**新版太阳神三国杀**

**武将扩展学习手册**

**（终结参考）**

编者：独孤安河

鸣谢：幽蓝风暴、女王受·虫、啦啦SLG

首发：百度太阳神三国杀Lua吧

**编写说明**

本手册从2016年8月1日开始编写。

编者：独孤安河

2016年8月1日

**目录**

第一部第一章：创建一个新武将

第一部第二章：添加技能

第一部第三章：武将美化

第一部第四章：创建一个新技能

第一部第五章：添加音效

第一部第六章：添加提示信息

第一部第七章：武将变更

第一部第八章：启用AI

第一部第九章：修改距离

第一部第十章：技能管理

第一部第十一章：修改手牌上限

第一部第十二章：体力变更

第一部第十三章：游戏进程的变更

第一部第十四章：启用标记

第一部第十五章：卡牌转换

第一部第十六章：卡牌传递

第一部第十七章：用卡牌实现技能效果

第一部第十八章：启用标志

第一部第十九章：涉足判定领域

第一部第二十章：私家牌堆

第一部第二十一章：卡牌的锁定

第一部第二十二章：卡牌免疫

第一部第二十三章：让房间提供卡牌

第一部第二十四章：启用标签

第一部第二十五章：濒死结算

第一部第二十六章：交换座次

第一部第二十七章：创建主公技

第一部第二十八章：强力技能

第二部第一章：聊聊天，叙叙旧

第二部第二章：摸牌堆控制术：观星与攻心

第二部第三章：隐藏技能

第二部第四章：触发机制：将干涉进行到底

第二部第五章：获得一个新技能

第二部第六章：限定技与觉醒技

第二部第七章：动画效果

第二部第八章：花样百出的摸牌手段

第二部第九章：不可回避的问题

第二部第十章：防具亦有悲剧时

第二部第十一章：不屈的战魂

第二部第十二章：死亡威慑

第二部第十三章：更改伤害来源

第二部第十四章：最中意的面孔

第二部第十五章：天下归心

第二部第十六章：限制卡牌的使用次数

第二部第十七章：异性之间的话题

第二部第十八章：卡牌强化

第二部第十九章：引入拼点机制

第二部第二十章：多张卡牌的移动

第二部第二十一章：卡牌的封禁

第二部第二十二章：连续发动技能

第二部第二十三章：选择花色

第二部第二十四章：询问使用卡牌

第二部第二十五章：多重视为技

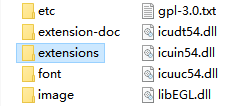
第二部第二十六章：手牌展示

第二部第二十七章：遗计定乾坤

第二部第二十八章：系统秘籍

**第一部第一章：创建一个新武将**

创建武将扩展包，首先要在游戏目录下找到extensions文件夹：



注意：

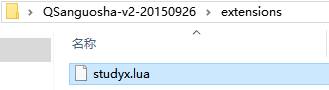
1、游戏目录，是指存放游戏主程序QSanguosha.exe的文件夹。也就是说，QSanguosha.exe和extensions文件夹应该是同级的。

2、如果游戏目录下没有extensions文件夹，可以自己新建一个。

3、文件夹的名字是“extensions”，最后有一个“s”，不能误写成“extension”。

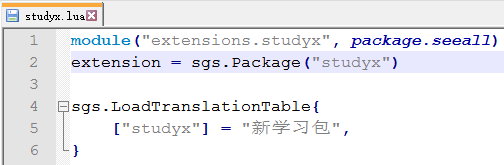
4、不要与上面的extension-doc文件夹混淆。extension-doc中存放的是早期神杀版本中保留下来的DIY教程，由hypercross和William915等前辈编写，对我们进行Lua武将扩展和AI扩展有很大的指导意义。但它绝对不是用来放Lua扩展包的地方！

在extensions中创建好Lua扩展文件后，就可以用Notepad++写扩展包了。



太阳神三国杀目前支持两种写Lua扩展包文件的方式。一种是如第一部中所说的，采用module关键字的方式，每个Lua文件对应一个Lua扩展包；另一种是不借助module而直接在文件结尾使用return关键字的方式，每个Lua文件可以写多个Lua扩展包。不过在本文中，我们仍然会沿用第一部的方法，通过module关键字组织Lua文件的内容。

创建扩展包的过程没有多少变化，所以现在我们的代码是这样的：



这里我们的扩展包名字是“studyx”，翻译为“新学习包”。

注意：

1、任何一款靠谱的文本编辑器都可以用来编写Lua扩展包，并不一定必须使用Notepad++，比如也会有人使用Sublime Text。可根据自己的喜好选择要用的编辑器。

2、不要忘了在“格式”菜单里修改编码格式为“UTF-8无BOM格式”，否则游戏中会出现乱码。

3、绝对不能使用记事本进行编辑！注意名字，“Notepad”是记事本，“Notepad++”才是我们可以用的编辑器。

写完代码之后要记得保存，通常使用“Ctrl＋S”组合键会是一个很方便的做法。如果不记得是否保存过，那么请留意一下上方标签里的磁盘图标的颜色，红色表示尚未保存，而蓝色表示已保存。

|  |  |  |
| --- | --- | --- |
| 标  签 |  |  |
| 含义 | 未保存 | 已保存 |

这里需要重点提一下文件的第2行：extension = sgs.Package("studyx")

赋值号“＝”左边的“extension”是固定的，不能改成别的东西。这是因为，太阳神三国杀在加载Lua的时候，如果遇到用module方式写成的Lua扩展包文件，就会直接去找里面的“extension”，将其作为文件中唯一的Lua扩展包加到游戏中去。

赋值号右边的内容则是创建一个扩展包的具体实现过程。它是通过函数sgs.Package()来实现的。这个函数的原型是：

sgs.Package(name, pack\_type)



包含两个参数name和pack\_type，其中：

①name：表示扩展包的名字。

②pack\_type：表示扩展包的类型，可以填以下几个值：

sgs.Package\_GeneralPack（武将包）

sgs.Package\_CardPack（卡牌包）

sgs.Package\_SpecialPack（特殊包）

不填的话，默认是sgs.Package\_GeneralPack，表示这是一个武将扩展包。所以通常我们是不用理会它的。

用sgs.Package()创建了一个武将包，通过赋值号“＝”交给左边的extension，这样extension就表示我们新创建的这个包了。然后太阳神三国杀在加载Lua的时候，找到这个extension，就会把我们创建的扩展包顺利加入游戏。

接下来是创建新武将“孙悟空”，方法没变，还是用sgs.General()函数。它的原型是：

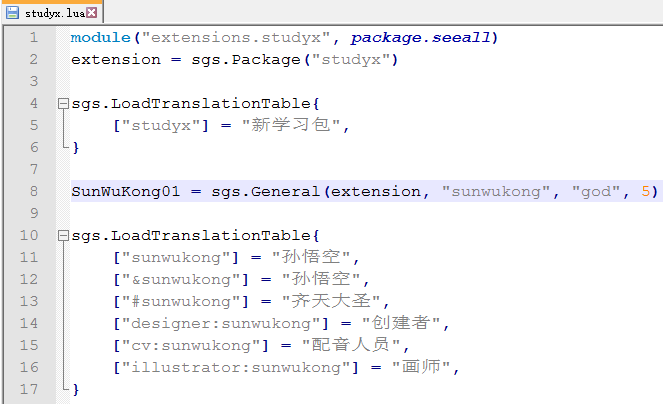
sgs.General(package, name, kingdom, max\_hp, male, hidden, never\_shown)



第一个参数package用于指定武将所在的扩展包，而我们刚才创建的武将包已经用extension表示了，所以这里要填extension，这样就可以将这个新武将加到新创建的扩展包中了。

第四个参数max\_hp表示武将的体力上限，在第一部中我们写的是带双引号的"5”，其实去掉两个引号、直接写成数字的形式也是可以的。这里我们就直接写数字5了。

现在文件中的代码应该是这个样子的：



注意：

1、翻译信息不一定要像第一部那样全写在同一个sgs.LoadTranslationTable段之中，每个Lua扩展包文件中可以拥有多个翻译表，出现在文件的任何合理的位置。

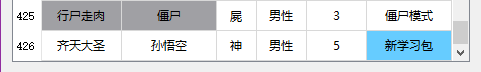
2、请一定注意武将名称和武将对象的区别（详情请见下面的3和4）。

3、出现在sgs.General()函数的第二个参数name位置上的是武将名称，本例中是”sunwukong”，所有关于武将的翻译信息都是根据这个名称决定的。

4、第8行赋值号“＝”左边那个没有双引号的变量SunWuKong01就表示一个武将对象，它代表着一个武将本身，所有在代码中对武将的操作（比如，为武将添加技能）都要通过这个变量完成。

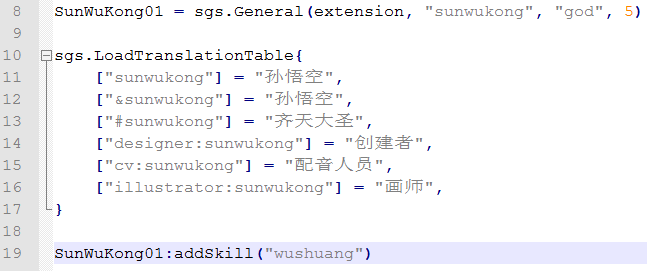
5、注意代码的顺序：由于sgs.General()函数创建新武将时需要用到表示扩展包的变量extension，所以一定要写在extension被赋值之后。即，第2行和第8行的代码是不能交换顺序的。

这样创建一个新武将的任务就完成了。



**第一部第二章：添加技能**

为武将添加技能的方法没有变化。所以为孙悟空添加完“无双”之后，代码应该是这样的：



注意：

1、在为武将添加技能之前一定要先分清楚，这个技能是否已经在游戏中存在。通过技能名称（带双引号的）添加已经存在的技能，通过技能对象（不带双引号的）添加尚不存在的新技能。

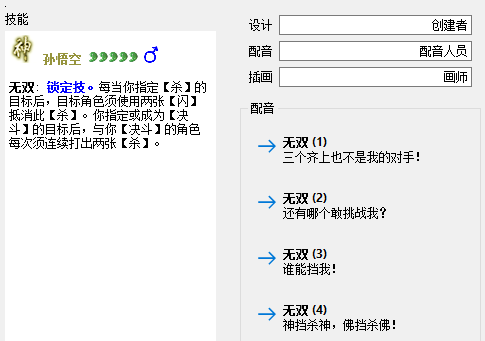
2、像“无双”这样写进源代码的技能就是已经存在的技能，所以只能通过技能名称进行添加。

3、一般来说，Lua扩展包中写的新技能，由于事先游戏中并不存在，所以要通过表示技能对象的变量进行添加。

4、如果要在Lua文件中为多个武将添加同一个新技能，第一次添加时通过技能对象添加（因为游戏中还不存在此技能），从第二次开始，要通过技能名称进行添加（因为第一次已经把新技能加入了游戏，成为了已存在的技能）。

5、关于技能名称与技能对象的具体分析，请稍后参考后续章节的内容。

这样就可以在游戏里看到孙悟空已经拥有技能“无双”了。



另外，本章节还有两个因为版本变化而引起的小问题，请留意一下。

首先，现在源代码链接已经改成了下面的这个地址：

<https://github.com/Mogara/QSanguosha-v2>

然后，添加技能“七星”的代码变为：

武将变量:addSkill(“qixing”)

武将变量:addSkill(“#qixing”)

武将变量:addSkill(“#qixing-ask”)

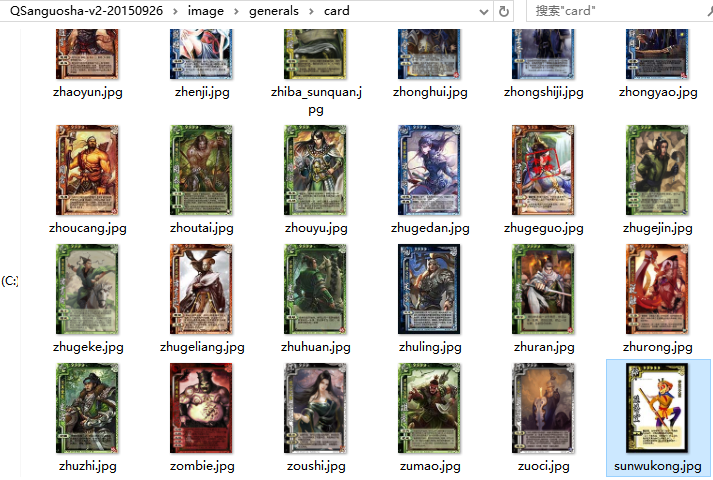
武将变量:addSkill(“#qixing-clear”)

武将变量:addSkill(“#qixing-fake-move”)

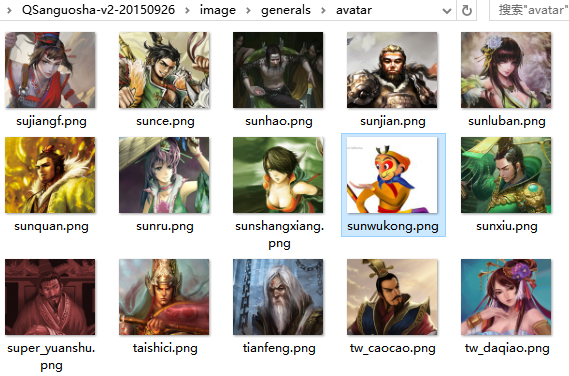
**第一部第三章：武将美化**

与第一部不同的是，目前太阳神三国杀只需要武将卡牌和武将头像两部分内容即可完成对一名武将的美化工作，而武将标签已经不再需要了。另外，武将卡牌的位置也发生了变化，同时还可以为武将添加全身图。

首先，武将卡牌的位置被移动到了游戏目录\image\generals\card\文件夹中，要在这里放入\*.jpg格式的武将卡牌。



然后，将\*.png格式的武将头像放到游戏目录\image\generals\avatar\文件夹中。



这样就可以正常游戏了。

至于武将全身图，相关信息是这样的：

像素尺寸：250×292

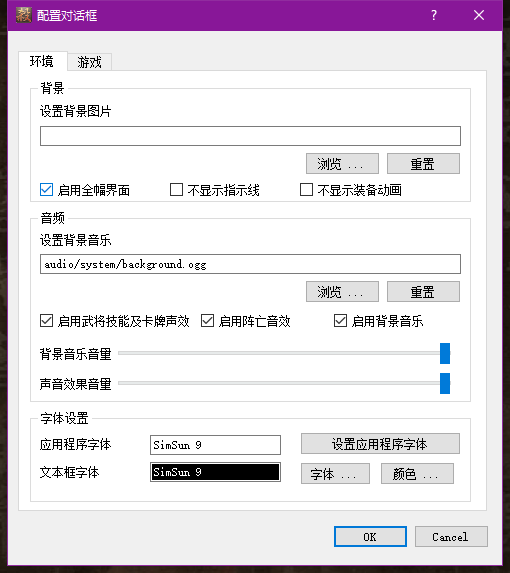
图片格式：png格式（与武将头像相同）

存放位置：游戏目录\image\fullskin\generals\full\文件夹

按照这些信息为孙悟空制作一张全身图，放到要求的文件夹中。



然后进入游戏，在主界面中点击“配置”按钮，在“配置对话框”中勾选“启用全幅界面”，就可以在游戏中体验武将全身图了。

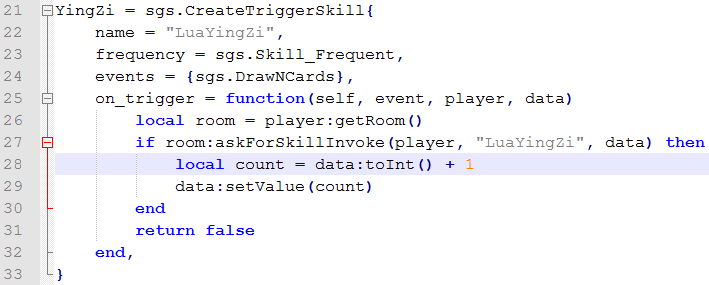


游戏效果（同将模式，并启用了作弊）：



**第一部第四章：创建一个新技能**

这一章的内容没有什么变动。首先找到原版英姿的代码：



然后在此基础上稍加修改，得到我们需要的技能“美王”的代码，并将之添加给孙悟空。



目标就达到了。

注意：

1、注意技能名称和技能对象的区别（具体请见2~5条）。

2、第22行中的”LuaMeihouWang”是这个技能的技能名称，所有与这个技能有关的翻译信息都是由这个名称决定的。

3、第21行通过sgs.CreateTriggerSkill{}方法创建了一个新技能，然后通过赋值号“＝”的作用，用左侧的变量MeiWang代表这个新技能本身，我们称其为技能对象。所有对技能本身的操作都要通过表示技能对象的变量进行。

