## 概述

### 基础要求

|  |  |
| --- | --- |
| 先尝试回答下面的问题：  什么是GIF动画序列？  什么是状态元和状态节点？  状态元和状态节点相互如何组合嵌套？  是否已完成 ”配置GIF文件并在游戏中播放” 流程？ | 需要先了解基础知识哦！ |

如果你对上述问题有疑问，那么说明你还不了解 动画序列。

需要去看看：”小工具 > GIF动画序列编辑器” 中 入门篇+高级篇 的知识。

|  |
| --- |
| 小工具>入门篇 教你如何用小工具配置GIF，并转入到游戏中并播放。  小工具>高级篇 介绍更复杂的GIF配置，并在游戏中执行复杂播放。  本文档 专门介绍行走图的**复杂播放机制**，  其中包含 标签播放+状态节点、全绑定、站桩/二方向/四方向 的配置介绍。 |

### 插件介绍

基于核心：

◆Drill\_CoreOfActionSequence 系统 - GIF动画序列核心

◆Drill\_CoreOfEventFrame 行走图 - 行走图优化核心

当前插件：

◆Drill\_EventActionSequence 行走图 - GIF动画序列

◆Drill\_EventActionSequenceAutomation 行走图 - GIF动画序列全标签播放

◆Drill\_EventActionSequenceBind 行走图 - GIF动画序列全绑定

你可以把行走图 **变成**GIF动画序列，也可以还原。

你可以先了解 [定义](#_定义) 和 [快速理解](#_快速理解) 。

全标签播放可以去看看：[动画序列标签表](#_动画序列标签表) 。

|  |
| --- |
| 插件：~~Drill\_EventFrameGif 行走图 - 多层行走图GIF~~  是行走图的个体装饰，独立的功能，与动画序列没有任何关系，注意区分。 |

|  |
| --- |
| 动画序列插件会把 行走图 **变成** 动画序列，  而另一类插件： Drill\_EventFrameNumber 行走图 - 多帧行走图  是对原行走图的**功能延伸**，可以看看文档：“7.行走图 > 关于多帧行走图.docx” |

### 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：

|  |  |
| --- | --- |
| 行走图动画序列 | [行走图动画序列](#行走图动画序列)  [站桩动画序列](#站桩动画序列) [四方向动画序列](#四方向动画序列)  [二方向动画序列](#二方向动画序列) [八方向动画序列](#八方向动画序列) |
| 全标签播放 | [全标签播放](#全标签播放) [标签播放](#标签播放)  [行走图的姿势](#_行走图的姿势) |
| 全绑定 | [全绑定](#全绑定) |

### 插件关系

行走图动画序列的关系如下图。

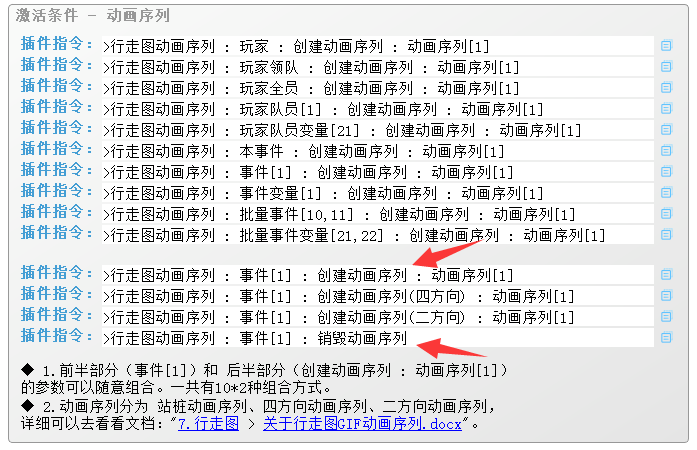


## 行走图动画序列

### 定义

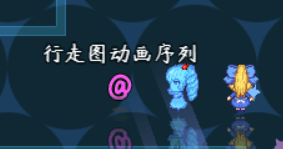
**行走图动画序列：**指专门作用于 行走图 的动画序列贴图结构。

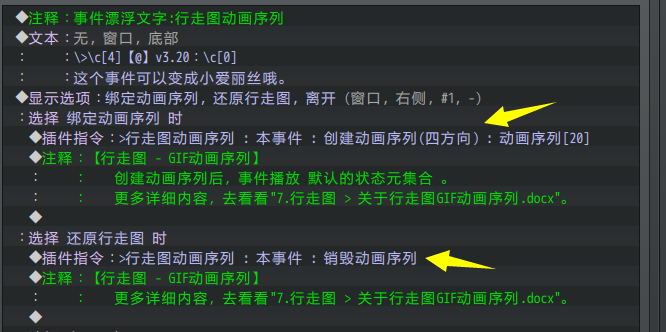
可以通过插件指令创建/销毁动画序列，并且能播放状态元/状态节点/动作元。



创建动画序列时，原行走图会被替换成动画序列的图像。

销毁动画序列时，会还原到原行走图。





动画序列创建后，会播放 默认的状态元集合。

注意，这个小爱丽丝移动时，不会播放行走动画。（需要开启 全标签播放 ）

### 类型

行走图动画序列分为下面几类：

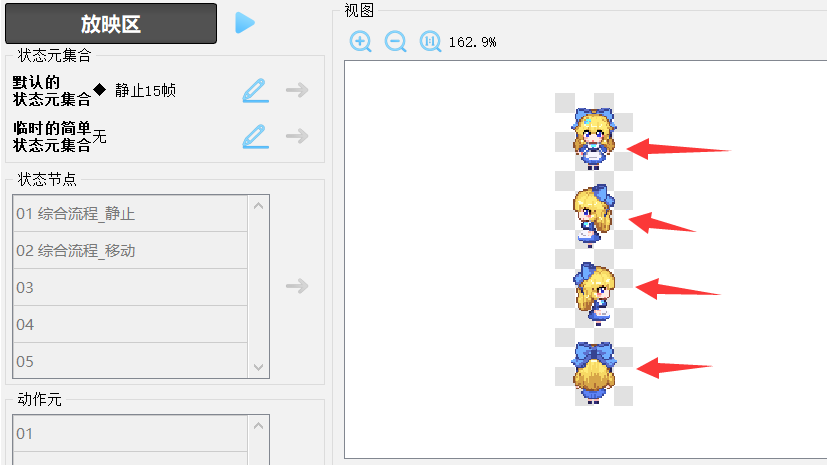
**站桩动画序列：**指通过插件指令“设置动画序列”来绑定的动画序列。

生成的动画序列就是单朝向的站桩图像。

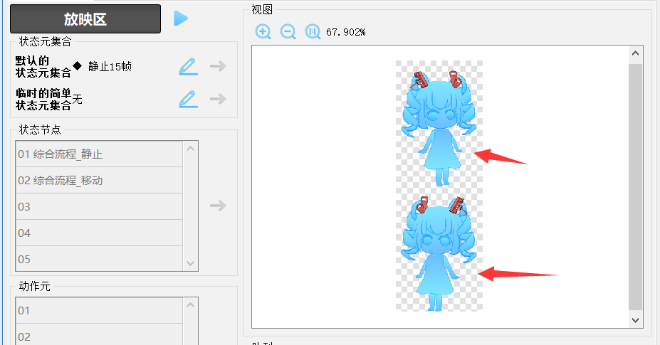
**四方向动画序列：**指通过插件指令“设置动画序列(四方向)”来绑定的动画序列。

生成的动画序列将根据 物体 的朝向，自动切割分配图像。

**二方向动画序列：**指通过插件指令“设置动画序列(二方向)”来绑定的动画序列。

动画序列只配置左右两个方向（资源朝向为先右后左），物体上下移动时，保持显示朝左或朝右的图像。

**八方向动画序列：**理论上存在该类型，但是插件暂时没实现此功能，等以后作者开坑。

### 快速理解

#### 1）物体与行走图

物体是指地图中的独立对象，也一般指 玩家/事件 。

每个物体都绑定一个行走图，而行走图的配置可以在事件页中设置。



|  |
| --- |
| 游戏中事件页切换时，物体会根据配置的行走图，改变自身的行走图。 |

玩家的行走图在角色配置中配置。



由于行走图每次只能绑定一个资源图片，

因此可以扩展的功能有限。

|  |
| --- |
| 插件： Drill\_EventFrameNumber 行走图 - 多帧行走图  是对原行走图的**功能延伸**，可以去了解下文档：“7.行走图 > 关于多帧行走图.docx” |

