**多方块合成结构：祭坛**

项目地址：<https://github.com/ENIACJushi/TouHouLittleMaidBE>

**1. 对功能的简单描述**

① 站在完整的未激活结构外，手持御币右击鸟居下梁，转为激活状态；

② 当结构被破坏，转为未激活状态；

③ 手持物品右键木桩最顶部一格，将物品放上去，空手+右键拿下来；

④ 当放置物品后达到合成条件，自动进行合成，不需要按顺序放置。

**2. 关于如何实现的思考**

需要考虑用方块还是实体实现方块的合成的核心组件。要能用物品改变方块或实体的数据，并获取和处理方块或实体中存储的这些数据。

① 实体的交互框小于方块，由方块存储物品数据（废弃方案）

能实现在方块被破坏时，触发销毁激活中祭坛的事件。但是注意，方块的on\_interact触发器只能处理事先设定好的物品值，不在合成表里的东西是不能处理的。为了可拓展性考虑，必须制作自动生成祭坛实体定义json的程序，不太可行。

② 实体的交互框大于方块，由实体存储物品数据（废弃方案）

破坏起来不方便，只能瞬间消失或者干脆无法破坏，但只有合成方块（木桩顶部）需要这样处理，其它的不需要，无伤大雅。可以用物品触发事件，也能通过score读出数据。使用小安粒的事件有希望支持全物品，是值得尝试的方案。

让小安粒能那东西的interact事件属性是give\_item，取下的是take\_item，可以通过脚本获取手上的物品。

监听放置/取下物品的事件比较绕，先用interact触发event，然后在脚本用DataDrivenEntityTriggerEvent监听，从而发起合成。

计划通。分别用6个类小安粒的实体包裹木桩最顶方块，实体持物的模型组模仿掉落物上下浮动，监听他们的交互事件以启动合成表的判定。用另一个实体提供祭坛的材质，没有交互框，因为对破坏祭坛的监听由方块完成；鸟居模型有些地方比一个方块小，新增一个技术性的透明方块，在激活祭坛时替换红色羊毛，它除外观以外的特性均与红色羊毛保持一致，即基本破坏时间、剪刀能加速破坏、破坏后掉落红色羊毛。木桩被完全包住了，但基岩版的谜之渲染会把距离相差不大的前后两个物体同时显示出来，也需要透明处理。

废弃原因：在测试时发现无论怎么做，实体都能被鱼钩钩走，所以必须将它们卡进方块。

③ 实体的交互框小于方块，由实体存储物品数据**（最终方案）**

用方块接收交互事件，然后使用脚本操作容器数据，将玩家手上的物品转移到实体里，这样做更加贴近Java版mod的操作方法，而且还可以使木桩顶部方块可交互。 大致实现如下：

分别将6个类小安粒的实体卡进木桩最顶方块，实体持物的模型组模仿掉落物上下浮动。

用另一个实体提供祭坛的材质，没有交互框，作为一个结构只拥有一个的实体，担任着类似“核心方块”的职务，称为“核心实体”；

对破坏祭坛的监听由方块完成，当结构被破坏时，方块会向核心实体发送销毁请求；

鸟居模型有些地方比一个方块小，新增一个技术性的透明方块，在激活祭坛时替换红色羊毛，它除外观以外的特性均与红色羊毛保持一致，即基本破坏时间、剪刀能加速破坏、破坏后掉落红色羊毛。

木桩被完全包住了，但基岩版的谜之渲染会把距离相差不大的前后两个物体同时显示出来，所以也需要透明处理。

脚本无法直接监听某个方块的事件……这可能会导致更多的性能开销，暂时用itemUseOn的监听好了。

**3. 具体的实现**

① 实体

~核心实体：touhou\_little\_maid:altar\_main

在祭坛被激活时位于鸟居下梁和上梁之间，从内向外看靠左的柱子内部。

为祭坛的非交互部分提供材质。

销毁事件touhou\_little\_maid:altar\_destroy：因为1.19.50方块的"minecraft:on\_player\_destroyed"事件并不能获取到方块破坏，并且无论是方块组件还是脚本事件（blockBreak），都只能检查来自玩家的破坏，而无法检测其它破坏，所以暂时令核心实体每游戏刻都检测结构完整性。若之后的版本推出检测所有破坏的组件，则可以由结构中被破坏的方块通过/event调用，检测结构是否完整，不完整则销毁。

