## 概述

### 插件介绍

插件基于核心：

◆Drill\_CoreOfBallistics 系统 - 弹道核心

◆Drill\_CoreOfGaugeMeter 系统 - 参数条核心

◆Drill\_CoreOfGaugeNumber 系统 - 参数数字核心

插件本体：

◆Drill\_GaugeForVariable UI – 高级变量固定框

该插件的扩展：

（暂无）

该插件的配置项非常多，需要花较长的时间耐心学习。

**这个插件的功能是实现 变量值 可视化而存在的，一切都是基于变量值变化而显示的。如果该框没有 绑定 的变量/物品，则没有任何显示的意义。**

### 相似文档

以下为设计方法相似的插件文档，可以结合理解：

|  |  |
| --- | --- |
| 插件名称 | 文档名称 |
| Drill\_EnemySimpleHud  战斗UI - 简单生命框 | 5.战斗UI > 关于战斗简单生命框.docx |
| Drill\_GaugeForBoss  UI – 高级变量固定框 | 13.UI > 关于高级BOSS生命固定框.docx |
| Drill\_GaugeForVariable  UI – 高级变量固定框 | 13.UI > 关于高级变量固定框.docx |
| Drill\_GaugeOfBufferTimeBar  UI - 缓冲时间条  Drill\_GaugeOfBufferTimeNum  UI - 缓冲时间数字 | 13.UI > 关于缓冲时间条与数字.docx |

