10-22

对象池模式

当角色现需要一次释放许多火球，并在之后销毁他们时，如果真的生成并销毁，那对cpu的开销将是巨大的。

所以我们使用对象池模式，在游戏一开始的时候就创建这些对象，只是是否使用他们的时候对他们的“是正在使用”属性进行修改。这样可以极大地减少cpu的开销，不过会增大内存的开销。

现在我们对角色的部分技能进行修改。

角色的一技能是释放一个扇形的区域的水球散弹。四技能是水球激光，五技能是水雨。我们实现技能是通过SpawnActor来实现的，水球技能是生成BP\_水球，水雨技能是生成BP\_水雨。



所以我们先关注水球技能。

按照PooledActor要求，应该首先修改BP-水球，添加一个bool型变量，命名为“正在使用”。

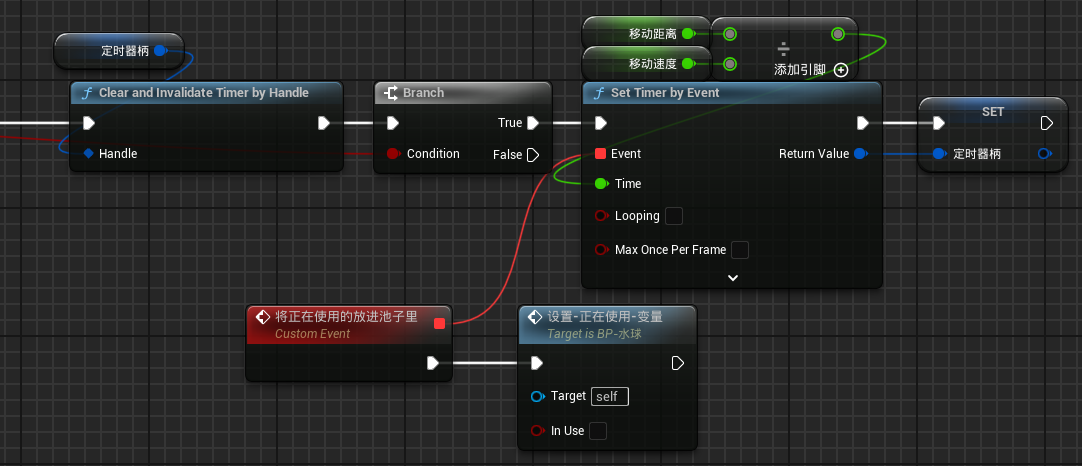
但是我们实现通用性效果，不在BP-水球上修改，创建BP-池对象。

添加一个bool型变量，命名为“正在使用”。添加float型变量，命名为存活时间

同时给他创建一个函数（或者自定义事件），命名为“设置变量-正在使用”，添加bool型形参InUse，对正在使用变量进行赋值。然后，如果他被使用，我们就启用碰撞，否则就禁用他。如果没有使用，我们也会隐藏它，或者以其他方式展示它。也有许多其他内容，具体见图片。



同时我们也要将计时器提升为变量，在我们创建一个新的之前，清除它并使其无效。



同时在EventBeginPlay时调用该自定义事件，确保任何Actor都是从不被使用开始的。



同时为实现更强的通用性，我们设计：所有发射物都是BP\_池对象的子类。在BP\_池对象中创建可能用到的变量，比如：创建者。而且还要可编辑实例、生成时公开。

上面是实现了PooledActor，也就是对象池里面的对象。接下来实现对象池。我们通过ActorComponent实现ActorPool

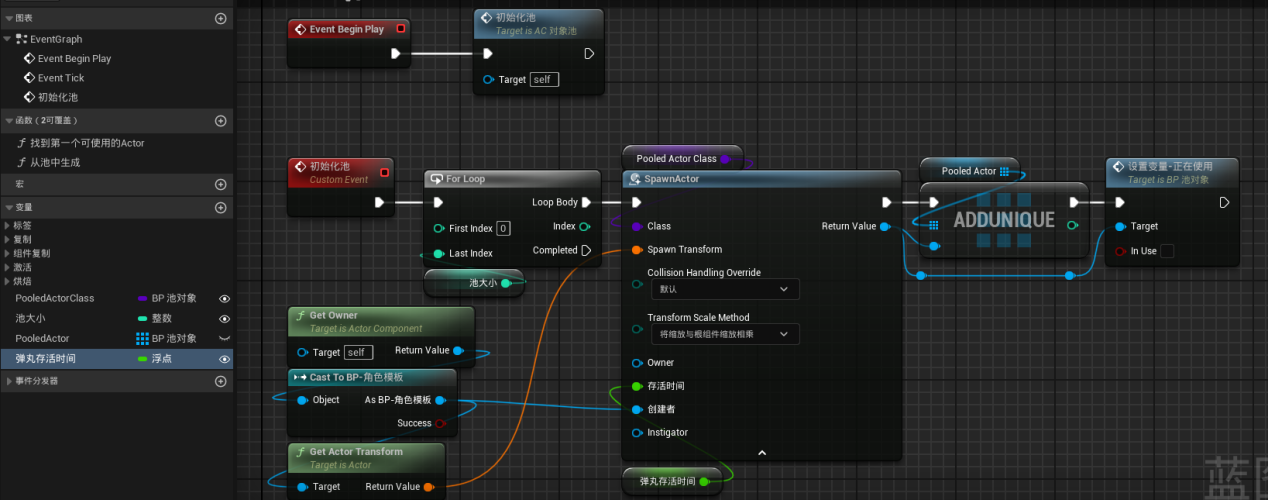
创建AC\_对象池，添加变量PooledActorClass，类型选择为BP-池对象的类引用，仅勾选可编辑实例。

