## 概述

### 相关插件

相关插件：

◆Drill\_BattleBackground 战斗 - 多层战斗背景

◆Drill\_BattleParticle 战斗 - 多层战斗粒子

◆Drill\_BattleCircle 战斗 - 多层战斗魔法圈

◆Drill\_BattleGif 战斗 - 多层战斗GIF

◆Drill\_BattleTiledGif 战斗 - 多层战斗平铺GIF

◆Drill\_BattleVideo 战斗 - 多层战斗视频

功能扩展插件：

◆Drill\_BattleCamera 战斗 - 活动战斗镜头

上述插件可以配置多层贴图，并装饰一个具体的战斗界面。

|  |
| --- |
| 以战斗界面的一个装饰为例，如下图，两个旋转的为战斗魔法圈，小爱丽丝后面左边的变化菱形为战斗GIF。其他的都是战斗背景。战斗界面中组合最多的都是战斗背景，通过控制不同的位移比形成空间感效果。 |

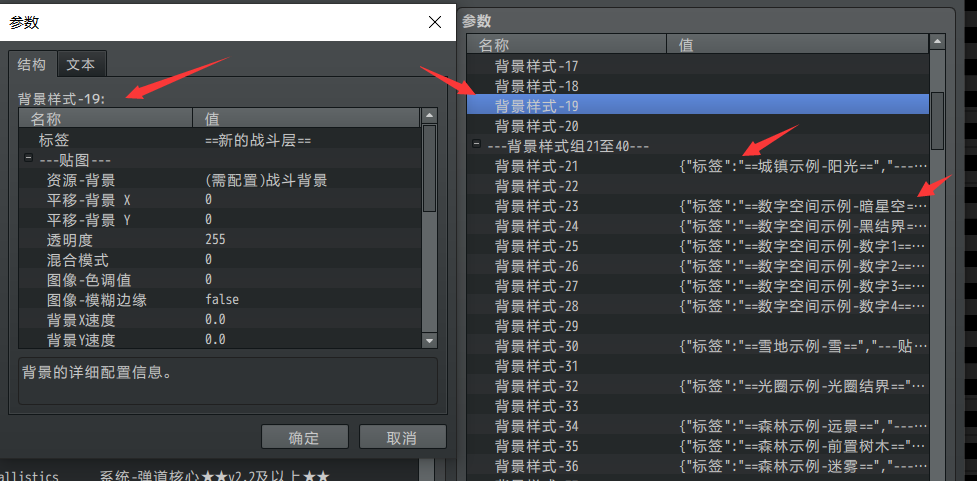
## 战斗界面设计

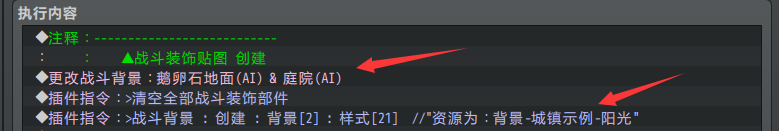
### 配置方法

#### 1）插件参数配置

战斗的装饰插件，打开参数配置即可配置。