### 插件关系

高级变量固定框的关系如下图。



## 固定框样式

### 结构

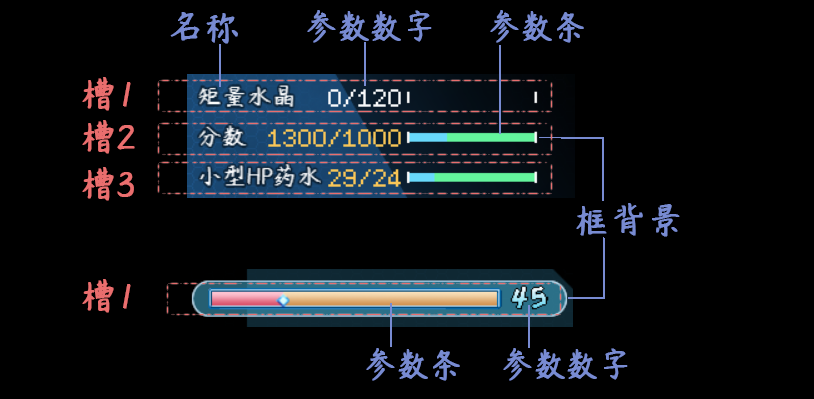
固定框的结构如下图所示。

**简单来说，变量框就是 外框+n个槽样式。每个槽样式都有1个参数条、1个参数数字、1个名称。参数条和参数数字要去相应的核心先设计好，再贴到框样式中**。



固定框样式需要用过变量框设置来绑定变量/物品数据。绑定后，变量/物品的数据值会提供给相应的 参数条和参数数字，参数条和参数数字会根据参数的值，进行自我变化。

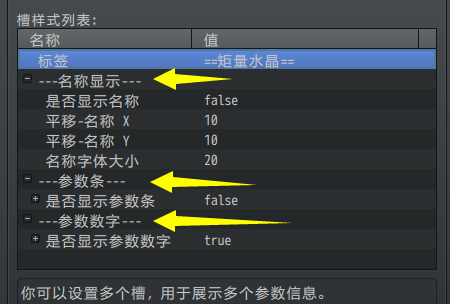
以 标准风格 和 垂直表单风格 为例，大致结构如下：



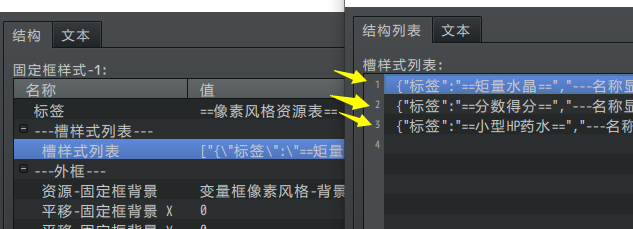
### 槽样式/槽数据

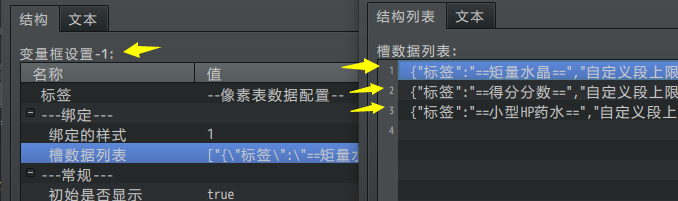
一个变量框可以有多个槽，每个槽都有1个参数条，1个参数数字，1个名称。

槽的数量没有限制，你可以在一个样式里面放很多个槽。



如果你在样式中配置了3个槽，那么在变量框设置中，需要填写3个配置数据。





### 参数条

**1) 参数条样式**

下图为背景框与参数条的组合关系。

**参数条的详细样式配置，可以去看看”关于参数条.docx”，这里不赘述**。

参数条：（参数条核心中配置，段资源+流动+遮罩）

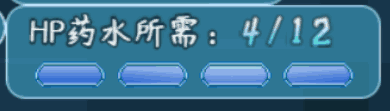
C:\Users\lenovo\AppData\Local\Microsoft\Windows\INetCache\Content.Word\变量框样式4-参数条.png

C:\Users\lenovo\AppData\Local\Microsoft\Windows\INetCache\Content.Word\变量框样式4-参数条遮罩.png

外框：（高级变量固定框中配置）



组合后形成：

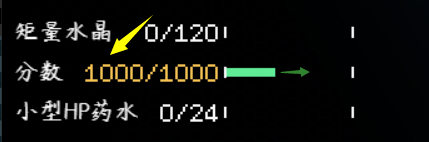


配置参数条坐标时，注意要使得参数条与背景框完美嵌入。

**2) 加满动画**

加满动画是指，框出现时，参数条加满过程的动画。

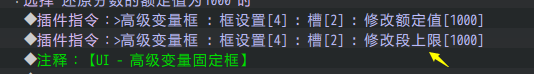
**注意，当你切换菜单界面、切换地图、进入新战斗时，参数条会重复播放一次加满动画**。



该动画在许多动作类游戏中的boss都有展现。而在变量框中不常用，一般建议关闭。

**3) 插件指令修改段上限**

变量框中，你可以通过插件指令临时修改段上限。



### 参数数字

**1) 参数数字样式**

**参数数字的详细样式配置，可以去看看”关于参数数字.docx”，这里不赘述**。





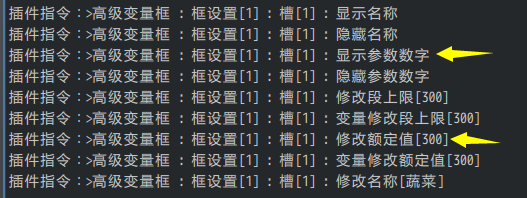
图中有背景框，1个参数条，1个参数数字。

图中的数字绑定了一个变量，该变量会随时间变化。

变量数字：右对齐 + 额定值关闭

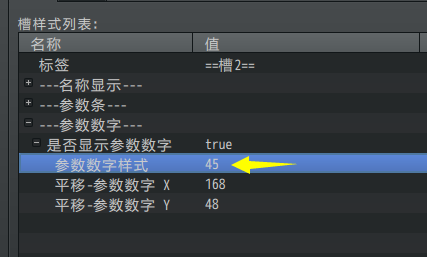
**2) 插件指令临时显示数字**

你可以使用插件指令设置参数数字在战斗或者地图中出现。



但是前提是，你必须配置样式，如果样式为0，则什么都看不见。

额定值也必须开启，不然修改额定值没有任何效果。



### 名称显示

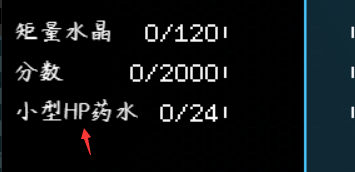
**1) 特殊字符**

名称暂时不能支持特殊字符，也不能支持高级颜色。

**2) 对齐方式**

注意，这里的名称是固定的左对齐，无法控制居中或者右对齐。

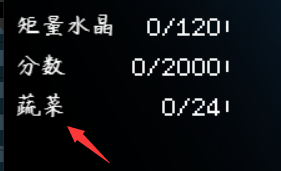
你只能手动控制坐标，调整名称移动至合适的位置。



**3) 插件指令修改名称**

你可以使用插件指令临时修改指定框的 对应槽 的名字。





## 变量框设置

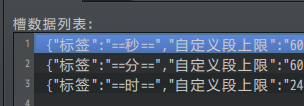
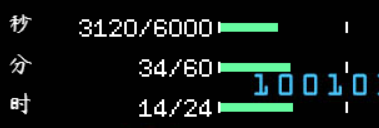
### 绑定关联

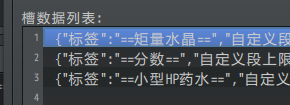
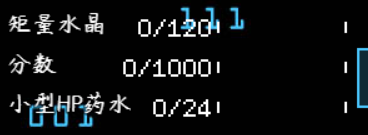
变量框设置的结构如下图所示。