添加变量“池大小”，int型，可编辑实例。

添加变量PooledActor，类型选择为BP\_水球的对象引用的数组。

接下来实现ActorPool的功能

首先是初始化池

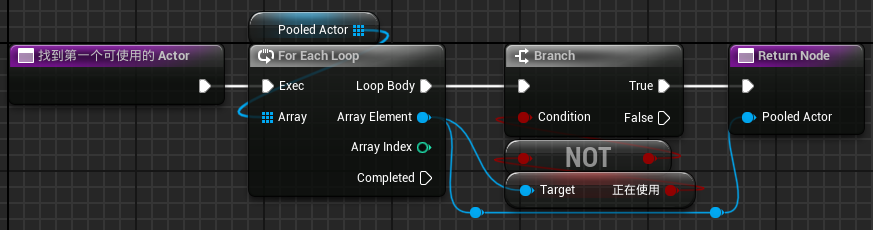


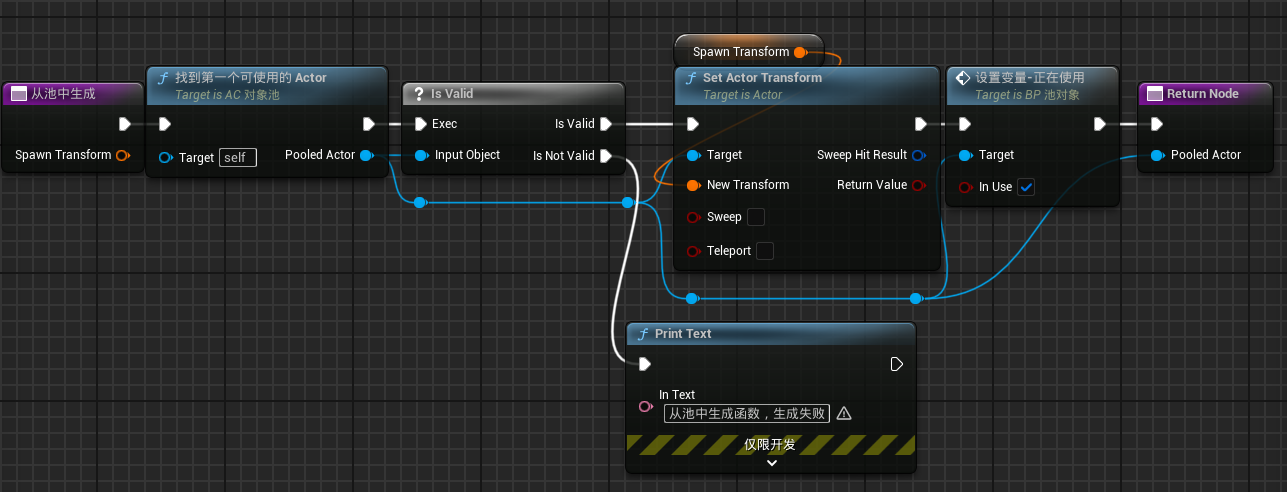
然后实现两个函数帮助我们在池中检索参与者

创建函数，命名为“找到第一个可使用的Actor”，返回值为PooledActor，类型为BP\_池对象的对象引用。

创建函数，命名为“从池中生成”。这个函数负责将参与者设置为正在使用。函数的输入为Transform（变换）类型的形参，返回值为PooledActor，类型为BP\_池对象的对象引用。