#### 2）物体与动画序列

动画序列是一个大的数据容器，

下面包含了 状态元、状态节点、动作元 等结构。

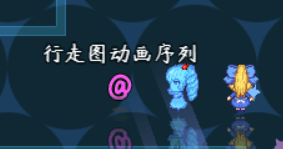


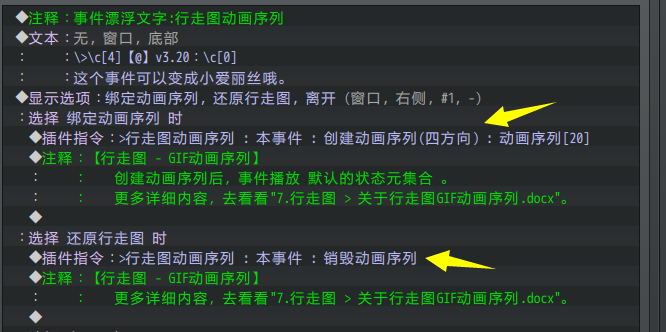
因此在编辑器中，动画序列不会主动绑定物体。

而需要 事件注释或者插件指令 手动控制绑定。

创建动画序列时，原行走图会被替换成动画序列的图像。

销毁动画序列时，会还原到原行走图。





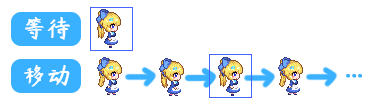
动画序列创建后，会播放 默认的状态元集合。

注意，这个小爱丽丝移动时，不会播放行走动画。（需要开启 全标签播放 ）

#### 3）动画序列与行走图的设计

这里简单介绍一下 多帧行走图 与 GIF动画序列的设计区别：

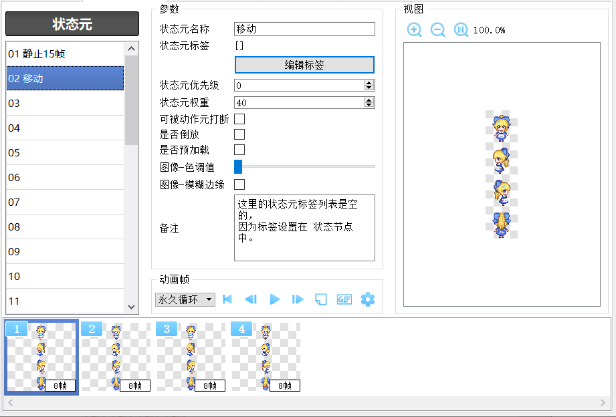
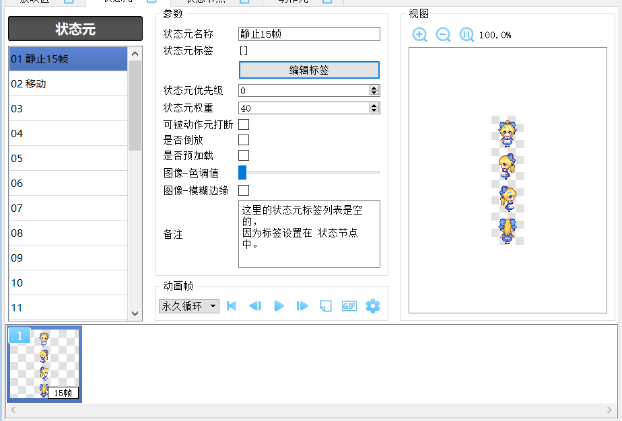
**多帧行走图设计：**行走图资源有 静止、移动 两种状态，只需要一张行走图即可设计，但是静止状态只有的一帧。



|  |
| --- |
| 如果你只希望让行走图播放单一的GIF，不需要动画序列那么复杂，  可以去看看多帧行走图的文档：“7.行走图 > 关于多帧行走图.docx” |

**GIF动画序列设计：**在小工具GIF动画序列编辑器中，定义两个状态元：静止、移动，并分别配置GIF。

完成配置后，分别使用标签：”<行走图-静止>” 和 ”<行走图-移动>” 对应到状态。



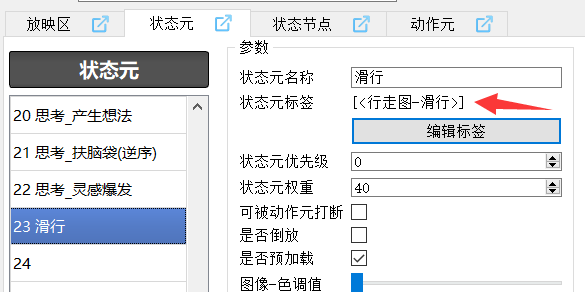
|  |
| --- |
| 通过动画序列的状态元设置与标签设置，实现了与 多帧行走图 一模一样的效果。  即根据 移动、静止 状态自动播放不同的图像。 |

#### 4）动画序列进阶配置 - 更多的标签

除了前面介绍的这两个基本标签，还有更多可以用的标签：

“<行走图-奔跑>”、“<行走图-跳跃>”、“<行走图-滑行>”……等。

这些标签都能在特定的动作条件下播放出来。





|  |
| --- |
| 上述的滑行动作，就来自于插件 图块-物体滑行 。  所有可用的标签以及所需的插件，可见后面章节：[动画序列标签表](#_动画序列标签表) |

#### 5）动画序列进阶配置 - 更复杂的动画

GIF动画序列有三类子结构：状态元、状态节点、动作元。

其中，状态节点与状态节点 之间能相互连接，

并且，状态节点与状态元 之间能相互连接。

连接能设置 随机播放或顺序播放 ，通过层层嵌套，播放出复杂的GIF动作。

比如在 <行走图-静止> 时，小爱丽丝会 照镜子、拍裙子、整理头发 等。



这些设置都在小工具内实现，

让动画序列播放出 随机组合的GIF切片。

可以去看看 小工具 中的文档 ”关于GIF动画序列核心编辑器.docx”。



### 完整配置流程

|  |
| --- |
| 此功能要求 **先配置** GIF动画序列的状态元/状态节点/动作元。  在进行 **后配置** 标签、绑定，作用于地图的所有物体的行走图。 |

完整的行走图的各个配置流程之间场景如下：



全标签可见后面章节：[动画序列标签表](#_动画序列标签表) 。

## 全标签播放

### 定义

**标签播放：**指 通过标签控制 播放 状态元/状态节点。

**全标签播放：**指行走图有一套自定义的标签，通过那些标签自动播放 状态元/状态节点。

来自插件：

◆Drill\_EventActionSequenceAutomation 行走图 - GIF动画序列全标签播放

|  |
| --- |
| 全标签播放通过标签控制，（如”<行走图-静止>”、”<行走图-移动>”）  动画序列如果没配标签，则全标签播放无效，只播放 默认的状态元集合 。 |

全标签播放插件会根据时机，调用指定标签，

然后执行 播放状态节点/播放状态元 功能，如下图。



|  |
| --- |
| 注意，全标签播放 只根据标签 播放GIF。  只要你的动画序列里有设置 标签 就能播放，  播放的动画序列可以是 [站桩动画序列](#站桩动画序列) ，也可以是 [四方向动画序列](#四方向动画序列) ，不要混淆了。 |

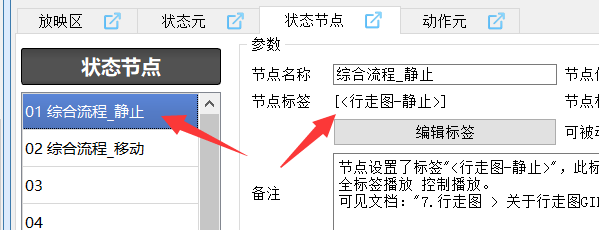
### 设置标签播放

标签播放的效果可结合示例 动画序列管理层 ：



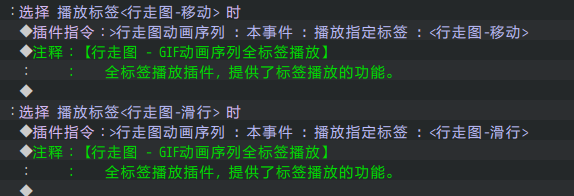
小工具中，在 状态元/状态节点 上加标签即可，可以加很多不同的标签。

注意，**相同标签**只加一个就能生效，如果多加，则会找编号最小的播放。

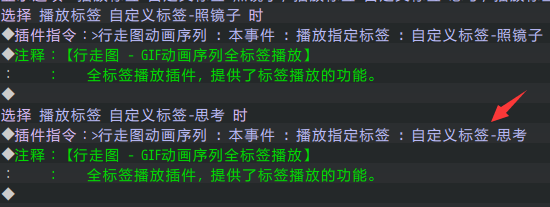
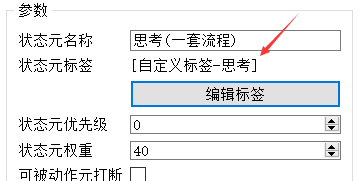


游戏中，通过插件指令即可 手动控制 标签播放。

如果没有配置对应的标签，那么物体执行播放不会有任何效果。



标签可以自由定义，只要文本对应即可。



但是行走图的标签固定为”<行走图-静止>”、“<行走图-移动>”，

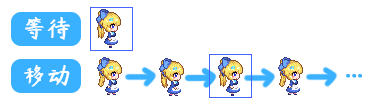
可见后面的标签表：[动画序列标签表](#_动画序列标签表) 。

### 行走图的姿势

#### 1）多帧行走图设计

原行走图资源有 静止、移动 两种状态，

只需要一张行走图即可设计，但是静止状态只有的一帧。

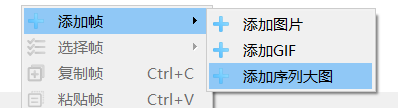


|  |
| --- |
| 如果你只希望让行走图播放单一的GIF，不需要动画序列那么复杂，  可以去看看多帧行走图的文档：“7.行走图 > 关于多帧行走图.docx” |