背景只能配置样式，在使用时，需要通过插件指令建立背景，并关联样式。



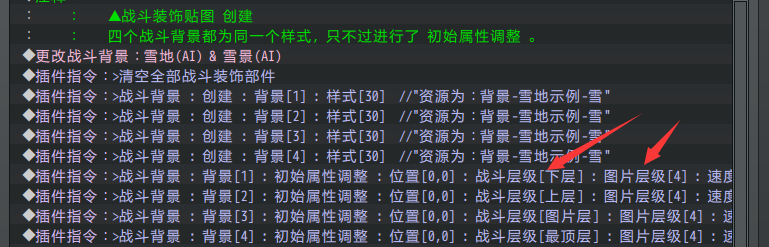


事件指令的战斗背景配置，和战斗背景可以放一起。

但是事件指令的战斗背景，固定放在最下面。（图片层级比0还低）

背景样式1、2、3之间相互独立，只在图片层级相同时，才会影响装饰的先后顺序，

你可以直接通过插件指令修改 [战斗层级](#_战斗层级) 。

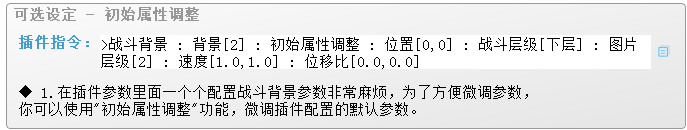


#### 2）插件指令调整

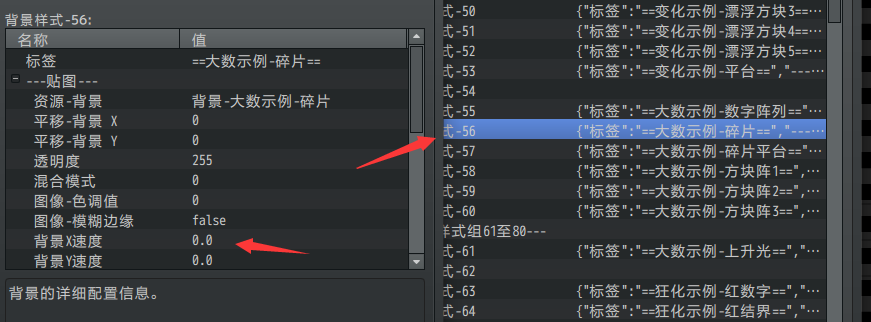
由于战斗界面要配置样式，还要配置插件指令。

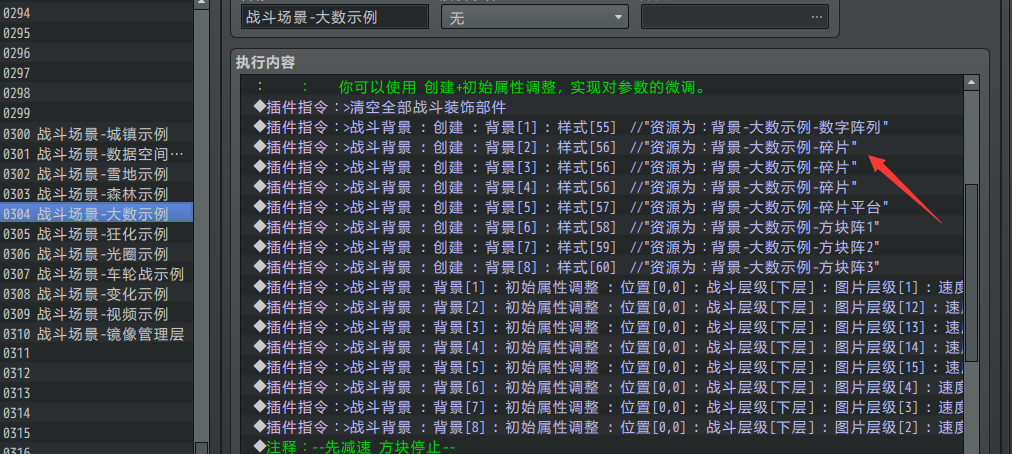
你可以只在样式中配置 图片资源，然后其他的配置，在插件指令中配置。

这样可以方便随时调整多个参数：



如下图，碎片默认不配置速度，速度在插件指令中初始化：





|  |
| --- |
| 性能方面，战斗背景设置二三十层都没问题，都不会比单个视频动画背景卡。  所以，能做成战斗背景和魔法圈的部分，尽量不要做成视频来作为背景。 |

