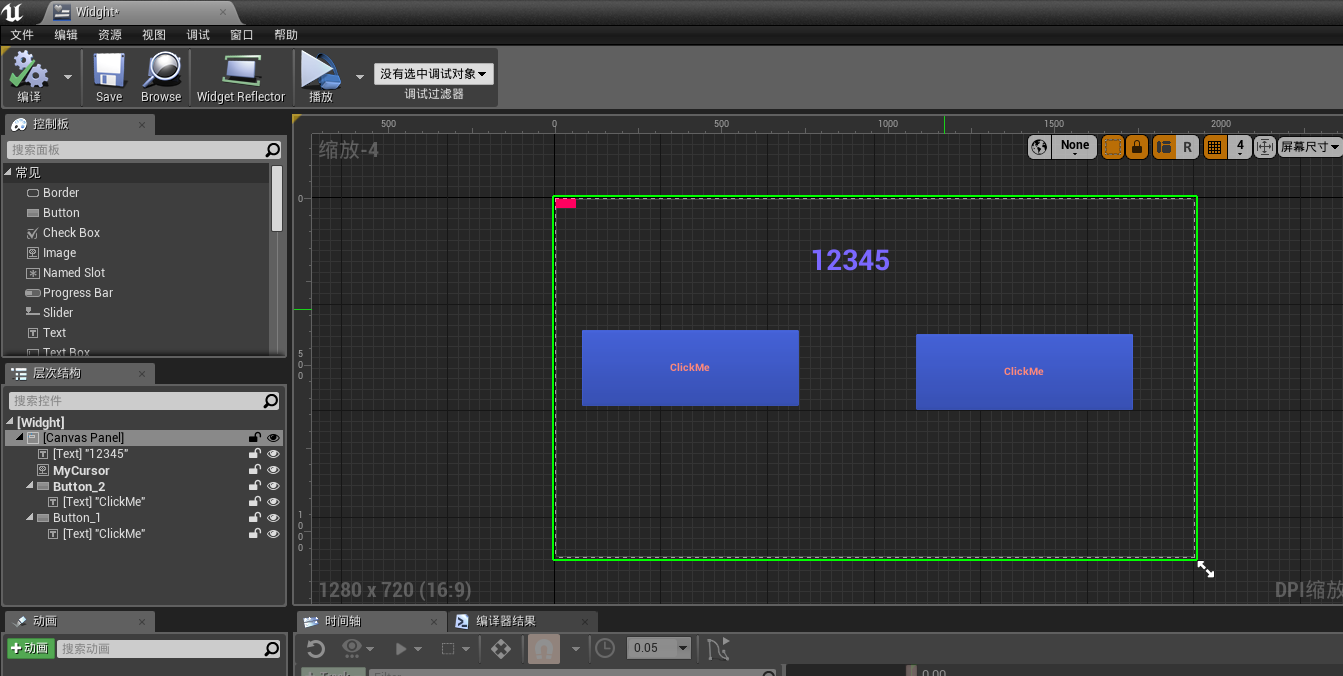
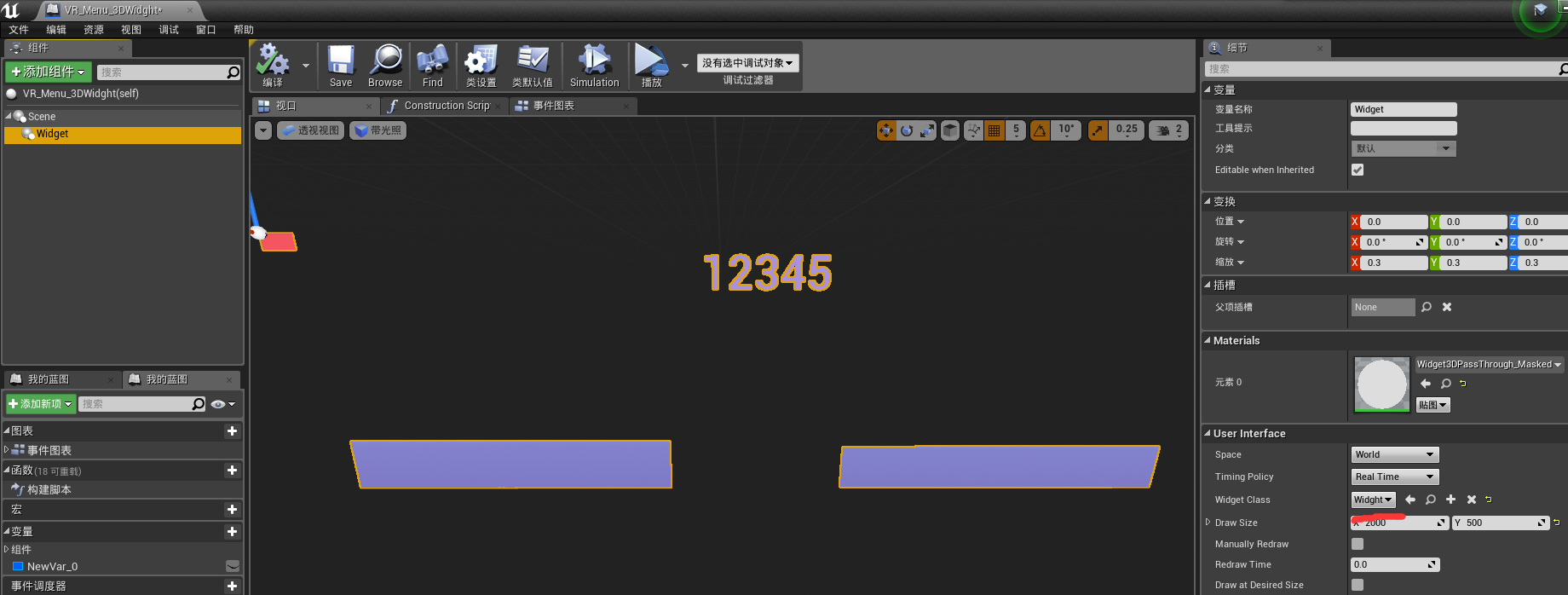
普通的控件通常可以直接使用按键事件直接触发，但是在VR模式下我们使用手柄无法直接触发控件的按键事件（至少我没找到），如果想要触发的话需要用射线照射到点的位置与控件的位置进行判断，看看这个点在没在控件的范围内，下面来看制作的详细步骤

