

无限过图关卡编辑教程

Lsy

在阅读本教程之前，请确认你对 xml 的数据结构有基本的认知。

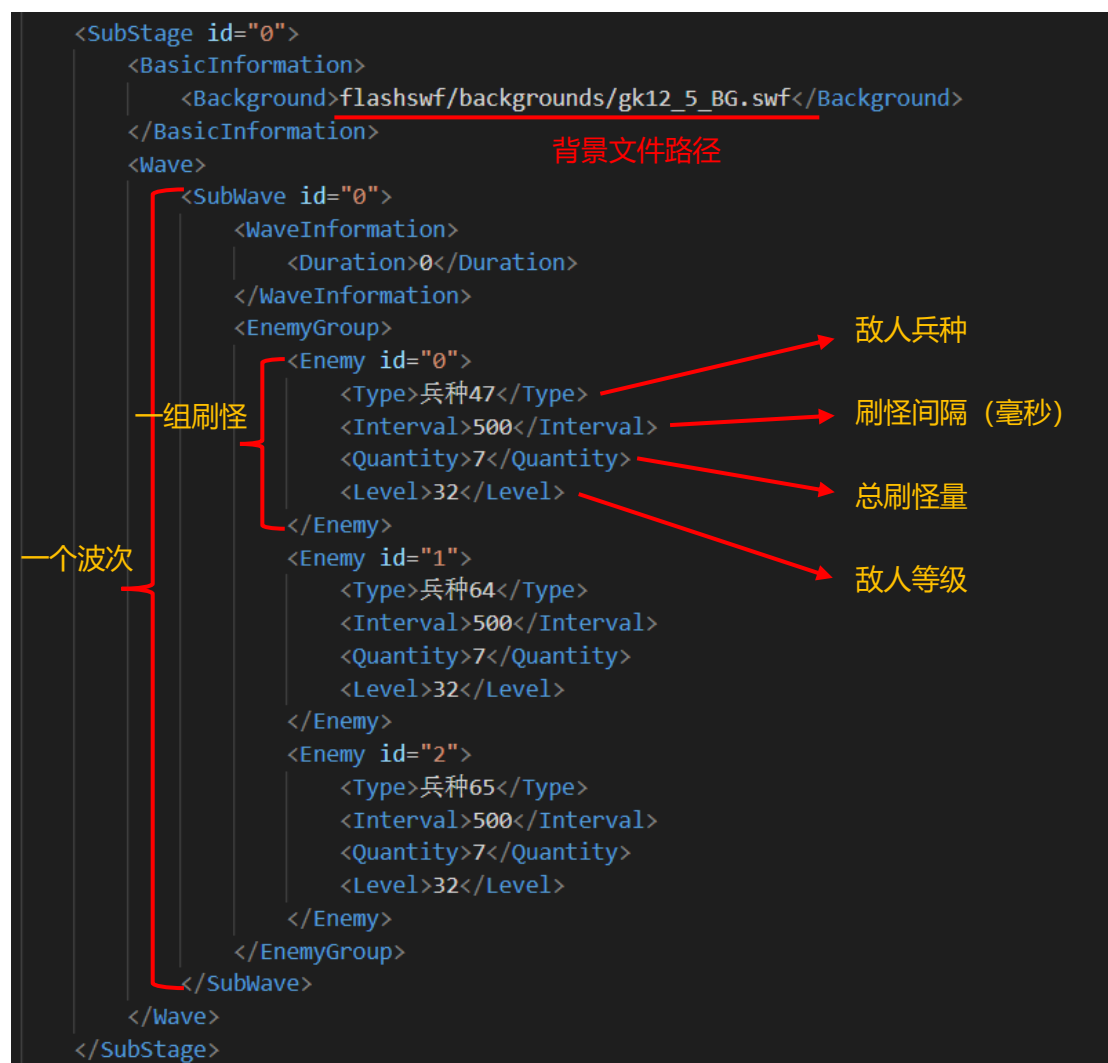
那么开始吧。

无限过图的关卡信息由多个<Substage>标签构成，每个标签代表一张图。那么，这一坨一坨数据代表什么呢？

```
<SubStage id="0">
  <BasicInformation>
    <Background>flashswf/backgrounds/gk12_5_BG.swf</Background>
  </BasicInformation>
  <Wave>
    <SubWave id="0">
      <WaveInformation>
        <Duration>0</Duration>
      </WaveInformation>
      <EnemyGroup>
        <Enemy id="0">
          <Type>兵种47</Type>
          <Interval>500</Interval>
          <Quantity>7</Quantity>
          <Level>32</Level>
        </Enemy>
        <Enemy id="1">
          <Type>兵种64</Type>
          <Interval>500</Interval>
          <Quantity>7</Quantity>
          <Level>32</Level>
        </Enemy>
        <Enemy id="2">
          <Type>兵种65</Type>
          <Interval>500</Interval>
          <Quantity>7</Quantity>