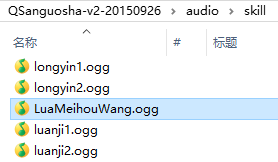
4、为武将添加技能时，如果是游戏中已存在的技能，通过技能名称（带双引号的）进行添加即可；而如果是新技能，就要通过技能对象（不带双引号的）进行添加，因为只有技能对象才包含了一个技能的所有信息，只提供一个技能名称是不能让太阳神三国杀知道这个技能究竟会有什么效果的。

5、本文目前已出现的游戏包、武将、技能三者类比：

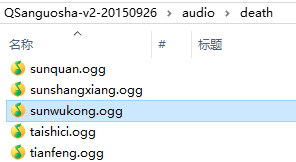
|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 对象 | 创建方法 | 名称 |
| 游戏包 | extension | sgs.Package() | “studyx” |
| 武将 | SunWuKong01 | sgs.General() | “sunwukong” |
| 技能 | MeiWang | sgs.CreateTriggerSkill{} | “LuaMeihouWang” |

**第一部第五章：添加音效**

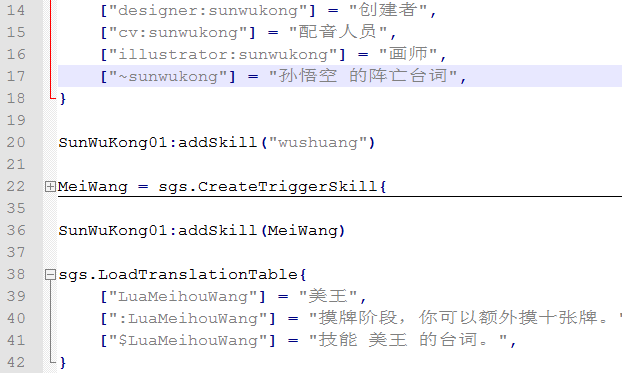
音效文件的格式和位置要求没有变化。制作好配音文件后，首先把技能“美王”的配音文件放到游戏目录\audio\skill\中。



再把孙悟空的阵亡配音文件放到游戏目录\audio\death\中。



然后写好技能台词和阵亡台词的翻译即可。



这样就把配音添加到游戏中去了。

不过对于技能配音而言，现在还缺少关键的一步。此时我们还不能在游戏中听到技能配音，因为太阳神三国杀并不知道要在什么时候去播放配音。因此，我们需要在技能执行过程中使用代码对技能配音的播放进行控制。这里要用到的是Room中的函数broadcastSkillInvoke()，它的原型是：

broadcastSkillInvoke(skillName)

broadcastSkillInvoke(skillName, category)

broadcastSkillInvoke(skillName, type)

broadcastSkillInvoke(skillName, isMale, type)



有四个原型，不过通常我们只用第1个和第3个。它们总共涉及两个参数：

①skillName：表示技能名称。

②type：表示配音文件的编号。

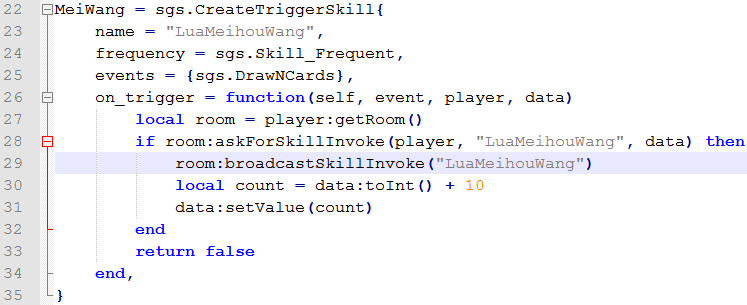
其实通过这两个参数找到对应的配音文件是很简单的，只需要把两个参数的值连接起来就是目标文件的文件名了。比如我们可以这么写：

room:broadcastSkillInvoke(“LuaMeihouWang”, 2)

就会播放LuaMeihouWang2.ogg这个配音文件（当然这个文件实际上是不存在的，只是举个例子而已）。

现在我们把下面这句话加到“美王”的技能代码中去，放到询问发动技能之后：

room:broadcastSkillInvoke(“LuaMeihouWang”)



这样就可以在决定发动技能后第一时间播放LuaMeihouWang.ogg这个配音文件了。

**第一部第六章：添加提示信息**

添加代码注释和利用sgs.LogMessage结构体显示提示信息的方法没有变化。但是在目前版本的太阳神三国杀中，显示全屏信息特效的方法已经与第一部的时候完全不同了。

这次我们要用Room中的doLightbox()函数显示全屏信息特效，它的原型是：

doLightbox(lightboxName, duration, pixelSize)



有三个参数，分别是：

①lightboxName：表示要显示的信息。

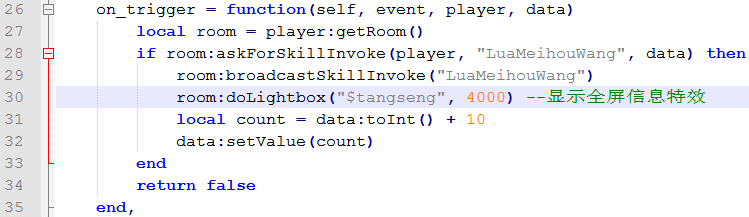
②duration：表示显示持续的时间，以毫秒为单位。默认值是2000。

③pixelSize：表示显示的范围大小，以像素为单位。默认值是0，表示充满全屏。

对比第一部中提到的写法"lightbox:$tangseng:4000"，lightboxName对应的就是“$tangseng”部分，duration对应的是“4000”部分，pixelSize则是新引入的参数。

在播放配音的代码之后加入这么一句：

room:doLightbox(“$tangseng”, 4000)



并按照第一部的写法写好对应的翻译信息，然后就可以在游戏中体验全屏信息特效了：



另外，与第一部时相比，此版本的太阳神三国杀还多出了一些新功能，比如下面这个——播放全屏技能特效。

播放全屏技能特效的方法，是利用Room的doSuperLightbox()函数。这个函数的原型是：

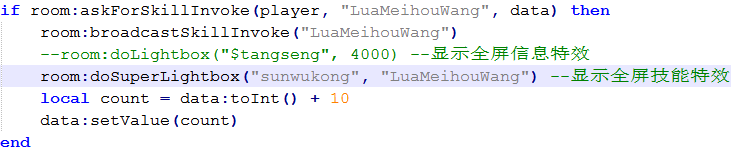
doSuperLightbox(heroName, skillName)



两个参数heroName和skillName分别表示要显示的武将的名称和技能的名称。

现在我们注释掉播放全屏信息特效的那行代码，然后加上一行播放全屏技能特效的代码：

room:doSuperLightbox(“sunwukong”, “LuaMeihouWang”)



这行代码可以为我们显示孙悟空的技能“美王”的全屏特效：



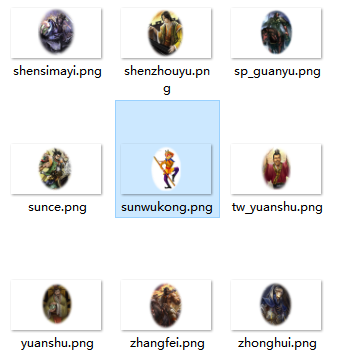
不过似乎少了张武将图片……没关系，按如下要求制作一张并放在指定位置就可以了。

像素尺寸：1000×550

图片格式：png格式

存放位置：游戏目录\image\animate\文件夹

文件名：与武将名称相同

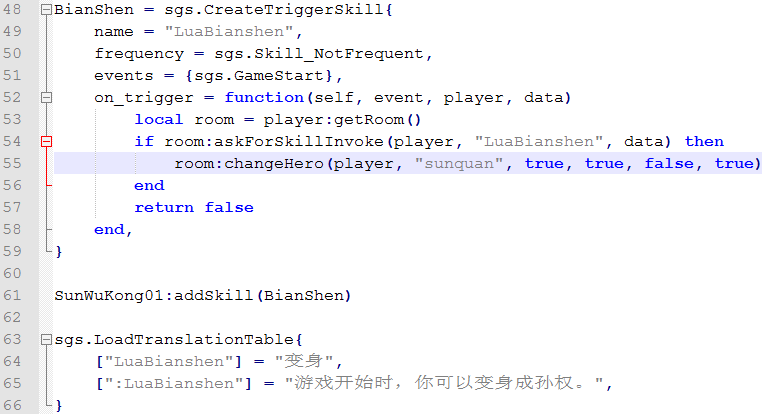


这样就可以在游戏中看到完整的全屏技能特效了。



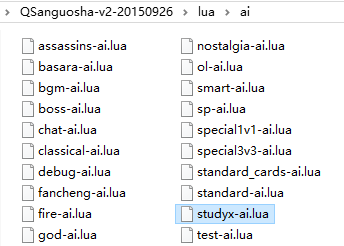
**第一部第七章：武将变更**

这一章的内容没有任何变化。因此技能“变身”还是要用Room的changeHero()函数实现。

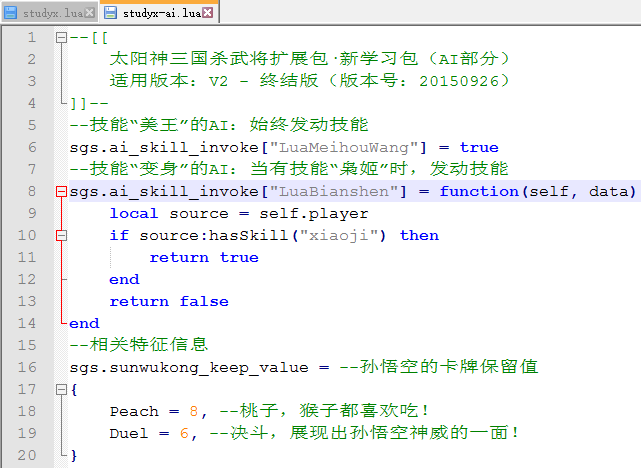


**第一部第八章：启用AI**

AI方面的变化也不大。首先还是要在游戏目录\lua\ai\文件夹中新建一个AI文件studyx-ai.lua。



然后使用Notepad++等编辑器打开此文件，修改编码格式、写入必要的AI代码即可。



注意：

1、第8行中使用的sgs.ai\_skill\_invoke[“LuaBianshen”]这种写法和第一部中使用的sgs.ai\_skill\_invoke.LuaBianshen的写法是等价的，不过对于Lua扩展包，编者更推荐使用方括号的这种形式。

2、同1，第6行也可以写成：sgs.ai\_skill\_invoke.LuaMeihouWang = true。

3、第9~13行可以简略地写成一句：return self.player:hasSkill(“xiaoji”)。这里分成了多行只是为了让读者朋友们看得更明白、对其中的逻辑关系理解得更准确。

4、目前版本的太阳神三国杀已经取消了“武将嘲讽值”的设定，所以表sgs.ai\_chaofeng已经被废弃，不建议再用。不过写了也没关系，只是不起作用了而已。

5、武将的卡牌保留值里需要写各种卡牌的类型名，而不是对象名。所以第18~19行写的是Peach和Duel，而第一部中的写法（peach和duel）则是错误的。有关卡牌类型名和对象名的区别，请见6~7。

6、卡牌的类型名，用于表示某一类型的卡牌，特点是首字母大写、中间没有连字符。

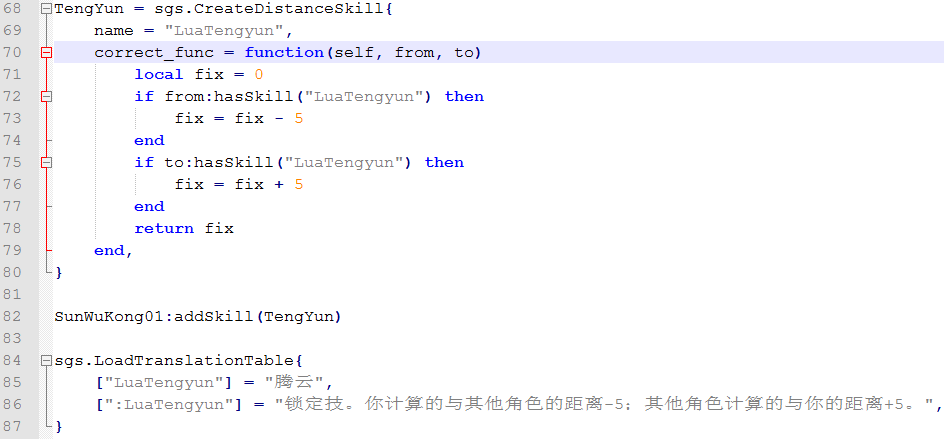
7、卡牌的对象名，用于表示某一特定卡牌的名称，特点是字母全小写，中间可能有连字符。