变量框设置的配置比较杂，因为你可以使用相同的样式，来显示多个不同的数据。你可以参考 垂直表单风格 的设置，同一种样式的风格，可以显示不同的数据。



变量 –> 样式 是一对一 关系， 样式 -> 变量 是一对多 关系。

物品 –> 样式 是一对一 关系， 物品 -> 变量 是一对多 关系。

也就是说，样式可以对应给多个物品/变量，而每个物品/变量只能对应一个样式。

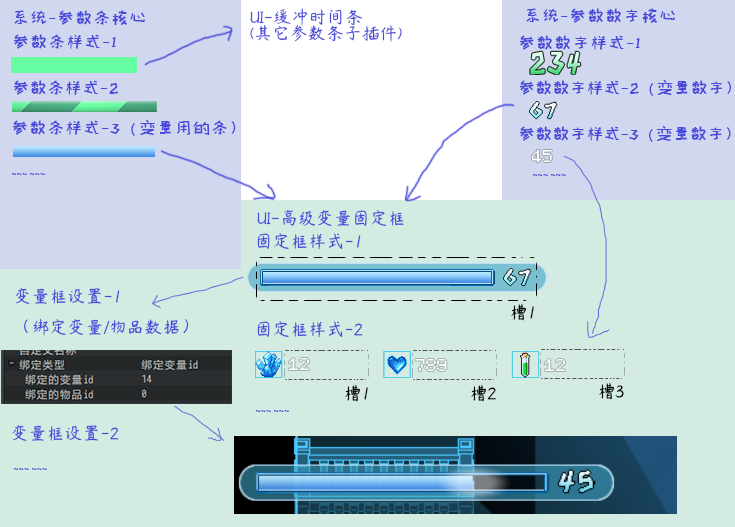
**变量框可以放置一大堆，不过你需要合理分配位置，使得变量框、boss框、菜单ui等各个部件不会拥挤在一起。**