          <Level>32</Level>
        </Enemy>
      </EnemyGroup>
    </SubWave>
  </Wave>
</SubStage>
```

例：翹虎堂外围第一张图

在经过一些形而上学的观察之后，不难理解（瞎猜）出这里每个参数和它们的层级关系代表什么。



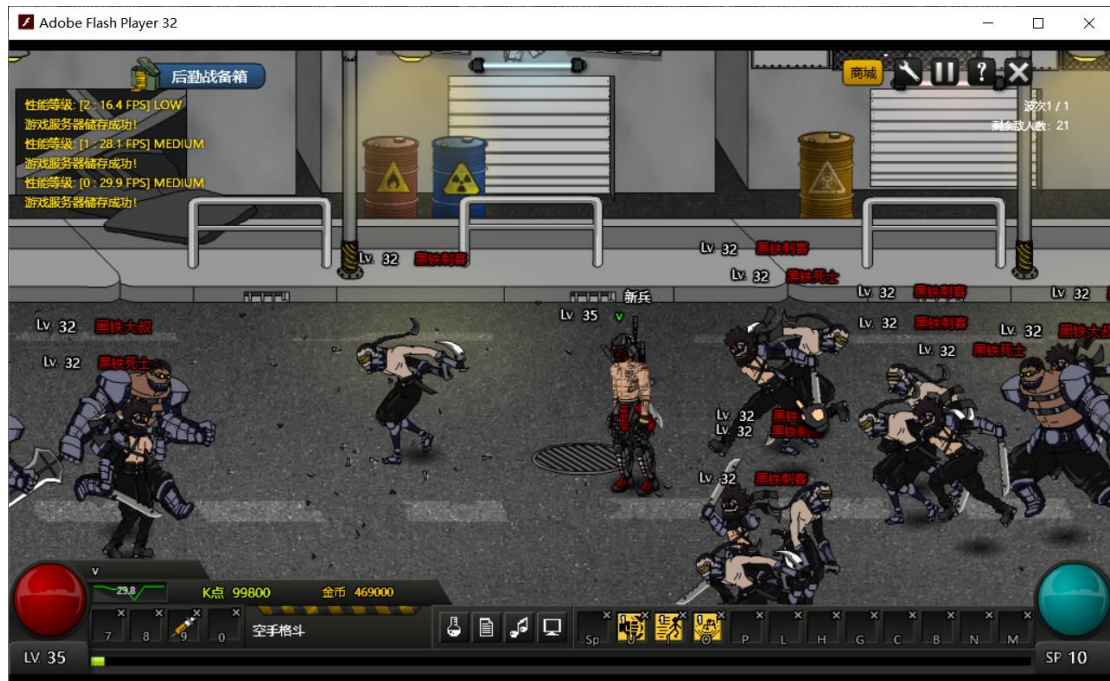
噯，看起来挺直观嘛。

然后我们来到 data/unit/unit.json 里查找兵种数据，

<pre>{ "id": 47, "spritename": "敌人-黑铁会死士", "level": 20, "name": "黑铁死士", "is_hostile": true, "height": 175, "data": { "primary": null, "secondary": null, "secondary2": null, "melee": null } }</pre>	<pre>{ "id": 64, "spritename": "敌人-黑铁会刺客", "level": 1, "name": "黑铁刺客", "is_hostile": true, "height": 175, "data": { "primary": null, "secondary": null, "secondary2": null, "melee": null } }</pre>	<pre>{ "id": 65, "spritename": "敌人-黑铁会大叔", "level": 1, "name": "黑铁大叔", "is_hostile": true, "height": 185, "data": { "primary": null, "secondary": null, "secondary2": null, "melee": null } }</pre>
--	---	---