下表给出了各卡牌类型之间的从属关系以及某些类型名对应的对象名。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 类型名 | | | | | 翻译 | 对象名 |
| Card | | | | | 卡牌 |  |
|  | SkillCard | | | | 技能卡 |  |
|  | DummyCard | | | 形式卡 |  |
| LuaSkillCard | | | Lua技能卡 |  |
| BasicCard | | | | 基本牌 |  |
|  | Slash | | | 杀 | slash |
|  | NatureSlash | | 属性杀 |  |
|  | ThunderSlash | 雷杀 | thunder\_slash |
| FireSlash | 火杀 | fire\_slash |
| Jink | | | 闪 | jink |
| Peach | | | 桃 | peach |
| Analeptic | | | 酒 | analeptic |
| Shit | | | 屎 | shit |
| TrickCard | | | | 锦囊牌 |  |
|  | GlobalEffect | | | 全局效果 |  |
|  | GodSalvation | | 桃园结义 | god\_salvation |
| AmazingGrace | | 五谷丰登 | amazing\_grace |
| AOE | | | 多目标攻击性锦囊牌 |  |
|  | SavageAssault | | 南蛮入侵 | savage\_assault |
| ArcheryAttack | | 万箭齐发 | archery\_attack |
| SingleTargetTrick | | | 单目标锦囊牌 |  |
|  | Collateral | | 借刀杀人 | collateral |
| ExNihilo | | 无中生有 | ex\_nihilo |
| Duel | | 决斗 | duel |
| Nullification | | 无懈可击 | nullification |
| Snatch | | 顺手牵羊 | snatch |
| Dismantlement | | 过河拆桥 | dismantlement |
| FireAttack | | 火攻 | fire\_attack |
| IronChain | | | 铁索连环 | iron\_chain |
| DelayedTrick | | | 延时性锦囊牌 |  |
|  | Indulgence | | 乐不思蜀 | indulgence |
| SupplyShortage | | 兵粮寸断 | supply\_shortage |
|  |  |  | Disaster | | 天灾 |  |
|  | Lightning | 闪电 | lightning |
| Deluge | 洪水 | deluge |
| Typhoon | 台风 | typhoon |
| Earthquake | 地震 | earthquake |
| Volcano | 火山 | volcano |
| MudSlide | 泥石流 | mudslide |
| YanxiaoCard | | 言笑牌 | YanxiaoCard |
| EquipCard | | | | 装备牌 |  |
|  | Weapon | | | 武器 |  |
|  | Crossbow | | 诸葛连弩 | crossbow |
|  | VSCrossbow | 连弩 | vscrossbow |
| DoubleSword | | 雌雄双股剑 | double\_sword |
| QinggangSword | | 青釭剑 | qinggang\_sword |
| Blade | | 青龙偃月刀 | blade |
| Spear | | 丈八蛇矛 | spear |
| Axe | | 贯石斧 | axe |
| Halberd | | 方天画戟 | halberd |
| KylinBow | | 麒麟弓 | kylin\_bow |
| IceSword | | 寒冰剑 | ice\_sword |
| Fan | | 朱雀羽扇 | fan |
| GudingBlade | | 古锭刀 | guding\_blade |
| MoonSpear | | 银月枪 | moon\_spear |
| SPMoonSpear | | SP银月枪 | sp\_moonspear |
| YitianSword | | 倚天剑 | yitian\_sword |
| YxSword | | 杨修剑 | yx\_sword |
| Armor | | | 防具 |  |
|  | EightDiagram | | 八卦阵 | eight\_diagram |
| RenwangShield | | 仁王盾 | renwang\_shield |
| Vine | | 藤甲 | vine |
| SilverLion | | 白银狮子 | silver\_lion |
| GaleShell | | 狂风甲 | gale\_shell |
| FiveLines | | 五道杠 | five\_lines |
| Horse | | | 坐骑 |  |
|  | DefensiveHorse | | 防御马 | jueying（绝影） |
| dilu（的卢） |
| zhuahuangfeidian（爪黄飞电） |
| hualiu（骅骝） |
| OffensiveHorse | | 进攻马 | chitu（赤兔） |
| dayuan（大宛） |
| zixing（紫骍） |
|  | Monkey | 猴子 | monkey |
| Treasure | | | 宝物 |  |
|  |  |  | WoodenOx | | 木牛流马 | wooden\_ox |

**第一部第九章：修改距离**

创建距离修正技的方法没有变化，依然要用sgs.CreateDistanceSkill{}方法。不过第一部中，技能“腾云”的实现过程稍微有些漏洞，这里给出修正后的代码，如下图所示。



因为“腾云”是一个对两个计算方向都有影响的技能，当两名角色都有技能“腾云”时（比如同将模式），按照第一部中的写法就会漏掉技能的后半部分对结果的影响。因此要对两个部分的影响进行叠加。如果是像“马术”、“飞影”等只对一个计算方向有影响的距离修正技，就不用这样处理了，还按第一部中的写法加以实现即可。

另一个问题，锁定距离的方法，也还是使用Room中的setFixedDistance()函数。而相应地，解除距离锁定的方法，目前版本的太阳神三国杀用到的是Room中的另一个函数removeFixedDistance()，它的原型是：

removeFixedDistance(from, to, distance)



三个参数from、to和distance的含义和setFixedDistance()中是一样的。

除非是打算永久性地锁定距离，否则这两个函数通常来说应该是成对使用的。即，在发动技能时使用setFixedDistance()函数进行距离锁定，然后在一个规定的时机，通过某个触发技使用removeFixedDistance()函数将锁定解除。

**第一部第十章：技能管理**

在目前版本的太阳神三国杀中，技能的类型又有了一些新变化。除了第一部中介绍过的触发技、视为技（含锁定视为技）、禁止技、距离修改技、手牌上限技，和第二部中介绍过的目标增强技之外，还新增加了技能失效技、攻击范围技这两种基本技能类型。

技能失效技，是指能令特定角色的某个技能在一定时间范围内不起作用、失去原有效果的技能。界限突破后的标准版·关羽所拥有的技能“义绝”，还有标准版·马超所拥有的技能“铁骑”，能够令目标角色的非锁定技失效，就是属于此类型的效果，需要通过技能失效技实现。

创建一个技能失效技用到的方法是sgs.CreateInvaliditySkill，它的原型是：

sgs.CreateInvaliditySkill{name, skill\_valid}

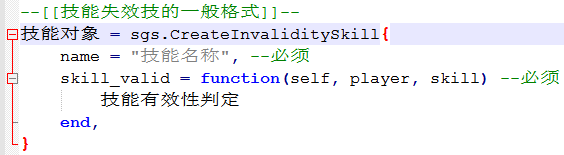
其中：

①name表示技能的名字；

②skill\_valid表示对技能是否有效的判定准则。对于某个特定的技能，只有当其符合这个准则时，才会生效；而如果技能不符合这个准则，它就是一个失效的技能了。

skill\_valid是一个函数，它的原型是：

function(self, player, skill)



攻击范围技，是指对一名角色的攻击范围进行修正的技能。其他·旱魃的技能“焚天②”（攻击范围＋武将牌上“焚”的数量）、倚天·倚天剑的技能“争锋”（无武器牌时，攻击范围＝体力值），都属于该类型的技能，需要使用攻击范围技实现。

创建一个攻击范围技用到的方法是sgs.CreateAttackRangeSkill，它的原型是：

sgs.CreateAttackRangeSkill{name, extra\_func, fixed\_func}

其中：

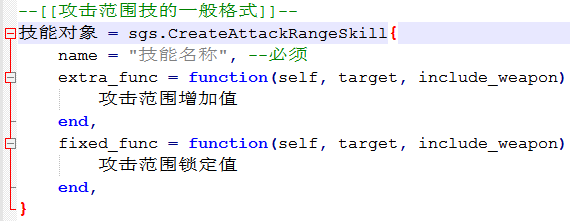
①name还是表示技能的名字；

②extra\_func表示额外获得的攻击范围；

③fixed\_func表示被锁定的攻击范围。

这里，extra\_func和fixed\_func都是函数，且函数原型是一样的，都是：

function(self, target, include\_weapon)



注意：

1、extra\_func和fixed\_func都不是必须的，但也不能都不写，至少要写其中的一个。

2、extra\_func函数中，如果不符合技能要求、不想修改攻击范围，最后请用一句return 0结束。增加值为0就表示不修改原有攻击范围了。

3、fixed\_func函数中，如果不符合技能要求、不想锁定攻击范围，最后请用一句return -1结束。锁定值为-1就表示不对攻击范围进行锁定了。

对于触发技的触发时机，目前版本的太阳神三国杀也有了一些调整。以下是目前的触发时机总结：

|  |  |  |  |
| --- | --- | --- | --- |
| 值 | 时机 | 解释 | 数据类型 |
| 0 | NonTrigger | 空时机 | —— |
| 1 | GameStart | 游戏开始时 | 无 |
| 2 | TurnStart | 回合开始前 | 无 |
| 3 | EventPhaseStart | 阶段开始时 | 无 |
| 4 | EventPhaseProceeding | 阶段进行时 | 无 |
| 5 | EventPhaseEnd | 阶段结束时 | 无 |
| 6 | EventPhaseChanging | 阶段交替时 | PhaseChangeStruct |
| 7 | EventPhaseSkipping | 跳过阶段时 | bool |
| 8 | DrawNCards | 摸牌阶段摸牌时 | int |
| 9 | AfterDrawNCards | 摸牌阶段摸牌后 | int |
| 10 | DrawInitialCards | 分发起始手牌时 | int |
| 11 | AfterDrawInitialCards | 分发起始手牌后 | int |
| 12 | PreHpRecover | 回复体力前 | RecoverStruct |
| 13 | HpRecover | 回复体力时 | RecoverStruct |
| 14 | PreHpLost | 失去体力前 | int |
| 15 | HpLost | 失去体力时 | int |
| 16 | HpChanged | 体力变化时 | RecoverStruct或int |
| 17 | MaxHpChanged | 体力上限变化时 | 无 |
| 18 | EventLoseSkill | 失去技能时 | QString |
| 19 | EventAcquireSkill | 获得技能时 | QString |
| 20 | StartJudge | 判定开始时 | JudgeStruct |
| 21 | AskForRetrial | 判定牌生效前 | JudgeStruct |
| 22 | FinishRetrial | 判定牌生效时 | JudgeStruct |
| 23 | FinishJudge | 判定结束时 | JudgeStruct |
| 24 | PindianVerifying | 亮出拼点牌时 | PindianStruct |
| 25 | Pindian | 拼点时 | PindianStruct |
| 26 | TurnedOver | 武将牌翻面时 | 无 |
| 27 | ChainStateChanged | 武将牌横置/重置时 | 无 |
| 28 | ConfirmDamage | 确定伤害时 | DamageStruct |
| 29 | Predamage | 造成伤害前 | DamageStruct |
| 30 | DamageForseen | 伤害结算开始时 | DamageStruct |
| 31 | DamageCaused | 造成伤害时 | DamageStruct |
| 32 | DamageInflicted | 受到伤害时 | DamageStruct |
| 33 | PreDamageDone | 扣减体力前 | DamageStruct |
| 34 | DamageDone | 扣减体力时 | DamageStruct |
| 35 | Damage | 造成伤害后 | DamageStruct |
| 36 | Damaged | 受到伤害后 | DamageStruct |
| 37 | DamageComplete | 伤害结算完成时 | DamageStruct |
| 38 | EnterDying | 进入濒死状态时 | DyingStruct |
| 39 | Dying | 处于濒死状态时 | DyingStruct |
| 40 | QuitDying | 退出濒死状态时 | DyingStruct |
| 41 | AskForPeaches | 濒死状态求桃时 | DyingStruct |
| 42 | AskForPeachesDone | 濒死状态求桃后 | DyingStruct |
| 43 | Death | 死亡时 | DeathStruct |
| 44 | BuryVictim | 处理死亡角色时 | DeathStruct |
| 45 | BeforeGameOverJudge | 游戏胜负判定前 | DeathStruct |
| 46 | GameOverJudge | 游戏胜负判定时 | DeathStruct |
| 47 | GameFinished | 游戏结束时 | —— |
| 48 | SlashEffected | 杀生效后 | SlashEffectStruct |
| 49 | SlashProceed | 杀结算时 | SlashEffectStruct |
| 50 | SlashHit | 杀命中时 | SlashEffectStruct |
| 51 | SlashMissed | 杀被闪避时 | SlashEffectStruct |
| 52 | JinkEffect | 闪判定生效前 | Card |
| 53 | NullificationEffect | 无懈可击判定生效前 | Card |
| 54 | CardAsked | 被要求使用卡牌时 | QStringList |
| 55 | PreCardResponded | 打出卡牌前 | CardResponseStruct |
| 56 | CardResponded | 打出卡牌时 | CardResponseStruct |
| 57 | BeforeCardsMove | 卡牌移动前 | CardsMoveOneTimeStruct |
| 58 | CardsMoveOneTime | 卡牌移动时 | CardsMoveOneTimeStruct |
| 59 | PreCardUsed | 使用卡牌前 | CardUseStruct |
| 60 | CardUsed | 使用卡牌时 | CardUseStruct |
| 61 | TargetSpecifying | 卡牌指定目标时 | CardUseStruct |
| 62 | TargetConfirming | 成为卡牌目标时 | CardUseStruct |
| 63 | TargetSpecified | 卡牌指定目标后 | CardUseStruct |
| 64 | TargetConfirmed | 成为卡牌目标后 | CardUseStruct |
| 65 | CardEffect | 卡牌生效前 | CardEffectStruct |
| 66 | CardEffected | 卡牌生效后 | CardEffectStruct |
| 67 | PostCardEffected | 卡牌结算完成进入弃牌堆时 | CardEffectStruct |
| 68 | CardFinished | 卡牌结算完成后 | CardUseStruct |
| 69 | TrickCardCanceling | 无懈可击使用资格判定时 | CardEffectStruct |
| 70 | TrickEffect | 锦囊牌生效时 | CardEffectStruct |
| 71 | ChoiceMade | 作出选择时 | QString |
| 72 | StageChange | 游戏阶段交替时 | —— |
| 73 | FetchDrawPileCard | 摸牌堆提供卡牌时 | 无 |
| 74 | ActionedReset | 行动状态重置时 | 无 |
| 75 | Debut | 登场时 | 无 |
| 76 | TurnBroken | 回合中断时 | —— |
| 77 | NumOfEvents | 时机总数 | —— |

注意：

1、NonTrigger和NumOfEvents不是真正意义上的时机。

2、PreCardUsed和CardEffect用于AI记录某些必要的游戏信息，一般不建议使用。

3、GameFinished、StageChange和TurnBroken是用于抛出的时机，请不要用于触发技能。

4、StageChange是虎牢关模式专用时机。

5、FetchDrawPileCard通常用于小型场景模式，一般不建议使用。

6、ActionedReset是3v3对战模式专用时机，一般不建议使用。

7、Debut是1v1系列模式专用时机，一般不建议使用。

8、上表中的数据类型为源代码中使用的C++数据类型。对应的Lua数据类型为：

①bool(C++)→boolean(Lua)

②int(C++)→number(Lua)

③QString(C++)→string(Lua)

④QStringList(C++)→table<string>(Lua)

另外，对于获得与失去技能，第一部中提到的Room中的acquireSkill()函数、attachSkillToPlayer()函数以及detachSkillFromPlayer()函数依然可用。不过在目前版本的太阳神三国杀中，更建议使用新推出的handleAcquireDetachSkills()函数进行技能管理。它的函数原型是：

handleAcquireDetachSkills(player, skill\_names, acquire\_only)



各参数的含义如下：

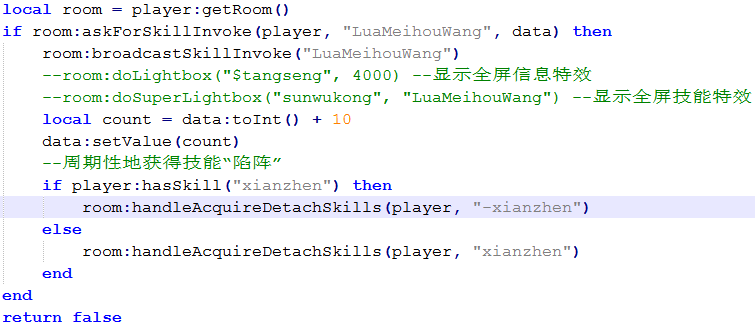
①player：表示进行技能管理的角色。

②skill\_names：表示所有涉及的技能及其相应的处理方式，多个技能项之间使用竖线“|”进行分割。每一项中，如果只写技能名称，则目标角色将获得此技能；如果在技能名称前添加一个减号“-”，则改为令目标角色失去此技能。

例如，写成”-zhiba|yingzi|yinghun”就表示该角色将失去技能“制霸”并获得技能“英姿”和“英魂”。

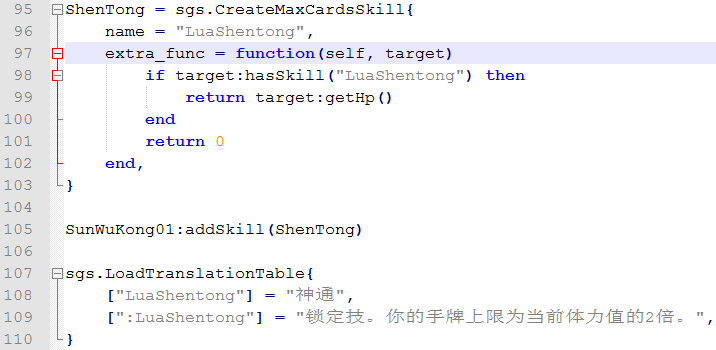
③acquire\_only：表示在执行失去技能时，是否只令角色失去额外获得的技能。如果为true（是），表示保留武将原有技能，即只有当某技能是游戏中获得的技能时，才失去此技能；如果为false（否），表示将无条件地失去技能，不论该技能是否是游戏中才获得的技能。不写的话，默认为false。