#### 2）GIF动画序列设计

在小工具GIF动画序列编辑器中，需要定义两个状态元：静止、移动。

可以导入原来的行走图，通过添加帧，导入 序列大图 来配置GIF。



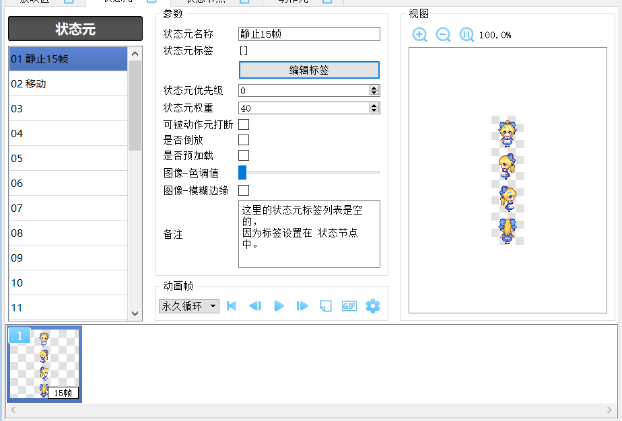
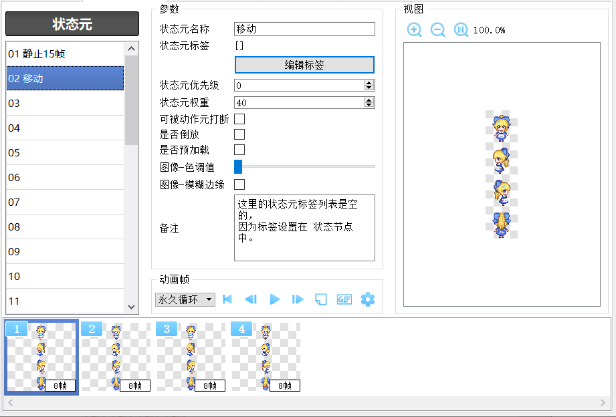


其中，静止设置1帧（也可以设置多帧），

移动设置4帧（静止->左脚->静止->右脚 的播放顺序），

设置后，分别加上标签：”<行走图-静止>” 和 ”<行走图-移动>” 即可。

详细配置细节可见 从零开始设计 章节：[3. 配置状态元](#_3._配置状态元_1) 和 [4. 状态节点与标签设计](#_4._状态节点与标签设计) 。

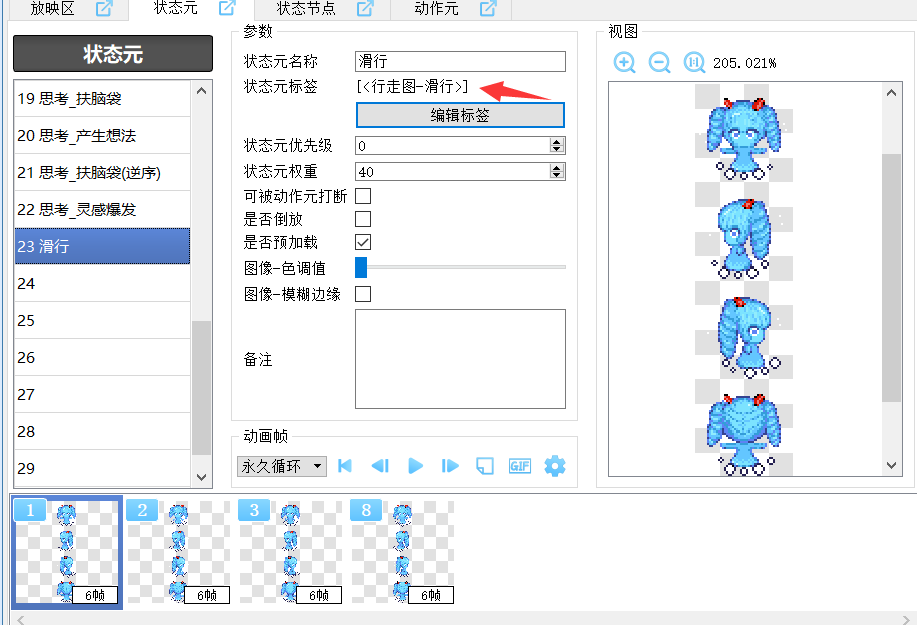
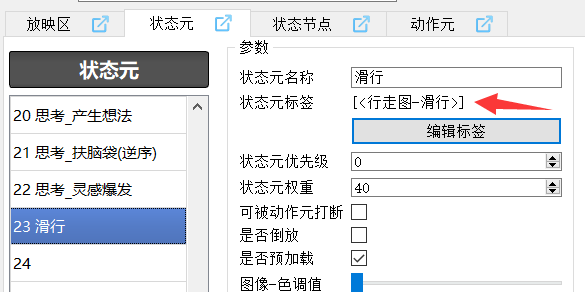
|  |
| --- |
| 通过动画序列的状态元设置与标签设置，实现了与 多帧行走图 一模一样的效果。  即根据 移动、静止 状态自动播放不同的图像。 |

#### 3）多标签设计

除了前面介绍的这两个基本标签，还有更多可以用的标签：

“<行走图-奔跑>”、“<行走图-跳跃>”、“<行走图-滑行>”……等。

这些标签都能在特定的动作条件下播放出来。





|  |
| --- |
| 上述的滑行动作，就来自于插件 图块-物体滑行 。  所有可用的标签以及所需的插件，可见后面章节：[动画序列标签表](#_动画序列标签表) |

### 动画序列标签表

所有可用的动画序列标签如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **物体行为** | **描述** | **播放标签** | **优先级** | **所属插件** |
| 静止 | 物体原地站立时。 | <行走图-静止> | 1 | 无 |
| 移动 | 物体正在移动时。 | <行走图-移动> | 1 | 无 |
| 奔跑 | 物体奔跑时。（事件无法奔跑，只有玩家能奔跑） | <行走图-奔跑> | 5 | 无 |
| 跳跃 | 物体跳跃时。 | <行走图-跳跃> | 10 | 无 |
| 滑行 | 物体站在冰面上，且正在移动时。 | <行走图-滑行> | 2 | Drill\_LayerSlipperyTile  图块 - 物体滑行 |
| 滑行 | 物体站在冰面上，且原地站立时。 | <行走图-滑行静止> | 2 | Drill\_LayerSlipperyTile  图块 - 物体滑行 |
| 被举起 | 物体被举起时。 | <行走图-被举起> | 15 | Drill\_PickThrow  互动 - 举起花盆能力 |
| 举花盆 | 物体举起花盆时。 | <行走图-举花盆> | 15 | Drill\_PickThrow  互动 - 举起花盆能力 |

优先级为固定值，播放时会选择优先级最高的标签。

基于 状态元/状态节点/动作元 的播放原理，上述物体的行为全都并列且独立，播放某种行为时，无法同时播放其它行为。

## 全绑定

### 定义

**全绑定：**指根据 行走图资源 自动绑定创建动画序列。

（你可以通过 事件注释 手动一个个加，但全绑定明显更适合游戏中大量个体的情况）

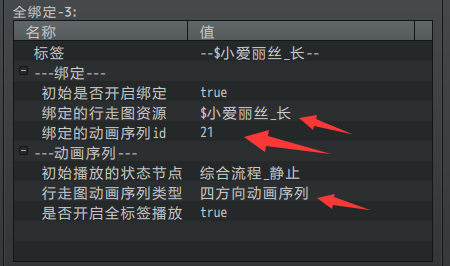
来自插件：

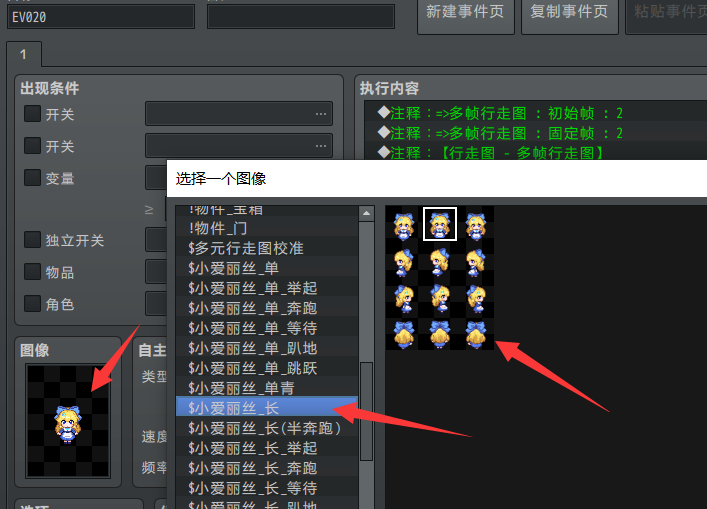
◆Drill\_EventActionSequenceBind 行走图 - GIF动画序列全绑定

|  |
| --- |
| 事件 默认都用简单行走图，不含动画序列。  全绑定能一次性给 所有事件 进行动画序列创建。 |

如下图的例子，全绑定后，所有行走图资源为 ”$小爱丽丝\_长” 的事件，

自动绑定创建动画序列，并 开启四方向、开启全标签播放 。





|  |
| --- |
| 全绑定还可以设置 站桩行走图、二方向行走图，  如果设置错了，你可能会看见被切成多片的行走图，或者多个不同朝向的行走图。  留意一下此细节，把全绑定设置改正确就好了。 |