销毁事件和合成事件的执行均由脚本完成。

③ 方块

~替换木桩顶部的方块：touhou\_little\_maid:altar\_platform\_block

被破坏时，触发销毁事件。

交互时触发放置物品判断，若方块实体没有物品，且玩家主手有物品，则将玩家主手物品转移到方块实体中，随后进行合成检测。若方块实体有物品，且玩家主手没有物品，则触发方块实体的弹出物品事件touhou\_little\_maid:altar\_pop\_item。

合成事件touhou\_little\_maid:altar\_craft：由方块通过命令/event调用，检测当前祭坛上的物品是否满足某个合成，满足则执行合成。

~替换红色羊毛的方块：touhou\_little\_maid:altar\_torii\_block

提供碰撞箱，只是一个占位置的方块。除外观、不可燃以外的特性均与红色羊毛保持一致，即基本破坏时间、剪刀能加速破坏、破坏后掉落红色羊毛、选择框的大小。

~替换木桩下两格的方块：touhou\_little\_maid:altar\_log\_block

提供碰撞箱，只是一个占位置的方块。

④ 物品

~灵梦的御币：touhou\_little\_maid:hakurei\_gohei

木棍（2）钻石（1）纸（2）在工作台合成

木棍（3）纸（3）在祭坛合成

⑤ 脚本

最主要的东西，因为量有点大（数据以外的代码700行），在这就不放具体流程了，注释写的很详细，可以直接查看源码。

主要有三个模块：激活、释放、合成。

~关于激活模块

因为实体转头有过渡动画，所以不同方向的实体用的是不同方向的模型，在生成时通过脚本设置哪个方向模型应该被显示。

~关于将物品放在平台上

方案1（弃用）

做法：将物品生成在平台实体的位置，令实体自己将物品装备在手上，从而显示出来。

因为实体防不了鱼钩，所以只能把它卡在方块里，用gt上物品。只能用gt而不能用replaceitem指令是因为放置物品还要考虑后续取下来以及用于合成时的判断，是必须要保留所有数据的，包括潜影盒里的物品甚至创造模式ctrl＋鼠标中键得到的那种nbt数据。指令做不到保留数据，而gt有由已有物品生成掉落物的API，那个可以做到保留所有数据地生成一个掉落物。

实体的背包、主手、盔甲栏是三个独立的东西，目前gt的接口只能访问背包里的物品，要让物品显示出来就得让实体自己装上，即丢个物品给他让他自己捡。，需要用到包括equipitem的四个组件，当时怎么做也没法阻止他们把盔甲装上盔甲栏，就弃用这条方案了。

实体组件可以实现手上只拿一个物品，但没法避免在自动装备到主手的同时不装备到盔甲栏。

还有一个更为重要的弃用原因是：比较复杂的合成表匹配问题必须要用gt脚本来实现。而gt不能访问主手和盔甲栏，显示出来了（即装备上了主手或者盔甲拦）就不能用于合成（即在背包栏供gt访问），能用于合成就不能显示出来。

方案2（最终方案）

做法：将玩家手中的物品变为掉落物，生成在平台中央。

itemStack数量（amount）无法被设置（1.19.50），也没有清空栏位的API，所以使用

了clear命令＋延时执行实现将手上任意数量物品中的一个放在平台上。

已知bug：放受损的物品会导致身上的物品不能清除，因为clear的data会检查物品耐久，而gt得不到这个值。因为新版推出了清空格子的方法，所以现版本暂不修复。

如果不给平台方块设置interact属性，手持方块与其交互时就会把方块放上去，所以这个属性必须加；潜行状态交互interact属性也防不了，所以在事件那里加了个玩家是否处于潜行状态的判断。

~关于合成

适配了原mod的合成表，但tag得自己设。

方块调用实体的合成事件，再让实体进行合成检测会导致调用者不明，为了p点系统考虑，得在方块这边就处理，因此给平台方块加了rotation和number属性，rotation记录旋转，number记录在结构中的位置（逆时针第几个）。

平台上已经有一件丢上去（没tag）的物品时，再放同类物品会堆叠在一起，所以合成时要检测数量，若有大于一的量，弹开这个物品并终止合成。

读取合成表使用动态引入方法，合成表文件提供一个合成json常量，列表文件提供一个合成表文件路径json常量，atarCarft类则根据列表文件动态导入合成表。

先匹配物品数量，再构建领接矩阵，使用匈牙利算法寻找最大匹配，找到的是最优匹配则返回true，生成合成结果并kill材料。