使用Room的handleAcquireDetachSkills()函数重新实现第一部中，在发动技能“美王”的同时周期性地获得技能“陷阵”的效果。代码如下图所示。



**第一部第十一章：修改手牌上限**

修改手牌上限用到的方法还是第一部时提到的sgs.CreateMaxCardsSkill，所以技能“神通”的实现也没有多少变化。

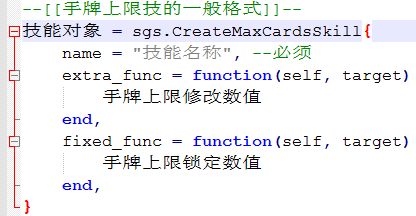


不过事实上，目前版本的sgs.CreateMaxCardsSkill多了一个新功能：锁定手牌上限。也就是说，如果没有写extra\_func函数，而是写了一个叫做fixed\_func的函数时，将会把目标角色的手牌上限锁定为fixed\_func函数的返回值（如果返回值是-1，表示不进行锁定）。

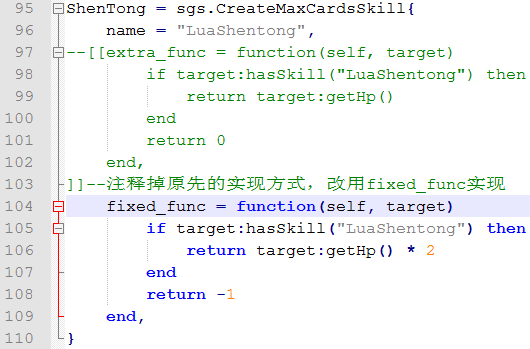
这个fixed\_func函数的原型和extra\_func函数的原型一样，都是：

function(self, target)

而且两个参数self和target的含义和extra\_func中的同名参数也是一样的。



所以根据技能描述，“神通”的最正确的实现应该是使用fixed\_func对手牌上限进行锁定。第一部时是因条件所限，才使用extra\_func进行模拟实现的。

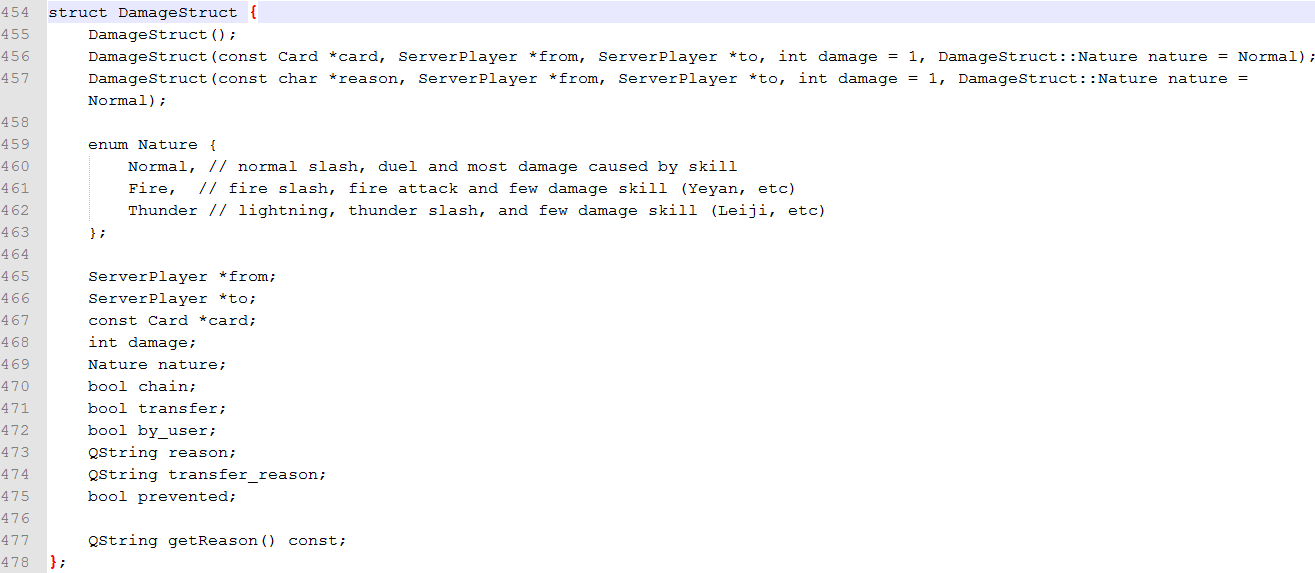


以上就是使用fixed\_func实现技能“神通”的具体过程。

**第一部第十二章：体力变更**

这一章涉及了失去体力、制造伤害、回复体力、扣减体力上限、增加体力上限等多种变化体力或体力上限的方式，内容虽然多，但却没有什么太大的变化，本章节中的两个技能“自满”和“灭妖”也不需要进行修改即可在目前版本的太阳神三国杀中良好地发挥作用。

仅有的几个变化，一个，在于伤害结构体（sgs.DamageStruct）。



与第一部时相比，当前版本中的伤害结构体，增加了两个构造函数：

sgs.DamageStruct(card, from, to, damage, nature)

sgs.DamageStruct(reason, from, to, damage, nature)

以及四个成员变量：by\_user、reason、transfer\_reason、prevented

还有一个成员函数：getReason()

同时，还去掉了一个成员变量：trigger\_chain

两个新的构造函数可以让我们更方便地定义和使用一个伤害结构体。其中的参数和伤害结构体中的同名成员变量的含义是相同的。在这里，构造函数的后两个参数damage和nature可以不写，默认值分别为1（造成1点伤害）和sgs.DamageStruct\_Normal（造成无属性伤害）。

新增加的四个成员变量的含义为：

①by\_user：表示当存在造成伤害的卡牌时，伤害来源是否为卡牌的使用者。

例如，当【决斗】的使用者因决斗失败而受到对方造成的伤害时，此伤害的by\_user值即为false，表示伤害来源（对方）与卡牌使用者（自己）不相同。

而通常情况下，by\_user的值都为true。

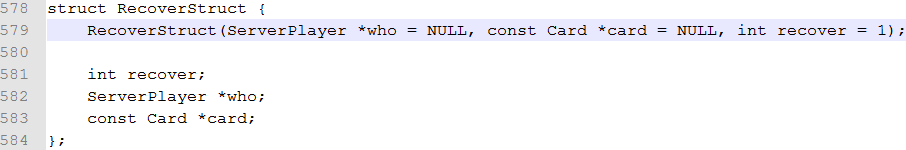
②reason：表示造成伤害的原因。

③transfer\_reason：表示转移伤害的原因。

④prevented：表示该伤害是否被防止。

当伤害在sgs.DamageForseen、sgs.DamageCaused、sgs.DamageInflicted三个时机因return true被防止后，prevented的值会被设置为true，否则其始终为false。

另一个，在于回复结构体（sgs.RecoverStruct）。



与伤害结构体类似，回复结构体也对构造函数进行了修改。新的构造函数为：

sgs.RecoverStruct(who, card, recover)

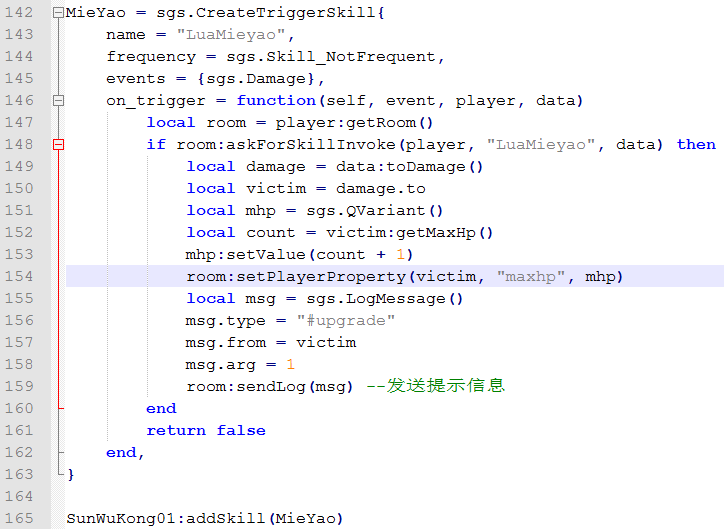
三个参数与回复结构体内的同名成员变量的含义相同，且都可以省略不写。

下面使用新的伤害结构体和回复结构体实现第一部中提到的技能“自满”。不过首先根据设计本意，修改一下技能描述：“自满：锁定技。回合开始前，你须对自己造成的一点伤害，然后回复一点体力。”



可以看出，使用了新的构造函数之后，代码变得更简洁了。

另一个技能“灭妖”的实现过程如下：



几乎没有任何变化。



**第一部第十三章：游戏进程的变更**

跟阶段有关的触发时机在目前版本中已经扩充到了八个，分别是：

① sgs.TurnStart（回合开始前）

② sgs.EventPhaseStart（阶段开始时）

③ sgs.EventPhaseProceeding（阶段进行时）☆

④ sgs.EventPhaseEnd（阶段结束时）

⑤ sgs.EventPhaseChanging（阶段交替时）

⑥ sgs.EventPhaseSkipping（跳过阶段时）☆

⑦ sgs.DrawNCards（摸牌阶段摸牌时）

⑧ sgs.AfterDrawNCards（摸牌阶段摸牌后）☆

另外还有以下七个与游戏进程相关的时机：

① sgs.GameStart（游戏开始时）

② sgs.DrawInitialCards（分发起始手牌时）☆

③ sgs.AfterDrawInitialCards（分发起始手牌后）☆

④ sgs.Debut（登场时）☆△

⑤ sgs.StageChange（游戏阶段交替时）☆△×

⑥ sgs.TurnBroken（回合中断时）☆×

⑦ sgs.GameFinished（游戏结束时）×

其中标注“☆”的为相对于第一部时新增加的时机，“△”为特定模式下的专用时机，“×”表示该时机用于抛出。

但与回合有关的阶段没有变化，用Player的getPhase() 函数还是会得到以下九个结果之中的一个：

① sgs.Player\_RoundStart（启动阶段）

② sgs.Player\_Start（准备阶段）

③ sgs.Player\_Judge（判定阶段）

④ sgs.Player\_Draw（摸牌阶段）

⑤ sgs.Player\_Play（出牌阶段）

⑥ sgs.Player\_Discard（弃牌阶段）

⑦ sgs.Player\_Finish（回合结束阶段）

⑧ sgs.Player\_NotAcitve（回合外）

⑨ sgs.Player\_PhaseNone（阶段间）

这里②~⑦就是我们常说的一个回合内包含的六大阶段了。①是太阳神三国杀为了处理某些数据或实现某些游戏效果而额外引入的一个阶段；⑧表示当前不为目标角色的回合，是角色的回合外；⑨表示目标角色正在进行阶段交替，既不属于前一个阶段，也不属于后一个阶段，而是处于两个阶段之间。

一个完整的回合所经历的阶段及其对应的时机如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 流程 | 阶段 | 时机 | 解释 |
| 1 | NotActive | TurnStart | 回合开始前 |
| 2 | PhaseNone | EventPhaseChanging | 启动阶段开始前 |
| 3 | RoundStart | EventPhaseStart | 启动阶段开始时 |
| 4 | EventPhaseProceeding | 启动阶段进行时 |
| 5 | EventPhaseEnd | 启动阶段结束时 |
| 6 | PhaseNone | EventPhaseChanging | 启动阶段结束后/准备阶段开始前 |
| 7 | Start | EventPhaseStart | 准备阶段开始时 |
| 8 | EventPhaseProceeding | 准备阶段进行时 |
| 9 | EventPhaseEnd | 准备阶段结束时 |
| 10 | PhaseNone | EventPhaseChanging | 准备阶段结束后/判定阶段开始前 |
| 11 | Judge | EventPhaseStart | 判定阶段开始时 |
| 12 | EventPhaseProceeding | 判定阶段进行时 |
| 13 | EventPhaseEnd | 判定阶段结束时 |
| 14 | PhaseNone | EventPhaseChanging | 判定阶段结束后/摸牌阶段开始前 |
| 15 | Draw | EventPhaseStart | 摸牌阶段开始时 |
| 16 | EventPhaseProceeding | 摸牌阶段进行时（此时执行摸牌） |
| 17 | EventPhaseEnd | 摸牌阶段结束时 |
| 18 | PhaseNone | EventPhaseChanging | 摸牌阶段结束后/出牌阶段开始前 |
| 19 | Play | EventPhaseStart | 出牌阶段开始时 |
| 20 | EventPhaseProceeding | 出牌阶段进行时（此时执行出牌） |
| 21 | EventPhaseEnd | 出牌阶段结束时 |
| 22 | PhaseNone | EventPhaseChanging | 出牌阶段结束后/弃牌阶段开始前 |
| 23 | Discard | EventPhaseStart | 弃牌阶段开始时 |
| 24 | EventPhaseProceeding | 弃牌阶段进行时 |
| 25 | EventPhaseEnd | 弃牌阶段结束时 |
| 26 | PhaseNone | EventPhaseChanging | 弃牌阶段结束后/回合结束前 |
| 27 | Finish | EventPhaseStart | 回合结束阶段开始时 |
| 28 | EventPhaseProceeding | 回合结束阶段进行时 |
| 29 | EventPhaseEnd | 回合结束阶段结束时 |
| 30 | PhaseNone | EventPhaseChanging | 回合结束阶段结束后 |
| 31 | NotActive | EventPhaseStart | 回合结束后 |

对于跳过阶段这件事，我们还是使用ServerPlayer的skip() 函数。但是这个函数有了些新的变化，原有的两个原型被合并成了一个原型：

skil(phase, isCost)



同时增加了一个新的参数：isCost。

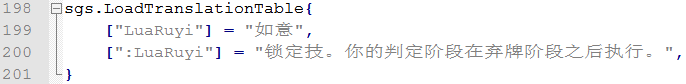
isCost参数表示本次跳过阶段是否为其他技能效果的消耗，有两个取值true（作为其他技能效果的消耗而跳过阶段，如技能“神速”）和false（只是一种纯粹的游戏效果而跳过阶段，如技能“克己”）。默认为false。

另外，检查阶段是否被跳过的isSkipped() 函数以及插入一个额外阶段的insertPhase() 函数都是没有变化的，可以继续放心使用。

技能“如意”中的跳过判定阶段是作为纯粹的游戏效果出现的，isCost可以不写而取默认值false，因此不需要修改。



对应的技能描述：



将武将翻面所用的Player的turnOver() 函数没有任何变化。但是获得额外回合的Player的gainAnExtraTurn() 函数，在目前版本去掉了参数，原型变为：

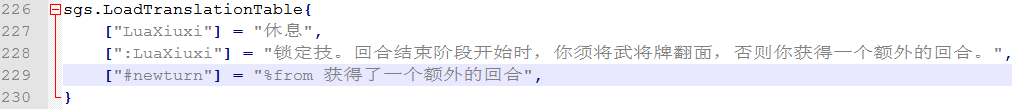
gainAnExtraTurn()