#### 3）公共事件规范

由于战斗界面需要使用大量插件指令，并且只作用于一种情况的战斗界面。

因此，将这些战斗界面配置，统一放在公共事件中，便于随时调用管理。



战斗场景装饰 与 敌群 是分离的两个部分。

因此，你可以在进入敌群战斗前，使用相同的战斗界面装饰。

如下图，示例中很多不同的 敌群 示例（战斗处理），

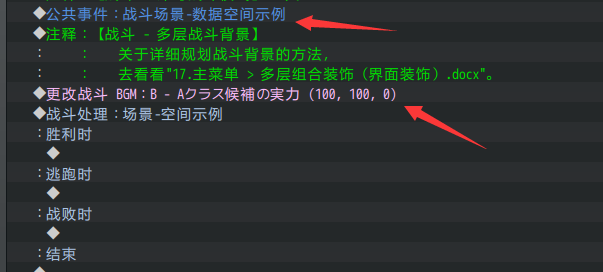
都使用相同的 战斗场景-数据空间 的装饰。



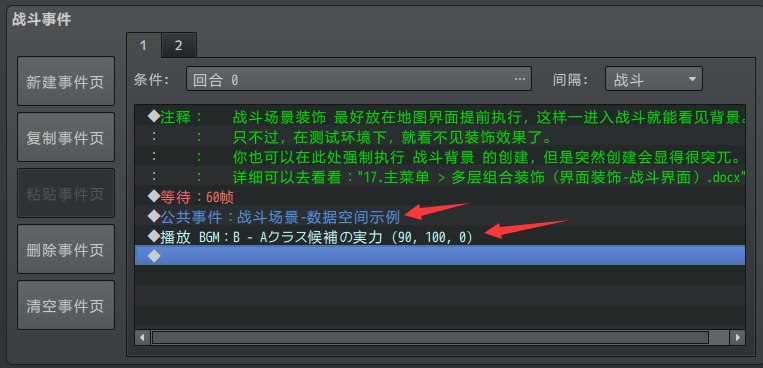
#### 4）提前创建 与 立即创建

**提前创建：**指在地图界面中，提前执行战斗场景的布置。

提前创建时，也可以修改战斗BGM。

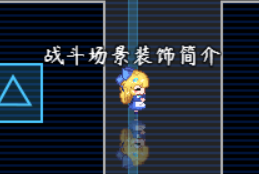


**立即创建：**指在战斗界面中，对战斗场景进行布置。



常规的战斗界面都应该提前创建，因为在战斗时立即切换背景会看到突兀的图片变换。

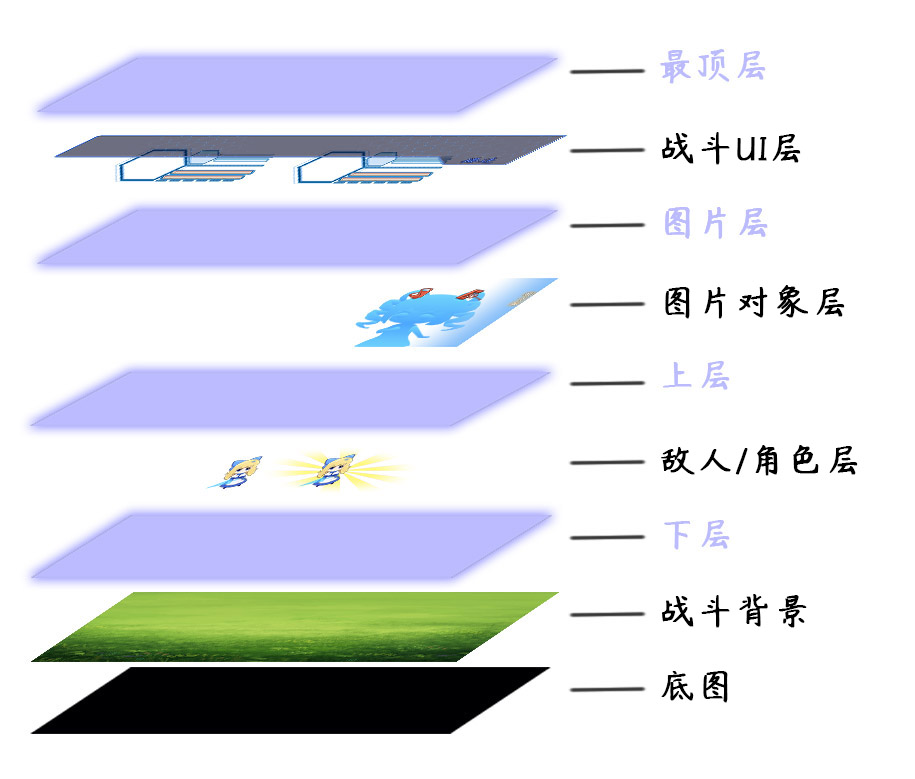
详细说明可以去看看示例中 战斗管理层 的 场景装饰简介 。



### 战斗层级

#### 1）层级介绍

**战斗界面分为四层，下层、上层、图片层、最顶层。**



> 由于游戏自身底层设置的关系，对话框、图片、敌人、背景都是不可分割的大层次，并且，对话框一定在图片上面。因此如果你想让图片被放在对话框上面，只能通过其他插件来辅助。