噯，原来指代的是这三个怪啊。

然后我们来到游戏里一看。



对照实际的刷新情况，可以看到 xml 里的数据被如何翻译成实际出怪：

1. 本关一共刷新均为 32 级的 7 个黑铁死士，7 个黑铁刺客，7 个黑铁大叔。
2. 三组刷怪的刷新间隔分别都为 500 毫秒，所以所有敌人都会在主角进图后的 3.5 秒内刷出。
3. 根据无限过图的默认设置，每次刷怪都有 2/3 概率在屏幕右侧生成，1/3 概率在屏幕左侧生成，而后者有可能导致在玩家进图时敌人直接刷新到玩家脸上。

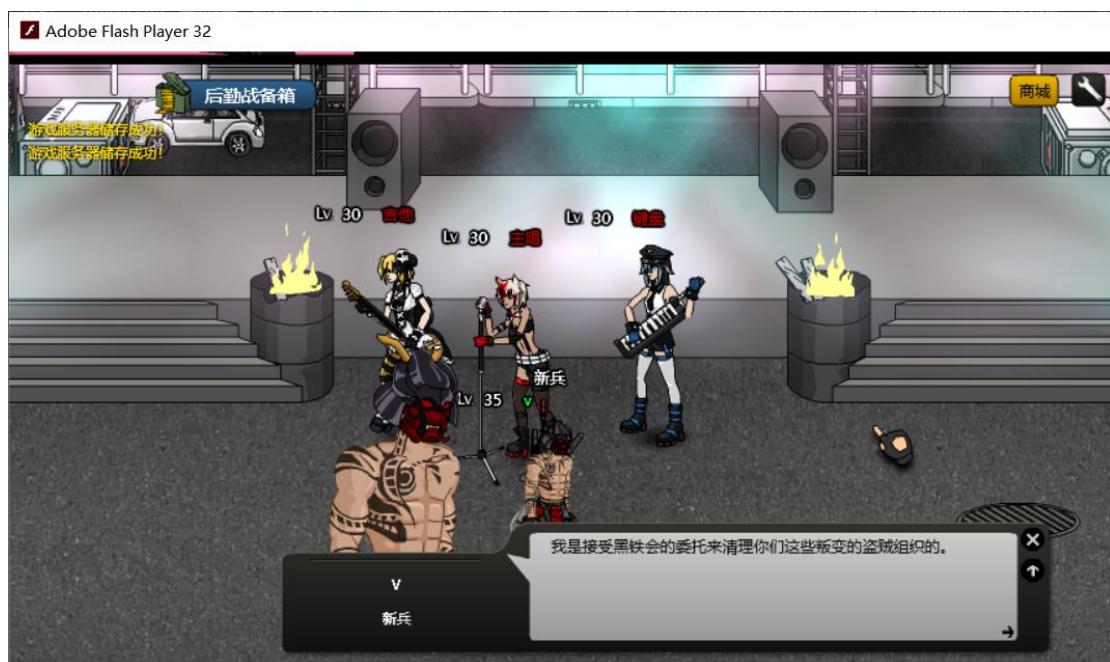
好了，现在你已经完全学会编辑关卡了！本次的教程就到这里。



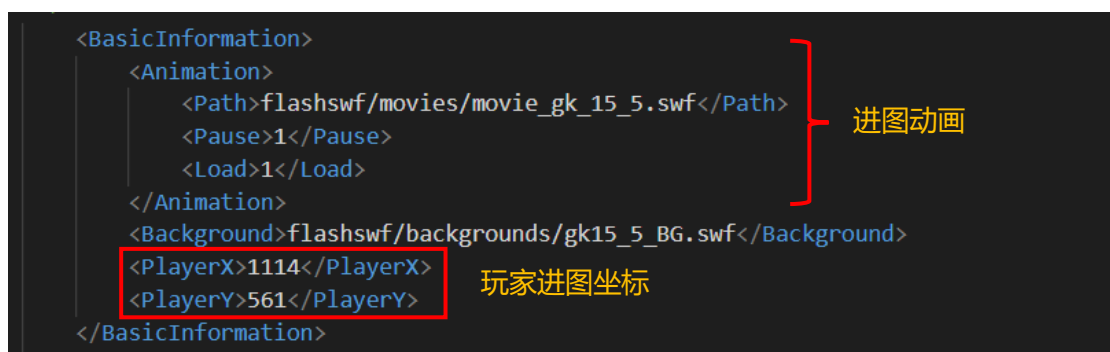
能不能让主角或者敌人在固定位置出现呢？

果然，大伙都觉得原版无限过图实在是太缺乏深度了。

没有问题。我们来看看摇滚公园的最后一张图。



主角和摇滚三姐妹在进图时都刷新在舞台中间。怎么实现呢？



简单地在<BasicInformation>中设置玩家坐标<PlayerX>和<PlayerY>即可达成。

顺便，进图或过图时的动画同样可以在<BasicInformation>中配置。

```

<EnemyGroup>
  <Enemy>
    <Type>兵种50</Type>
    <Interval>100</Interval> 最小刷新时间
    <Quantity>1</Quantity>
    <Level>30</Level>
    <x>1007</x> 敌人坐标
    <y>434</y>
    <Parameters>不掉钱:true</Parameters>
  </Enemy>

```

敌人的坐标也可以简单的通过在<Enemy>里加上<x>和<y>来配置。注意，如果需要通过让敌人在玩家进图时就被看到，请在<Interval>中填写 100。



有没有办法像老关卡一样让怪物死一个刷一个呢？

没有问题。

仍然以摇滚公园为例。与三姐妹同时出现的，还有左上角不断生成的狗狗。



狗狗一共 5 只，每死一只就会刷新一只。为了达到这个效果，我们需要定义敌人的出生点。

```

<SpawnPoint>
