## 概述

### 相关插件

基于核心：

◆Drill\_CoreOfDynamicMask 系统 - 动态遮罩核心

插件本体：

◆Drill\_LayerIllumination 地图 - 自定义照明效果

◆Drill\_MouseIllumination 鼠标 - 自定义照明效果

你可以用自己画的照明资源图片，绑定在某些物体上，实现发光。

### 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：

|  |  |
| --- | --- |
| 黑暗层 | [黑暗层](#_黑暗层) [黑暗层开关](#黑暗层开关) [黑暗层颜色](#黑暗层颜色) |
| 光源/照明 | [物体照明](#物体照明) [高级照明](#高级照明) [限时动态照明](#限时动态照明)  [光源颜色](#光源颜色) [光源图片层级](#光源图片层级) [遮挡设计](#遮挡设计) |

### 插件关系

照明效果的插件关系如下，实线表示必须插件，虚线表示可断开的扩展插件。



## 自定义照明

### 黑暗层

**1) 定义**

**黑暗层：**指将地图全部涂黑的层级。

**照明/光源：**指将涂黑的层级反向画亮的贴图对象，该贴图对象可以绑定事件，也可以单独控制变化，实现不同的光源照明效果。



**2)** **黑暗层开关**

下面以一个实际的例子说明。

白天如下图。（白色的地图表示黑暗层关闭，蓝色的表示黑暗层开启。）

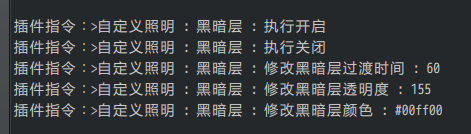


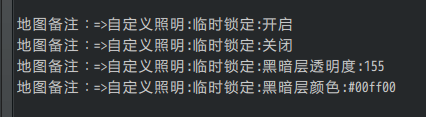
黑夜如下图。（白色的地图表示黑暗层关闭，蓝色的表示黑暗层开启。）



**由图可知，无论外界如何变化，商店和小屋都是亮的，矿场和黑暗森林都是暗的。这四张地图 不受外界影响，所以是锁定的黑暗层状态**。

那么，外界 和 锁定 分别对应插件中的 插件指令 和 地图备注 控制。





插件指令无法影响 锁定地图 的黑暗层。但是可以影响默认的黑暗层。

换句话说，就是临时锁定不会影响外面的昼夜运作，但是临时锁定无法修改。插件指令可以控制外面的昼夜运作，且在任何地图执行都有效。

比如，屋内添加锁定注释，屋外地图没有注释。

那么，在屋内执行插件指令，屋内不会变黑，去了屋外，会发现已经变黑。

**3) 复杂黑暗层开关**

如果你想制作：一个房间，可以开灯关灯，然后另一个房间，也可以开关灯。

按照黑暗层开关来设计，就变得复杂的多了，最好避免这种设计。

你应该使用 灯事件的大范围照明开关 来实现，而不是控制整个黑暗层开关。

想办法通过事件照明来绕开复杂情况，思路会清晰许多。

**4)** **黑暗层颜色**

黑暗层的底层原理是滤镜，所以不能修改混合模式。

黑暗层默认是固定黑色"#000000"，你可以设置整体为蓝色、红色、绿色。

比如黑暗层改为#000099：

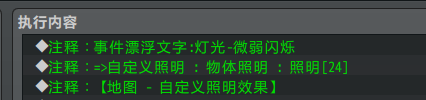


注意，只有 黑暗层开启后 才能有黑色夜晚的效果。

另外，黑暗层不要设置太多的透光，否则与 直接地图滤镜 就没什么区别了。

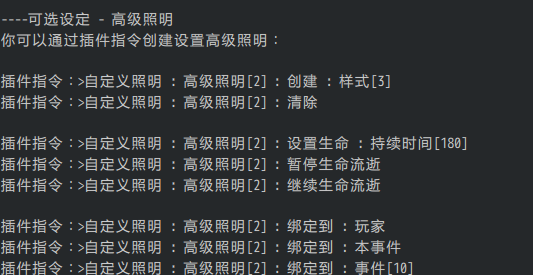
### 照明

**物体照明**：指绑定在具体事件、玩家身上的照明对象，但每个玩家/事件只能绑定一个。物体照明可以用事件注释设置绑定，切换事件页后，需要添加关闭的注释。



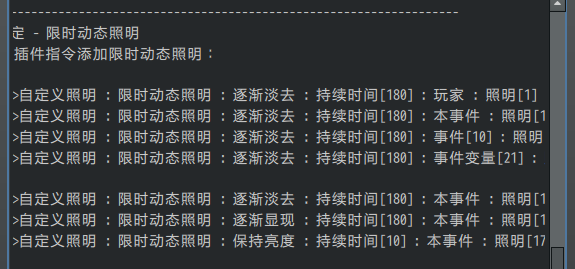
**高级照明**：指独立的照明对象，可以依附于事件、玩家身上。多个高级照明可以绑定在同一个物体上，也可以单独控制移动、透明度变化。