下面就来实现这些函数。





接下来对BP\_水球进行设置

首先修改水球的父类为BP\_池对象。

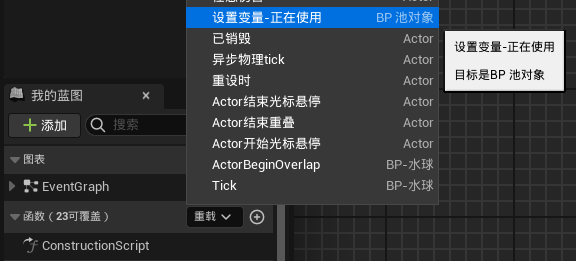
这之后需要对水球进行较多的修改。

将水球的存活时间（来自父类的变量）设置为移动距离/移动速度

在DestroyActor时不destroy，而是设置“正在使用”变量

然后是设置速度。设置速度不应该在水球的EventBeginPlay时设置。因为当Actor在池中生成时，开始游戏就会被调用。相反，我们希望它在Actor正在使用时应用。

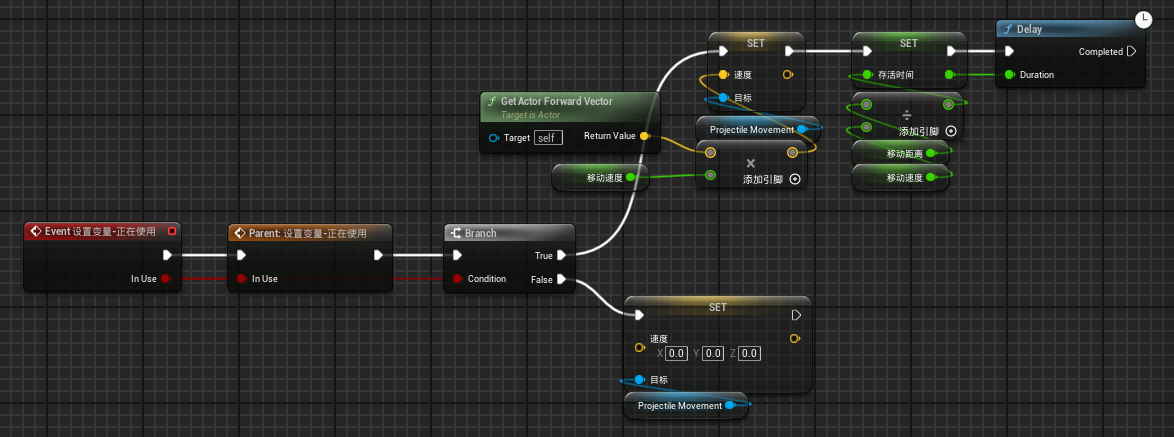
重载函数，选择“设置变量-正在使用”