  <Point id="0">
    <x>50</x>
    <y>490</y>
    <QuantityMax>1</QuantityMax>
  </Point>
</SpawnPoint>

```

出生点序号，从 0 开始

出生点坐标

最大同时在场数

这是摇滚公园里对狗狗出生点的定义。所有在这个出生点刷新的敌人都会会在 x=50, y=490 生成，并且每次刷怪前会检测场上该出生点的存活人数，若超过 <QuantityMax> 的值则停止生成。

然后，我们只需要简单地将狗狗绑定到这个出生点：

```

<Enemy>
  <Type>兵种13</Type>
  <Interval>1000</Interval>
  <Quantity>5</Quantity>
  <Level>10</Level>
  <SpawnIndex>0</SpawnIndex>
</Enemy>

```

出生点索引

多组刷怪可以绑定到同一个出生点，一起受到同一个 <QuantityMax> 的约束


出生点的用法还有很多，在此仅简单展示一下。

```

<SpawnPoint>
  <Point id="0">
    <x>450</x>
    <y>360</y>
    <Identifier>出生点-钻头</Identifier>
  </Point>
  <Point id="1">
    <x>700</x>
    <y>500</y>
    <Identifier>出生点-钻头</Identifier>
  </Point>
  <Point id="2">
    <x>510</x>
    <y>650</y>
    <Identifier>出生点-钻头</Identifier>
  </Point>
</SpawnPoint>

```

门元件的标识符




<Identifier>标签定义钻头作为出生点并自动调用开门动画。例：A 兵团试炼场第一张图。

```

<SpawnPoint>
  <Point id="0">
    <x>915</x>
    <y>422</y>
    <NoCount>true</NoCount>
  </Point>
</SpawnPoint>