高级照明需要用插件指令单独创建并控制属性。



**限时动态照明**：指只能存在一段时间的照明，时间结束后会被清除。

本质上来看限时动态照明就是 预设的高级照明 。该预设 会从100-200中获取未创建的编号，然后创建一个高级照明，并设置生命时间、绑定对象、透明度变化。



### 光源与颜色

**1) 标准资源**

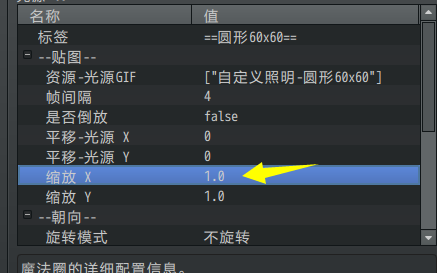
**标准照明资源为白色和透明色。**

因为白色会作为黑暗层相减的颜色，在黑暗层中剪出一个区域。

按照一对一的资源，每个资源都对应一种光源，也就是说你要画很多资源。

你可以偷懒，用重复的资源进行缩放，达到不同程度的光源效果：



**2)** **多种颜色**

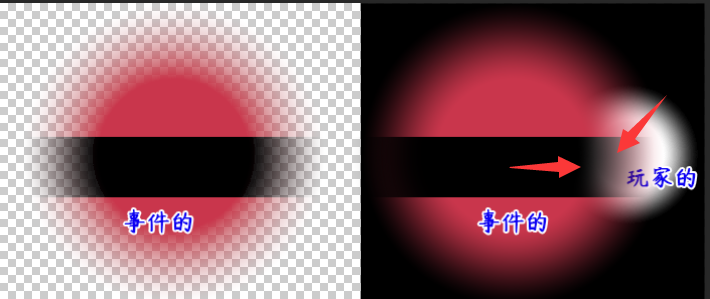
插件与纯色滤镜的功能相似。

光的三原色是：红、绿、蓝。

黄=红+绿。紫=红+蓝。青=蓝+绿。白=红+绿+蓝。

注意，你可以在资源图片中画黑色，黑色和透明的效果一样，不影响黑暗层颜色。

但是基于之前光源相互盖住的问题，你会发现黑色会盖住别的光源，强行画黑。



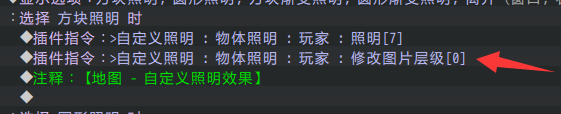
### 光源叠加

**1)** **图片层级**

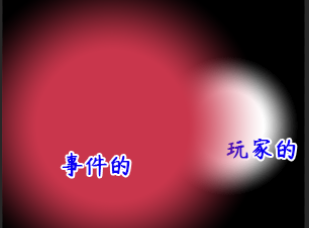
光源之间是可以相互叠加的，但是叠加方式和资源图片的先后顺序有一定的关系。

你可以自定义光源的图片图层，也可以通过插件指令临时修改。





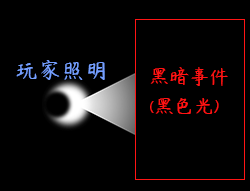
**如果光源只是白色，上述先后关系是看不出来的，不需要区分**。因为白色叠加后还是白色。而这里以红色光源和白色的为例：

因为玩家光源在1层，事件光源在2层，玩家光源处于下方，所以事件会盖住玩家的白色光源。

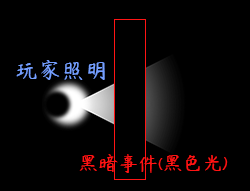
**2)** **遮挡设计**

如果你需要设计室内的墙壁，很简单，绘制一个纯黑色图片资源，然后绑定到事件身上，即可实现遮挡效果。



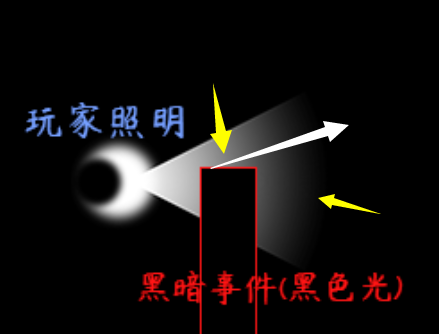
但是，由于都是图片资源，这样存在一些局限性，比如，下图的情况。

如果目标墙壁太薄，玩家的照明是可以直接透过墙壁的。



这个问题是**无解**的，因为电脑无法识别两张 资源图片 之间怎样实现光线遮挡。

比如，下图的情况。



系统需要对中心点进行切割放射，只有矢量光线(点列表)才能做到，而对于图片资源（色彩矩阵）来说，这显然做不到。

该插件使用图片资源进行照明配置，能极大地节省光线计算的性能，还能完美自定义。而选择了图片资源，就必须抛弃 光线遮挡 这类消耗极大性能的计算算法。毕竟鱼和熊掌不可兼得。