### 行走图切换时

全绑定也支持 事件切换其他的行走图，

切换时，全绑定会 立即关闭动画序列 并还原行走图。

|  |
| --- |
| 由于一次性影响所有事件，  如果你希望一些小爱丽丝不被 全绑定，那么配置不一样的资源名称即可。 |

### 误用提示

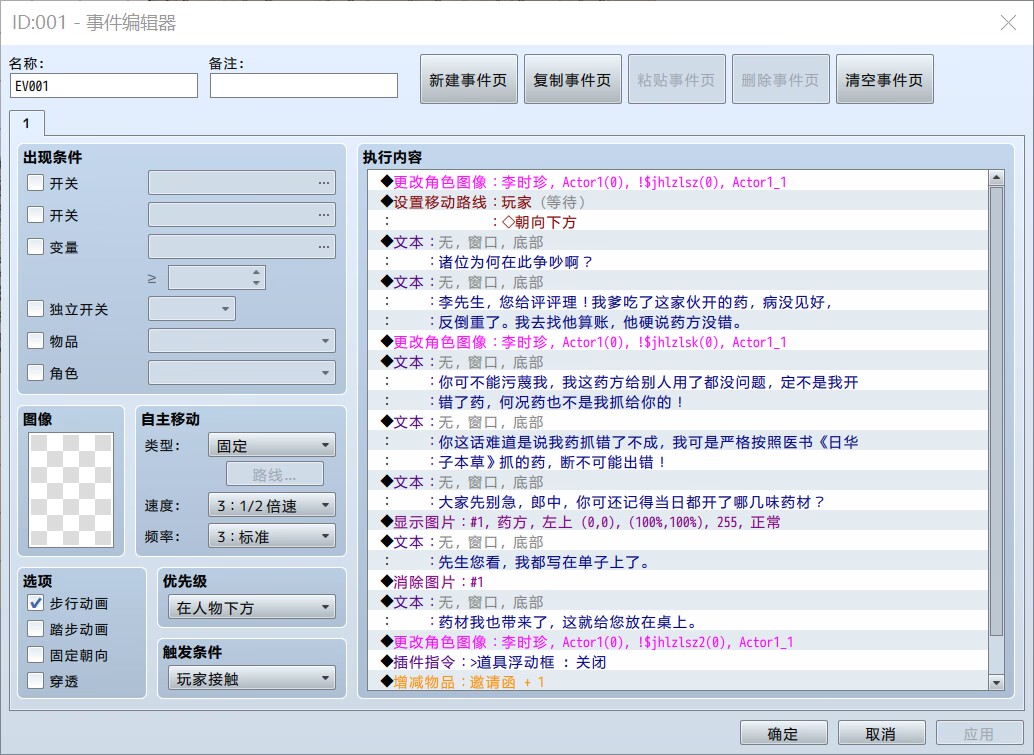
|  |
| --- |
| 根据前面介绍的基本定义，这里小结一下：  **行走图动画序列：**指专门作用于 行走图 的动画序列贴图结构。  可以通过插件指令开启/关闭绑定，并且能播放状态元/状态节点/动作元。  **全标签播放：**指 通过标签控制 播放 状态元/状态节点/动作元。  **全绑定：**指根据 行走图资源 自动绑定创建动画序列。 |

下面有这样一个问题：

如果我想设置一个人物，在剧情模式下，能控制他移动，并且能控制播放喝水、捧腹大笑等动作，应该如何设计？

部分群友的思维还停留在 多帧行走图 的设计上，如下图。

因为 多帧行走图+切换行走图的图像，的确能实现不同的行走图GIF动作。



但是，绑定了动画序列之后，此操作会**非常浪费性能**。

因为程序中走的流程是这样的：

|  |
| --- |
| 切换行走图图像 -> 全绑定插件检测到了资源变化 -> 销毁动画序列 -> 重新创建新匹配的动画序列 -> 动画序列播放默认的状态元集合 |

动画序列只要绑定一次就可以播放很多GIF了，此操作非常浪费游戏计算性能。

因此，图中的 **切换行走图图像**，应该改成 **插件指令播放状态节点/状态元集合** 才对。

## 从零开始设计（DIY）

### 配置站桩动画序列并播放

#### 1. 设置一个目标

2021年3月6日，作者我玩了一下live2d的制作工具。

学艺不精，只弄了头发和眼睛，然后导出了小爱丽丝图片集。



然后，作者我想把这些图片，放到地图界面中。

单独作为一个行走图的动画序列来播放。

#### 2. 结构规划/流程梳理

使用行走图动画序列插件就可以实现。

◆Drill\_CoreOfActionSequence 系统 - GIF动画序列核心

◆Drill\_EventActionSequence 行走图 - GIF动画序列

这里推荐使用 GIF动画序列编辑器 小工具进行设计，不推荐直接在插件里直接配置。

这里先单独设置一个 站桩动画序列，并且只播放 默认的状态元集合。

此操作相当于将一个 事件行走图 转变为 GIF 的功能。



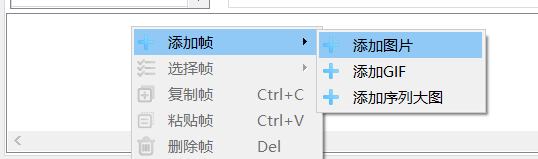
#### 3. 新建动画序列

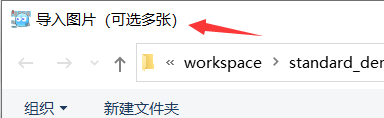
新建一个动画序列，起名为”小爱丽丝”。



#### 4. 新建状态元

新建状态元，起名为“小爱丽丝静止1”，导入多张图片





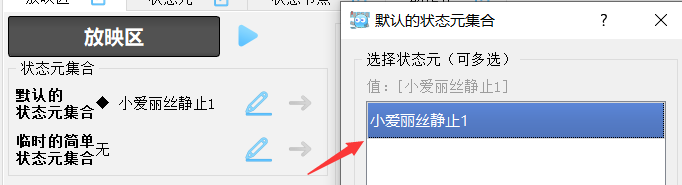
导入多张图片，如下。

（图片大小可以是任意的，不一定要被 行走图默认的48x48大小 所拘束）



#### 5. 设置 默认的状态元集合

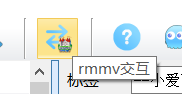
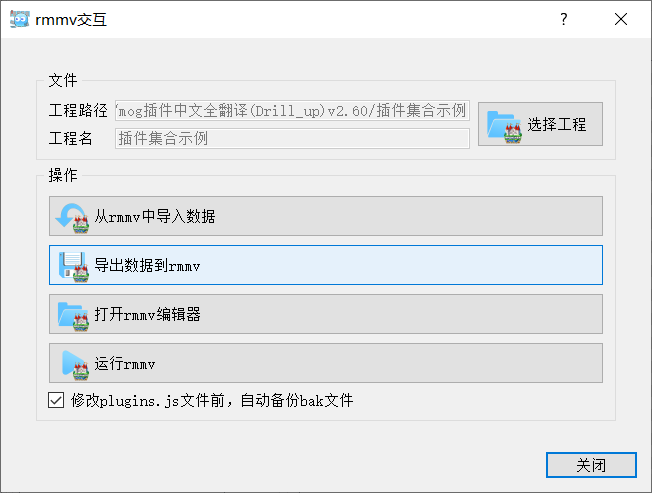
切换到放映区，然后设置默认的状态元集合，为刚才添加的状态元。



#### 6. 导出配置rmmv

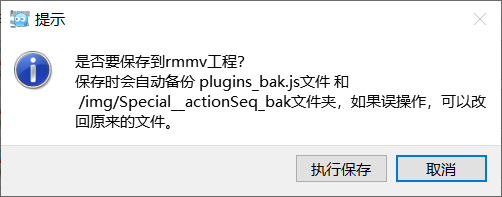
小工具中，点击rmmv交互按钮。

点击“导出数据到rmmv”按钮即可（需要关闭rmmv才能导出）。

由于导出配置是非常重要的操作，此操作会删除编辑器里面之前的动画序列数据。

因此小工具会给你备份之前的配置。

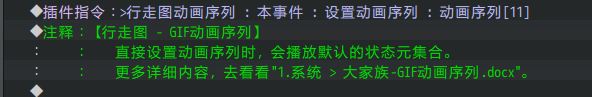


|  |
| --- |
| 备份最多备份一次，第二次导出保存时，第一次导出保存的备份文件会被清除。  另外如果你要发布你的游戏，要记得删除这些占容量的备份文件。 |

