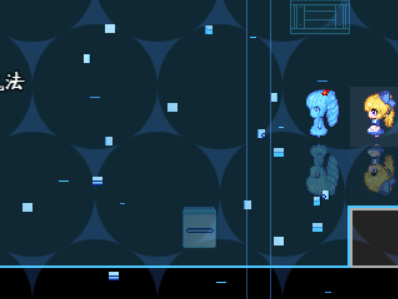
暂停后永久保持，切换菜单、保存游戏等操作都不会使碎片消失。



**2）初始就保持粉碎状态**

你可以通过下面的插件指令，使得事件在初始就处于粉碎状态。

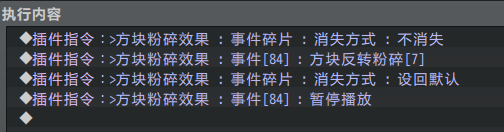
如果要播放复原动画，执行“继续播放”即可。

### 碎片滞留设置

碎片播放可以暂停/继续，再加上设置碎片不消失，就可以使碎片长期滞留在界面中。

切换菜单、保存读取时，碎片都不会还原。



通过这种滞留设置，你可以制作可打碎的花盆，并且还能将花盆复原。

可以去 互动管理层 看看。



但注意，离开地图后重新进入地图，碎片会消失。

（因为地图会重新载入事件，事件播放的粉碎碎片 也被重新载入了）

### 指令时差问题（对话框粉碎）

指令时差问题 是对话框粉碎中一个特殊的情况。

◆Drill\_DialogShatterEffect 对话框 - 方块粉碎效果

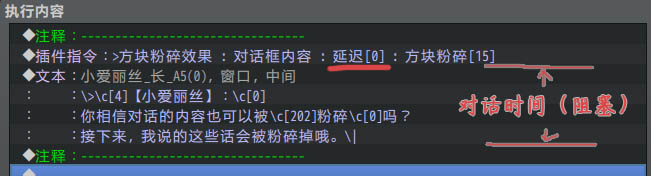
由于对话框会阻塞插件指令，不能及时播放粉碎效果。

所以只能预先设置粉碎效果，然后再执行 对话框 指令。

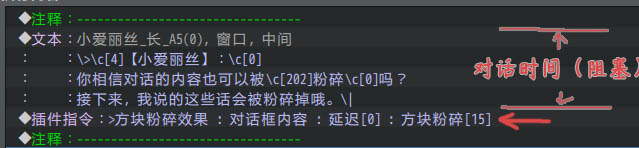
**1) 阻塞说明**

进入对话框时，对话框会阻塞事件指令，指令不会并行生效。

如果粉碎提前执行了，或者在对话框关闭后才执行，那么效果会非常差。



**过早执行**（文字还没输出来，就已经碎消失了）



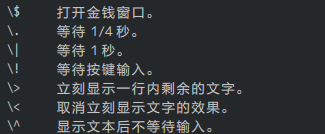
**过晚执行**（对话已经结束且关闭了，才执行粉碎）

如果执行的是 粉碎倒放 ，设置延迟[0]是可以的。

**2) 使用 消息输入字符**

对话框的持续时间是 不稳定 的，玩家如果不停地按 确定键 或者 加速键，那么很有可能因为时间太短，粉碎效果根本就执行不到。

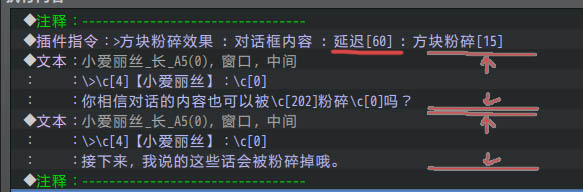
因此你需要添加一些 消息输入字符 来调整时差。



**3) 多个对话的时差**

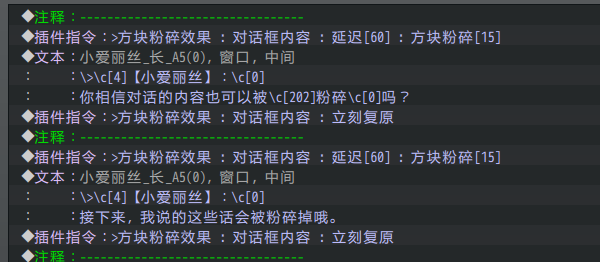
对话框粉碎有个特性：粉碎执行后永久有效。

因此，如果在第一个对话中，很短时间内就按了下一个对话，那么粉碎将会在第二个对话中执行；**如果不按第一个对话，等第一个对话粉碎，第二个对话，就是已经粉碎且消失的状态，你将看不见任何内容**。