总而言之，要绕开这个问题，可以把墙壁做成至少3个图块的厚度，这样看起来就不那么突兀了。

### 性能影响

因为黑暗层和光源是在整个地图画面的基础上，绘制一层遮罩。

照明效果与滤镜相比，滤镜是每个事件、图片都加上一层滤镜。所以如果事件一多，滤镜的性能消耗会上涨的特别快，而照明在事件多的情况下上涨不会那么明显。





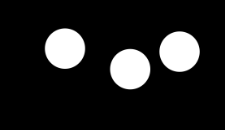
经过数次优化，照明插件的性能还是比较难压下去，因为主要消耗GPU的能力，**通过客户端打开的游戏没有掉帧问题，而用浏览器进行游戏会比较吃力**。

## 从零开始设计（DIY）

### 建立黑暗地图并添加照明

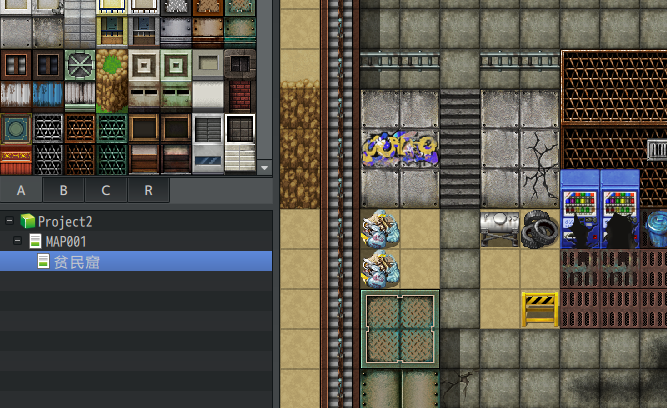
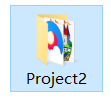
#### 1. 设置一个目标

这里作者我想建立一个地图，进入地图后为全黑，看不见玩家，但有部分照明。



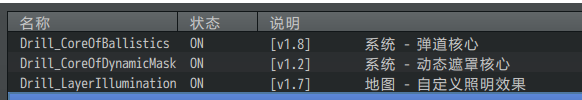
#### 2. 新建工程

新建一个工程，导入随机一张地图。



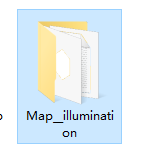
#### 3. 添加插件

在列表中添加照明插件，照明插件中有一些默认的照明配置。





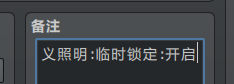
把示例中，img/Map\_\_illumination 文件夹全部内容复制到工程文件下。



#### 4. 开启黑暗层

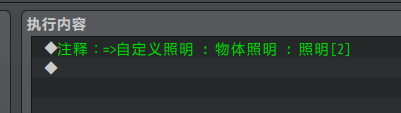
在地图备注中，添加黑暗层指令，确保黑暗层被开启。

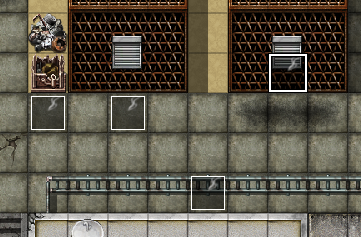
=>自定义照明:临时锁定:开启



#### 5. 添加事件

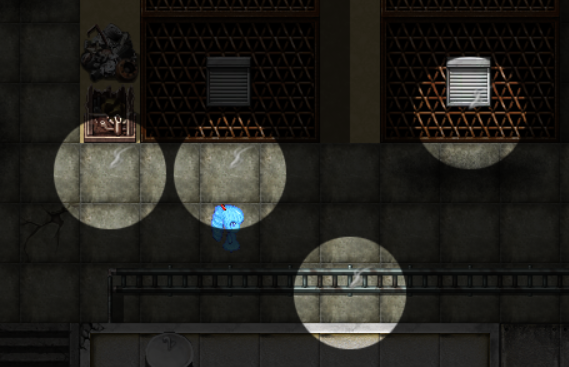
在地图中添加几个事件，并且设置绑定照明1。





#### 6. 测试游戏

进入游戏中，可以看到这些事件都能正常发光了。



#### 7. 细节调整

照明设置中，可以对相关的配置进行设置。



你也可以在一些特殊情况下，用插件指令添加“限时动态照明”，创建只能发光一小会的光源。