#### 7. 插件指令设置

添加一个事件，然后给事件设置动画序列11。

（设置动画序列后，会自动播放 默认的状态元集合 ）

#### 8. 测试游戏

在游戏中测试，可以看到行走图被换成了 站桩动画序列 的图片。

另外注意，事件的中心锚点在正下方。

|  |
| --- |
| 如果你还设置了动画序列的 状态节点或动作元，  你可以通过插件指令手动控制播放 状态节点或动作元。 |

### 配置行走图并手动播放标签

#### 1. 设置一个目标

示例中能够设置行走图 标签播放 。



这里作者我想把 配置动画序列、创建行走图动画序列、通过插件指令手动播放标签 等的流程，详细说明一下。

|  |
| --- |
| 可以参考小工具中的 动画序列[20] 的配置。 |

#### 2. 结构规划/流程梳理

需要用到下面的插件。

◆Drill\_CoreOfActionSequence 系统 - GIF动画序列核心

◆Drill\_EventActionSequence 行走图 - GIF动画序列

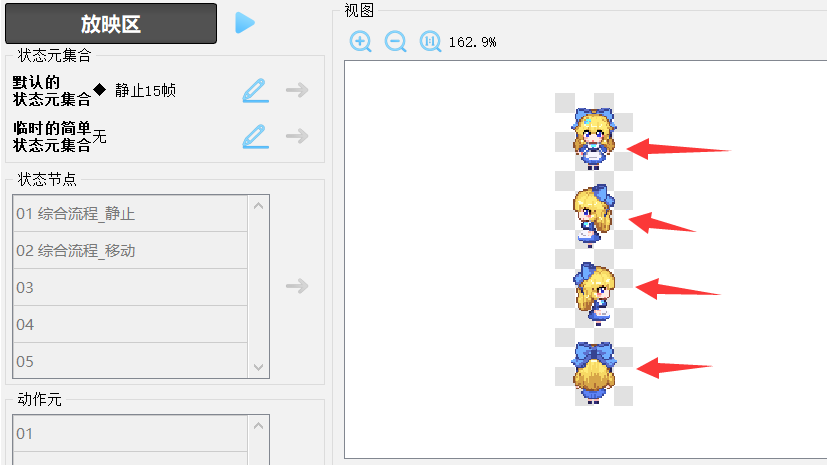
◆Drill\_EventActionSequenceAutomation 行走图 - GIF动画序列全标签播放

其中，注意一下 站桩动画序列 和 四方向动画序列的区别。

详细定义可以见：[站桩动画序列](#站桩动画序列) 和 [四方向动画序列](#四方向动画序列) 。

这里 四方向动画序列 的配置，注意行走图要为 纵向 的四等份合并，

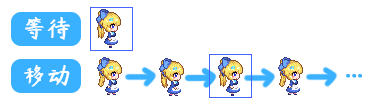
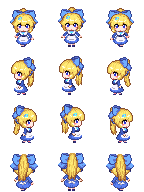
分别对应 朝下、朝左、朝右、朝上：



|  |
| --- |
| 由于此功能 状态元、状态节点 非常多，  推荐使用 GIF动画序列编辑器 小工具进行设计，  不推荐直接在插件里直接配置。。 |

#### 3. 配置状态元

这里我们以一个基础的 小爱丽丝行走图 为例：

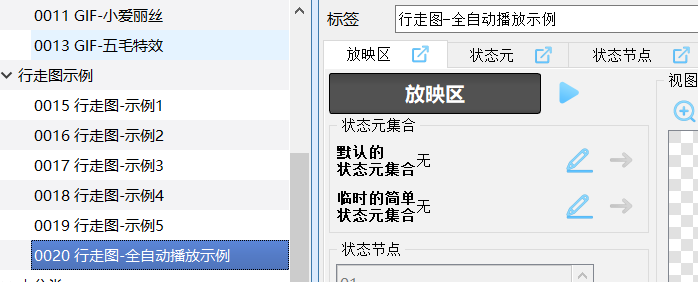


该行走图融合了两个不同的行为： 静止、移动

因此，需要先分开为两个状态元，对应 静止和移动 。

主要考虑来自 [行走图的姿势](#_行走图的姿势) ，目前先只考虑 静止和移动。

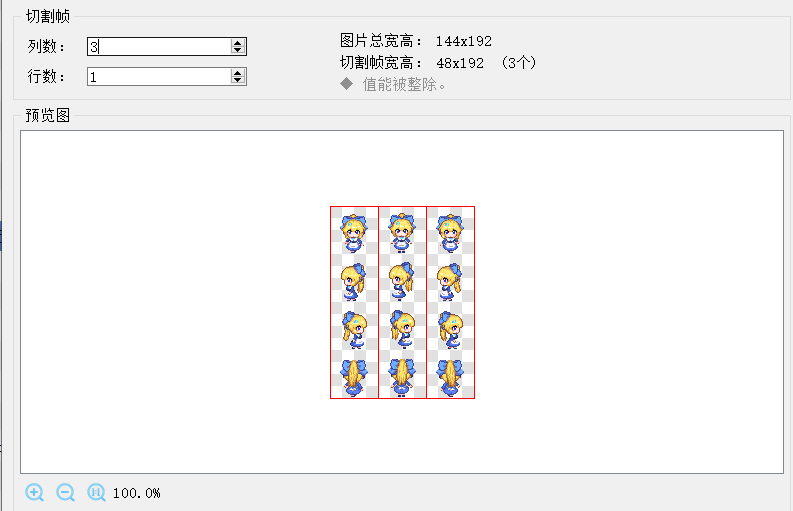
先建立动画序列：



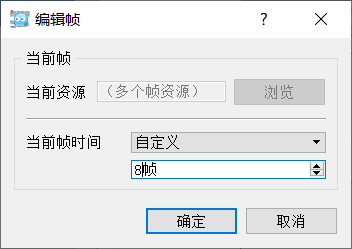
然后设置两个状态元，对应 静止和移动：



注意，状态元添加资源时，只切竖的，不要切横的。



然后把动画帧播放的速度放慢，



注意，小爱丽丝移动时，为 站立->踏步->站立->踏步->……

因此对应四个动画帧，下图1和3都是站立的动画帧。



这样，就完成了两个基本的状态元。



#### 4. 状态节点与标签设计

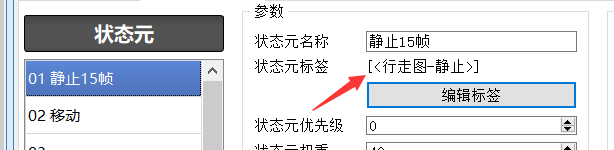
状态节点暂时不需要太复杂的设计，直接一对一连接状态元即可。

但注意 状态节点 的**起名规则**，这里起名为 ”综合流程\_xx”，

然后配置 顺序播放 “静止15帧”状态元。



你可以直接给 状态元 添加 <行走图-静止>和<行走图-移动> 标签。



也可以给 静止的状态节点 添加标签。

相同标签在 状态元/状态节点 中最多加一个，设置多了没有效果。



目前只考虑<行走图-静止>和<行走图-移动>标签，

其他标签后面章节再说。

后面绑定全标签播放时，插件会根据 当前行为，自动播放 符合标签 的状态节点。

#### 5. 放映区测试播放

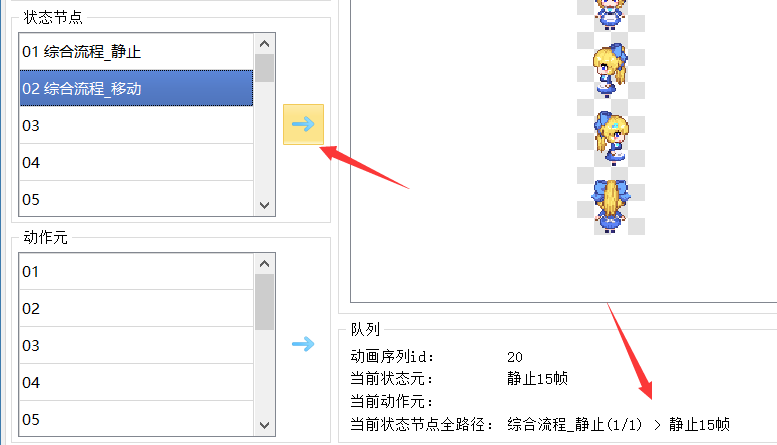
在放映区中，对添加的这两个状态节点进行测试播放。



注意，放映区测试有时候会发生播放没有反应的问题。

你需要检查一下是不是 状态节点 没配置 状态元。

然后检查一下是不是没有刷新数据，点击保存，在切换一下其他动画序列，再切换回来，点击播放，试试看是否有效。

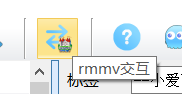
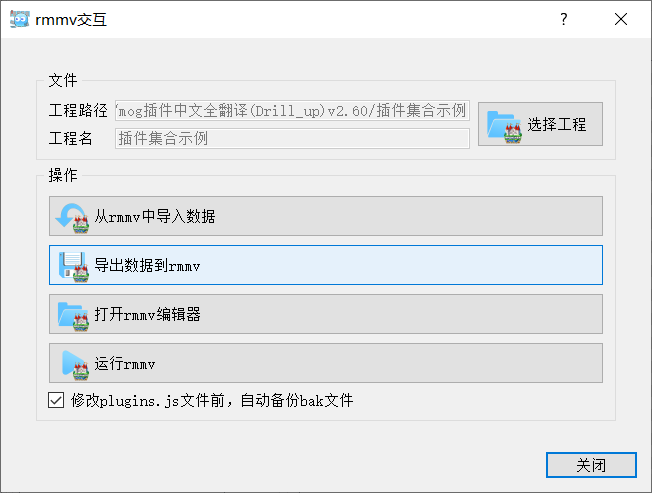


