# AfterDawn的eevee教程

如果你仅希望自己快速入门，只需要看涂色的内容

**红色加粗字体**是一些我觉得比较重要的内容

目录

学习eevee能做什么？ 2

eevee是怎么工作的？ 4

Attached Container（依附控制器） 7

Collidable Modifier（碰撞属性修饰器） 12

Depth Modifier（深度属性修饰器）

Debug Container（Debug控制器）

Flag Gata Container（Flag硬币门控制器）

Flag Toggle Modifier（实体属性修饰器）

Floaty Container（月亮块属性控制器）

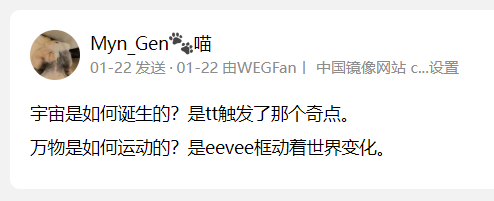
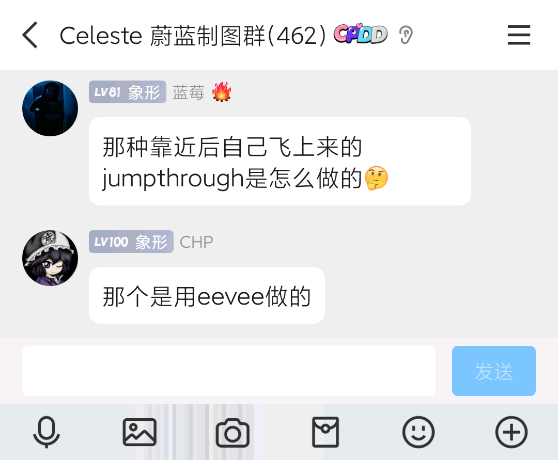
Global Modifier（全局属性修饰器）

Holdable Container（抓取物控制器）

SMW Track Container（SMW轨道控制器）

# 学习eevee能做什么？

也许，你以前没有听说过eevee

也许，你曾经在被制图前辈推荐eevee

eevee是什么？

为什么大佬们都在推荐eevee？

我们日常说的eevee，指的是**eevee helper提供的那些控制器**（而不是精灵伊布）。

Helper链接：https://gamebanana.com/mods/53765

你可以通过上面的链接基础的了解eevee helper实体的基本情况。

eevee极大的拓展了我们能使用的实体，并丰富了实体的用法，让我们可以在不懂代码不会写新实体的情况下，创造出无穷的实用实体，以及无数实体的新特性。

eevee可以自由的控制大部分实体，更改那些实体的各种特性以便满足各种需要。

举例如下

简单的，比如使错位的单向板正常依附在实体上，使实体消失时摧毁相关装饰物，更改碰撞效果，检测实体状态并输出到日志中，丰富场景层次，制作剧情演出，无效化实体，隐藏实体，自定义任何实体的浮动特性，全局化加载实体，抛接实体，任意实体应用SMW轨道运动。

复杂的，比如抓住水母时无视电网，建立全局计时器，制作限时关卡，制作弹幕游戏，抛接重生点。

# eevee是怎么工作的？

eevee的工作主要有两个部分组成：**选择实体，影响实体**。

本节仅会说明eevee会如何选择实体，关于他们如何影响实体我会在那些eevee控制器的具体说明中讲解。

eevee的本体被称为**容器（Container）**，eevee会尝试包含**容器中的所有实体。**

容器位置，大小与Ignore Container Bounds属性决定了容器**包含实体的范围**；Contain Mode决定了容器**包含实体的时机**；而白名单与黑名单决定了容器**包含实体的种类**。

勾选Ignore Container Bounds属性后，eevee会无视容器大小，试图房间内包含所有满足条件的实体，因此你可以将它放在房间外好好整理各种控制器，并让它可以正常包含房间内所有白名单内的实体。