> 部分插件的图片可能会直接被固定在界面的某个层中，比如视频背景，是固定在下层，并且图片层级为0。

（图片层级也称”堆叠级”，可以去看”0.基本定义 > 界面.docx”的层级与堆叠级章节）

> 如果你不想要战斗背景，可以直接在下层设置完全不透明的背景，挡住就可以了。

|  |
| --- |
| 游戏中还存在一个特殊的层级：天窗层。  此层级比最顶层还高，在该层级中能够使用 动态屏幕快照 。  详细可以去了解下文档：“1.系统 > 大家族-屏幕快照.docx” |

#### 2）底图说明

底图是战斗层级的底层功能，用于显示 地图界面 的**模糊图像**。

（清空所有战斗装饰时，能看见底图）

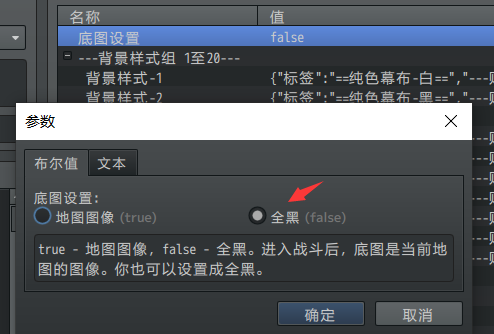
这个图像的作用是让你感觉自己像在地图界面。

然而，战斗界面的敌人图像明显大的多，所以这里的底图反而很突兀，所以默认涂黑。



如果你不想看到底图，你可以把它涂黑，通过插件修改：

◆Drill\_BattleBackground 战斗 - 多层战斗背景



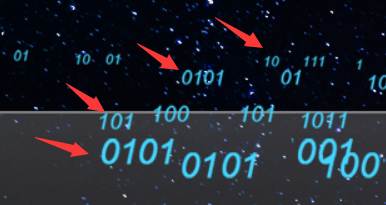
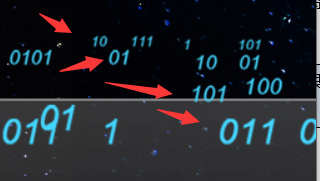
### 位移比设计

**位移比：**是指 战斗活动镜头 移动时，战斗各个装饰贴图 跟随镜头一起移动的距离比。 通过位移比可以实现不同图层之间的距离感。

|  |
| --- |
| 根据近大远小的关系，物体离你越远，物体越小，并且你移动画面时，也一直紧跟着你的镜头。位移比越接近1，物体越远。 |

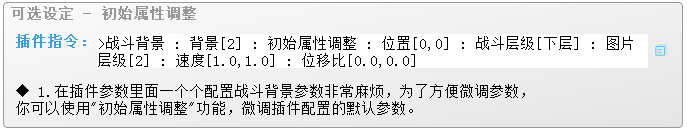
比如 战斗场景-数据空间示例 中，

如果你的镜头上移，会发现背景数字会 向下/向上 展开小范围幅度。

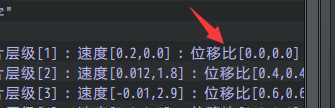


|  |
| --- |
| 由于战斗界面的镜头移动幅度明显小的多（镜头架矩形 减去 镜头矩形 的活动范围），所以在位移比为0.9、0.8之间时，几乎看不出层次效果。  所以一般都刻意加大位移比的变化，比如0.25，0.5，0.75这种跨度，不同背景之间有一定距离的位移，就能产生层次感了。 |