可以看看 ”小工具 > 关于GIF动画序列核心编辑器.docx” 的 常见问题。

#### 6. 导出配置到rmmv

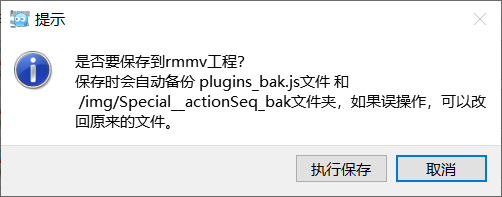
小工具中，点击rmmv交互按钮

点击“导出数据到rmmv”按钮即可（需要关闭rmmv才能导出）。

由于导出配置是非常重要的操作，此操作会删除编辑器里面之前的动画序列数据。

因此小工具会给你备份之前的配置。



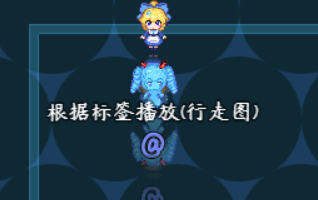
|  |
| --- |
| 备份最多备份一次，第二次导出保存时，第一次导出保存的备份文件会被清除。  另外如果你要发布你的游戏，要记得删除这些占容量的备份文件。 |

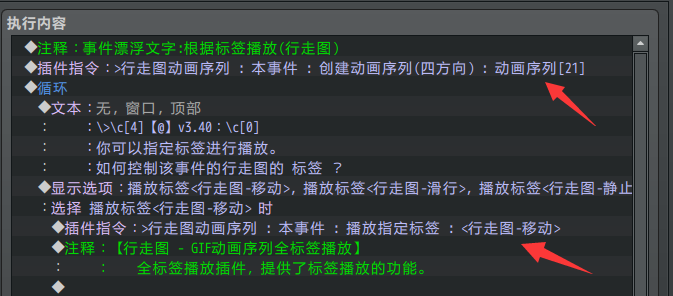
#### 7. 插件指令测试标签播放

接下来，在游戏中定义一个@符号事件，

接触事件时，创建动画序列，将其变成小爱丽丝，

然后执行插件指令 “播放指定标签” 。





可以看到动画序列能够根据标签”<行走图-移动>”播放移动动画了。

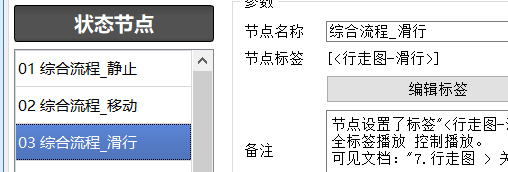


|  |
| --- |
| 如果你只想播放指定的 状态元/状态节点 ，  使用插件指令 “播放简单状态元集合” “播放状态节点” 即可。  可以去看看示例中的 行走图-示例 的5个例子。 |

#### 8. 新建滑行的标签

前面章节配置了 <行走图-移动> 和 <行走图-静止>，

这里可以再配置 <行走图-滑行> 。





但是需要添加插件：

Drill\_LayerSlipperyTile 图块 - 物体滑行

这时，使用插件指令播放标签即可。



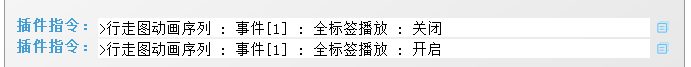
#### 9. 全标签播放的开关

注意，这里整个过程，只介绍了 手动播放标签 的方法。

这里都是使用插件指令手动执行的 标签播放。

虽然用到了 全标签播放 插件，但流程中都 **没有开启** 全标签播放；

所以事件不会自动变换行走图。



使用上面的插件指令，开启全标签播放后，行走图就能自动根据标签播放了。

|  |
| --- |
| 可以参考小工具中的 动画序列[20] 的配置。  里面提供了最简单的 <行走图-静止>、<行走图-移动>、<行走图-滑行> 的标签配置。 |

### 配置完整的行走图并全标签播放

#### 设置一个目标

‎2023‎年‎2‎月‎22‎日，作者我偶然点开了魔兽地图编辑器的单位动作面板，

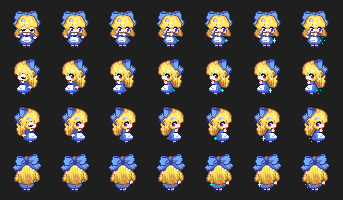
这个面板展示了 动画名称和动画对应的动作 。



许多单位的待机动画比攻击动画都丰富，并且有 Stand、Stand\_2、Stand\_3 等标签。

于是作者我心想，待机动作做那么复杂，那么小爱丽丝也不能一直站在那一直发呆。

开坑。



|  |
| --- |
| 可以参考小工具中的 动画序列[21,22,23] 的配置。  注意，这里的配置比较复杂，你可以先看看下前面章节：[配置行走图并手动播放标签](#_配置行走图并手动播放标签)  来理解 标签播放和全标签播放 的定义。 |

#### 2. 结构规划/流程梳理

需要用到下面的插件。

◆Drill\_CoreOfActionSequence 系统 - GIF动画序列核心

◆Drill\_EventActionSequence 行走图 - GIF动画序列

◆Drill\_EventActionSequenceAutomation 行走图 - GIF动画序列全标签播放

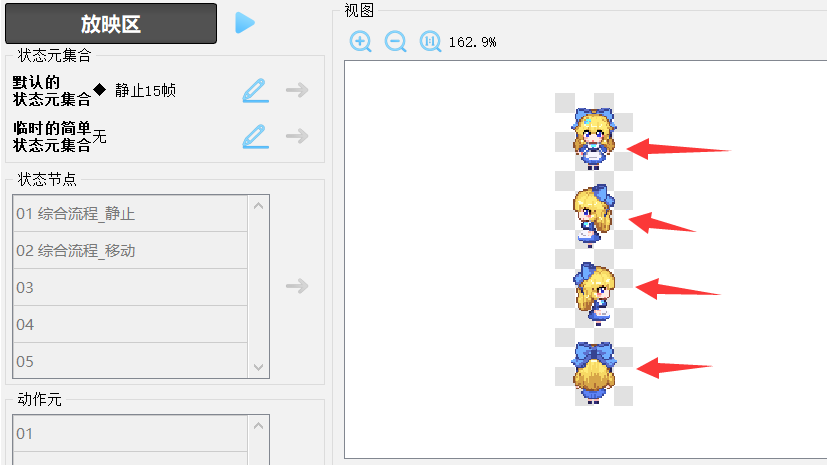
◆Drill\_EventActionSequenceBind 行走图 - GIF动画序列全绑定

其中，注意一下 站桩动画序列 和 四方向动画序列的区别。

详细定义可以见：[站桩动画序列](#站桩动画序列) 和 [四方向动画序列](#四方向动画序列) 。

这里 四方向动画序列 的配置，注意行走图要为 纵向 的四等份合并，

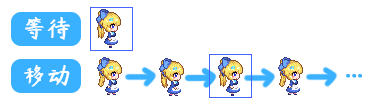
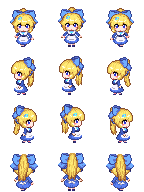
分别对应 朝下、朝左、朝右、朝上：



