

# 钢铁雄心四模组学前须知

## 前言：

大家好！我是胡桃。在开始介绍 HOI4 模组中的基本要素之前，我有一些话要说：

1、本文本面向于**初学者**和**学艺不精者**，对于那些在 mod 领域有所建树或是有深刻理解的大佬，这篇教学文章很幼稚。大佬们若看到这段文字，也请不要歧视或是贬低初学者们（尤其是某些人仗着新人没有属于自己的 mod 就嘲讽和谩骂新人，这种人我不做评价），这是十分没有意义的事情，谁都是从懵懂无知走向成熟的。

2、请学习者有一颗**谦虚**的心。既然你想要学习钢四的模组制作，那么想必你肯定是热爱这个游戏并且想给游戏社区做出贡献的，有不懂的地方可以问，但**不要伸手夺食**。

3、请学习者一定要对游戏本体有一定游戏理解，否则对制作 mod 是绝对的**大忌**。

## 正文

### 基础三要素

#### 效果 修正 条件

## 效果(Effect)：

要介绍的第一个要素便是 效果，这可以说是钢四模组里最重要的要素。

那么，我们首先要去理解 **效果** 的定义和用处：

【**维基百科描述**】效果（通常称为命令）用于在特定作用域内动态影响游戏。它们是对游戏当前状态的一次性变动，无法产生持续影响。若要实现以数值表示的、持续且永久性的游戏状态改变，需使用修正值(modifiers)。效果块(Effect blocks)不能直接应用修正值，但可以通过添加能够施加修正值的内容（最常见的是通过 add\_ideas）间接实现。

【**个人解释**】效果即瞬间的强制修改。

【**用处**】是游戏内最基本、最常用的修改数值的方法。（注意：效果绝大多数情况下修改基础数值）

结合游戏实际能更好的让初学者理解。我们在玩游戏的时候，通过国策可以得到政治点数、增加基础稳定度等等，这些便是效果最常见的用法和体现。

这种对游戏内的一些关键组成部分进行一次修改、添加或删减的行为（代码），就是效果。

## 修正(Modifier)：

介绍完效果后，我们便开始介绍模组内第二重要的要素：**修正**

**修正** 的定义与用处：

【**维基百科描述**】修正值本质上是一种用于内部计算的变量。但与 Defines（定义值）不同，修正值可以在任意修正块(modifier block)中动态调整。通常情况下，修正值用于创建可逆的、持续且长久的效果，这些效果能够通过数值形式直观体现。

【**个人解释**】修正是一种相对动态的附加数值

【**用处**】是游戏内最主要的修改数值的方法，其可以修改的数值也最广泛的。

而结合游戏的话，有时候我们的基础稳定度明明已经是 100%了，但稳定度一栏还是显示 90%，这时候就是修正发力了，因为修正是作为一种额外的数值附加上去的。

在最终计算数值的时候，修正会产生相当大的影响，并且可以这么说，在计算数值的时候，修正是最后参加计算的。

## 条件(Trigger):

我们刚刚讲了效果和修正，接下来要讲讲条件了。

在讲条件之前，我得说一些有趣的东西，Trigger 只翻译成“条件”其实有点片面，它还可以是“扳机”，也就是“触发器”（枪射出子弹的触发器就是扳机）。有些“效果”是满足了 Trigger 就立刻实现，不管你愿不愿意，这可以称之为 触发器；而有些“效果”需要满足了 Trigger 你才能去以自己的想法决定是否触发，这也许可以称为 条件。但接下来我会统一使用“条件”这个译名。