如果要设计位移比，可以设置初始属性调整，将所有参数铺开，再依次修改位移比。



位移比分为X方向和Y方向，写一样的值即可。



### 粒子与背景设计

有时候背景也可以作为粒子来使用，因为战斗场景有可能会飘大量粒子，但是粒子数量太多会加大战斗界面的消耗。

◆Drill\_BattleBackground 战斗 - 多层战斗背景

◆Drill\_BattleParticle 战斗 - 多层战斗粒子

远处的粒子可以使用背景来画，近处的用粒子来实现。

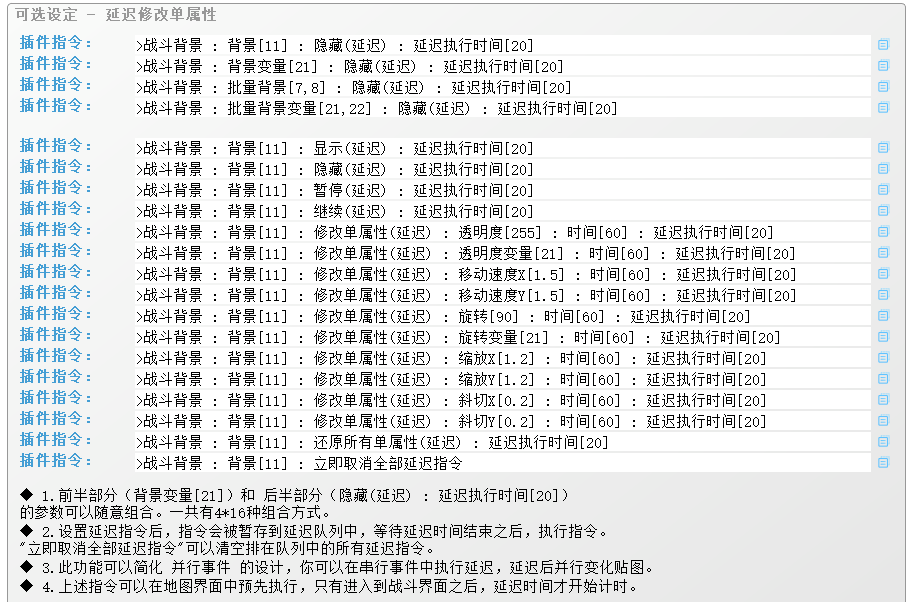


### 延迟指令设计

|  |
| --- |
| 战斗装饰插件的指令较多且使用频繁，建议使用小工具：插件信息查看器。  在开启游戏编辑器时，可以并行使用读取器复制指令。 |

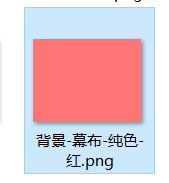
简单的延迟指令可以看看公共事件 战斗场景-城镇示例 和 战斗场景-雪地示例 。

复杂一点的可以看公共事件 战斗场景-变化示例 。



总之，战斗开始前提前创建背景、魔法圈、GIF等。

一开始不出现的装饰贴图，设置透明度为0即可，延迟在特定的时间，再显现。



示例中提供了一些纯色的幕布，使用 纯色幕布+混合模式2 可以制作出简单滤镜的效果，通过延迟指令，可以让这些幕布在战斗音乐的特定位置变色。



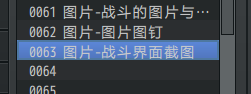
## 扩展效果

### 截取空白战斗界面

需要用到下面的插件：

◆Drill\_PictureSnapShot 图片 - 临时屏幕快照

示例中可以看看下面这个敌群设置：



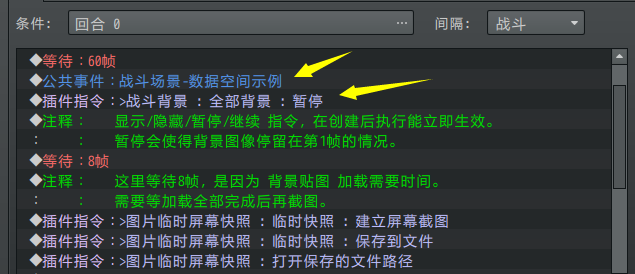
你可以把小爱丽丝敌人放在屏幕左上角藏起来。



在战斗界面中建立战斗场景，然后设置 暂停 所有背景。

等8帧左右（等贴图加载完毕），再截图。

截图可以在测试战斗的时候执行。



截图完毕后你能得到空白的战斗截图。

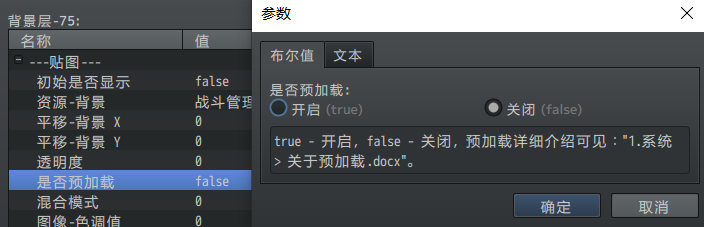