|  |
| --- |
| 由于此功能 状态元、状态节点 非常多，  推荐使用 GIF动画序列编辑器 小工具进行设计，  不推荐直接在插件里直接配置。。 |

#### 3. 配置状态元

这里我们以一个基础的 小爱丽丝行走图 为例：

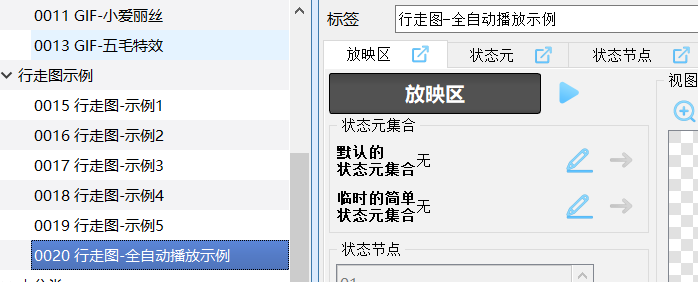


该行走图融合了两个不同的行为： 静止、移动

因此，需要先分开为两个状态元，对应 静止和移动 。

主要考虑来自 [行走图的姿势](#_行走图的姿势) ，目前先只考虑 静止和移动。

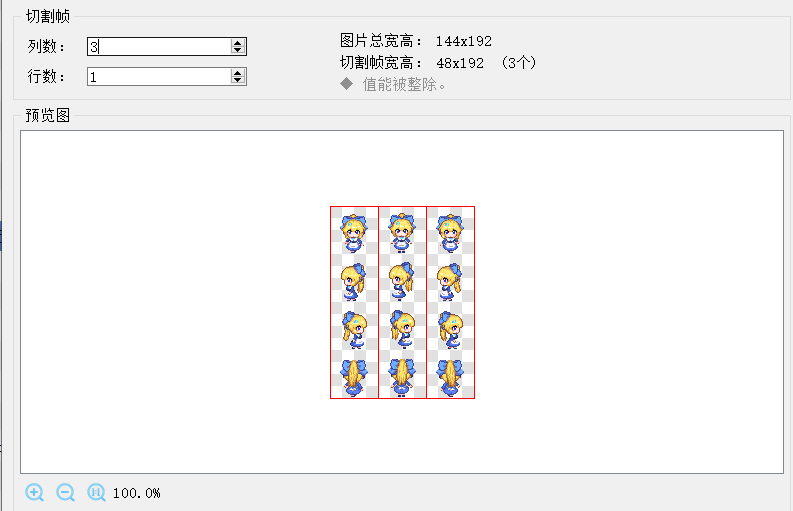
先建立动画序列：



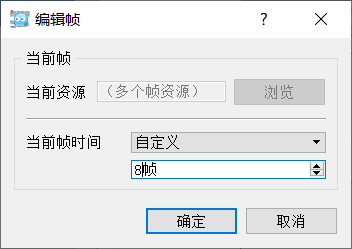
然后设置两个状态元，对应 静止和移动：



注意，状态元添加资源时，只切竖的，不要切横的。



然后把动画帧播放的速度放慢，



注意，小爱丽丝移动时，为 站立->踏步->站立->踏步->……

因此对应四个动画帧，下图1和3都是站立的动画帧。



这样，就完成了两个基本的状态元。



#### 4. 状态节点与标签设计

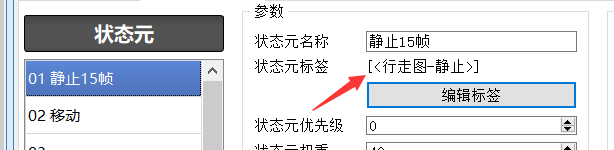
状态节点暂时不需要太复杂的设计，直接一对一连接状态元即可。

但注意 状态节点 的**起名规则**，这里起名为 ”综合流程\_xx”，

然后配置 顺序播放 “静止15帧”状态元。



你可以直接给 状态元 添加 <行走图-静止>和<行走图-移动> 标签。



也可以给 静止的状态节点 添加标签。

相同标签在 状态元/状态节点 中最多加一个，设置多了没有效果。



目前只考虑<行走图-静止>和<行走图-移动>标签，

其他标签后面章节再说。

后面绑定全标签播放时，插件会根据 当前行为，自动播放 符合标签 的状态节点。

#### 5. 放映区测试播放

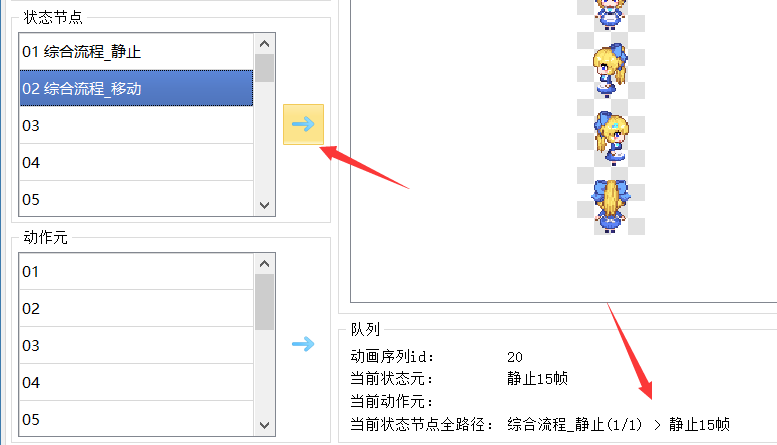
在放映区中，对添加的这两个状态节点进行测试播放。



注意，放映区测试有时候会发生播放没有反应的问题。

你需要检查一下是不是 状态节点 没配置 状态元。

然后检查一下是不是没有刷新数据，点击保存，在切换一下其他动画序列，再切换回来，点击播放，试试看是否有效。

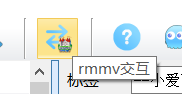
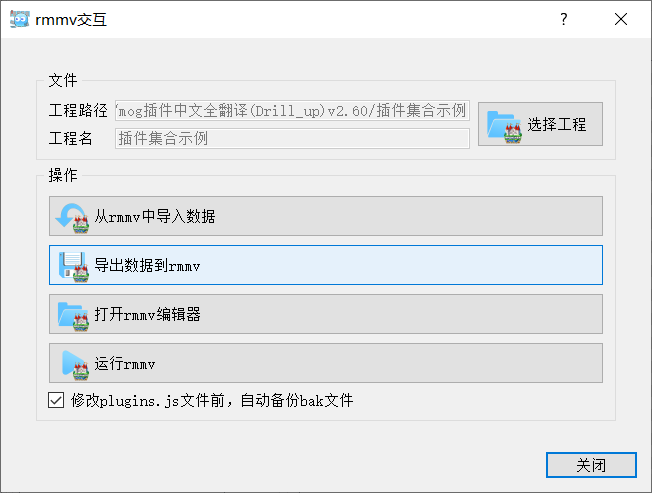


可以看看 ”小工具 > 关于GIF动画序列核心编辑器.docx” 的 常见问题。

#### 6. 导出配置到rmmv

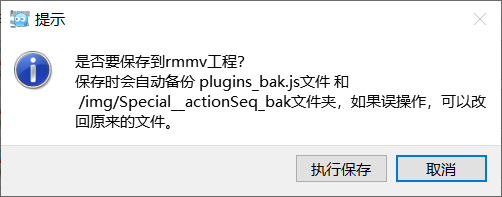
小工具中，点击rmmv交互按钮。

点击“导出数据到rmmv”按钮即可（需要关闭rmmv才能导出）。

由于导出配置是非常重要的操作，此操作会删除编辑器里面之前的动画序列数据。

因此小工具会给你备份之前的配置。

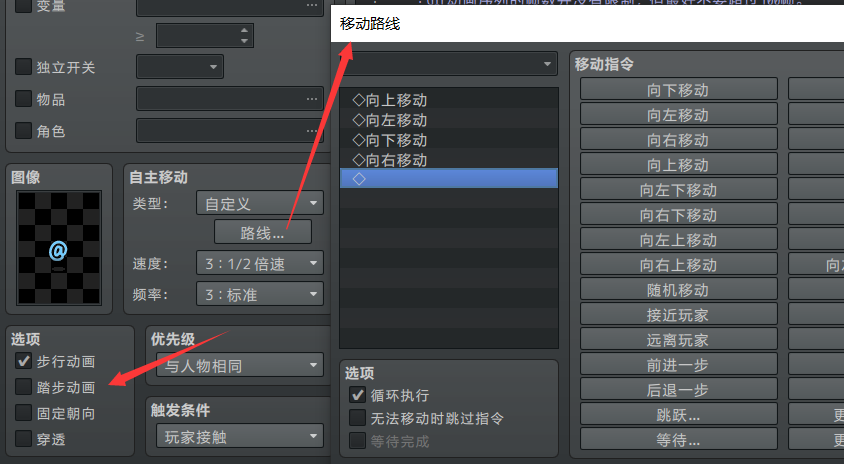


|  |
| --- |
| 备份最多备份一次，第二次导出保存时，第一次导出保存的备份文件会被清除。  另外如果你要发布你的游戏，要记得删除这些占容量的备份文件。 |