```

不计入击杀要求

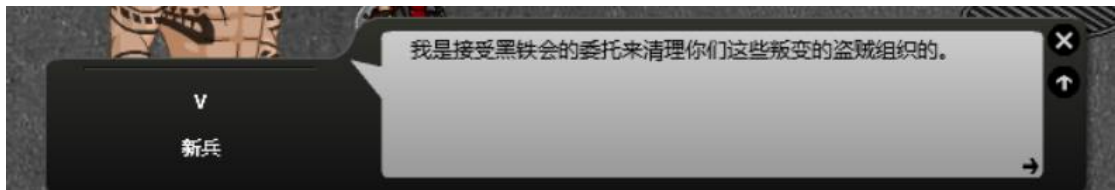


<NoCount>标签填 true 可使该出生点的敌人不计入击杀要求。例：AVP 第五张图的铁血鹰。



能不能在关卡中加上对话呢？

没有问题。再回到摇滚公园这张图。



关卡对话的格式和任务对话基本相同，在这里不再赘述。

```
<Dialogue>
  <SubDialogue id="0">
    <Name>$PC</Name>
    <Title>$PC_TITLE</Title>
    <Char>$PC_CHAR#严肃</Char>
    <Text>我是接受黑铁会的委托来清理你们这些叛变的盗贼组织的。</Text>
  </SubDialogue>
  <SubDialogue id="1">
    <Name>Singer</Name>
    <Title>摇滚女皇-主唱</Title>
    <Char>Singer</Char>
    <Text>我们才不是什么叛变组织，我们只是看不惯黑铁会和自由革命军狼狈为奸屠杀难民抢劫资源，所
  </SubDialogue>
  <SubDialogue id="2">
    <Name>Keyboard</Name>
```

注意：目前的模板支持在进图时，或任意波次结束后添加对话。把整个<Dialogue>标签放在<Substage>目录下表示进图对话，放在<Subwave>目录下表示该波次结束后的对话。

```
    <Text>结束了...</Text>
  </SubDialogue>
  <SubDialogue id="5">
    <Name>Singer</Name>
    <Title>摇滚女皇-主唱</Title>
    <Char>Singer</Char>
    <Text>这次真是多亏了你了。</Text>
  </SubDialogue>
  <SubDialogue id="6">
    <Name>Guitar</Name>
    <Title>摇滚女皇-吉他</Title>
    <Char>Guitar</Char>
    <Text>你挺厉害的嘛，以后不说你坏话了。</Text>
  </SubDialogue>
  <SubDialogue id="7">
    <Name>Keyboard</Name>
    <Title>摇滚女皇-键盘</Title>
    <Char>Keyboard</Char>
    <Text>...谢谢。</Text>
  </SubDialogue>
</Dialogue>
</SubWave>
</Wave>
</SubStage>
```

摇滚内战的结束对话。注意其依附的目录



能不能做成不把敌人杀完就过关呢？

没有问题。有两种方法可以实现。

1. 再次回到摇滚公园。许多早期关卡都不要杀完，包括摇滚公园的第三张图。

```
<Wave>
  <SubWave id="0">
    <WaveInformation>
      <Duration>0</Duration>
      <FinishRequirement>1</FinishRequirement>  击杀要求
    </WaveInformation>
    <EnemyGroup>
      <Enemy>
        <Type>兵种46</Type>
        <Interval>100</Interval>
```

<FinishRequirement>标签默认为 0，填写正整数 n 则会在剩余敌人数小等于 n 时直接判定本波通过。本张图在剩余一个敌人的情况下就可通关。

2. 使用出生点和非出生点分离需要击杀的敌人。刚才介绍了使用<NoCount>标签使得某个出生点的刷怪不计入击杀要求。除此之外，还有一个反向的方法：

```
<Wave>
  <SubWave id="0">  地图不计入击杀要求
    <WaveInformation>
      <Duration>0</Duration>
      <MapNoCount>true</MapNoCount>
    </WaveInformation>
```

<MapNoCount>标签为 true 会将所有不从出生点刷新的敌人排除出击杀要求外。

在袭杀与圈套第三张图，击杀唯一从出生点刷出的单位牛仔即可获胜。





我比较懒，不想填一堆敌人坐标怎么办？

没有问题，我也嫌填坐标累，因为需要先搞清楚地图尺寸。来看看摇滚公园的第二张图。

```
<Enemy>
  <Type>兵种43</Type>
  <Interval>100</Interval>
  <Quantity>3</Quantity>
  <Level>20</Level>
  <SpawnIndex>back</SpawnIndex>
</Enemy>
</EnemyGroup>
```