## 设计

### 配置流程

**由于高级变量框、参数条核心、参数数字核心中都有不同的关联配置，这里建议使用**

**”风格”来区分不同的变量框样式**。防止各个配置纠缠在一起，难以区分。



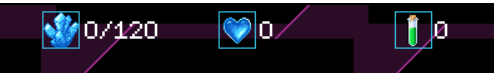
在文件夹中查看时，建议将内容平铺，方便看图片名称和分辨率。



### 示例风格

以下为示例中提供的风格：

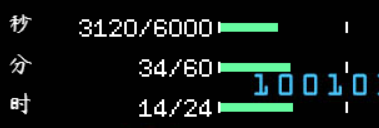
**像素风格资源表**：



**像素风格资源表(文字)**：



**垂直表单风格**：



**标准风格**：



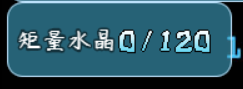
**标准风格(多段)**：



**凹槽板风格**：



**计分板风格：**



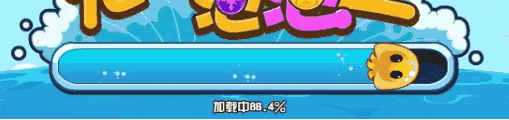
### 参考对象

**变量框的设置非常多，由于与变量直接关联，只要你有想法，都可以模仿实现出来**。

以下是一些游戏的变量框参考：

《q宠企鹅》无敌泡泡王的加载条

背景 + 参数条（1段 + 不流动 + 粒子效果 + 游标）



《星露谷物语》 的体力条

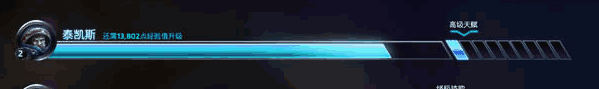
结构为：背景 + 参数条（1段 + 不流动 + 旋转90度 + 弹出条）



《星际争霸2》 的经验条

结构为：背景 + 参数条（1个层级 + 不流动 + 游标）

这里的游标，是非常亮的那一圈光线。（后面的指针做不了……）



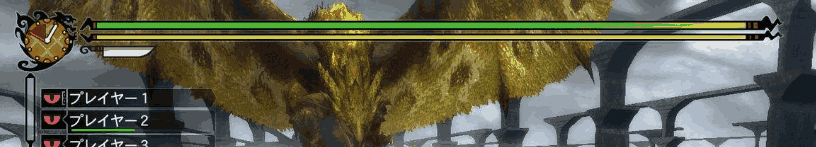
《火炬之光2》的某魔法buff的持续时间显示框（用缓冲时间条也可以做）

背景 + 参数条（1段 + 不流动）



《怪物猎人》的生命条和耐力条

背景 + 槽1：生命条（1段 + 不流动） + 槽2：耐力条（1段 + 不流动）



《魔兽争霸3》的任务计分板

背景 + 名称 + 参数数字（右对齐）



## 从零开始设计

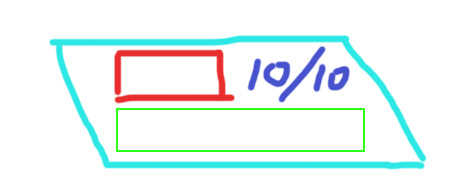
在开始设计前，你可以先了解一下 [变量框的配置流程](#变量框的配置流程) 。

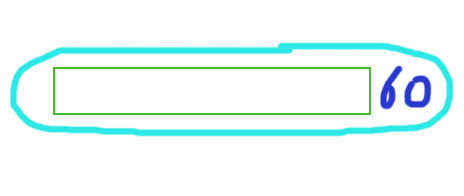
|  |
| --- |
| **关于审美：**  经过群友提醒，作者我才注意到，设计出来的框架可能【不好看】。  这是因为作者我编写文档时，是以一个开荒者的角度思考问题的。开荒者关注的侧重点是一个满足功能的框架和一个完整的教程说明，美感是次要因素，所以自然显得难看。  作为读者的你们，更多要关注的是游戏的细节和UI的设计。所以，千万不要把我的拙劣的美术当成UI的能力上限。希望你们能在此教程基础上设计出更漂亮的UI。 |

### 1.设置一个目标

首先要想明白自己想创建一个什么样的boss生命框，这一点非常重要。Boss框的功能并不是堆得越多越好，重点还是要贴合你游戏的画风。

之前也做过一些简单的变量框，比如下面是一些灵感草稿，基本都是名字、参数条、参数数字三者强行加上：





这里，我想做一个不一样的。

比如，示例游戏中有收集矩量水晶的小任务，我想把它做成下面这样：



### 2.结构分解，规划区域

你需要将 参数条、参数数字、外框 的结构分离。因为三者都在不同的插件中配置。

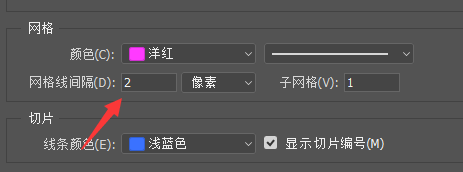
经过分析，需要设置3个槽样式，每个槽样样式只勾选参数数字结构，姓名和参数条都关闭。由于变量固定框没有图标文本功能，这里将图标画在外框背景中即可。

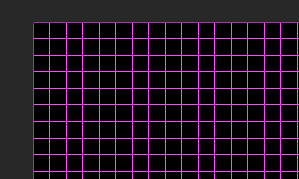


### 3.起草资源

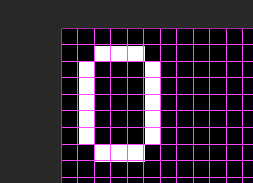
**1）参数数字资源**

参考图中的数字非常简洁干净，这里我打算使用自己画的像素数字来进行模仿，先开启2x2的网格：

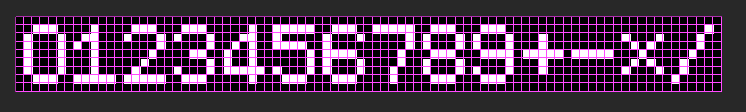




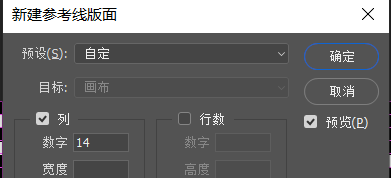
由于之前在 ”关于缓冲时间条与数字.docx” 中画过数字点阵图，受到其灵感的影响，这里为画的第一个数字0：