但这其实对我们的影响并不大，毕竟那个参数当初也是不需要写的。所以技能“休息”的写法与第一部时一样。

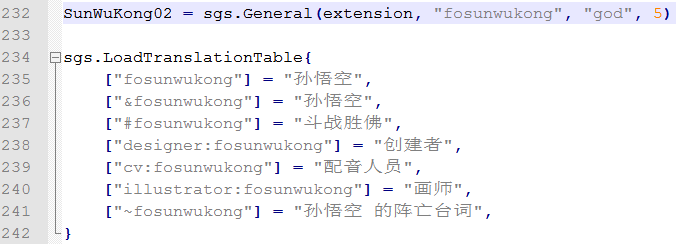


注意这里自定义了一条提示信息”#newturn”，因为gainAnExtraTurn() 在执行过程中并不会主动向游戏日志发信息提示在场玩家，需要我们自己实现这个效果。



**第一部第十四章：启用标记**

依照第一部的思路，先创建一个新的武将。

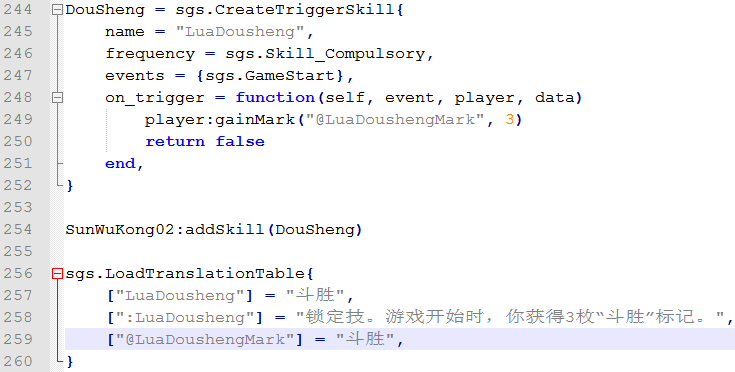


目前版本中用到的标记如下表所示。

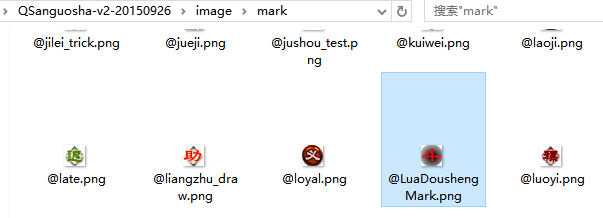
|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 标记 | 标记名称 | 来自技能/效果/功能 |
| 1 | @arise |  | 雄异 |
| 2 | @beam | 平衡木 | 武将测试<历史功能> |
| 3 | @bear | 忍 | 忍戒 |
| 4 | @benxi |  | 奔袭 |
| 5 | @bossExp |  | 闯关模式 |
| 6 | @burn |  | 焚城 |
| 7 | @burnheart |  | 焚心 |
| 8 | @chanyuan |  | 缠怨 |
| 9 | @chaos |  | 乱武 |
| 10 | @chou | 仇 | 誓仇<历史版本> |
| 11 | @chuanqi |  | 传奇模式<历史模式> |
| 12 | @chuanxin |  | 穿心 |
| 13 | @cihuai |  | 刺槐 |
| 14 | @collapse | 崩坏 | 毒士 |
| 15 | @conspiracy |  | 共谋 |
| 16 | @defense | 守 | 镇卫 |
| 17 | @defensive\_distance\_test |  | 距离计算<测试版本> |
| 18 | @duanchang |  | 断肠 |
| 19 | @earth | 五灵·土 | 五灵 |
| 20 | @fenwei |  | 奋威 |
| 21 | @fenyong |  | 愤勇 |
| 22 | @fight |  | 战神 |
| 23 | @fire | 五灵·火 | 五灵 |
| 24 | @flame |  | 业炎 |
| 25 | @fog |  | 大雾 |
| 26 | @frantic |  |  |
| 27 | @gale |  | 狂风 |
| 28 | @handover |  | 献州 |
| 29 | @hate | 誓 | 誓仇 |
| 30 | @hate\_to | 仇 | 誓仇 |
| 31 | @hengjiang |  | 横江 |
| 32 | @huashen |  | 化身 |
| 33 | @jilei\_basic | 鸡肋·基本牌 | 鸡肋 |
| 34 | @jilei\_equip | 鸡肋·装备牌 | 鸡肋 |
| 35 | @jilei\_trick | 鸡肋·锦囊牌 | 鸡肋 |
| 36 | @jueji |  | 绝汲<历史版本> |
| 37 | @jushou\_test |  | 据守<测试版本> |
| 38 | @kuiwei |  | 溃围 |
| 39 | @laoji | 老骥 | 伏枥 |
| 40 | @late |  | 智迟 |
| 41 | @liangzhu\_draw |  | 良助 |
| 42 | @loyal |  | 忠义 |
| 43 | @luoyi |  | 裸衣 |
| 44 | @max\_cards\_test |  | 手牌上限计算<测试版本> |
| 45 | @nightmare | 梦魇 | 武魂 |
| 46 | @nirvana |  | 涅槃 |
| 47 | @nosburn |  | 焚城<怀旧版本> |
| 48 | @offensive\_distance\_test |  | 距离计算<测试版本> |
| 49 | @qianxi\_black | 潜袭·黑色 | 潜袭 |
| 50 | @qianxi\_red | 潜袭·红色 | 潜袭 |
| 51 | @rescue |  | 解烦 |
| 52 | @round | 退治 | 僵尸模式 |
| 53 | @shouye |  | 授业 |
| 54 | @skill\_invalidity |  | 技能失效技效果 |
| 55 | @sleep |  | 醉乡 |
| 56 | @songci |  | 颂词 |
| 57 | @struggle |  | 死战 |
| 58 | @substitute |  | 替身 |
| 59 | @substitute\_hp |  | 替身 |
| 60 | @suiren |  | 随仁 |
| 61 | @thunder | 五灵·雷 | 五灵 |
| 62 | @tied |  | 连理 |
| 63 | @twine |  | 挟缠 |
| 64 | @waked | 觉醒 | 觉醒技效果 |
| 65 | @water | 五灵·水 | 五灵 |
| 66 | @wen | 文 | 谋断 |
| 67 | @wind | 五灵·风 | 五灵 |
| 68 | @wrath | 暴怒 | 狂暴 |
| 69 | @wu | 武 | 谋断 |
| 70 | @xiemu\_qun | 协穆·群 | 协穆 |
| 71 | @xiemu\_shu | 协穆·蜀 | 协穆 |
| 72 | @xiemu\_wei | 协穆·魏 | 协穆 |
| 73 | @xiemu\_wu | 协穆·吴 | 协穆 |
| 74 | @yongsi\_test |  | 庸肆<测试版本> |
| 75 | @zhenggong |  | 争功 |

ServerPlayer中有关标记的几个函数，包括用于获得标记的gainMark()、用于失去标记的loseMark()，还有用于失去所有标记的loseAllMarks()，都没有什么变化，可以放心使用。

但是技能“斗胜”的代码还是需要稍微修改一下，因为” @fight”这个标记名字已经被太阳神三国杀用掉了，所以这里把“斗胜”标记的名字改成了” @LuaDoushengMark”。



当然，标记文件也应对应地重命名为：@LuaDoushengMark.png。



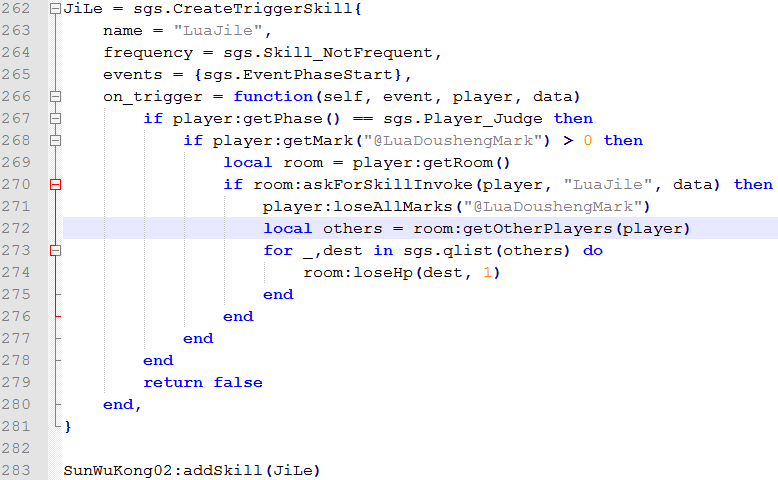
获得所有其他角色用到的是Room中的getOtherPlayers() 函数。这个函数在目前的版本中添加了一个新的参数include\_dead，所以现在它的原型变成了：

getOtherPlayers(except, include\_dead)

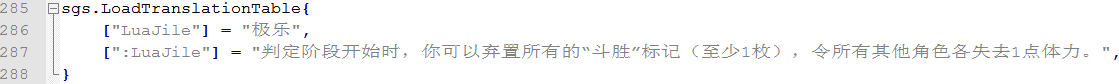


新参数include\_dead表示在统计所有角色时是否包括已死亡的角色，有两个取值：true（包括死亡角色）和false（不包括死亡角色）。不写的话，默认为false，也就是只考虑所有存活的角色，因而这样将得到所有其他存活角色。

获得指定标记数目的函数，也就是Player的getMark() 函数，在目前的版本中没有什么新变化。所以技能“极乐”的实现也没有太大的变化。



对应的技能描述：



**第一部第十五章：卡牌转换**

这一章主要围绕着创建视为技时所用到的方法——sgs.CreateViewAsSkill{}函数展开。这个函数随着版本的变迁，也有了新的变化，增加了许多功能以适应后来新涌现的技能特征。在目前版本的太阳神三国杀中，这个函数的原型如下：

sgs.CreateViewAsSkill{

name,

response\_pattern,

response\_or\_use,

expand\_pile,

n,

view\_filter,

view\_as,

guhuo\_type,

should\_be\_visible,

enabled\_at\_play,

enabled\_at\_response,

enabled\_at\_nullification,

}

与第一部时相比，新增加了五项内容。这些新增项的含义如下：

① response\_pattern：

表示该视为技可以响应的卡牌样式，是一个字符串，默认为空。这一项可以认为是enabled\_at\_response函数的简化版本，即，如果所要求的卡牌样式与此处的设定相同，就可以发动该视为技进行响应。

② response\_or\_use：

表示是否可以使用类似“木牛流马”等牌堆中视为手牌的牌发动该视为技。因为卡牌【木牛流马】的效果中含有“如手牌般使用或打出”的字样，这一项便因此而得名。

这是一个布尔值，可以取true（表示可以使用视为的手牌）或false（表示不能使用）。而默认值则是false，即通常情况下发动技能是不能使用牌堆中的牌的。在源代码中，只有像“武圣”、“国色”、“乱击”这些视为单张真正卡牌的情形会将此项的值设为true。

③ expand\_pile：

表示与该技能相关的私家牌堆的名称，是一个字符串，默认为空。如果设置了这一项，那么在选择卡牌发动技能的时候，太阳神三国杀会有一个把该牌堆中的牌移入手牌区以供玩家选择的视觉效果。

④ guhuo\_type：

表示该视为技发动时所用到的“蛊惑对话框”的结构类型。是一个由”!lrsd”这五个字符中的某几个字符所组成的字符串，默认为空。各个字符的含义如下：

（a）字符’!’：这是一个英文状态下的感叹号，表示允许在出牌阶段出牌时以外的情形发动该视为技视为目标卡牌。如果没有字符”!”，则这个视为技只能在出牌阶段主动出牌时发动，而不能用于响应等其他场合。请注意，这个字符必须放在guhuo\_type字符串最左侧的起始位置，如果位于字符串的中间或末尾，则没有任何效果。

（b）字符’l’：表示”left”，也就是位于左侧的基本牌部分。含有字符’l’表示该视为技可以用于视为基本牌。

（c）字符’r’：表示”right”，也就是位于右侧的锦囊牌部分。含有字符”r”表示该视为技可以用于视为锦囊牌（包括延时性锦囊牌和非延时性锦囊牌两类）。

（d）字符”s”：表示”slash”，也就是把各种类型的“杀”合并成一个选项。如果不含有这个字符，那么在基本牌部分将出现“杀”、“雷杀”、“火杀”、“普通杀”等多个与杀有关的选项。

（e）字符”d”：表示”delayed”，也就是锦囊牌中的延时性锦囊牌部分。含有字符”d”可以把右侧的锦囊牌部分分为两栏，分别为非延时性锦囊牌和延时性锦囊牌。如果没有”d”，则只会产生非延时性锦囊牌这一栏。

⑤ should\_be\_visible：

表示是否需要在游戏界面中显示该技能对应的按钮。这是一个函数，函数原型是：function(self, player)。如果该函数给出一个true的结果，则会在界面中显示此视为技对应的技能按钮；如果结果为false，则不显示按钮。另外，默认的返回结果通常可以视为true。

在源代码中，技能“黄天送牌”和“制霸拼点”中用到了这个函数。作为响应对应主公技的视为技，这两个技能对角色的势力有严格的要求。所以虽然游戏中的所有角色都会被添加这些技能，但只有特定势力的角色的界面上会出现技能按钮、允许发动技能，而其他势力的角色则如同该技能不存在一般，界面上没有任何显示。



而由于新增加了response\_pattern这一项，原有的enabled\_at\_play和enabled\_at\_response这两个函数的默认行为也发生了变化。

① enabled\_at\_play：原来默认情况下将直接产生一个true的结果，而现在要首先判断response\_pattern是否存在，只有当response\_pattern为空的时候才默认产生true的结果。

② enabled\_at\_response：原来默认情况下将直接产生一个false的结果，但现在如果response\_pattern存在，那么默认的结果为”pattern的值与response\_pattern相等”。

可以看出就是多了一步对response\_pattern的判断；而不写response\_pattern的话，这两个函数的默认结果和第一部时其实是相同的。

这一章的另一个变化在于Engine的成员函数cloneCard。在第一部时cloneCard函数有三个原型，而在目前版本中将其中第二、第三这两个原型合并成了一个，并为一些参数添加了默认值。于是，现在这个函数的原型变成了：

① cloneCard(card)

② cloneCard(name, suit, number, flags)



其中，第一个原型没有变化，还是根据一张给定的卡牌创造一张一样的卡牌。而第二个原型作为原先两个原型的综合，可以根据卡牌的名称、花色、点数、标志信息创造指定的卡牌。

注意这第二个原型中，除了name参数外，另外三个参数都有了默认值，这就允许我们有时只需要给出卡牌的名称即可迅速创造出需要的卡牌，因而用起来就显得更加的方便。

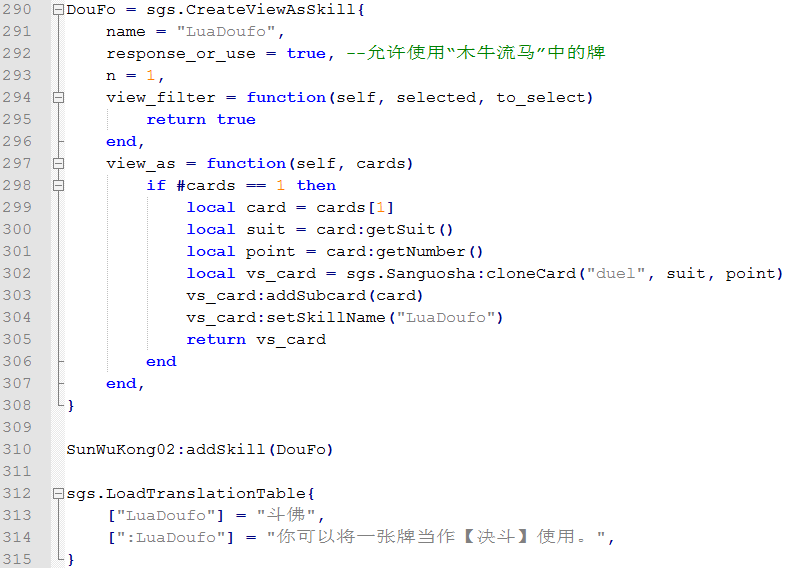
对后三个参数及其默认值的解释如下：

（a）suit：表示卡牌的花色，默认值为sgs.Card\_SuitToBeDecided。使用默认值时，太阳神三国杀会在稍后合适的时机，根据该卡牌的子卡信息确定卡牌的花色，而不再需要我们自己考虑卡牌的花色问题了。