### 预加载

战斗装饰插件可以设置预加载。

预加载一般用于游戏中多次、反复使用的贴图，

比如战斗的动画转场用的贴图。



但是注意，并不是所有装饰贴图都要配置预加载。

预加载会使得贴图创建后一直占着内存，直到游戏关闭才释放内存。

具体去看看： “1.系统 > 关于预加载.docx”。

## 从零开始设计（DIY）

### 设计一场战斗的场景效果

#### 1. 设置一个目标

2018年8月13日，作者我建立了插件示例集合最早的v1.01版本。

使用了默认雪景素材+早期mog战斗背景插件，制作了雪地与雪花图层飘落的场景。

后来经过一系列的插件更新，推翻mog战斗背景插件，还把素材换成了AI生成的雪景。

虽然很多东西进行了替换，但是雪地战斗场景的设计并没有变多少。

因此，这里作者我整理并讲解一下，如何使用当前素材+插件，在新的工程中设计雪地战斗场景。





#### 2. 结构规划/流程梳理

首先战斗界面需要考虑的问题比较多，需要按配置方法步骤来：

配置样式 > 建立公共事件 > 编写插件指令创建 > 根据创建时机执行公共事件

这些步骤也分别对应了前面章节的讲解：

[1）插件参数配置](#_1）插件参数配置)

[2）插件指令调整](#_2）插件指令调整)

[3）公共事件规范](#_3）公共事件规范)

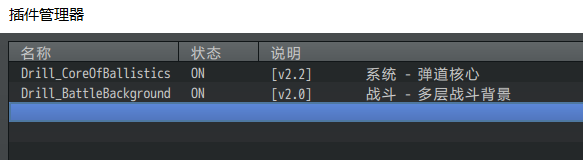
[4）提前创建 与 立即创建](#_4）提前创建_与_立即创建)

#### 3. 空白工程准备

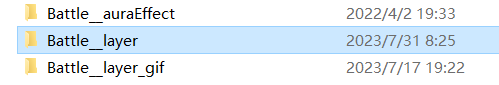
首先新建一个工程。



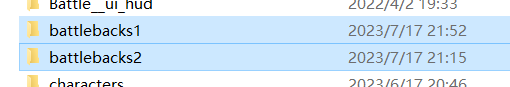
在工程中加入多层战斗背景插件。



将插件相关资源文件夹复制到工程中。



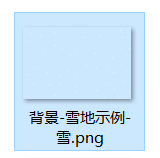
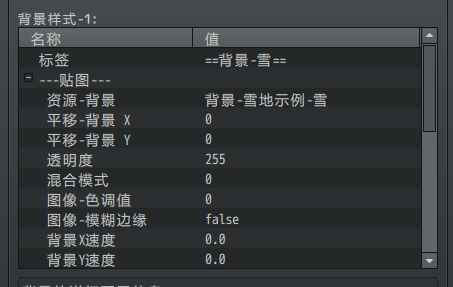
别忘了默认文件夹中的背景。