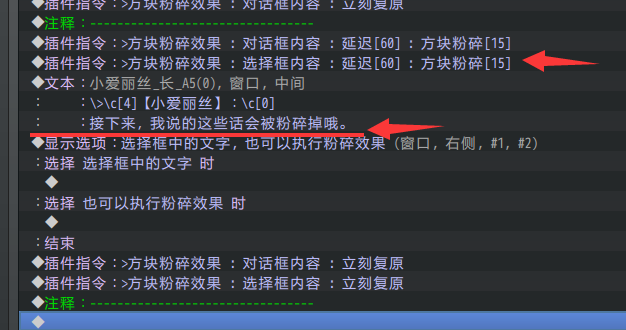
所以，一般每个对话都会执行一次 粉碎 和 立即复原。

这样可以确保每个对话都能够看见，且隔一段时间粉碎掉。



**4) 对话框子窗口的时差**

有时候 对话框和选择框 需要一起出现，如果对话框和选择框中间有任何其他指令，则这两个框都不会同时出现。为了避免分开问题，你需要提前对两个窗口进行粉碎。



**从零开始设计（DIY）**

### 设计提示

由于弹道本身的配置比较复杂，

这里建议最好先复制现成默认的配置，然后修改部分参数来看效果。



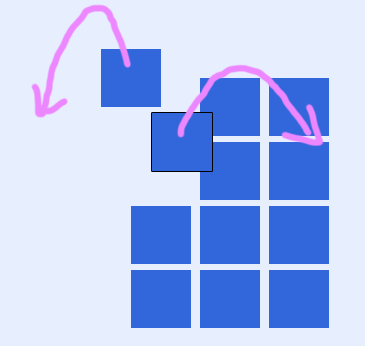
你也可以去了解下 ”32.数学模型 > 关于弹道.docx”。

### 设计一个图片粉碎效果

#### 1. 设置一个目标

首先你需要先设想或看到一个目标，来根据具体情况来尝试模拟你想象的粉碎效果。

比如作者我这里想到一种粉碎方法，碎片一个个依次掉落的动画：



#### 2. 结构规划/流程梳理

首先，要实现掉落，就要用到抛物线的弹道。

这里可以用示例中现成的，也可以自己设定。

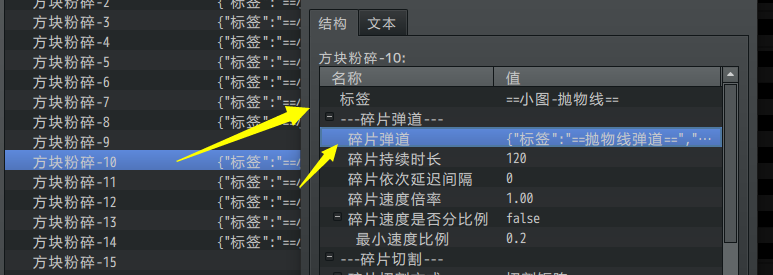
然后，由于是依次掉落，那么需要设置依次间隔。

最后，安排一个图片，并播放配置好的粉碎效果。

#### 3. 弹道设置

首先，拿一个示例的弹道作为参考。

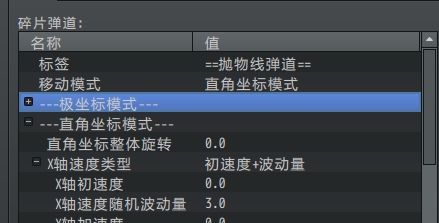
复制抛物线弹道设置，然后贴到新的弹道配置中。



其实你可以点开看看，里面使用的是直角坐标模式。

即x轴做随机匀速，y轴做加速运动。

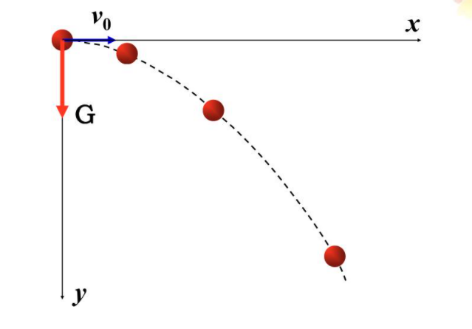
如果你想自定义碎片的速度，可以稍微修改x轴波动量和y轴初速度、加速度。





直角坐标模式，即x轴速度和y轴速度分开自定义。

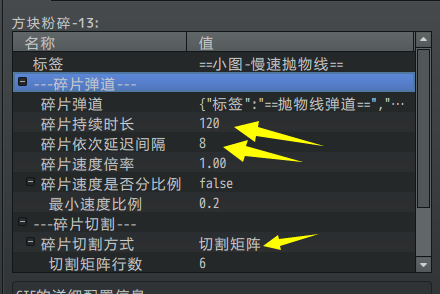
如果y初速度为0，则为平抛运动。



具体可以去了解下 ”32.数学模型 > 关于弹道.docx”。

配置了弹道之后，设置碎片的持续时长，以及依次延迟间隔。

注意，间隔不要设置的太长了，设置太长碎片会播放得很慢。

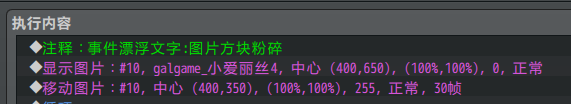


另外，这里使用切割矩阵，因为后面还需确定 碎片数量，以此推算播放动画播放时间。

如果为固定大小，那么碎片数量就不好确定，要根据图片大小来计算。

#### 4. 准备图片

编写事件指令，将一个小爱丽丝图片放在界面上。



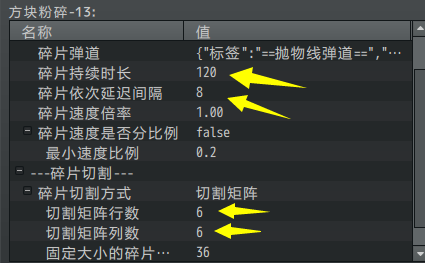
执行方块粉碎指令。



由于方块粉碎是并行效果，如果要执行复原指令，需要计算粉碎时间。

动画播放的时间 = 持续时长 + 碎片数量x间隔

即：120 +（6x6）x8 = 408帧。



在方块粉碎后，等408帧，然后再播放反转粉碎。



#### 5. 测试效果

完成上述内容，即完成了方块粉碎的设置。

测试效果图如下。



其他速度、播放时长的细节可以再回头打开 方块粉碎 的配置，去改改值。