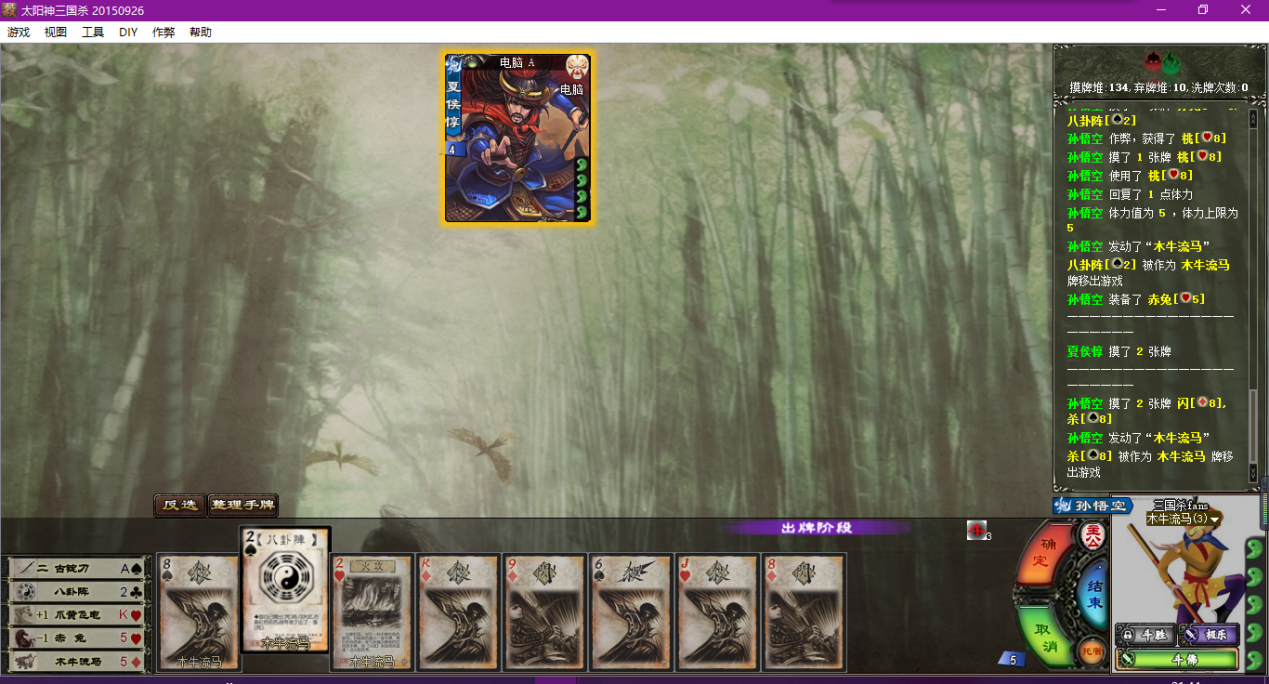
（b）number：表示卡牌的点数，默认值为-1。

（c）flags：表示卡牌所携带的标志，默认值是一个空表（{}），也就是没有任何标志。

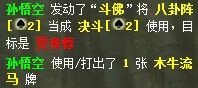
为了能让“斗胜”技能使用“木牛流马”中的牌，我们现在使用新版本的sgs.CreateViewAsSkill函数重写这个技能。不过其实只是多加了response\_or\_use这一项而已……



游戏效果：



游戏日志：



另一个技能“浮云”则可以利用response\_pattern简化成如下的模样。



**第一部第十六章：卡牌传递**

技能“自修”的代码没有什么变化，沿用第一部时的思路即可。

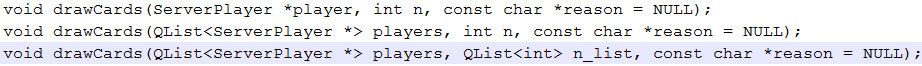


另一方面，Room的成员函数drawCards有了一点小变化，增加了一个新的原型，于是现在原型变成了三个：

drawCards(player, n, reason)

drawCards(players, n, reason)

drawCards(players, n\_list, reason)



新增加的这第三个原型是在第二个原型上的进一步扩张。第二个原型可以让多个角色一起摸同样数目的牌，而这个新原型可以让多个角色依次摸指定数目的牌。也就是说，两个sgs.QList类型的参数n\_list和players中的元素是一一对应的。

不过说实话这个原型应该用得不多，也很容易被前两个原型所代替，因此我们就不过多地研究它了。

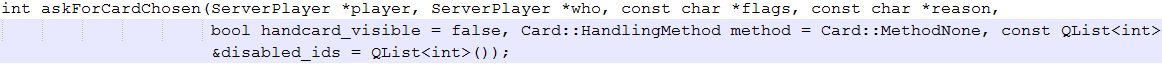
弃牌方面，Room的成员函数throwCard还是老样子，没有什么变化；但是另一个函数askForCardChosen一口气添加了三个新参数，改动较大。

现在askForCardChosen函数的原型为：

askForCardChosen(

player, who, flags, reason, handcard\_visible, method, disabled\_ids

)



新增加的三个参数含义是：

① handcard\_visible：表示选牌时，目标角色手牌区的牌是否对选牌角色可见。它有两个取值true（表示可见）和false（表示不可见）。默认为false。

② method：表示选牌之后，对选中卡牌的处理方式。默认为sgs.Card\_MethodNone，也就是不作任何处理。

③ disabled\_ids：表示在选牌时不能被选中的卡牌的编号列表。默认为sgs.IntList()（空列表），也就是所有卡牌都可以选择。

注意：

1、参数method表示对选中卡牌预计的后续处理方式，类型是sgs.Card\_HandlingMethod。因为从新标准版·黄月英的“奇才”技能中引入了“卡牌不能被弃置”的新元素，所以便有了这样的参数，使得能够对那些可能的弃牌操作进行控制。在这里，如果method的值被设定为sgs.Card\_MethodDiscard（弃置），而某些备选卡牌不能被弃置，就会导致这些卡牌按钮变灰、不能被选中。

2、作为卡牌的处理方式，method可取下列值：

① sgs.Card\_MethodNone（无）

② sgs.Card\_MethodUse（使用）

③ sgs.Card\_MethodResponse（响应）

④ sgs.Card\_MethodDiscard（弃置）

⑤ sgs.Card\_MethodRecast（重铸）

⑥ sgs.Card\_MethodPindian（拼点）

不过根据1中的分析可知，在askForCardChosen()中只有①和④是最常用的。

还有一处变化，是在判断一名角色判定区是否有牌时，目前版本的太阳神三国杀又为Player新增加了一个成员函数getJudgingAreaID，用于获取判定区所有卡牌的编号列表。它的函数原型为：

getJudgingAreaID()



和getJudgingArea()一样，这个函数也没有任何参数。而且由于获取的是卡牌的编号而不是卡牌本身，所以在那些应用卡牌编号的场合可能会更方便一些。

不过，这些变化并没有影响到技能“引渡”，它的实现思路还是和第一部时是一样的。



接下来，关于移动场上的卡牌，moveCardTo函数没有变化。倒是askForPlayerChosen函数增加了几个参数，现在它的原型为：

askForPlayerChosen(player, targets, reason, prompt, optional, notify\_skill)



新增加的三个参数为：

① prompt：表示询问角色选择时显示的提示语。默认为nil，此时将在提示框中显示“请选择一名角色”。

② optional：表示是否可以放弃选择，有两个取值true（表示可以不选择角色）和false（表示必须选择一名角色）。默认为false，也就是必须作出选择。

③ notify\_skill：表示选择角色后是否显示技能发动，也是两个取值true（表示显示发动了技能）和false（不显示）。默认为false。如果填true的话，将在武将头像上显示reason参数的翻译结果（通常为对应技能的名字，只能在其他玩家视角下看到）。

这三个参数的加入使得我们有些时候能够获得更好的游戏体验。在第一部中，技能“挖角”的实现受当时条件限制，需要先选择是否发动技能、然后选择目标角色，操作起来较为不便，而且确认发动技能后没有后悔的机会。现在我们将使用一种新的思路来重写这个技能。

首先，判断是否满足技能发动的条件，即，当前阶段是否为弃牌阶段、以及自身是否装备了防具，这和第一部中是一样的。

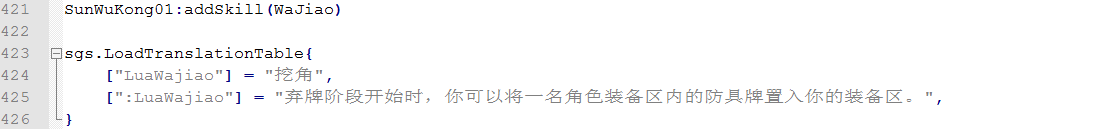
然后，筛选出场上所有已装备防具的角色作为备选角色。这可以用第二部中提到的sgs.SPlayerList()实现。

接着，如果存在这样的备选角色，从中选出一名角色作为目标角色。这里就用到了askForPlayerChosen函数，我们把optional参数的值改成true，如果在这一步选择了目标角色，就认为是发动了技能，反之则认为放弃发动；然后把notify\_skill参数的值也设为true，表示在选择角色的时候显示技能发动。

最后，如果选出了目标角色，则将其防具移动到自己装备区。

具体代码如下：





注意：

1、用于获取角色防具的方法是Player的getArmor()函数。类似的还有：

① 获得角色武器：getWeapon()。



② 获得角色防御马：getDefensiveHorse()。



③ 获得角色进攻马：getOffensiveHorse()。



④ 获得角色宝物：getTreasure()。



这些函数都没有参数。

2、还可以用以下几个函数进行相关的判断：

① 判断角色是否拥有某武器的效果：hasWeapon(weapon\_name)。



② 判断角色是否拥有某防具的效果：hasArmorEffect(armor\_name)。



③ 判断角色是否拥有某宝物的效果：hasTreasure(treasure\_name)。



以上三个函数的参数均为对应装备牌的对象名，另外hasArmorEffect还可以用于判断“八阵”、“玲珑”、“蛮甲”这些通过技能视为的防具牌的效果。

3、sgs.SPlayerList()表示角色列表，对应ServerPlayer元素。类似的还有：

① sgs.PlayerList()，对应Player类型元素。

② sgs.CardList()，对应Card类型元素。

③ sgs.IntList()，对应number中的整数元素。

④ sgs.BoolList()，对应boolean元素。

⑤ sgs.SkillList()，对应Skill类型元素。

⑥ sgs.DelayedTrickList()，对应DelayedTrick类型元素。

⑦ sgs.CardsMoveList()，对应CardsMoveStruct类型元素。

⑧ sgs.PlaceList()，对应sgs.Player\_Place类型元素。

⑨ sgs.PhaseList()，对应sgs.Player\_Phase类型元素。

⑩ sgs.GeneralList()，对应General类型元素。

其中④和⑩是目标版本中新增加的表示方法，而第一部中提到的sgs.ItemList则已经被去除了。

**第一部第十七章：用卡牌实现技能效果**

在目前版本的太阳神三国杀中，创建技能卡的方法sgs.CreateSkillCard{}比起第一部时也有了较大的变化。这时这个函数的原型是这样的：

sgs.CreateSkillCard{

name,

skill\_name,

target\_fixed,

will\_throw,

can\_recast,

handling\_method,

mute,

filter,

feasible,

about\_to\_use,

on\_use,

on\_effect,

on\_validate,

on\_validate\_in\_response,

}

比起第一部时，参数个数增加了一倍之多，当然功能也因此变得更强大了。以下便是对各个新增参数的解释。

① skill\_name：技能卡对应的技能的名字。默认值设定得有点复杂，是先去除name中的”Card”部分，再全部转化为小写字母而得到的。比如技能卡的名字是”MySuperCard”，那么这里skill\_name的默认值就是”mysuper”。但这并不意味着技能卡的名字一定要以”Card”结尾。

② can\_recast：技能卡是否允许重铸。填true表示允许重铸，填false表示不能重铸。默认为false。

③ handling\_method：对技能卡的处理方式，sgs.Card\_HandlingMethod类型。如果will\_throw的值为true，那么这一项的默认值为sgs.Card\_MethodDiscard，表示弃置；否则的话，默认值为sgs.Card\_MethodUse，表示使用。

④ mute：表示使用技能卡时是否关闭自动播放的技能配音。填true表示关闭自动播放，填false表示开启自动播放。默认值为false。但实际上，由于源代码的问题，不论这里填什么值，只要skill\_name有对应的配音，那么在使用技能卡时都一定会自动播放的，所以这一项可以忽略。

说明：

mute参数失效的原因在于，源代码的src\core\lua\_wrapper.cpp中，位于第101行的LuaSkillCard::clone函数在复制技能卡时，遗漏了mute这一项，导致技能卡的mute参数始终保持其原始的默认值false，从而无法完成关闭自动播放配音的功能效果。

⑤ about\_to\_use：表示在使用这张技能卡的时候要执行的动作。是一个函数，原型为function(self, room, use)。其中self表示技能卡本身，room表示当前的房间对象，use表示当前要处理的卡牌使用结构体。通常这一项可以忽略，这样就会执行正常的卡牌使用流程，包括确认目标生效顺序、显示提示信息、移动卡牌、对目标进行效果结算等。

⑥ on\_validate：表示实际使用的卡牌。也是一个函数，原型是function(self, use)，其中self表示技能卡本身，use表示原有的卡牌使用结构体，而要产生的是一个sgs.Card类型的结果也就是实际要使用的卡牌。这个函数为技能卡提供了一个在使用前“变成”其他卡牌的机会，从而使得一张技能卡在不同的场合下产生不同的效果。

⑦ on\_validate\_in\_response：表示实际用于响应的卡牌。还是个函数，原型是function(self, user)，其中self表示技能卡本身，user表示利用该技能卡进行响应的角色，通常也就是技能拥有者自己。和⑥一样，这个函数也是为了让技能卡能够应对不同的场景、“变成”不同的卡牌而引入的。



注意：

1、五个和技能卡效果有关的部分的作用流程为：

① about\_to\_use：结算总控制。

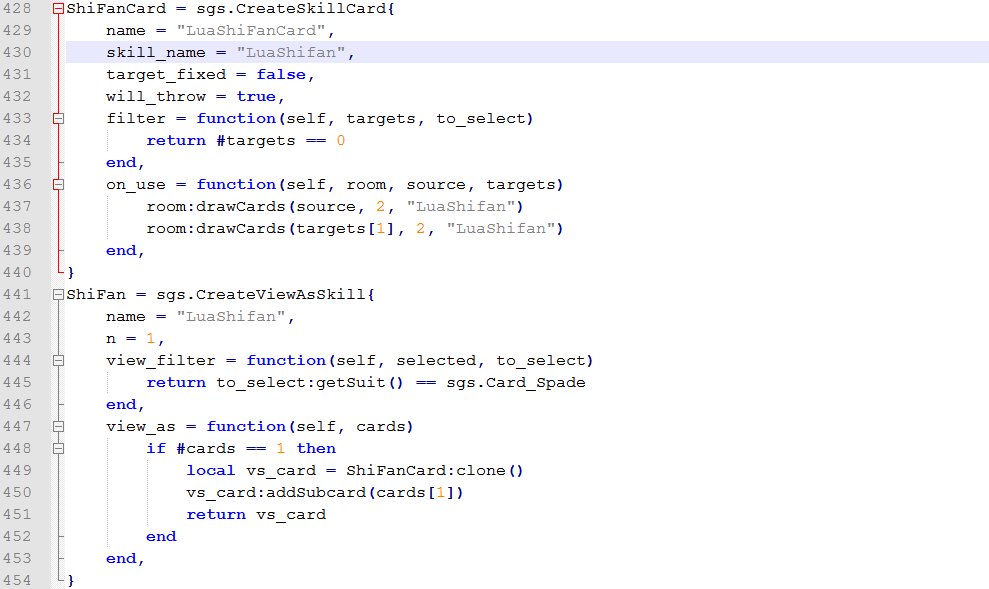
② on\_validate或on\_validate\_in\_response：确认要用的卡牌。