**eevee包含实体的时间由****Contain Mode决定**，下面是4种Contain Mode的介绍：

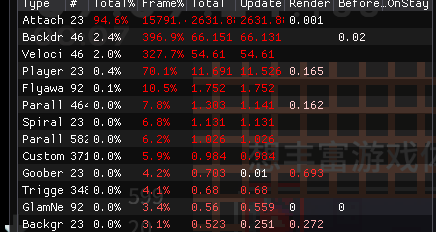
RoomStart：**房间开始时**立即进行包含判断

FlagChanged：每次当**flag启用时**进行包含判断

Always：它会**每帧**都进行包含判断，只要满足包含条件它都会重新进行包含，以使范围内实体始终被包含。

**警告！不当的使用会大大增加性能负担**

DelayedRoomStart：在首次Update后尝试附着，避免在房间初始化阶段处理复杂逻辑适用于需要等待其他实体初始化的场景。有什么RoomStart处理不了的就用DelayedRoomStart。

（警告！如果你忽视容器边界还把包含模式调为always，eevee就会每帧翻遍你房间内所有实体去寻找自己要控制的东西，导致性能占用爆炸）

是的，那个性能占用达到94.6%的东西就是这样的一个eevee，谨记不要乱尝试这样的危险行为（对电脑危险）。

如果写了白名单（Whitelist），它仅会包含白名单中的实体；如果写了黑名单（Blacklist），它就会排除这些实体不包含。

那么……名单要写什么？entity type！

entity type是实体的身份证，每个实体的entity type独一无二，我们可以通过它选择我们要控制的实体；

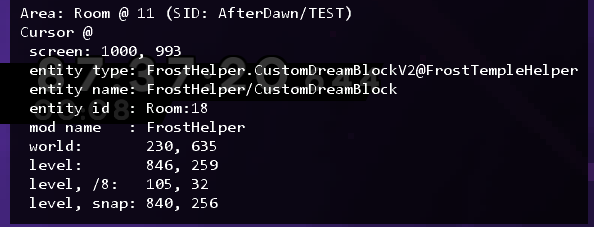
注意：在Loenn的实体详细信息上面出现的是helper name/entity name，尽管entity type和entity name大部分时候相同，但并非绝对一致，如果随意的使用entity name，eevee也可能随意的偷懒罢工。

如果你想查看一个实体的entity type，你需要以下几个步骤

在开启Celeste TAS这个mod的情况下进入游戏

按~键进入调试模式

鼠标点击对应实体（多点几下可以切换展示同一位置下的其他实体属性）

把下面图中红线标注的文字抄下来（大小写严格）



当你把entity type抄到eevee的Whitelist或Blacklast后，eevee就会选择性的控制你想控制的实体，而不是控制容器范围内所有实体

可以使用英文逗号分隔多个entity type，以此控制多种实体。

顺带一提……eevee能包含的不止有entity，还有控制trigger和player。

# Attached Container

依附控制器，可以让容器及其包含实体，依附于节点指向的另一个实体。

**属性列表**

Contain Mode：包含模式，具体见前文，**不会就用RoomStart**

Contain Flag：Contain Mode为FlagChanged时，用于进行**包含判断的条件flag**

Whitelist：容器**仅会包含白名单中的实体**，具体见前文

Blacklist：容器**不会包含黑名单中的实体**，具体见前文

Attach Mode：与Contain Mode类似，但是**仅用于目标实体**

Attach Flag：与Contain Flag类似，但是**仅用于目标实体**

Relative Attach X/Y：不填写时，容器与目标实体的**位置关系**与**包含判断时**的位置关系相同；填写后，容器与目标实体的**位置关系**以这个属性的值为准。**Only X/Y会影响它的作用**。