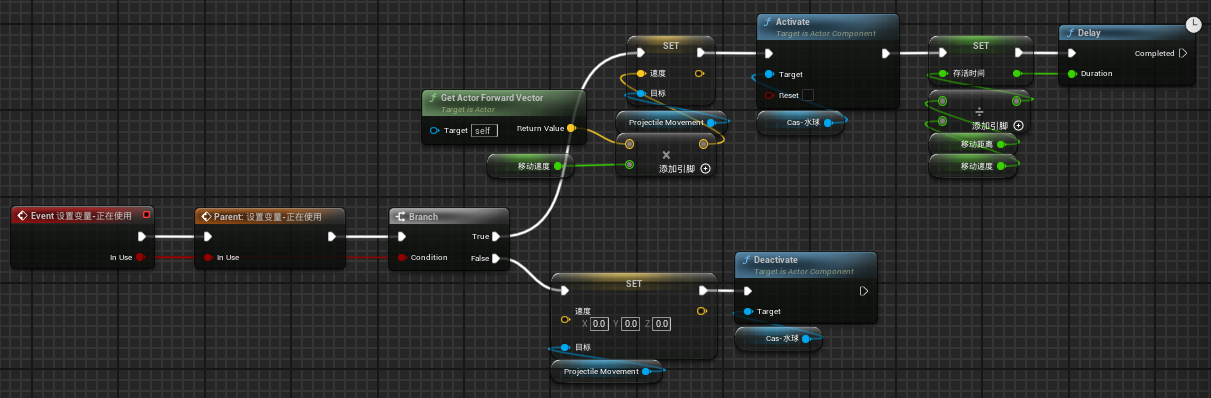


我们希望在父类函数的基础上SetVelocity，右键事件节点，选择

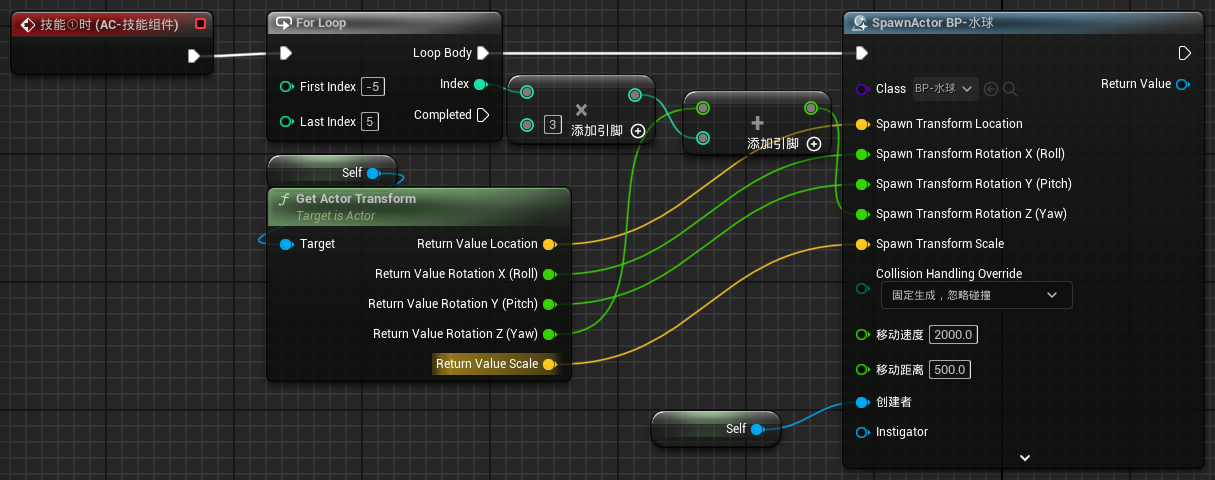




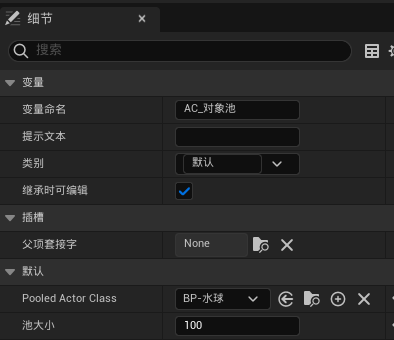
同时在InUse的不同情况下时设置粒子系统的开关



现在需要回顾壁的技能。它的一技能是放一个散弹，需要ForLoop来SpawnActor。

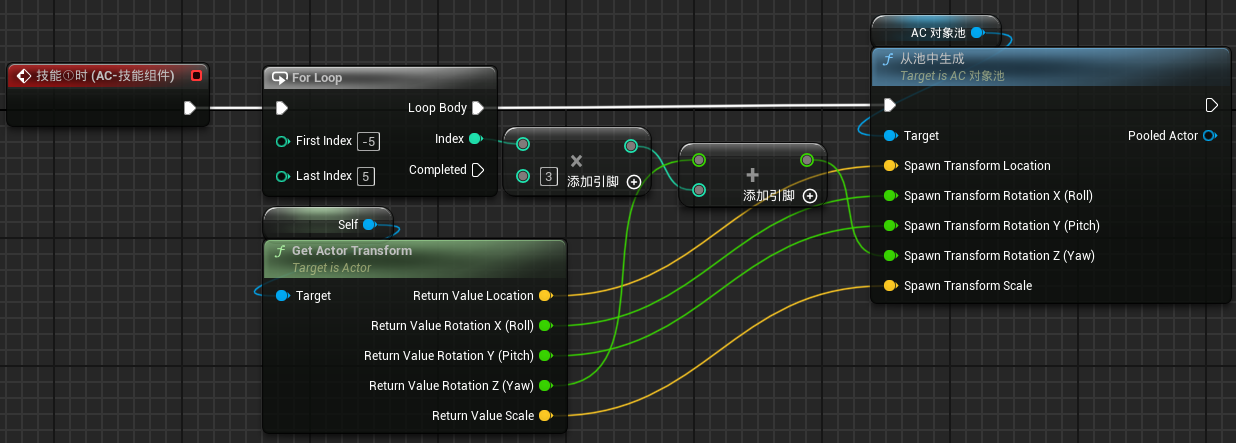


现在我们用AC\_对象池组件来实现它。添加这个组件，在细节中找到默认，会有组件中的可编辑实例的变量。



然后我们就可以用组件来替换SpawnActor

这里给一技能进行替换。



水球的伤害目前的计算公式为“伤害=发起人攻击力”

