## 概述

### 相关插件

主要插件：

◆Drill\_LayerCommandThread 地图-多线程

◆Drill\_BattleCommandThread 战斗-多线程

公共事件包含三种分类情况，对应后面章节：[多线程](#_多线程)，[执行渠道](#_执行渠道) 。

如果不仔细区分，很容易混淆概念，该文档需要反复阅读才能理解。



### 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：

|  |  |
| --- | --- |
| 串行与并行 | [串行](#串行) [并行](#并行)  [地图事件](#地图事件) [战斗事件](#战斗事件) [对话框阻塞](#_对话框阻塞) |
| 公共事件 | [多线程](#_多线程) [执行渠道](#_执行渠道)  [事件指令执行](#_2）事件指令执行) [自身触发条件执行](#_3）自身触发条件执行) [技能执行](#_4）技能执行) [插件执行](#_5）插件执行) |

## 串行与并行

### 定义

**串行**：是指指令必须按照顺序依次执行，如果途中出现了等待的情况，那么指令都会被阻塞，必须等到 指令完成后 才能执行下一条指令。

（比如地图事件中的 自动执行、确定键 的执行过程。）



**并行：**是指指令可以在同一个时间中，分别执行。指令之间相互独立，互不干扰。

（比如地图事件中的 并行处理 的执行过程。）

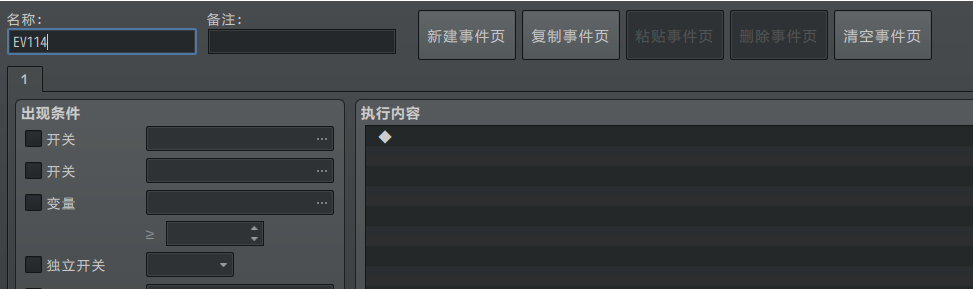


|  |
| --- |
| 注意，串行和并行是对于 **事件页** 而言的，事件页与事件页之间可以并行或串行。  事件页自身中的所有指令，都必须从上往下按顺序依次执行。  另外，**对话框**事件指令是特殊的指令体，只要执行对话框，就会强制串行，阻塞其他所有事件的线程，这里的 并行串行 不包含该指令体造成的影响。 |