#### 4. 配置背景样式

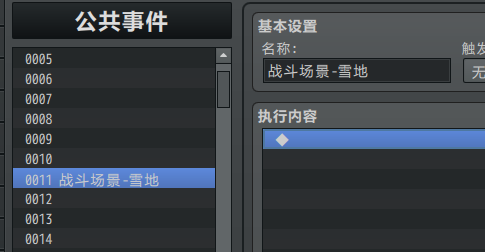
配置战斗背景样式前，可以先了解下配置技巧：[配置方法](#_配置方法) 。

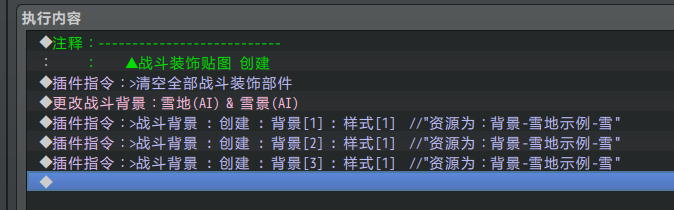
在背景样式中，配置资源路径即可。



建立一个公共事件，这个公共事件专门打包雪地的战斗场景。

公共事件中编写插件指令去创建最基本的雪背景。





完成公共事件后，在地图界面，建立一个事件用于测试战斗。

先执行 公共事件来初始化雪地场景，再执行战斗处理 即可。



#### 5. 测试效果

测试游戏，可以看到 静态的 雪景和雪花。



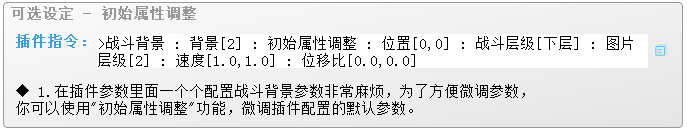
这是因为初始背景的X速度和Y速度都为零，所以雪花没有移动。

而且三个背景叠在一起，看不出来。

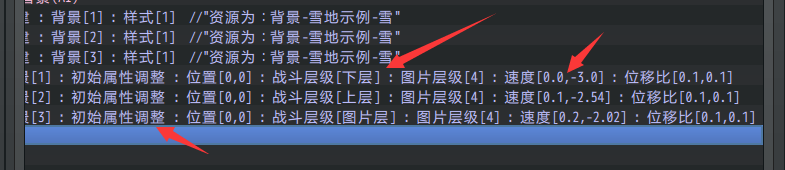
#### 6. 初始属性调整

你可以在背景样式中设置不同的X速度和Y速度，但这样就要配置三个背景样式。

为了节约样式，你也可以在设置一个样式后，使用下面的插件指令修改：



通过设置 初始属性调整 指令，修改不同背景的速度。（这些背景都是相同的样式）



这时候在游戏中测试，可以看到雪花的背景开始移动了。



初始化参数中还有位移比的设置，位移比与战斗活动镜头有关。

◆Drill\_BattleCamera 战斗 - 活动战斗镜头

由于当前没有配置战斗镜头，所以此参数不会生效。

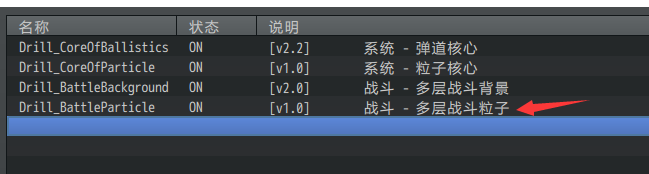
（你可以加上镜头插件）

#### 7. 添加多层粒子

除了平铺的背景雪，这里可以再加上零散的粒子雪。

◆Drill\_BattleParticle 战斗 - 多层战斗粒子

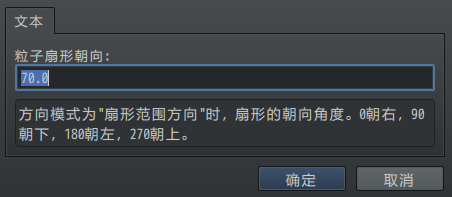
将战斗粒子插件加到工程中。



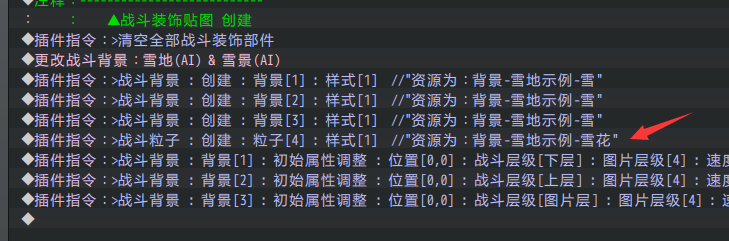
粒子样式的配置不太一样，基于 速度+方向 来控制。

粒子从上方落下，飘落方向为右下（90度~0度左右）。