因为在初始化池的时候给创建者赋值了，所以这里的水球能完美的接收到父类成员的赋值。

接下来也要给其他的技能进行修改。修改的比较多，这里不做展示

为更好的实现通用性和便用性

创建BP\_池对象的空白子类，我们来通过组件的函数实现对子类的设置。

并且从此之后，BP\_池对象当做所有发射物的父类，在后期可以改名为BP\_通用发射物。

创建空白子类，命名为BP\_test。

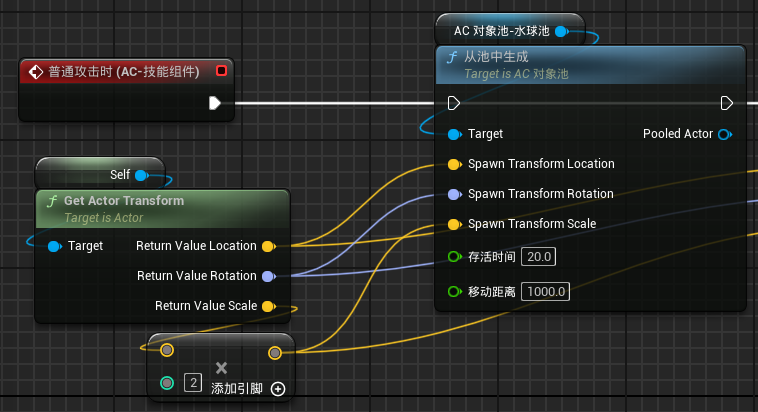
给池对象中加上发射物移动组件，重力设置为0。创建变量命名为移动距离，可编辑实例，生成时公开。

注意，因为水球中本来就有发射物移动组件，所以两个都会生效。注意找时间将水球中的该组件删除。

接下来思考我们要怎么设置

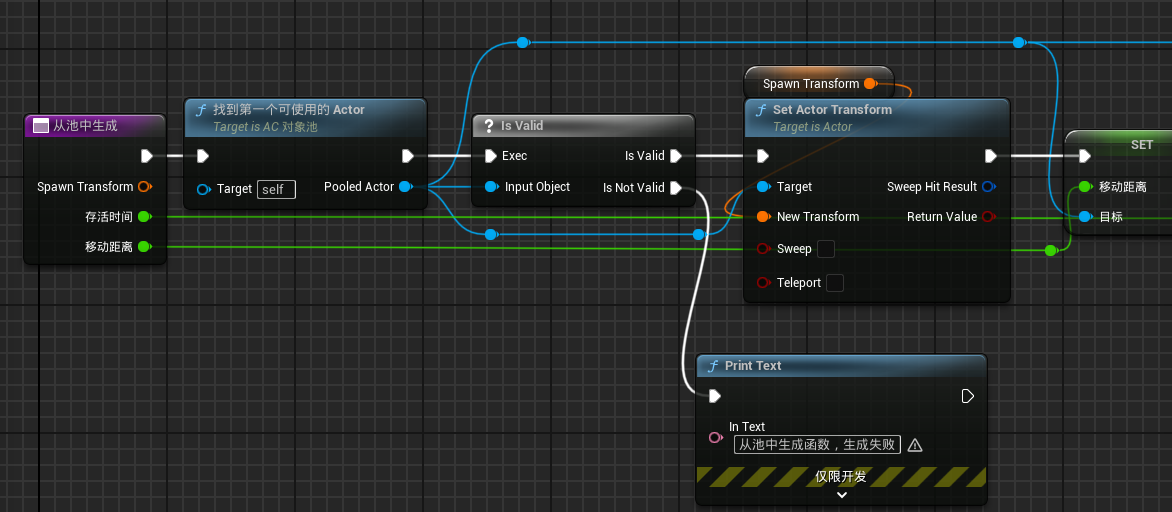
我们生成水球的时候是在BP\_角色中调用AC\_对象池的“从池中生成”函数，我们希望这个函数可以像SpawnActor那样可以通过将某个变量生成时公开来对这个变量进行赋值。

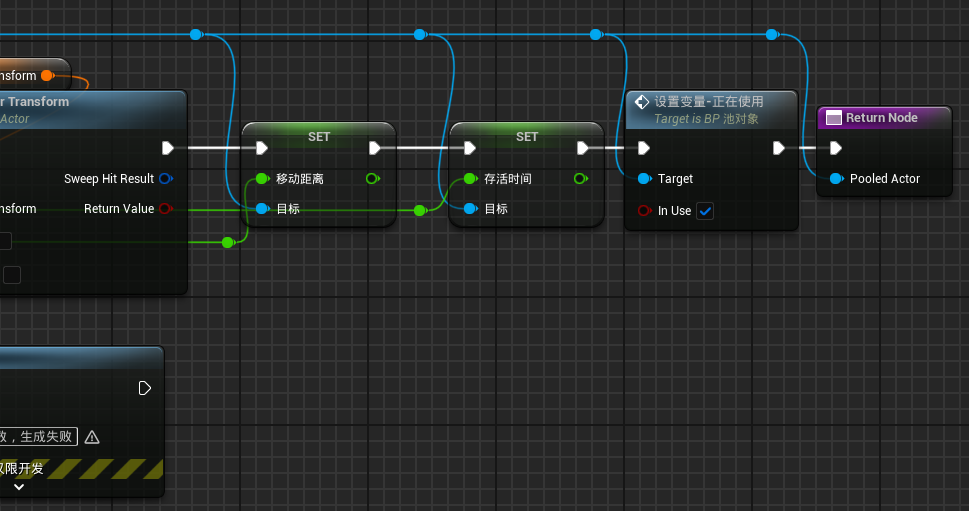
现在我们思考：我们需要什么参数来进行传递。这里我们选择的是“存活时间、移动距离”两个参数。这两个参数比较直观。