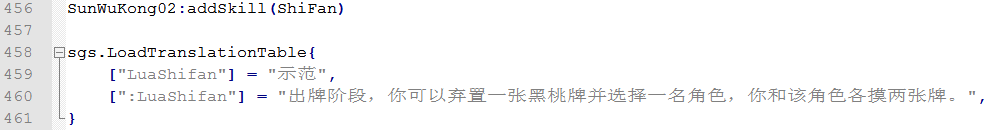
③ on\_use：效果总控制。

④ on\_effect：对每名使用目标角色的效果。

2、如果技能卡在结算过程中使用on\_validate或on\_validate\_in\_response“变成”了其它卡牌，那么就不会再进入技能卡自身的on\_use和on\_effect部分、而是去执行它“变成”的卡牌的效果了。

对于技能“示范”来说，原有的代码依然可用。



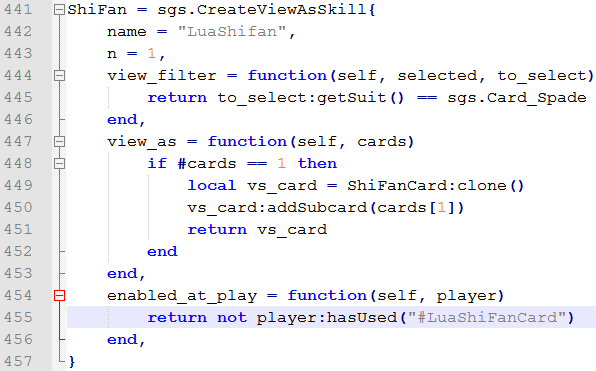


在创建技能卡时多写了一个skill\_name项而已。

**第一部第十八章：启用标志**

Room的成员函数setPlayerFlag()在目前版本中没有任何变化，所以设置标志和清除标志的方式与第一部时是一样的。

但是我们还是打算重写一下“示范”技能，因为控制技能卡使用次数还是使用第二部中提到的hasUsed()或者usedTimes()方法更方便一些。至于说setPlayerFlag()函数，在记录数据方面其实是可以发挥更大作用的。

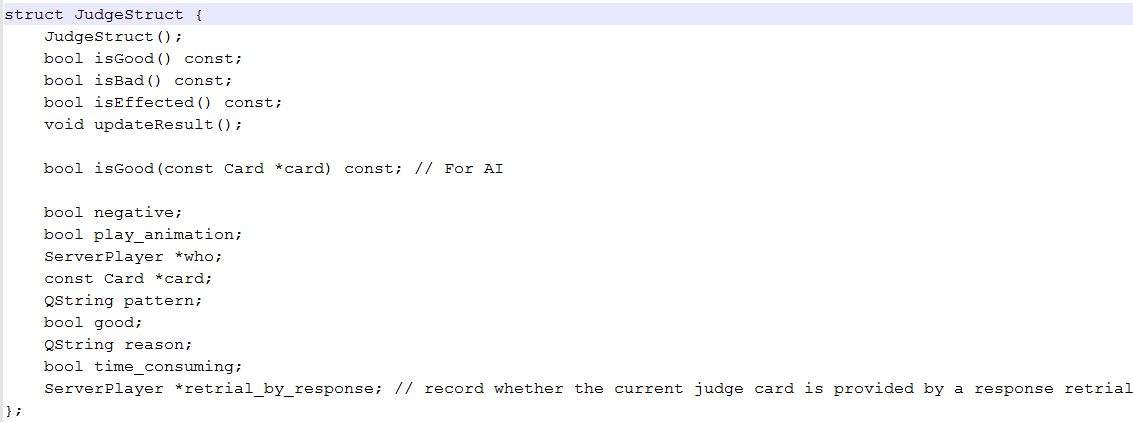


注意：

1、Lua语言对字母的大小写是非常敏感的！这里技能卡的名字（也就是用sgs.CreateSkillCard{}创建技能卡时填写的name项）是”LuaShiFanCard”，如果错写成了”luaShiFanCard”或者”LuaShifanCard”之类的就会出错，得不到想要的效果了。

**第一部第十九章：涉足判定领域**

与第一部时相比，判定结构体sgs.JudgeStruct的变化还是比较大的。它在目前版本中的结构如下图所示：

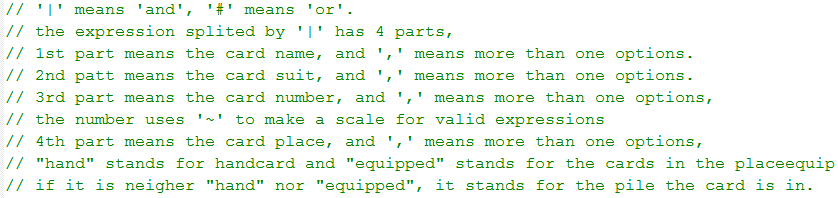


多了一个成员变量retrial\_by\_response，看代码中的注释似乎是表示提供判定牌的来源角色；不过这个变量并没有在实际中用到过，只是会在判定牌移动时对卡牌移动的原因造成一点影响，所以这里就不过多研究它了。至于另外的8个成员变量，其含义与第一部时应该是相同的。

然而最大的变化却来自于表示判定规则的成员变量pattern。在第一部中，pattern是一个sgs.QRegExp对象。但在目前版本中，pattern被简化成了一个字符串，它的常用格式是：”（卡牌判定结果）|（花色判定结果）|（点数判定结果）”。例如，要表示卡牌【闪电】的判定规则（花色为黑桃、点数为2~9），就可以这样写：”.|spade|2~9”。

注意：

1、以上只是一个常用的pattern变量的构成格式，也就是说，通常情况下用这个格式就足够了。关于pattern变量完整的格式说明，可以参考源代码中src\package\exppattern.cpp中第17行~第25行的一段注释。



这说明pattern实际上主要是由四部分构成的：卡牌名称、卡牌花色、卡牌点数、卡牌位置。并且用竖线”|”分割多个需要同时满足的规则、用井号”#”分割多个并列的只需满足其一的规则、用逗号”,”分割每个规则中的只需满足其一的条件。

2、pattern变量中，不需要的部分，如果其后续部分也都不需要，那么从这一部分开始都可以省略；如果后面还有需要的部分，那么这一部分不能省略，要用英文状态下的点号”.”代替。省略不写的部分将不会影响到判定。

比如，卡牌【乐不思蜀】的判定规则（红心牌），只提到了花色，那么就可以写成：”.|heart”。其中，第三部分“卡牌点数”是不需要的，后面也没有需要的部分了，所以我们省略了这部分内容，整个判定规则只写到了第二部分就结束了。而第一部分“卡牌名称”虽然也是不需要的，但我们需要它后面的第二部分，所以这第一部分不能被省略，只能用点号代替。

3、pattern变量的第一部分“卡牌名称”，有三种不同的写法：直接写卡牌的类型名、先写”%”符号再接着写卡牌的对象名、以及直接写卡牌的编号。

假如判定规则是卡牌【火攻】，那么pattern的值可以写成”FireAttack”（类型名）或者”%fire\_attack”（百分号和对象名）；如果要求必须是卡牌编号为137的【火攻】，那么此时pattern的值就应该写成”137”（卡牌编号）。

4、pattern变量的第二部分“卡牌花色”，除了可以要求具体的花色，还可以用于要求卡牌的颜色。其中“black”表示黑色牌、“red”表示红色牌。

例如卡牌【八卦阵】的判定规则（红色牌），就可以直接写为”.|red”（当然，写成”.|heart,diamond”也是一样的）。

5、pattern变量的第一部分“卡牌名称”和第二部分“卡牌花色”中，都可以用”^”符号表示否定，比如可以用”.|^heart”表示“不为红心牌”的判定规则。不过这种写法其实并不常用，通常会通过把sgs.JudgeStruct结构体的good变量设置为false来达到否定当前判定规则的目的。

6、不过”^”符号也不是完全没有用处，因为pattern变量的第一部分“卡牌名称”可以使用”+”符号分割两个需要同时满足的条件。

假如，判定规则是“非延时性锦囊牌”，说明符合条件的判定牌必须同时满足“是锦囊牌”和“不是延时性锦囊牌”两个条件，因此这时的pattern变量就要写成：”TrickCard+^DelayedTrick”。

7、pattern变量的第三部分“卡牌点数”，有两个特殊的规定。首先，如果规则中使用了符号”~”（包含多个连续的点数），那么可以省略作为其中最小点数的1点和作为最大点数的13点。然后，如果没有使用符号”~”（只包含一个点数），那么可以用字母”A”、”J”、”Q”、”K”分别表示1点、11点、12点和13点。

假如，判定规则是“点数不大于7”，那么就可以把pattern变量写作”.|.|-7”。当然，非要写成”.|.|1-7”甚至”.|.|1,2,3,4,5,6,7”也都是正确的。

8、pattern变量的第四部分“卡牌位置”，可以写”equipped”（表示角色装备区）、”hand”（表示角色手牌区），以及”%”连接牌堆名（表示角色武将牌上的某个具体的牌堆）。不过作为判定牌，生效时肯定不会在这些位置，所以为了不影响判定，这一部分通常是不写的。

Room的成员函数askForDiscard也增加了一个参数pattern，现在它的原型变成了：

askForDiscard(

target,

reason,

discard\_num,

min\_num,

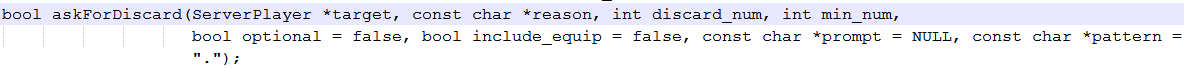
optional,

include\_equip,

prompt,

pattern

)

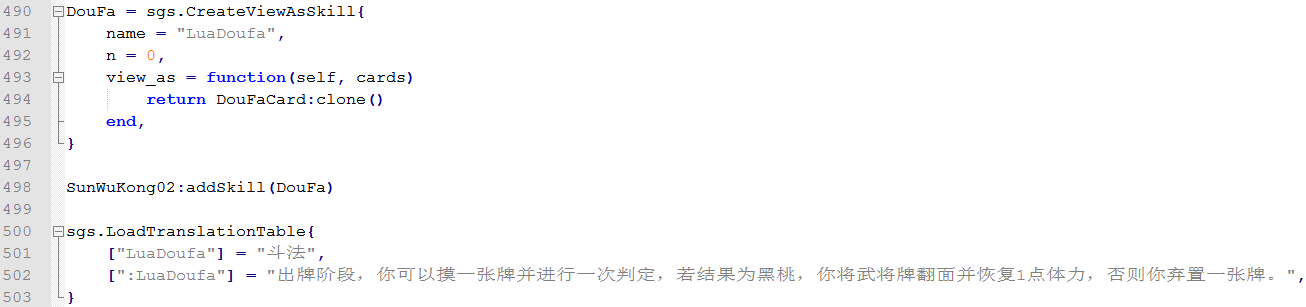


这里新增加的参数pattern表示对弃置卡牌的具体要求，写法和判定结构体sgs.JudgeStruct中的pattern变量的写法是一样的。默认为”.”，表示任意卡牌（也就是没有要求）。如果写了具体的pattern，那么目标角色只能弃置符合pattern的卡牌。

现在我们重写“斗法”技能。首先是技能卡部分，注意判定规则的新写法。



然后是对应的视为技等内容。

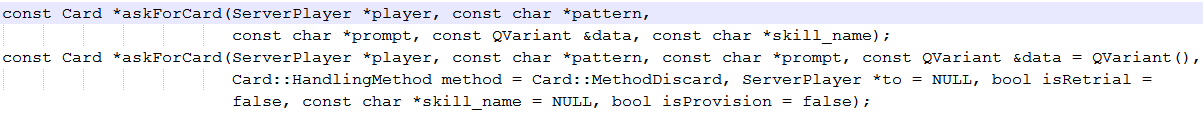


这样就完成了。

询问使用或打出一张卡牌所用到的askForCard函数也有了不小的变化。在目前的版本中，这个函数的原型是这样的：

askForCard(player, pattern, prompt, data, skill\_name)

askForCard(player, pattern, prompt, data, method, to, isRetrial, skill\_name, isProvision)



公有的前四个参数player、pattern、prompt、data以及第二个原型中的参数to，其含义和第一部时都是一样的。另外其他参数的含义如下：

① skillname：表示本次询问对应的技能名字。不写的话，默认为空。

② method：表示对询问卡牌的处理方式。比如，填sgs.Card\_MethodResponse就是询问打出一张牌，填sgs.Card\_MethodUse就是询问使用一张牌，等等。默认为sgs.Card\_MethodDiscard，表示询问弃置一张牌。

③ isRetrial：表示本次询问的卡牌是否是用于修改判定牌。有两个取值，true（表示用于改判）和false（表示其它用途）。默认为false。

④ isProvision：表示本次询问的卡牌是否是为他人提供的卡牌。也是有两个取值true（表示是为他人提供的卡牌）和false（表示与他人无关）。如果填true的话，该卡牌将首先进入处理区sgs.Player\_PlaceTable，等待后续结算；如果填false，则会直接进入弃牌堆sgs.Player\_DiscardPile。默认值为false。在目前版本的太阳神三国杀中，只在激将、护驾、连理这三个技能中用到了这个true值。

第一部中，技能“引导”采用了“触发技＋视为技＋技能卡”的实现方式，这种方式在选牌规则较为复杂的时候会比较方便。不过考虑到这个技能本身并没有那么复杂，所有内容全写进一个触发技也没有问题，而且更为简洁。下面我们就利用这个新版本的askForCard函数重写一下这个技能。



这里由于askForCard函数需要的参数比较多，把括号中的内容分成三行写了。

**第一部第二十章：私家牌堆**

目前版本中用到的私家牌堆如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 牌堆名称 | 牌堆代码 | 来自技能 | 对应游戏包 |
| 马 | “xlhorse” | 孝廉 | 台版 |
| 司敌 | “sidi” | 司敌 | 一将成名2014 |
| 逆 | “counter” | 陷嗣 | 一将成名2013 |
| 醇 | “wine” | 醇酪 | 一将成名2012 |
| 权 | “power” | 权计 | 一将成名 |
| 米 | “ytrice” | 义舍 | 倚天 |
| 倨傲 | “hautain” | 倨傲 | 智 |
| 不屈 | “buqu” | 不屈 | 风 |
| 遗计 | “yiji” | 遗计 | 标准版 |
| 谦逊 | “qianxun” | 谦逊 | 标准版 |
| 木牛流马 | “wooden\_ox” | 木牛流马 | 界限突破卡牌 |
| 忠义 | “loyal” | 忠义 | 3v3 |
| 笔伐 | “bifa” | 笔伐 | SP |
| 咒缚 | “incantation” | 咒缚 | SP |
| 引兵 | “yinbing” | 引兵 | SP |
| 伏兵 | “ambush” | 设伏 | SP |
| 兵 | “zhenweipile” | 镇卫 | SP |
| 焚 | “burn” | 焚天 | 其他 |
| 节钺 | “jieyue\_pile” | 节钺 | OL |
| 粮 | “food” | 输粮 | OL |
| 破军 | “olpojun” | 破军 | OL |
| 米 | “rice” | 义舍 | OL |
| 清俭 | “olqingjian” | 清俭 | OL |
| 不屈 | “nosbuqu” | 不屈 | 怀旧·风 |
| 田 | “field” | 屯田 | 山 |
| 幻 | “sorcery” | 千幻 | 国战·阵 |
| 星 | “stars” | 七星 | 神 |
| 梦 | “dream” | 醉乡 | ☆SP |
| 锦 | “brocade” | 银铃 | ☆SP |
| 军威 | “junwei\_equip” | 军威 | ☆SP |
| 诏 | “edict” | 汉统 | 贴纸 |
| 扈 | “retinue” | 义从 | 贴纸 |