**条件** 的定义与用处：

【**维基百科描述**】条件用于检测游戏当前状态是否满足特定要求，且不会对游戏状态产生任何改变。

【**个人解释**】这个还需要解释吗？

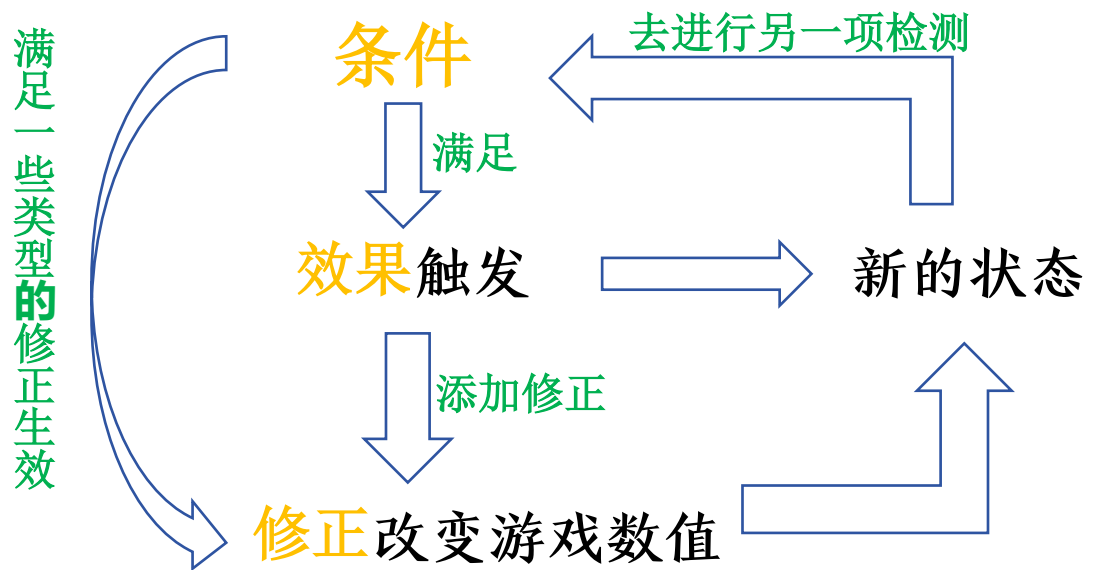
【**用处**】用于检测游戏当前状态是否满足指定要求。

结合游戏，有些国策和决议是需要条件才能选择的，甚至要使某些修正生效都是需要满足条件的。

所以，修正是决定游戏内一个板块是否可用的前提，间接或直接影响了效果和修正是否能出现。

## 效果、修正、条件的关系：

前三个要素在我看来是互相捆绑和影响的。我画个简单的关系图给你们看：



基本就是这样，大家应该看得懂吧。（坏笑）。

我们也可以用一种抽象又形象的方式来讲解：比如你去住宾馆，你超过了原本订房时约定的退房时间，退房给钱的时候，你要给 250 元——200 元是订房的价钱，50 元是额外手续费用。在这个抽象的事情中你一共达成了 2 个条件，得到了 1 个效果和 1 个修正。第一个条件：你预定了宾馆的房间；第二个条件：你超时了。因为满足了第一个条件，所以你要交 200 元。（更抽象一点我们可以这样说，你满足了 `book_room = yes`，所以你触发了一个效果 `add_money = -200`）然后因为满足了第二个条件，你需要交额外的 25% 的手续费。（抽象点说，满足了 `over_time = yes`，修正生效 `extra_cost = 0.25`）

这样能看懂吗？真的（捂脸），我这辈子不会再写出这么难蚌的文案.....

## 重要要素

### 定义 域 本地化 标记 变量 数组

## 定义（Defines）：

在讲完了基础三元素后，就轮到重要要素了。

**定义** 的含义有点难理解：

【**维基百科描述**】定义是游戏内部使用的公开常量，用于控制几乎所有的游戏机制。它们为模组开发者提供了一种途径，可对缺乏直接脚本接口的系统进行调整和配置。

【**个人解释**】定义是决定游戏内部分底层数据的代码，比如最高可储存的 PP 点数（默认为 2000），这是无法用修正和效果在游戏内直接修改的，只能在文件中定义，作用于整局游戏。

【**用处**】用于控制几乎所有的游戏机制，它们为模组开发者提供了一种途径，可对缺乏直接脚本接口的系统进行调整和配置。（摘抄 wiki）

（请 mod 制作者完成小学教育且具有一定计算机知识储备或游戏知识储备再看以上有关定义的解释，否则难以理解。）

那么 定义 如何在游戏中的体现呢？如原版游戏中，在一个地区建造同种建筑最多同时使用 15 个民用工厂。通过修改定义文件，我们可以实现同时使用 150 个工厂建造建筑。又或者原版游戏中情报机构需要升级满 5 次才能解锁一个额外的特工位，我们可以通过修改文件改为自己想要的数字。

所以，定义决定了游戏很多基础玩法的数值。如果说，效果和修正改变了游戏的发展走向，那么定义直接改变了游戏的玩法。

## 域（Scopes）：

域，这个名字对于普通人而言有点拗口，我们也可以称其为 范围 或 作用域。

**域** 的含义：

【**维基百科描述**】作用域用于切换当前选定的实体，以确定效果的应用对象或触发条件的检测对象。

【**个人解释**】表示范围，可以是国家、地区甚至是人物。

【**用处**】辅助效果的使用和条件的检测。

对于玩家而言，域 可能绝大多数玩家都察觉不到其存在或其用处。假设有一个国策的奖励是在莫斯科地区增加一个民用工厂、苏维埃联盟增加 2 个地图外工厂，朱可夫失去“畏惧斯大林”特质，那么在这个奖励里，一共出现了 3 个 域。