那么我们就需要对“从池中生成”函数进行修改。而且，我们也需要设置：当正在使用为true时，设置速度。

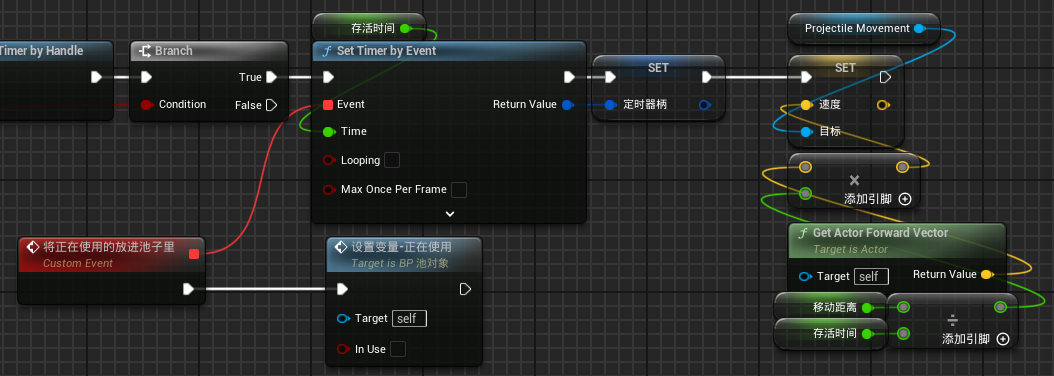
所以首先改“从池中生成”函数。这个函数应该是修改子类中的，也就是修改通用发射物的函数。





我们希望，在发射物正在使用时，就进行速度的设置。所以

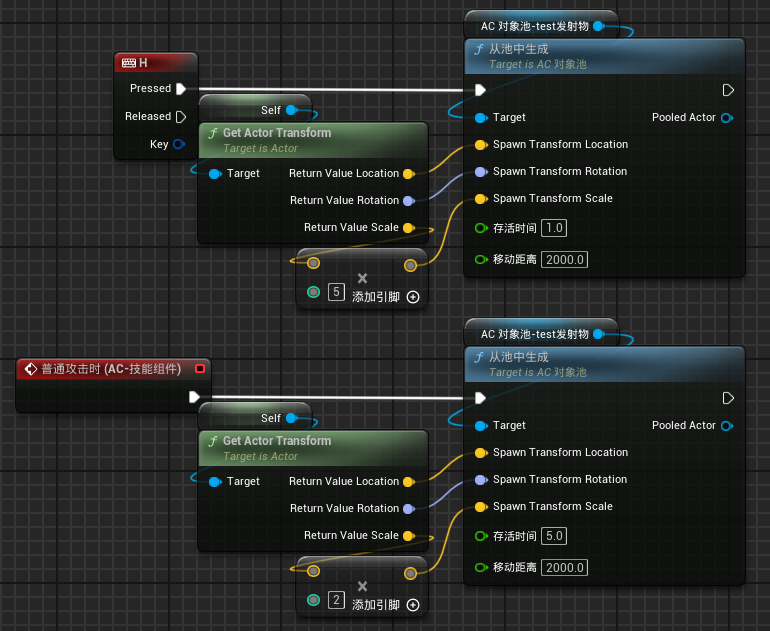
在池对象的“设置变量-正在使用”中，如果InUse为真，则SetVelocity