Attach to：**目标实体的白名单**，与Whitelist类似

Destoryable：勾选后，目标实体被摧毁时，**容器及其内部实体会被一起摧毁**

Fit Contained：当容器内容发生变化时，自动调整容器大小，同时固定容器中心位置，避免**容器位置突变**，确保容器能够自动适应内容变化，同时保持视觉一致性和物理特性

Force Standard Behavior：打开后强制使用标准行为，适用于**需要统一行为的场景**（比如大范围/多种实体）；关闭后优先采用自定行为，会降低性能，但更加稳定。标准行为不允许实体实现特殊附加逻辑，因此打开属性后可能有一些不确定的特殊行为。

Ignore Anchors：忽视实体节点，容器**不对实体节点进行处理**，适用于实体节点有自己的处理方式的情况。

Ignore Container Bounds：**忽视容器边界**，本实体一般不建议使用：如需使用，推荐打开Force Standard Behavior。

Match Collidable：**匹配碰撞特性**，如果目标实体不可碰撞，则容器内实体不可碰撞，和Flag Toggle Modifier影响有关。

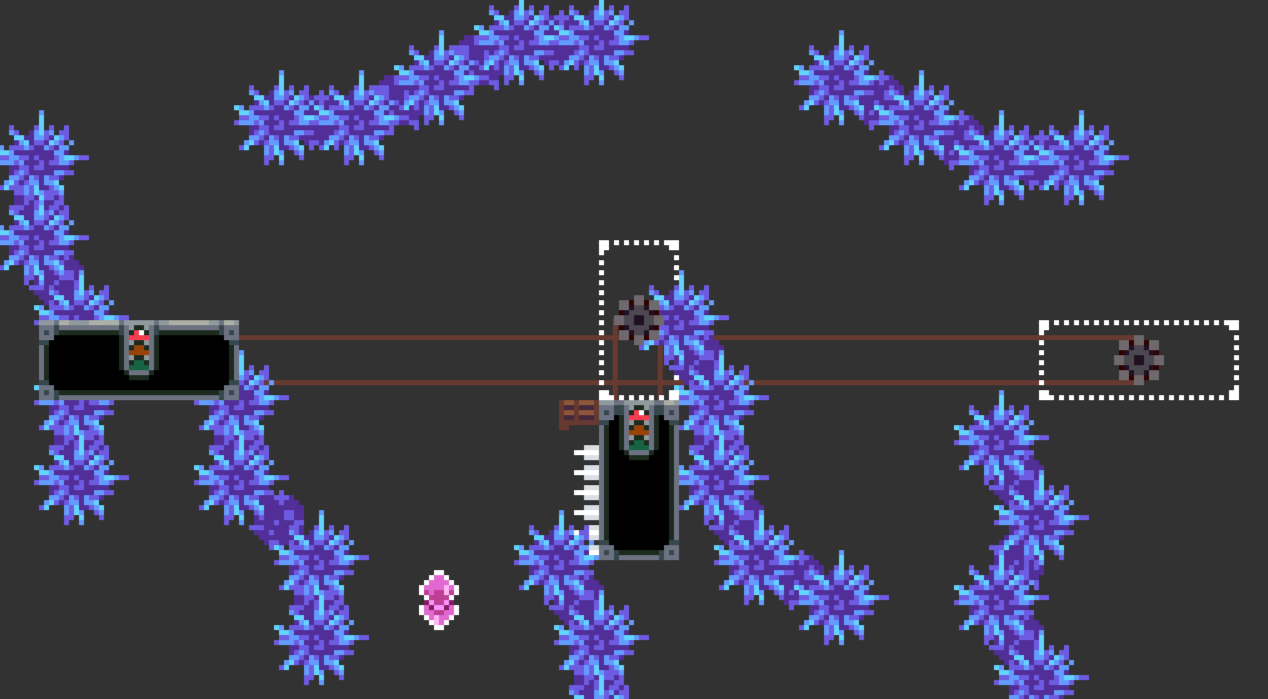
Match Visible：**匹配可见行为**，如果被目标实体不可见，则容器内实体不可见，和Flag Toggle Modifier影响有关。