### 事件

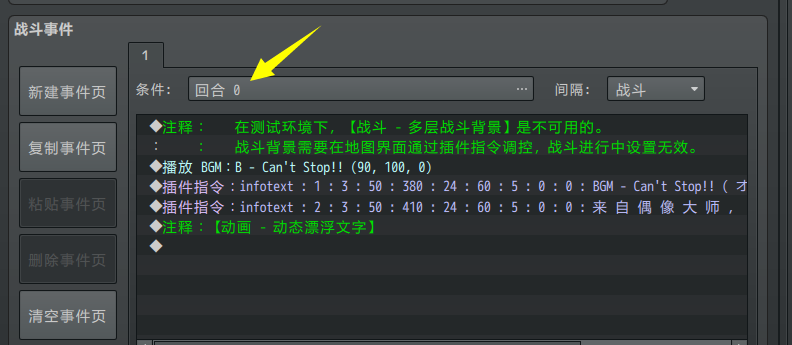
**地图事件**：即地图界面中的事件，在地图中双击即可放置一个事件。

地图事件有行走图、坐标、自主移动、触发条件等各个属性。



**战斗事件**：即战斗界面中的事件，在敌群中进行设置。

注意，战斗事件没有并行的能力，并且触发条件都是根据战斗阻塞串行的流程进行的。



### 对话框阻塞

对话框事件指令 是特殊的指令体，

只要执行对话框，就会强制串行，阻塞其他所有事件的线程。

因此，如果你的指令中有 对话框 出现，那么用串行。

如果没有对话框，那么 串行并行 随意。



因为并行中如果使用对话框，会造成数据执行错位的问题。

相反，如果你必须使用并行，那么需要尽可能避免出现对话框的指令。

|  |
| --- |
| 既是对话框又可以并行的功能插件也有，就是：临时漂浮文字。  可以去看看“13.UI > 大家族-漂浮文字.docx”。 |

### 触发与时间

#### 1）同一帧的先后顺序

触发的先后顺序，与事件id有关。

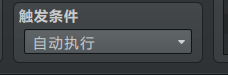
假设触发发生在同一帧内，那么：

目标事件id小的触发，会被先执行，

目标事件id大的触发，会被后执行。

进入地图后，程序会从id=1的事件开始，如果这个事件是自动执行的，则加入串行的队列，如果这个事件是并行的，则 并行执行指令 。





#### 2）时间差

串行与并行存在 **多个事件** 之间的 **时间差** 的问题。

如果初始化的事件都为串行(自动执行)，那么所有串行的事件指令都能够根据id顺序先后执行事件指令。

而并行执行就不一定了。等待指令需要时间、角色移动需要时间、角色转向需要时间、地图淡入淡出也需要时间。如果这些指令使用并行来执行，那么你无法确定哪个事件最先执行完毕，哪个最后执行。

|  |
| --- |
| 添加等待后，那么触发的先后顺序就发生了改变，  因为触发过程与执行处于不同的帧。  （”8.物体 > 触发的本质.docx” 文档中也有 触发与时间 的介绍。） |