这样，我们如果想创建一个普通的可飞行子弹，我们只需要创建池对象的子类，给他加一个Cascade就可以了，非常的简单。

在对象池中的两个变量“弹丸存活时间和弹丸移动距离”就用不到了。

经测试，可以生成不同效果的发射物



这时会发现，即使BP\_Test什么都没有设置，也能有正确的子弹移动。

所以我们可以删除掉水球中的部分速度设置的内容。

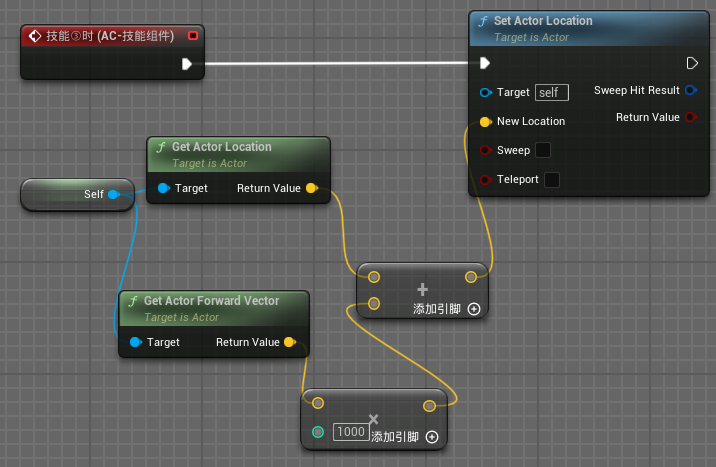
使用方法：

打开角色蓝图，添加AC\_对象池，选择AC\_对象池的实例，在细节中给默认进行赋值

优化：技能优化

闪现技能优化

原来的技能是



现在描述三类移动技能：快移技能、瞬移技能、闪现技能。

我们将为BP\_总类实现这三种技能，并封装为函数。然后把壁的位移技能改为相应的函数。

描述一下三种技能

快移技能，表示能快速向目标方向移动。如果遇到墙或某单位则停止移动。

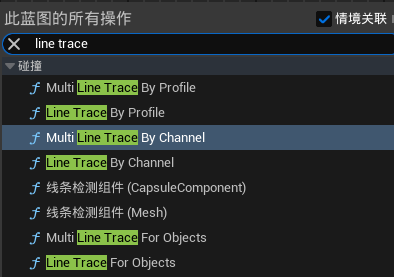
瞬移技能则表示快速向目标方向冲刺，遇到墙穿墙，遇到其他的也不会停止。

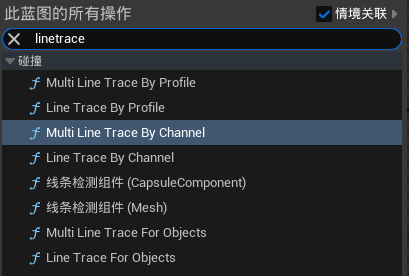
闪现技能则表示闪现到目标位置，中间不会有自己身体的实体。

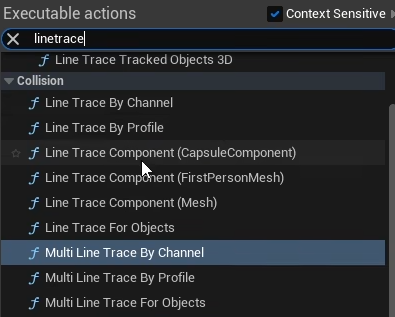
而对于快移技能，在碰到某类单位（如英雄）时停止移动，也需要来实现。

首先实现快移技能。我们要用射线的方法进行检测。