Only X/Y：目标实体**只会影响容器的X/Y移动**。

**不要都打开！**

Restrict To Node：**严格依附在节点位置的对应实体上**，如果关闭，则会依附在离节点最近的实体上，默认就好。

**使用实列（选读部分案例即可）**

1. 基础运用其一：错位单向板正常依附于实体

**目的：**本处需要**让不在实体边缘的单向板正常依附于红绿灯**。（批注：本处实际上不需要单向板提升容错，可能需要应用的情况应为用于蹭墙修正的多个侧单向板）我们可以想到用Attached Container让它正常依附。

**对需要实体的分析（本段请重视）：**

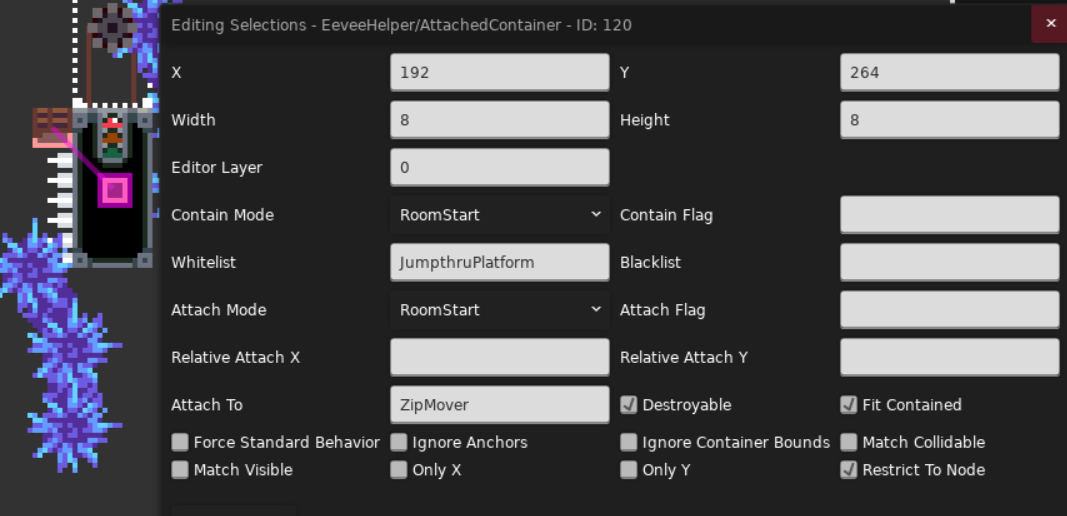
Contain Mode：无需更改包含情况，且红绿灯与单向板的加载行为都比较简单，应用RoomStart即可。