接下来画好剩余的数字。

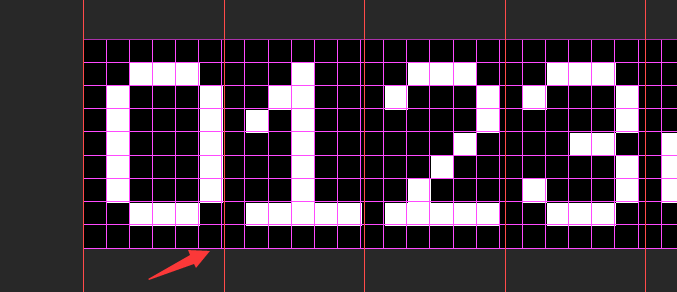


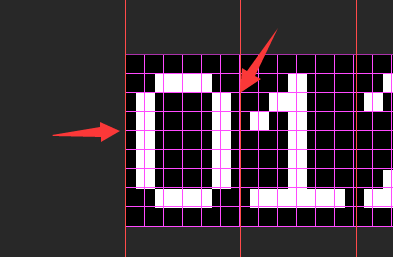
由于数字会被平均切割成14份，这里画好后，需要检查一下。

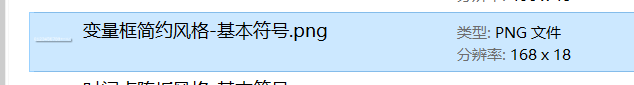


发现网格和等份线并没有贴合。后来发现是数字之间像素间距的问题。

这里每个数字的大小为12x18。



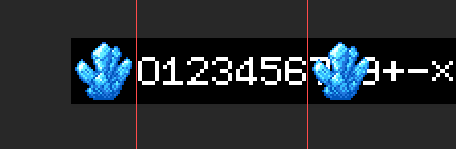




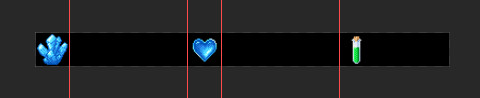
**2）外框背景**

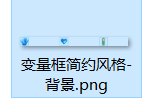
考虑到示例中要收集100多个水晶，那么字符的间距至少要”100/100”，7个字符，那么一个槽的宽度至少为：12\*7 = 84像素。

把矩量水晶放入，由于矩量水晶的图标是32x32大小，所以这里看起来间距需要加大。



初步考虑后，将分成下面三种表示物：矩量水晶、得分分数、小型HP药水数量。



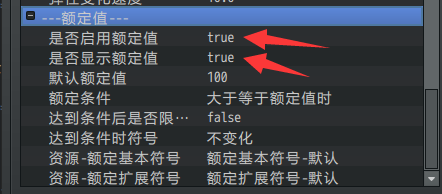
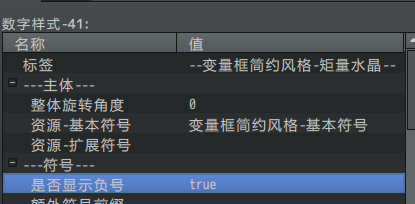


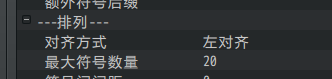
### 4.配置样式

**1) 参数数字配置**

进入参数数字核心。首先是矩量水晶的符号，矩量水晶的数字是”100/100”的格式，所以这里需要启用额定值。注意是左对齐。

另外，这里的默认额定值不需要考虑，因为变量框插件的数据后面会覆盖这个值。





一般数字就简单配置就可以。



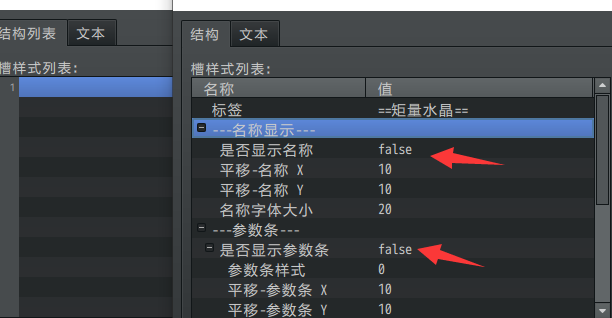
**2) 变量框配置**

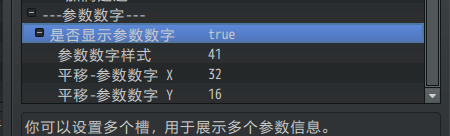
进入 高级变量固定框 插件。进入固定框样式，先把背景配上。

这里提及一下，固定框样式 和 变量框配置 之所以分开，是因为前者是样式，后者是数据。*数据有可能会在游戏中临时修改，而样式是固定之后不再变化的。*

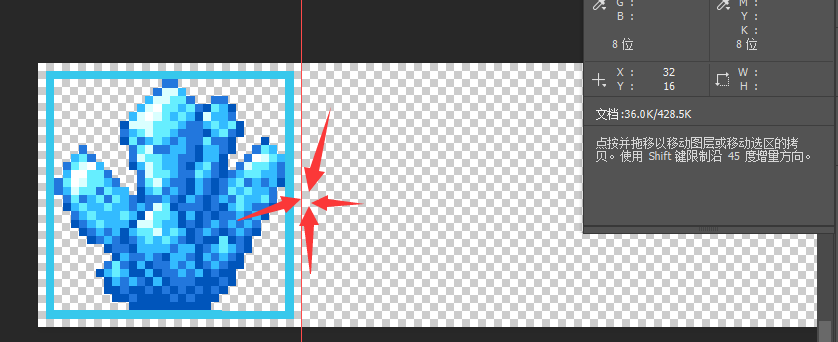


接下来，进入槽样式列表，把3个槽加上。每个槽的名称和参数条都关闭。

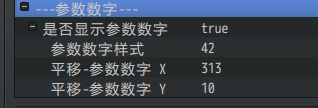
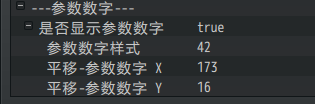