完成样式后，在公共事件中添加粒子即可。

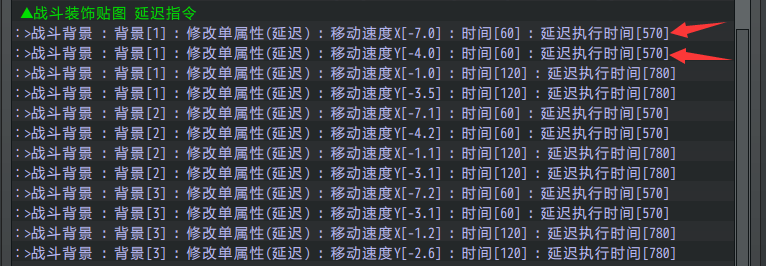


|  |
| --- |
| 由于粒子的参数配置比较多，需要对很多配置进行微调。  如果你觉得麻烦，那么跳过粒子的配置，就只用背景即可。 |

#### 8. 延迟指令设计

在战斗开始后的一段时间，你可以设置雪突然被风吹并移动了一下的效果。

在570帧时，加速移动，210帧后，减速并恢复正常的飘落速度。



注意，由于粒子不具备修改移动速度的功能，需要使用其他的插件指令来实现。

比如增减速移动，也能大概模拟出被风吹的效果。

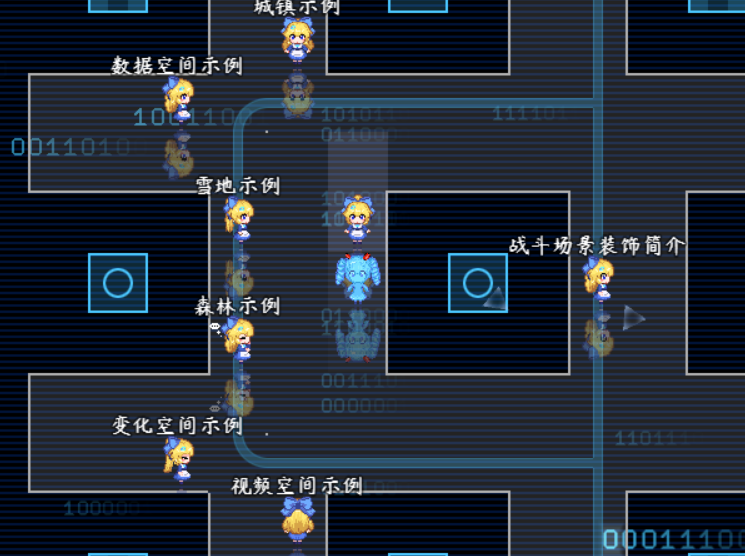


#### 9. 示例中可参考对象

示例中，已将所有战斗场景打包整合到了公共事件中。



详细可以去战斗管理层看看：



### 设计一个无缝切换的遇敌动画转场

#### 1. 设置一个目标

2023年7月23日，最近看主播玩游戏《取景器》。

这个游戏，只要看传送门的屏幕，点击进入后，里面的屏幕就会放大，

无缝切换到新的地图场景中。

作者我之前一直都没有 地图界面 转 战斗界面 的设计灵感。

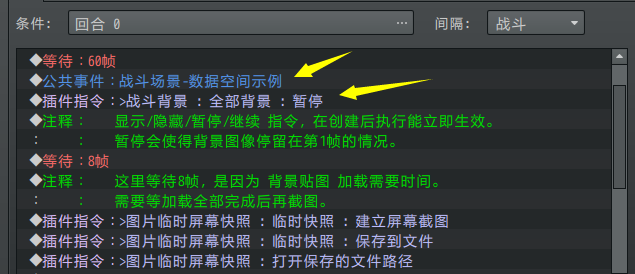
看到这个游戏之后，突然就有灵感了，想开坑了。

#### 2. 结构规划/流程梳理

【未完成】

图在加载时，需要消耗时间，无论性能再好的电脑，都需要至少1帧的加载时间。

因此，我们需要在创建背景后，立即设置暂停，确保所有贴图都处于第1帧的状态。



因为暂停，所以背景始终处于第1帧的状态。

然后用第1帧的截图作为过渡图片，能实现无缝画面同步的效果。

|  |
| --- |
| 战斗装饰插件脚本的底层 使用的是 数据-贴图 分离的结构。  你可以直接对背景数据设置显示/隐藏/暂停/继续，不需要等贴图加载完了才操作。 |

### 设计动画转场

魔法圈、GIF 也可以通过变换，实现转场动画效果。

示例中 战斗管理层 介绍了 平移、旋转、缩放、斜切 的变换功能。



并且，动画转场效果在示例中 动画转场管理层 也有设计介绍。

可以去看看：“16.图片 > 特效设计-动画转场.docx”。