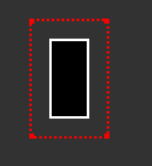
Whitelist：容器需要包含的实体为单向板，写入JumpThru。

Attach to：依附目标为红绿灯，写入ZipMover。

Destoryable：不需要摧毁单向板，其红绿灯不会自行摧毁，无需在意开关状态。

**参考布设方式以及实体属性如下图**

1. 基础运用其二：一次性实体消失时摧毁装饰

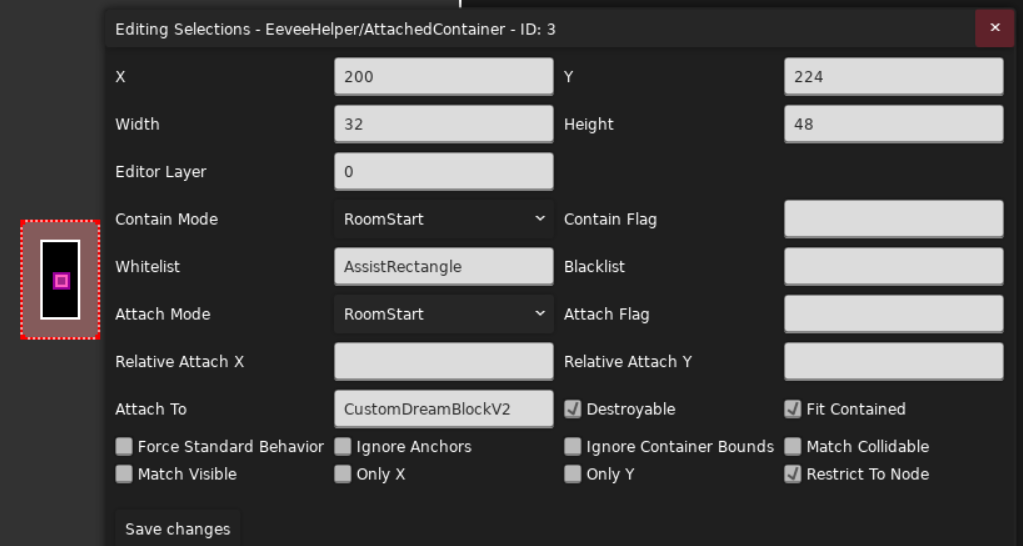
**目的：**本处需要**让果冻消失时，用于标注其为一次性果冻的装饰实体消失**。我们可以想到用Attached Container让装饰实体随着果冻消失而消失。

**对需要实体的分析（本段请重视）：**

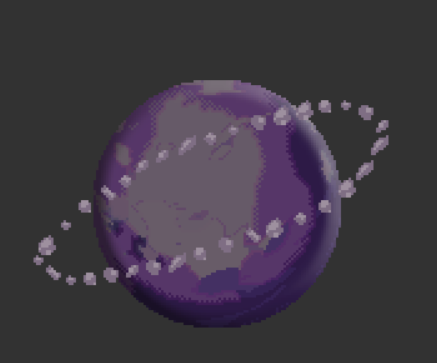
Contain Mode：无需更改包含情况，且实体边框与一次性果冻的加载行为都比较简单，应用RoomStart即可。

Whitelist：需要包含的实体为实体边框，写AssistRectangle。

Attach to：依附目标为一次性果冻，写CustomDreamBlockV2。

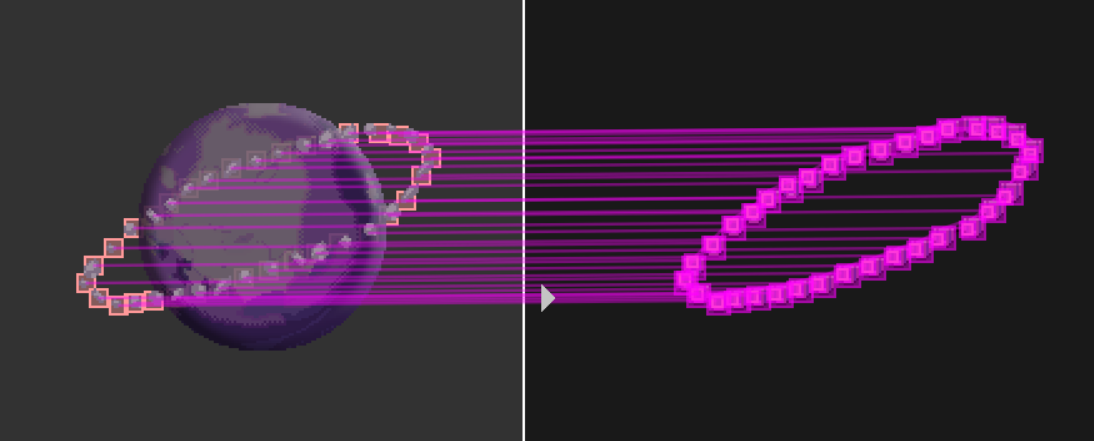
Destoryable：本处需要让一次性果冻消失时实体边框一同消失。属性设置为true。