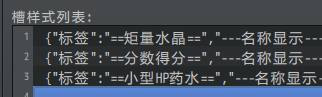


**PS对坐标点**：这里比较关键的是要对齐数字的位置与外框的坐标，由于设置了左对齐，参数数字的中心锚点在左中心，所以你需要确定下图的坐标。



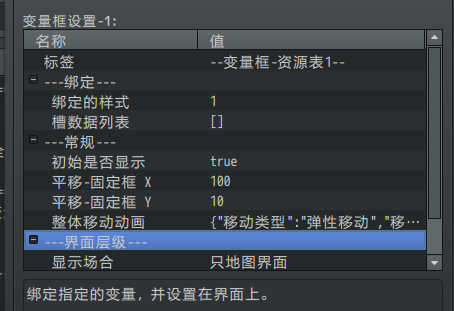
接下来，把后面两个数字的坐标也对上，并填入：



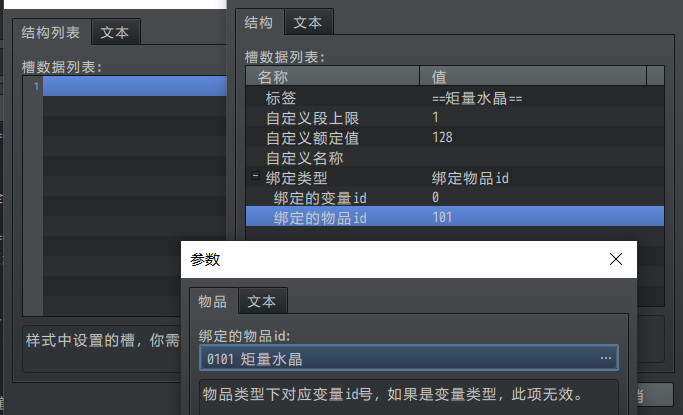


### 5.放置一个实例

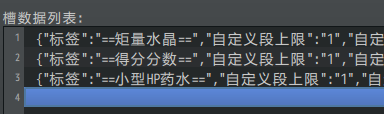
下面，进入变量框设置，配置一个实例：



因为样式1中配置了3个槽，这里的槽数据也要配置3个。这里，段上限和名称都是用不上的参数。（这里我忘了地图里面有多少个水晶，额定值先暂定为128个吧。）



其中得分分数和小型HP药水都没有开额定值，所以也不需要配置。

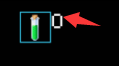


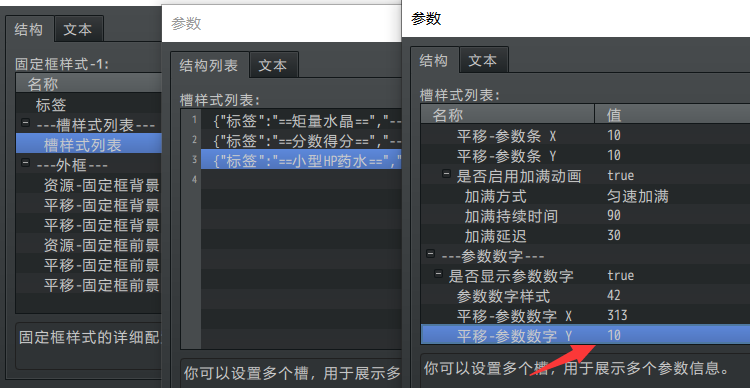
可以看到游戏中出现了框，但是有些参数需要微调。



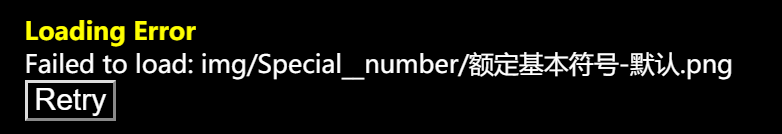
### 6.细节调整

首先，这里的药水数字的y值没有对齐，打开槽样式调整好。设为16。

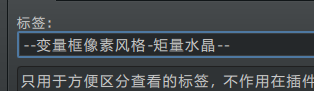


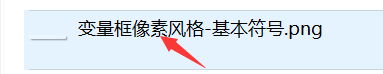


测试前，有时候经常会忘了一些默认资源没有配，这里需要回去找到配置，把资源的设置设为空：



原本制作时觉得会是简约风格，但是实际画出来之后，发现像素感非常浓，所以这里改名为像素风格啦。