射线，搜索line trace即可看到多种射线检测的方法。





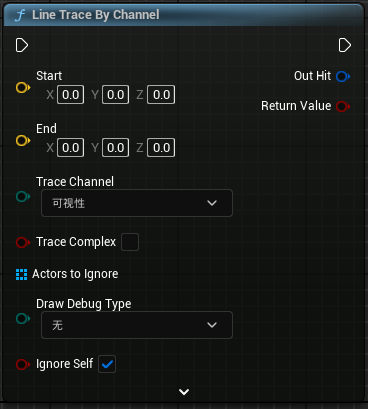


Line trace by channel是指射线射出之后检测到的第一个碰撞到的物体

Line trace by profile是指检测特定的物体

Line trace component就是按组件来分

这里我们使用by channel，只要中间碰到有物体就返回。



函数需要传递start和end两个参数。

Start即为Actor location

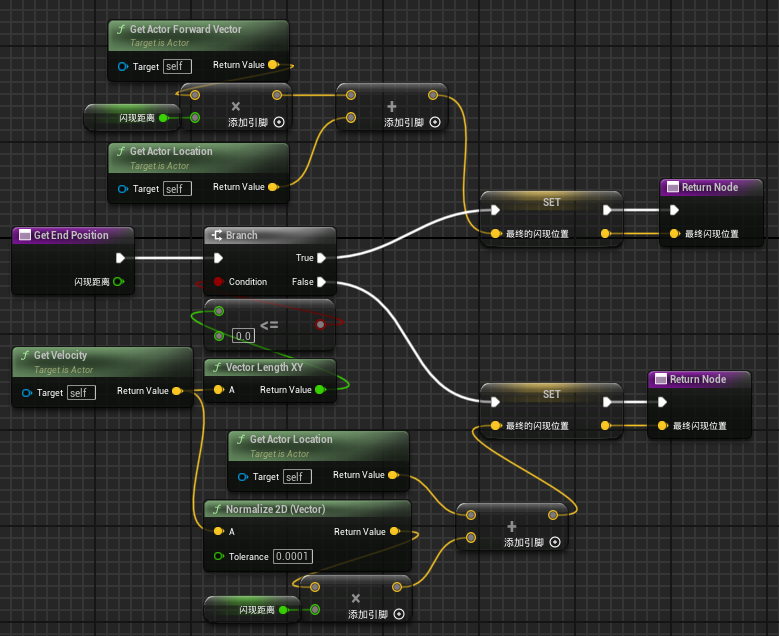
End为位移终点。

在第一人称和第三人称下，位移终点要考虑位移方向，位移方向应该是移动的方向，当没有移动的时候，位移方向为朝向方向。

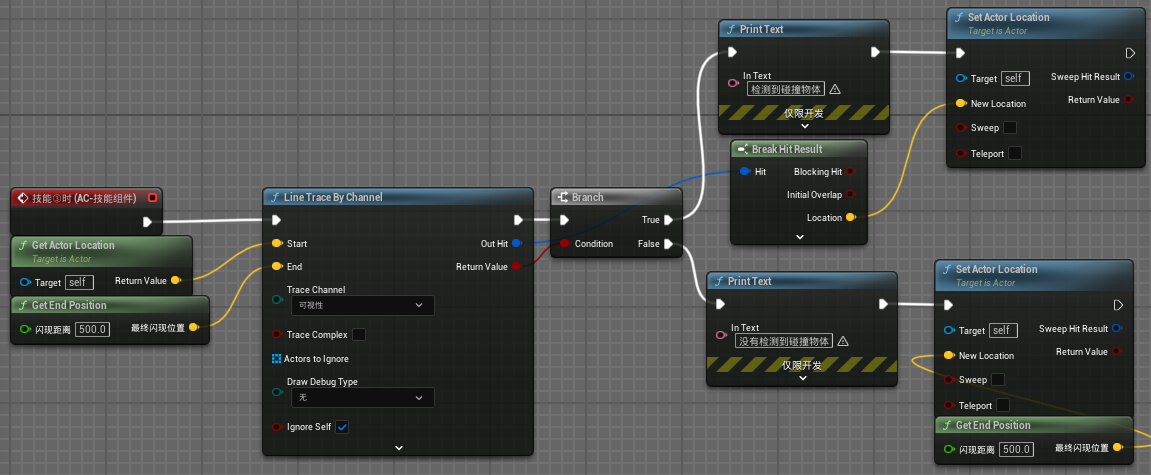
在上帝视角下，位移方向与上面同理，不设置为右摇杆方向。

所以我们需要一个方法来获得EndPosition。纯函数。

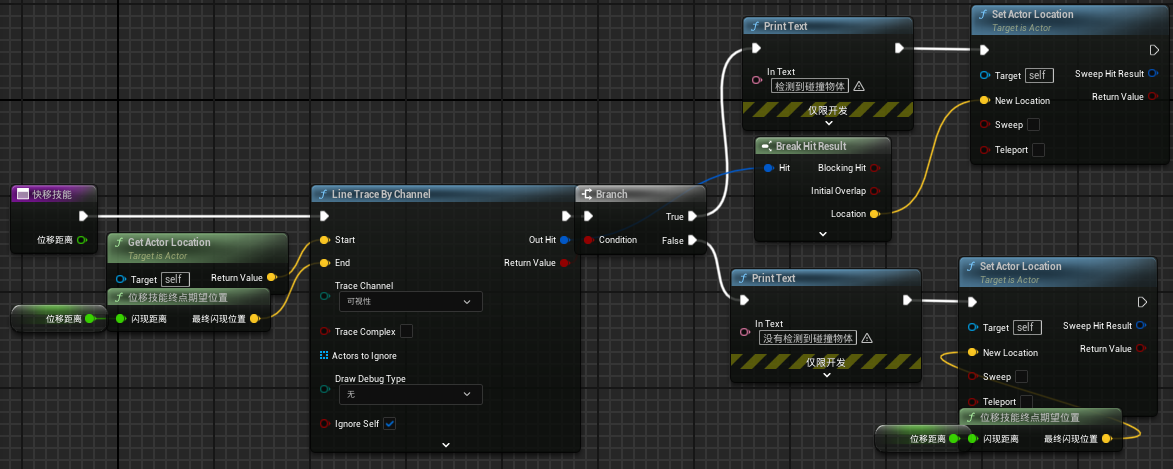
函数命名为“位移技能终点期望位置”。

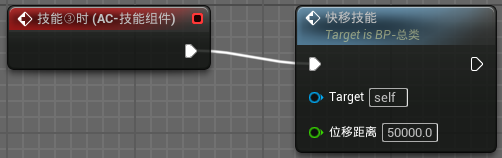


然后配置射线检测节点，并根据射线检测结果进行判断



成功。将上述节点封装成函数。





闪现技能的实现则比较简单



现在实现下面的效果：（潘森Q）

瞬间释放技能：快移技能

蓄力后释放技能：闪现技能