1. 复杂运用其一：旋转着的星环（上半）

**目的：**该案例中，**我们希望这些小石子能按照轨道绕着星球移动**。

目标拆解：1.自定义轨道。2.维持合理的图层关系（将于Depth Modifier补全）

这些石子无法自定义轨道，我们可以想到用Attached Container让石子依附于可以自定义轨道的实体。本处使用的是多节点周期板（这很麻烦，类似的东西你会很少见）。

 本处不对实体属性进行分析，可参考“基础运用其一：错位单向板正常依附于实体”，其他属性使用默认属性。

## Depth Modifier

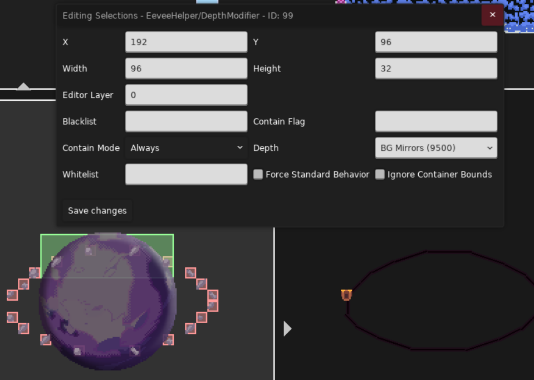
深度控制器，它能改变实体的深度，使mapper可以自由的调节各个实体显示的上下层级关系

声明：如果实体自带Depth属性的话，以实体Depth为主最好，用多了这个怕有bug

——底龙

（如果你在疑惑我为什么在讲Depth Modifier而不是Collidable Modifier的话，可以看看上一页结尾）

上文的土星环，所有debris都显示在星球贴图之上，这样的土星环显然是很奇怪的，我们需要把应该显示在星球后面的debris上面放上一个Depth Modifier，改变他的深度。

你需要在右图位置放下Depth Modifier，只用把白名单（Whitelist）写上（该区域未放入其他实体，因此没有写白名单，偷懒了），写好控制模式为always，放在应该显示在“后面”的石子那里，把depth调高到在depth后面就行了

属性列表（没兴趣深入可以只管标红的属性）

Contain Mode：包含模式，前面用一整节讲了，不会就用默认的RoomStart

Contain Flag：Contain Mode为FlagChanged时重新进行包含判断的条件flag

Whitelist：容器包含实体的白名单，写入你想控制的实体的entity type

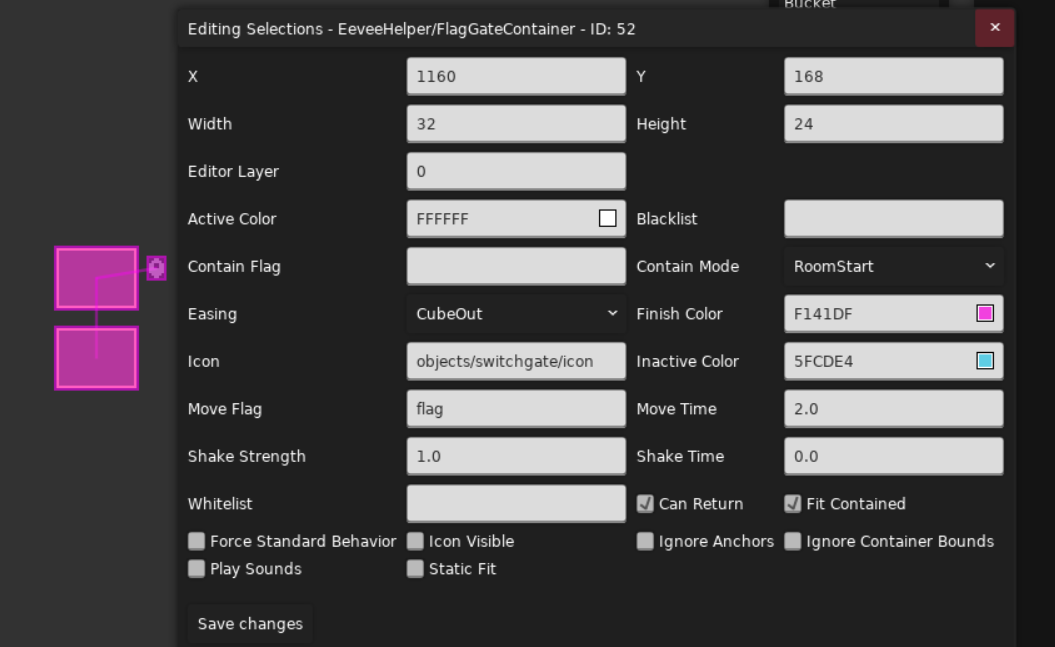
Blacklist：容器包含实体的黑名单，写入你不想控制的实体的entity type