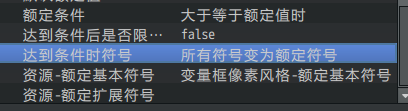


### 7.收尾工作

功能已经完成了大半，还剩下矩量水晶任务完成时，额定值资源的变化。这里可以设置值满了后，变成黄色。









如果有必要，扩展符号也可以加上。

### 8.其他说明

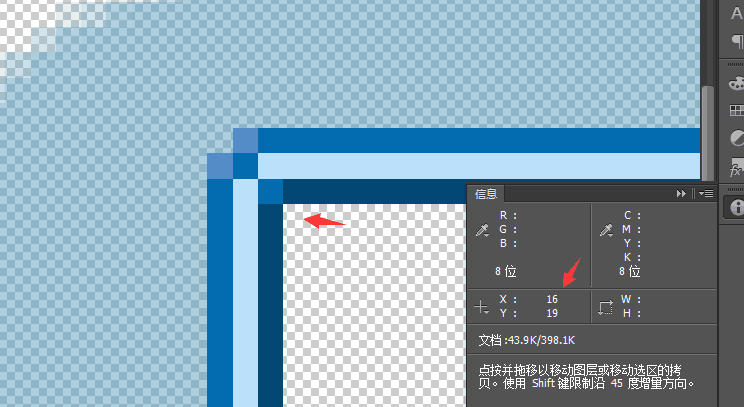
**1) 含有参数条的变量框**

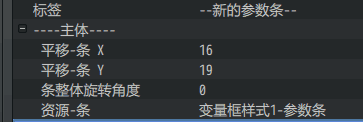
参数条比较麻烦的地方，就是需要要求外框和参数条本身完全吻合。

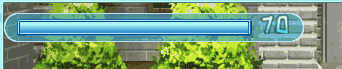


这些都需要通过 [PS对坐标点](#PS对坐标点) 的方式对应。

确定参数条的基本坐标，然后设置条的位置。







**2) 将之前设计的像素风格的图标换成名称**

因为涉及了不同的资源图片，所以这里只能做成两种固定框样式来控制。

这里需要注意的是，绘制的名称是固定左对齐的，不能居中不能右对齐。需要慢慢偏移对齐到坐标。比较麻烦。

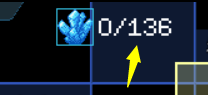




## 常见问题

### 水晶最大值在哪修改

比如，要修改示例中的136数字。



**1）基本理解**

在找这个配置前，你一定要有以下基本理解：

- 这个136是参数数字，不是单纯的贴图。

- 这个136在配置里面有一个参数的值为“136”。

- 这个136参数有一个专有名称，叫 额定值 ，是 参数数字核心 里面的定义。

这些理解都是在该文档和相关文档中有基本的定义，如果你还对基本内容不理解，再去翻翻看。