## 公共事件

### 多线程

公共事件分成下面两种方式，如下图。



如果不加任何插件，游戏默认支持情况是下面这样的。

（战斗界面无法实现并行）



通过下面插件，能实现 地图界面/战斗界面 的 串行/并行，可见：[5）插件执行](#_5）插件执行) 。

◆Drill\_LayerCommandThread 地图-多线程

◆Drill\_BattleCommandThread 战斗-多线程



### 执行渠道

#### 1）定义

**执行渠道：**指公共事件的执行渠道，分为下图的四种方式。



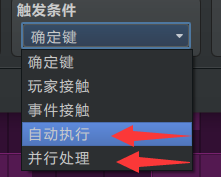
将 地图界面/战斗界面、串行/并行 继续细分，如下图。



#### 2）事件指令执行

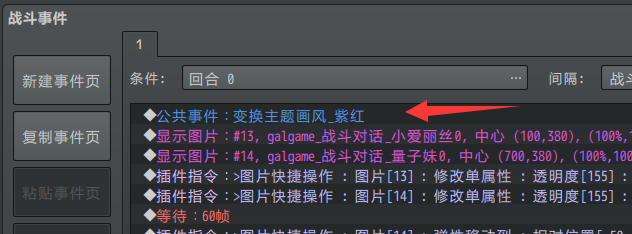
a）添加方法（地图事件）

你可以在地图事件中，添加新的公共事件运行，串行并行与事件设置相关。



b）添加方法（战斗事件）

你可以在战斗事件中，添加新的公共事件运行，但这里只能在战斗界面串行执行。



c）细节说明

简单来说，公共事件在这里相当于一个便捷的函数封装库。

能够封装一系列事件操作。

公共事件中可以有“本事件”的指令，比如开启独立开关操作，“本事件”指向的是执行这个公共事件的地图事件。其它情况下，由于没有“本事件”，所以这些操作无效。

另外，如果你在公共事件中使用插件指令“事件[110]”，

那么对应的事件是你当前所在地图中，id为110的事件。

d）串行与并行

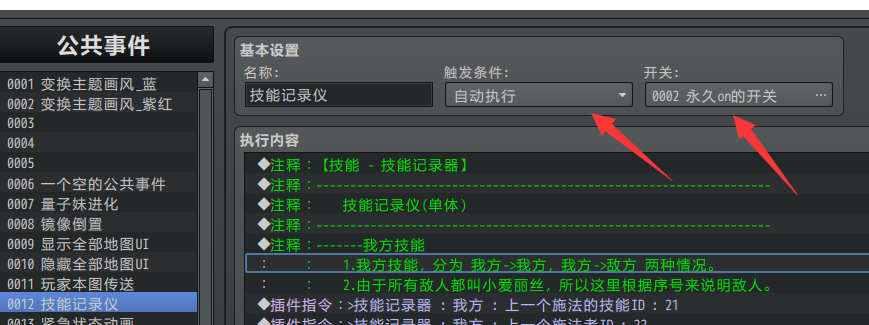
如果事件的设置是自动执行(串行)或其它功能，那么公共事件会阻塞，依次运行。

如果事件的设置是并行处理，那么公共事件与事件指令一起并行运行。

#### 3）自身触发条件执行

a）添加方法

你可以在公共事件自身中，添加触发开关或者并行触发条件。



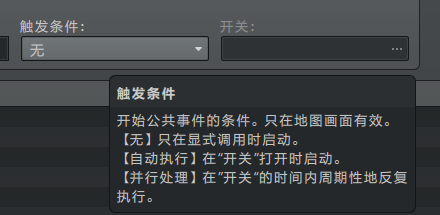
b）细节说明

必须要设置触发条件，才会执行，

如果触发条件为无，则只能通过 事件指令或技能 来执行。

c）串行与并行

注意，自动执行(串行)和并行处理的功能，只能在地图界面中有效。



#### 4）技能执行

a）添加方法

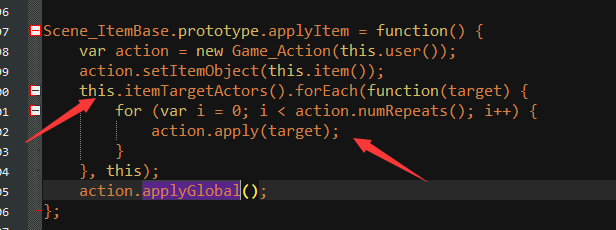
你可以在技能效果中，添加公共事件执行。



b）细节说明

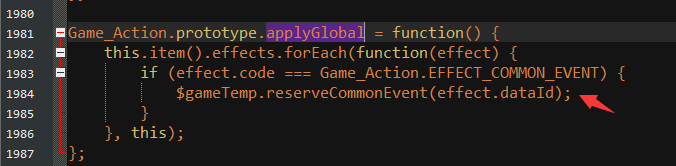
技能中如果写了多个效果和公共事件，那么先顺序执行效果，再顺序执行公共事件。

这里需要从脚本层面上解析，因为公共事件和一般的hp增减功能不一样：



技能范围确定了多个使用者，所以必须先让使用者顺序执行其它效果，再执行公共事件。

图中的ApplyGlobal即执行公共事件函数。



如果有多个公共事件，公共事件之间根据顺序排序执行

c）串行与并行

**通过技能执行的公共事件无法并行**。

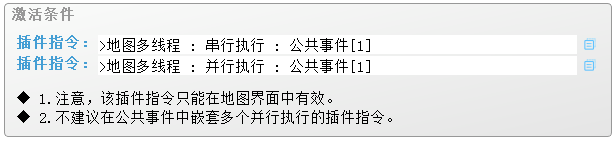
#### 5）插件执行

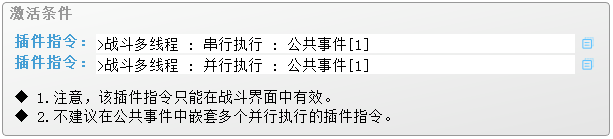
通过下面插件，能实现 地图界面/战斗界面 的 串行/并行：

◆Drill\_LayerCommandThread 地图-多线程

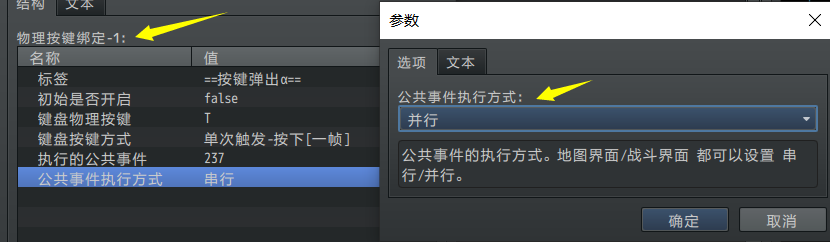
◆Drill\_BattleCommandThread 战斗-多线程







基于多线程的子插件，也能设置 地图界面/战斗界面 的 串行/并行 设置。



## 子插件功能

### 【控件 - 主菜单选项按钮管理器】

子插件：

◆Drill\_WindowMenuButton 控件 - 主菜单选项按钮管理器

从原理上看，插件的本质是走 [事件指令执行](#_事件指令执行) 的流程，在地图中创建一个”自动执行”或”并行处理”的事件，并使其执行对应的公共事件。



|  |
| --- |
| 注意，**菜单界面中不能执行公共事件**。  主菜单选项按钮管理器 的虽然是作用于菜单界面，但是在主菜单的按钮执行后，会立即离开菜单界面，进入地图界面来执行相关公共事件。（在物品界面使用物品公共事件也是一样的原理，会立即回到地图界面执行公共事件。） |