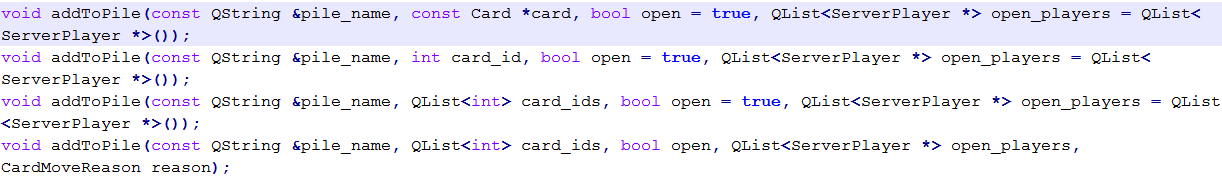
将卡牌添加到私家牌堆的函数addToPile()比起第一部时也有了不小的变化。现在它的原型是：

① addToPile(pile\_name, card, open=true, open\_players=sgs.SPlayerList())

② addToPile(pile\_name, card\_id, open=true, open\_players=sgs.SPlayerList())

③ addToPile(pile\_name, card\_ids, open=true, open\_players=sgs.SPlayerList())

④ addToPile(pile\_name, card\_ids, open, open\_players, reason)



相比之下，增加了两个新参数open\_players和reason，其中：

① open\_players：表示将卡牌公开的目标角色列表。不写的话，默认取值是sgs.SPlayerList()，也就是空列表，没有对任何角色公开当前卡牌。

② reason：表示卡牌移动的原因。

注意：

1、如果要添加的卡牌需要对在场一部分角色公开、对另一部分角色保密，则会用到open\_players参数。把open参数的值写成false，以实现保密效果，然后令open\_players参数的值为需要公开的角色列表即可。

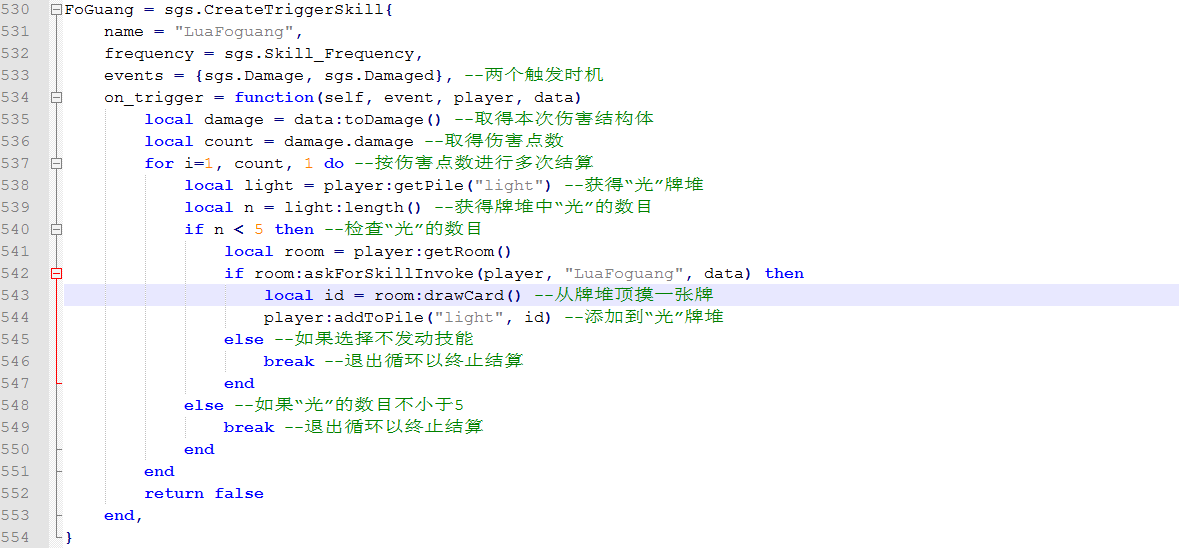
2、由于参数open\_players在前三个原型中可以省略不写，所以通常情况下，这个函数的用法和第一部时是完全相同的。

3、

虽然addToPile()函数的这个变化对我们的影响不大，但是我们还是决定要始重写技能“佛光”：“每当你造成或受到1点伤害后，你可以摸一张牌并将其置于你的武将牌上，称为‘光’（最多5张）。”

因为这是一个按伤害点数发动的技能，而且有数量上的限制（最多5张），所以在遇到一次多点伤害时，按照原先的写法就容易出现游戏效果与技能描述不符的情况。比如武将牌上已有4张“光”，此时受到了【闪电】造成的3点伤害，应当进行3次结算（第一次，发动技能，“光”达到5张；第二次和第三次不能发动技能）；而原先的写法，只是在最开始的时候进行了一次判断，然后一次性获得同等点数的“光”，就会造成只进行了一次结算，发动技能后得到7张“光”的效果。

解决的办法是，首先取得一次伤害的点数，然后利用循环进行重复的结算。把对“光”的数目的判断和询问进行技能发动的过程写在循环内部。代码如下：





因为每次结算只需要摸一张牌，而不是一次性摸多张牌，所以这里用到的是Room中的drawCard()函数，而不是第一部中提到的getNCards()函数。当然，这两个函数在目前版本的太阳神三国杀中都没有什么变动。

除了addToPile()函数以外，与五谷丰登选牌有关的一系列函数也有了新的变化。首先是填充选牌窗口用到的fillAG()函数，现在它的原型是：

fillAG(card\_ids, who=nil, disabled\_ids=sgs.IntList())



新增了一个参数disabled\_ids，表示禁选的卡牌编号列表。这个参数可以省略不写，此时将采用默认值sgs.IntList()，也就空列表，表示填充到窗口中的所有卡牌都可以被选择。

说明：

1、

然后，对于清空并关闭选牌窗口，在第一部时我们是通过用ServerPlayer中的invoke()函数发送”clearAG”信号来实现的。在目前版本的太阳神三国杀中，这个操作已经被包装成了Room中的一个函数clearAG()，它的原型为：

clearAG(player=nil)



唯一的参数player，表示要关闭选牌窗口的角色。如果不写的话，将取默认值nil，表示关闭在场所有角色的选牌窗口。

说明：

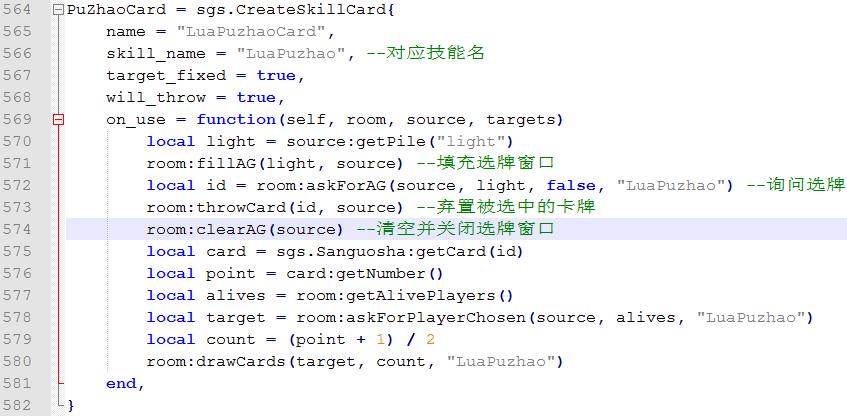
1、有时候选牌窗口是对所有角色可见的，在一段操作完成后我们需要关掉所有角色的选牌窗口（如卡牌【五谷丰登】的效果），此时只需要省略player参数，直接写上一句clearAG()即可。

2、还有的时候，选牌窗口只对特定的角色可见（即，窗口是不公开的，如技能“心战”的效果）。那么在一段操作完成后，我们需要关掉的仅仅是这些角色的窗口，此时就会用到player参数了。找到这些要关闭窗口的角色，然后分别对其使用clearAG(player)即可。

3、ServerPlayer的invoke()函数依然存在，而且仍然可以用于关闭选牌窗口。但是其用法与第一部时已大不相同，这里不再推荐使用。

现在我们开始重写技能“普照”。

其实只是修改了技能卡部分而已。



而视为技部分几乎没有什么变化。



这样就完成了。

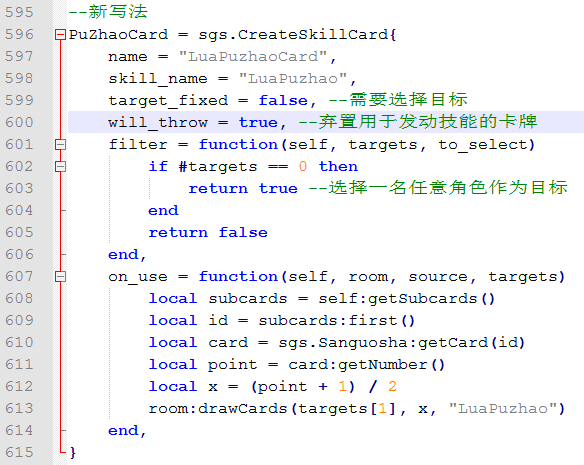
不过，第一部中说过，这个技能是仿照邓艾“急袭”设计的。在目前版本的太阳神三国杀中，邓艾的“急袭”及类似的技能都已经改成了将牌堆直接移入手牌区进行操作的形式，不再使用窗口选牌了。另一方面，根据“普照”的技能描述，应该是先选择弃置“光”然后再选择目标角色，而现在的代码效果是先选择了目标角色再在技能卡中通过窗口弃置“光”的，与描述其实并不完全相符。所以我们应当考虑换一种新的方法重写这个技能，“与时俱进”。

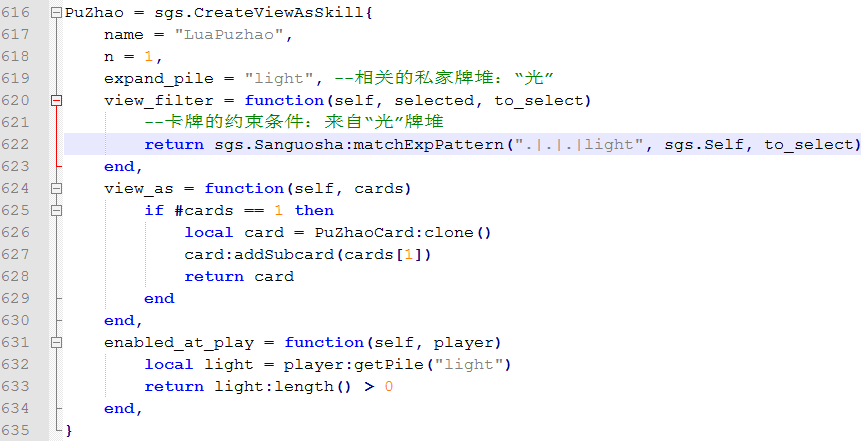
原先的写法是，在视为技中不选择卡牌，而是把选择卡牌作为技能效果放进技能卡里。现在我们把选择卡牌作为技能消耗写在视为技中，而技能卡里只有确定目标的过程和摸牌的效果。按照这个思路，技能描述可以改成这个样子：

普照：你可以将一张“光”当作“普照技能卡”对一名角色使用。

普照技能卡：目标角色摸X张牌，其中X为本技能卡点数的一半。

对应地，代码方面就变成了如下模样。





这里用到了一个新的函数：matchExpPattern()，来自Engine。它的作用是判断某角色的卡牌是否符合特定的约束条件。原型如下：

matchExpPattern(pattern, player, card)



其中：

①pattern：表示指定的约束条件。

②player：表示目标角色。

③card：表示待检查的卡牌。

注意：

1、pattern的写法和判定中的pattern相同。

2、player的类型为Player，而不是ServerPlayer。

**第一部第二十一章：卡牌的锁定**

锁定视为技的实现方式与第一部时是相同的，代码无需改动仍然有效。



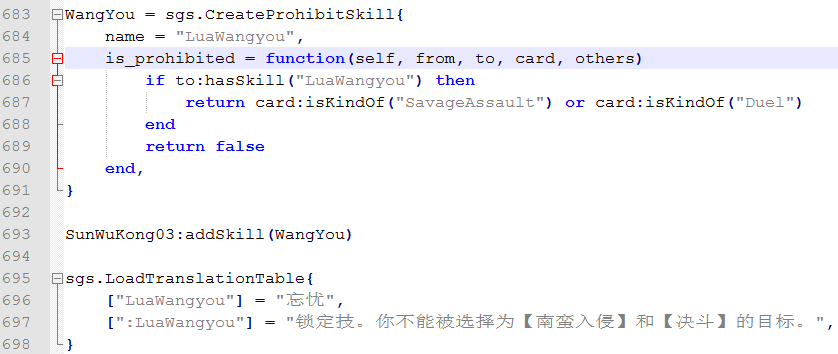


注意：

1、如果是像技能“红颜”那样只是修改卡牌颜色，没有重新视为其它卡牌的情形，可以跳过takeOver有关的步骤，直接修改getWrappedCard的结果即可。

**第一部第二十二章：卡牌免疫**

创建禁止技用到的sgs.CreateProhibitSkill方法比起第一部时变化不多，所以技能“忘忧”的代码仍然有效，可以直接使用。



唯一的不同在于，is\_prohibited函数多了一个参数others，表示其他有关的角色。但是这个参数可以省略，省略时将取其默认值sgs.PlayerList()。

**第一部第二十三章：让房间提供卡牌**

**第一部第二十四章：启用标签**

**第一部第二十五章：濒死结算**

**第一部第二十六章：交换座次**

**第一部第二十七章：创建主公技**

**第一部第二十八章：强力技能**

**第二部第一章：聊聊天，叙叙旧**

**第二部第二章：摸牌堆控制术：观星与攻心**

**第二部第三章：隐藏技能**

**第二部第四章：触发机制：将干涉进行到底**

**第二部第五章：获得一个新技能**

**第二部第六章：限定技与觉醒技**

**第二部第七章：动画效果**

**第二部第八章：花样百出的摸牌手段**

**第二部第九章：不可回避的问题**

**第二部第十章：防具亦有悲剧时**

**第二部第十一章：不屈的战魂**

**第二部第十二章：死亡威慑**

**第二部第十三章：更改伤害来源**

**第二部第十四章：最中意的面孔**

**第二部第十五章：天下归心**

**第二部第十六章：限制卡牌的使用次数**

**第二部第十七章：异性之间的话题**

**第二部第十八章：卡牌强化**

**第二部第十九章：引入拼点机制**

**第二部第二十章：多张卡牌的移动**

**第二部第二十一章：卡牌的封禁**

**第二部第二十二章：连续发动技能**

**第二部第二十三章：选择花色**

**第二部第二十四章：询问使用卡牌**

**第二部第二十五章：多重视为技**

**第二部第二十六章：手牌展示**

**第二部第二十七章：遗计定乾坤**

**第二部第二十八章：系统秘籍**