**2）缩小插件范围**

确认思路后，我们缩小相关插件范围：

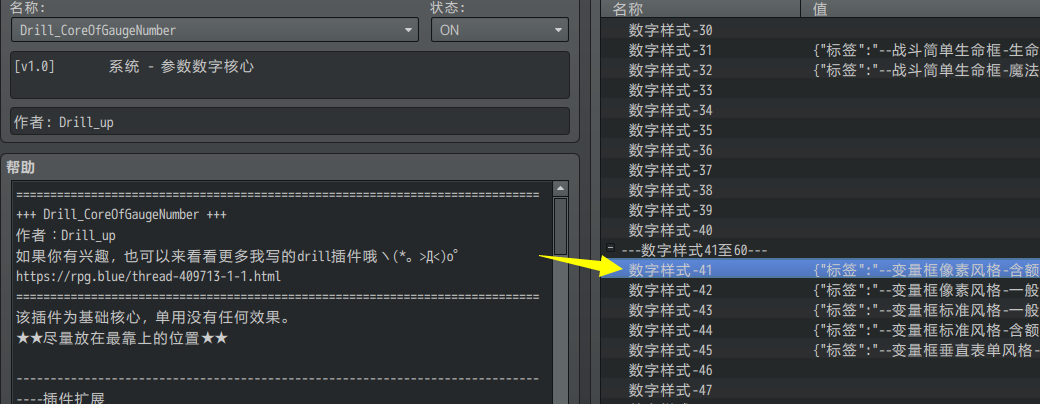
◆Drill\_CoreOfGaugeNumber 系统 - 参数数字核心

◆Drill\_GaugeForVariable UI – 高级变量固定框

配置只在这两个插件里面。

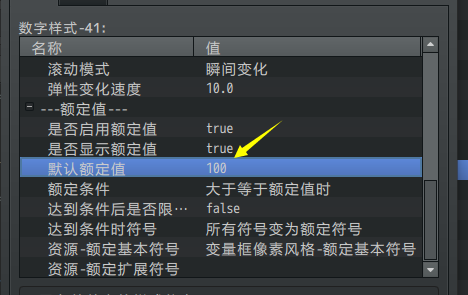
**3）在核心中找**

一般来说，我们配置了参数数字的资源，都会设置一下额定值，所以取核心配置里面找找”136”的额定值。



在这里发现，额定值为默认的100，并不是136。

也就是说，核心的参数数字 额定值配置，在子插件中有赋值。



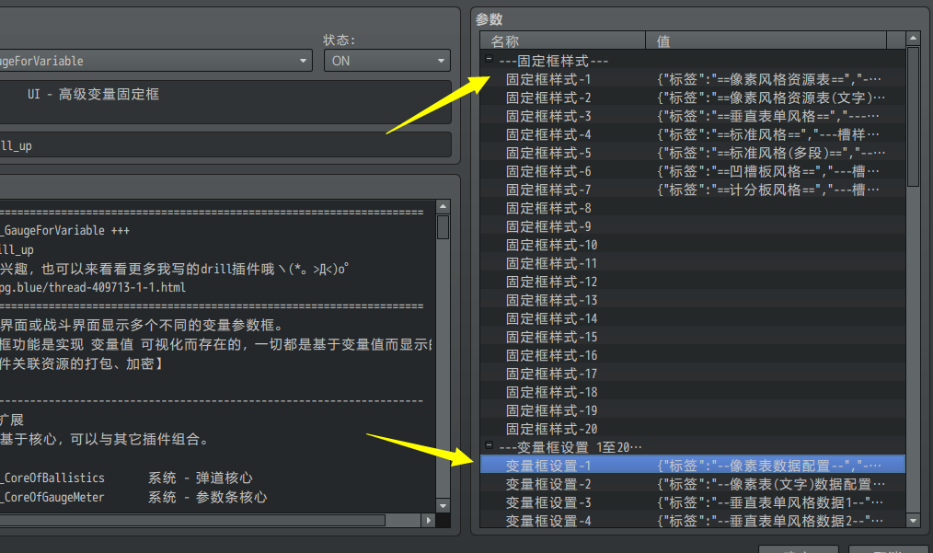
这种情况比较常见，因为 样式和实际框 是 一对多 的关系。

实际的框 不可能永远用同一个额定值，所以子插件多半都会用自己的值覆盖，比如 某个敌人的生命上限，某个角色的魔法上限，或某个框的变量值。

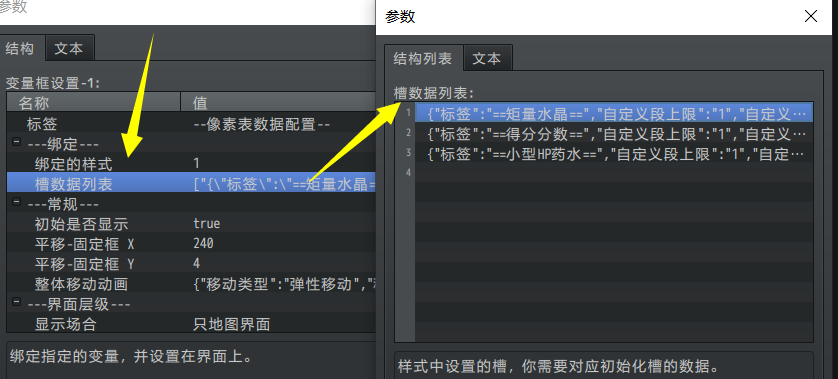
**4）在子插件中找**

打开高级变量固定框，有样式 和设置 两种配置。

根据之前找的过程，可以推断就在 设置 里面。



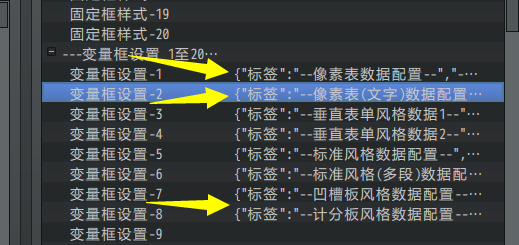
点开槽，可以看到多个配置的槽关系，并且发现有 自定义段上限 的配置。



点开，就发现了额定值的配置了。



另外，需要注意的是，由于示例里面，你可以切换好几个种变量框的样式，这就意味着有好几个框的 额定值 配置都需要改。



所以如果你没有特殊要求，尽量不要设计 ”多个框表示同一个数据” 。