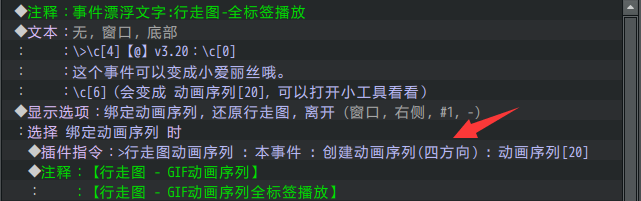
#### 7. 插件指令测试

接下来，在游戏中定义一个@符号事件，

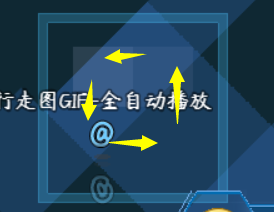
设置步行动画，不要原地踏步，频率为适中（走走停停）。



在这个事件中，添加 对话时 执行插件指令： “**创建动画序列(四方向)**”。



然后在游戏中测试可以看到：

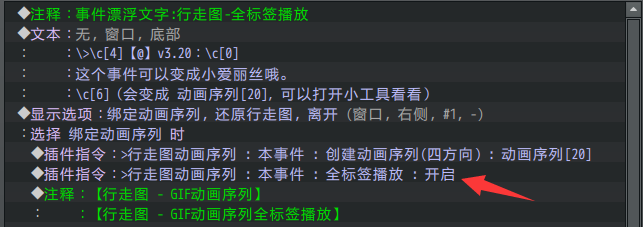
接触并 对话 后，选择”绑定动画序列”，可以变成小爱丽丝，

但这个小爱丽丝只会改变朝向，不会播放移动动画。

这是因为当前播放的只是 **默认的状态元集合** ，没有被 全标签播放的功能 控制。

#### 8. 全标签播放测试

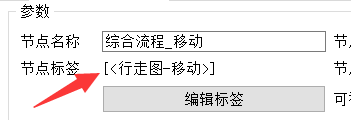
接下来，把 全标签播放 的插件指令加上。



可以看到如下图，小爱丽丝能正常行走了：



如果小爱丽丝仍然没有移动，那么检查一下状态节点是不是忘了加 移动标签。



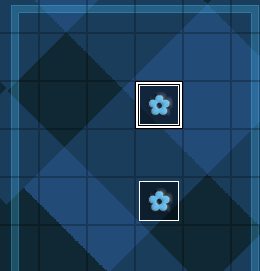
#### 9. 全绑定测试

全绑定是指 指定行走图资源 ，自动变成 动画序列。

◆Drill\_EventActionSequenceBind 行走图 - GIF动画序列全绑定

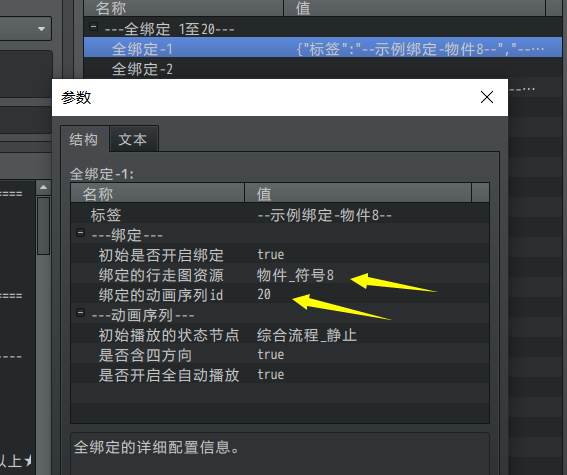
比如这里添加两个事件，资源为“物件-符号8”：

全绑定 根据资源 直接将行走图 变成 动画序列，行走图朝向、帧数 **不影响**动画序列。



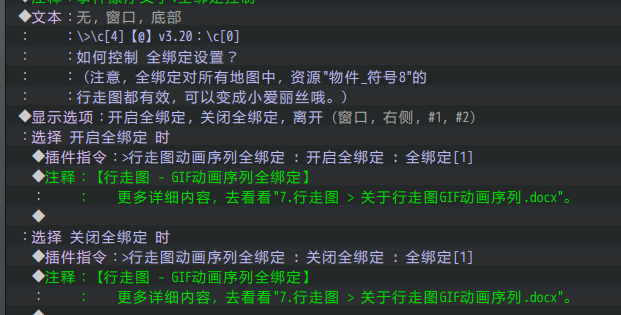
比如下图，在全绑定插件中，设置”物件\_符号8”的资源，绑定刚才添加的 动画序列20。

注意设置参数 “是否含四方向” 为true。



全绑定配置后，直接生效。

当然，你也可以通过插件指令手动控制开启/关闭。



如下图，左边为 关闭全绑定 时的效果，右边为 开启全绑定 的效果。

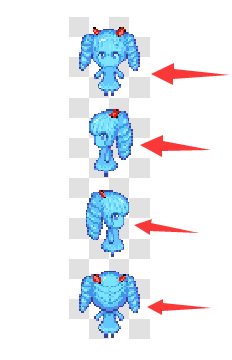
|  |
| --- |
| 如果你希望小爱丽丝不被 全绑定，那么配置不一样的资源名称即可。 |

#### 10. 二方向与四方向

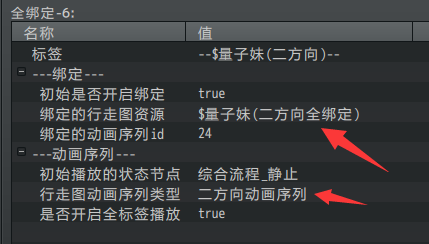
前面章节讲解了 配置状态元到全绑定的流程。

同样的流程也可以设计 二方向动画序列 。（注意资源朝向为先右后左）

只不过配置的资源不一样，如下图。



完成配置后，在全绑定插件中，配置类型为 二方向动画序列 即可。



#### 11. 细化更多标签

前面的章节 [8. 新建滑行的标签](#_8._新建滑行的标签) 中介绍了 添加 <行走图-滑行> 标签，

并且所有标签可以见：[动画序列标签表](#_动画序列标签表) 。

这里不再赘述。

#### 12. 细化更复杂的状态元/状态节点

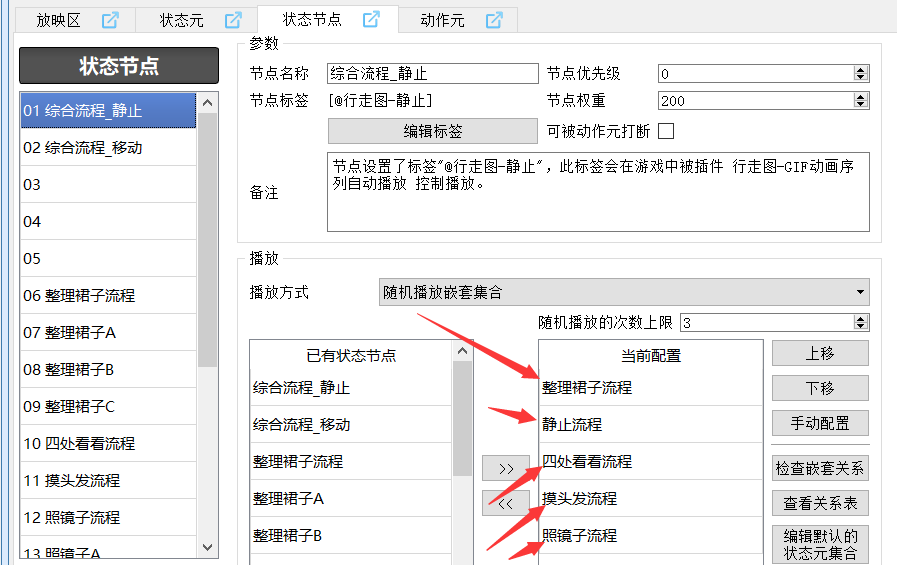
上述已经完整描述了 四方向动画序列、全标签播放、全绑定 的内容。

这里，使用默认的小爱丽丝的 静止和移动 动画，显得比较单一。



你可以添加更多的 状态元/状态节点。

比如小工具示例中的：



静止过程中，还可以时不时播放一些小爱丽丝 整理裙子、四处看看、摸头发、照镜子等的待机动画。

|  |
| --- |
| 全标签播放根据 <行走图-静止>等标签 来播放状态节点。  **状态节点的复杂程度 与 游戏中配置的 全标签播放/全绑定 无关。**  因此你可以先确保游戏中能正常绑定 行走图 之后，进一步对行走图动画加强细化。 |

#### 13. 复杂动画序列设计

详细内容可以去看看：”小工具 > 关于GIF动画序列核心编辑器.docx”

你也可以结合 插件示例 中，动画序列管理层 的五个小爱丽丝，来了解具体内容。



|  |
| --- |
| 复杂动画序列+全标签播放+全绑定 的最终效果，  可以参考小工具中的 动画序列[21,22,23] 的配置。 |

## 常见问题（FAQ）

### 移动时不播放移动动画

|  |  |
| --- | --- |
| **问题名称** | 移动时不播放移动动画 |
| **问题图示** |  |
| **问题描述** | 1. 如图添加上述事件注释后，@符号会变成小爱丽丝。 2. 小爱丽丝移动时会旋转朝向，但不会走动，像个石像一样。 |
| **原理解析** | 看注释可知，  注释只绑定了行走图动画序列，所以只播放 默认的状态元集合 是正常的。  因为你忘了还要添加 全标签播放 的注释。 |
| **解决方案** | 添加全标签播放的注释即可。      另外还要注意有没有在动画序列中写<行走图-移动>的标签。 |

### 设置动画序列后变成了切片/多个

|  |  |
| --- | --- |
| **问题名称** | 设置动画序列后变成了切片/多个 |
| **问题图示** |  |
| **问题描述** | 创建动画序列并且开启了全标签播放。  但是小爱丽丝显示成了切片/多个。 |
| **原理解析** | 创建动画序列有下面几种类型：  [站桩动画序列](#站桩动画序列) 、[四方向动画序列](#四方向动画序列)、 [二方向动画序列](#二方向动画序列) 、[八方向动画序列](#八方向动画序列)  这些类型的原理就是把资源图片进行 纵向切割。  如果出现了 切片/多个 情况，说明资源与动画序列设置 没匹配。 |
| **解决方案** | 检查插件指令或事件注释。  如果资源配置有 四个方向，那么应该使用指令“创建动画序列(四方向)”。  如果资源只是单图，那么应该使用指令“创建动画序列”。  如果你是用全绑定设置的，检查全绑定的类型是否对应： |

### 进入地图后图像会闪

|  |  |
| --- | --- |
| **问题名称** | 进入地图后，图像会闪 |
| **问题图示** |  |
| **问题描述** | 新配置了站桩动画序列，进入地图会闪一段时间，然后就不闪了。  四方向动画序列、二方向动画序列也会出现一样的问题。 |
| **原理解析** | 动画序列在创建时，需要加载所有相关的图片资源。  如果没来得及加载就提前播放图片，  那么就会出现图片先是空的，然后突然又加载显示了的现象。  此问题一般都不会遇到，但是如果遇到了（比如有大量行走图要加载），则可以开启预加载功能。  预加载的详细介绍可以去看看：“1.系统 > 关于预加载.docx” |
| **解决方案** | 开启预加载的设置，就不会闪了。    注意，还有一种情况，  有可能你不小心吧 创建动画序列 放在了并行事件中执行。  因为动画序列创建时，会闪一下，放并行事件中执行过多，就会反复闪。 |

### 其他问题

如果你在配置过程中遇到了无法解决的问题，

可以去看看 ”1.系统 > 大家族-GIF动画序列.docx” 文档中的 常见问题 章节。