哪 3 个呢？1、莫斯科地区；2、苏维埃联盟；3、朱可夫。

因此，这些对象便是域。域可以简单分成效果域、条件域和通用域。我刚刚列举的 3 个例子可以用通用域代表，也可以用效果域代表（一般会使用通用域，更方便）。最简单的区分即效果域最终要服务于效果实现，条件域服务于条件判定，通用域既可以用在效果也可以用在条件。

总结，域决定了效果施加的对象和条件检测的对象。

（以下要素介绍的时候会相对简略）

## 本地化 (localization)

本地化，这是一个说简单不简单，说难不难的部分。我们所写的代码是供钢四的游戏理解的脚本语言，比如 `1 = { add_core_of = SOV }`，对于钢四这个游戏而言，它就会理解这句代码的意思然后让苏联获得省份编号为 1 的地块的核心。而对于人类而言，需要更直观的书面语言才能理解这个代码的意思，所以，我们需要“本地化”这个“1”、“SOV”、“add\_core\_of”（这里 本地化作动词），将它们转换为人类能理解的语言。总结，本地化就是将“游戏能理解的语言”转换成“人类能理解的语言”。

其中，人类能理解的语言也分很多种：英语、法语、德语、西班牙语、巴西葡萄牙语、波兰语、俄语、日语、简体中文(english、french、german、spanish、braz\_por、polish、russian、japanese、simp\_chinese)。以上是钢铁雄心 4 直接支持的语言。

题外话：如何使代码本地化成钢四不支持的语言呢？

我们举繁体中文为例子，若你直接在本地化文件夹加上一个 Traditional\_Chinese 的文件夹，并在里面进行了本地化，那么没有任何用处。因为钢四启动器中并不支持以繁体中文为语言的模式下进入游戏，你只能从我上面列举的这些语言中选择一个进入游戏。为了达成在游戏里看到繁体中文的目的，我们只能选择一种语言，比如英文，然后在英文的本地化文件里修改。这样，虽然你是以英文模式进入的游戏，但因为你在英文本地化中写了繁体中文，那么显示的也会是繁体中文。（注意，本地化并不只是简单的翻译，顾名思义，“本地化”，是将代码语言翻译成“当地人”的语言，使用繁体中文的地区与使用简体中文地区的人们虽然都用着“中文”，但是语言习惯和某些构词是不一样的，所以本地化也是一种学问。）

## 标记(flags):

标记在这里我只讲两个最简单的用法：

- 1、给国家、地区、人物等 域 打一个标签（记号），在检测和快速查找的时候有很大用处，一般与域搭配使用。（比如你是一个面试官，你在你心怡的人选的简历上都贴上了黄色的贴纸，并且记下了他们的名字。但是过了几天，你突然发现我忘记了这些人的名字，所以你翻开了那些简历，把有黄色贴纸的简历挑出来，并录用这些人作为你公司的员工。我们可以将这个黄色贴纸理解为标签。）
- 2、给国家换名字、颜色、国旗也是使用标签完成的。

## 变量与数组(Variables & Arrays):

初学者并不需要大量使用变量，因此我讲的简单一些。变量即是可以随时改变的“量”。比如，1、2、3、4……之类的数字我们称之为常量，这些“量”是固定不变的。但假设我们定义一个变量“a”，这个 a 的值是可以变的，可能它一开始是 1，但你可以在游戏过程中通过效果随时修改它的值。不要觉得变量没用，变量有用的很，同时灵活运用也能防止你的代码变成一片屎山。

数组即多个变量的组合。

## (附) On\_actions:

目前此条目并没有什么好的翻译，真要直译的话，可以说成“行动”。其虽然没有专门的翻译，但作用相当大。

维基百科描述：“On\_actions”是在特定情况发生时执行的一段代码块，例如一个国家向另一个国家宣战时或一个州的控制权发生变更时触发某些效果。

On\_actions 可使用的范围不止上面列举的两个，还有更多可在维基百科上自行查询和了解。

# 附录 1：如何自己寻找这些要素或其使用方法？

很多人会因为找不到自己想要的效果、修正等要素或是某些功能的使用方法而烦恼，我会在此教授一些有用的技巧。

## 寻找游戏内提供的要素（如寻找效果）

- 1、通过本地化文件寻找；
- 2、通过维基百科寻找；
- 3、通过钢四备忘录寻找（这是我自己起的名字）【路径：Hearts of Iron IV\documentation】
- 4、上网找（不推荐）。

## 寻找使用方法

- 1、通过维基百科寻找；
- 2、某些功能蠢驴在游戏本体文件夹里会专门给一个文件向你介绍此功能的使用方法；
- 3、某些功能在备忘录有提及；
- 4、上网找。

# 附录 2：Modder 群群号推荐

好的交流群和老师有助于你的学习

- 1、霜泽图书馆（雪煤馆）：1026294736 ；

（闲聊群可来 950718988）