我们已经学会了使用<SpawnIndex>将刷怪绑定到出生点。而在没有出生点的图里，<SpawnIndex>仍然可以用来定义怪物生成位置。这张图有一个包含 3 个街舞少女的怪物组，<SpawnIndex>填写了 back，这意味着这三个街舞少女会在地图右侧屏幕外的随机位置生成。

以下是<SpawnIndex>常用的字段：

left：在地图左侧生成

right：在地图右侧生成

front：在地图左侧屏幕内的随机位置生成，会自动避免生成在和玩家出生点同一条直线上

back：在地图右侧屏幕外的随机位置生成

door：在地图通关出口处生成



能不能像生存模式一样设置刷新时间？

没有问题。回到摇滚公园……不，回到摇滚公园 DLC：堕落城的骚动！



可以看到，大部分的波次都有计时，计时结束或击杀所有敌人都会刷新下一波。

注意：

1. 同一个图里并非每波都必须计时，可以只有一部分波次计时。比如，最后一波一般来说没有必要计时。
2. <Duration>和<FinishRequirement>可以同时作为通过条件，并且在波次之间不断交替使用。在僵尸逆袭副本中有大量交替使用的例子。



还能不能加更多好玩的东西？

没有问题。还是摇滚公园！



这个只有在特定主线进度刷出的箱子，是通过<Instance>标签加入的。

```
<Instances>
  <Instance id="0">
    <x>900</x>
    <y>264</y>
    <Identifier>跳动的火焰</Identifier>
  </Instance>
  <Instance id="1">
    <x>1313</x>
    <y>265</y>
    <Identifier>跳动的火焰</Identifier>
  </Instance>
  <Instance id="2">
    <x>1681</x>
    <y>462</y>
    <Identifier>资源箱</Identifier>
    <Parameters>最小主线进度:27,最大主线进度:27,内部物:黑暗吉他,数量:1,等级:0,名字:木箱</Parameters>
  </Instance>
</Instances>
```

元件的标识符

额外参数

通过<Identifier>，可以导入库中注册了标识符的任何影片剪辑。

用<Parameters>导入额外参数的手段并不成熟，这里等一波 fs 的 eval 解析器。

```

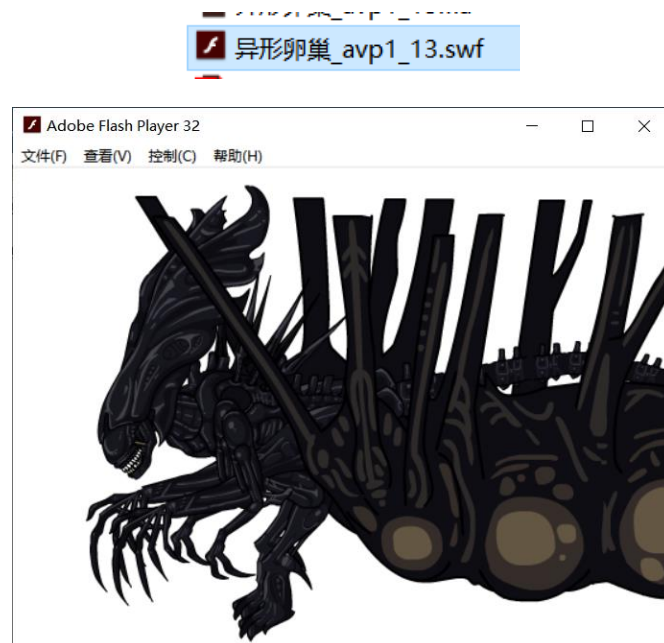
<Instances>
  <Instance id="0">
    <x>1225</x>
    <y>-68</y>
    <url>flashswf/movies/异形卵巢_avp1_14.swf</url>
  </Instance>
</Instances>

```

swf 文件路径

除了用<Identifier>从库中导入元件外，还支持直接使用<url>标签加载外部 swf。

AVP 最后两关的异形卵巢就是使用该方法加载。



以上内容虽然不是无限过图模板的全部，但相信足够满足目前大部分编辑关卡的需求，教程到这里真的要结束了。

什么，想要了解更多？

猜猜为什么刚才我反复提及摇滚公园？

目前无限过图模板的完整文档，就放在摇滚公园.xml 里。要解决更多的问题，

不妨打开看看吧！