**Depth**：你可以通过更改这个数值更改实体的显示位置，在eevee的属性列表中列出了一些常见的东西的depth，你可以选择合适的depth或者自己写一个数值

因为这个属性**非常重要**，因此我在这里插入一段wiki原话（和翻译）

You can change the order in which an entity is rendered/updated by changing its Depth.  
您可以通过更改实体的 Depth 来更改实体的渲染/更新顺序。  
Lower values mean that the entity is *closer in front* and updates *later*. Higher values mean that the entity is *further back* and updates *earlier*. Madeline uses depth 0.  
较低的值意味着实体显示时*更靠近前面*，并且*更新时间较晚*。较高的值意味着实体显示时位于*更靠后*的位置并*更早*更新。Madeline 使用的深度为 0。

## Flag Gate Container

  
flag硬币门，我这边不讲硬币门模式了（反正可以用依附控制器加普通硬币门做……允许吧）   
先放一个警告声明，尽量别试着移动一些砖类实体什么的，他们没有做移动适配导致贸然移动它概率导致爆破游戏（

flag硬币门的应用场景非常广泛，无论是搭配flag硬币做各种实体的硬币门，还是使用它做一些触发事件让某个实体出现/消失都都非常好用

先基本说明几个不是让你打勾的常见选项

easing（运动模式，可以参考本网站https://easings.net/）

icon， color（参考普通硬币门）

move flag

shake（运动开始前的原地震动，有两个属性）

move time

whitelist

can return（触发后关掉flag是否会原路返回）

icon visible和play sounds（能不能看见图标，能不能听见声音）

有这几个基本就可以把大部分实体当硬币门用了

## Flag Toggle Modifier



实体功能控制器，可以控制实体的各种功能

在包含目标实体后，你可以用那个Flag属性来启动它（没有flag设置即默认启动）

注意后面三个Toggle属性，toggle active是实体自身是否处于活跃状态，非活跃状态实体不会接受任何更新信号，toggle collidable是否接受玩家的碰撞，toggle visible是实体可见性

一般都是把前面两个关掉写上白名单放在对应位置，然后隐藏某个实体

## Floaty Container

  
月亮块控制器，让实体变成月亮块的样子，简单明了中间六个设置月亮块属性，disable spawn offset让月亮块固定开始位置防止你掉下单向板

ignore container bounds让全图目标实体可以像月亮块一样

## Global Modifier



⭐全局属性控制器，可以让选定的实体在整张地图都保持加载，而不只是在当前房间加载。

属性列表（这么短你直接全部看完吧）

Whitelist：容器包含实体的白名单，写入你想控制的实体的entity type

frozenUpdate：开启时，实体在游戏冻结状态下仍会执行更新逻辑（如粒子效果、动画等）

pauseUpdate：开启时，实体在游戏暂停状态下继续更新（如电网的运动）

transitionUpdate：开启时，实体在场景过渡期间（切板）保持活动状态（如持续播放的环境音效）

实践案例