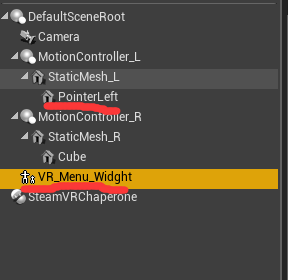
1. 创建一个Pawn类，Actor类，一个Widght类。
2. 在Widght类中搭建控件，有一个额外的image命名为MyCursor，用来表示射线指向的位置，制作完后可以隐藏或者改为装饰物

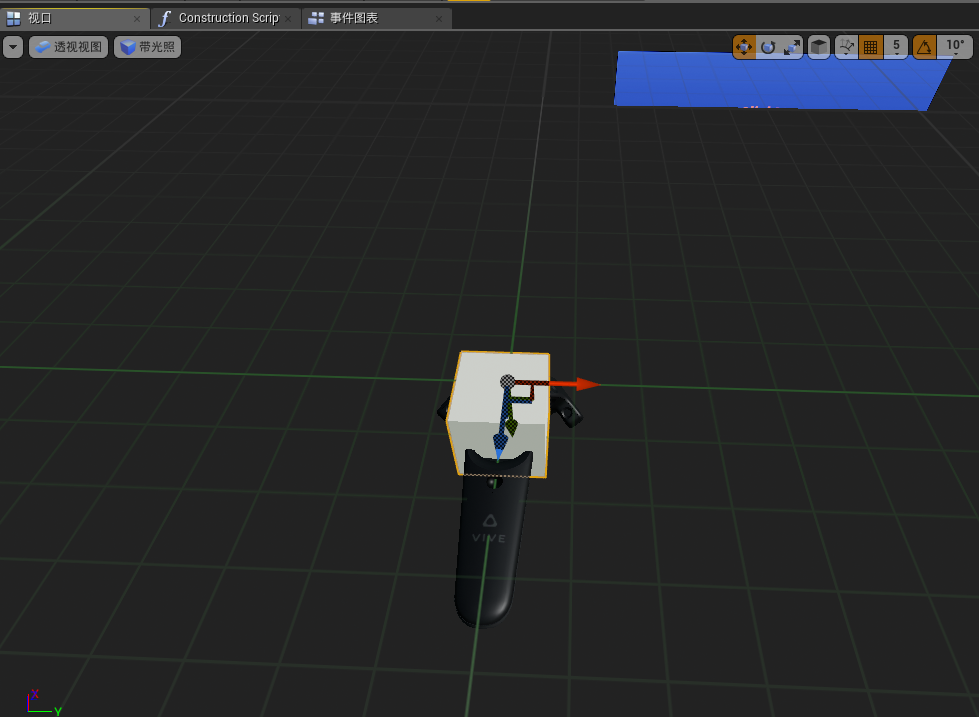


1. 将Actor类命名为VR\_Menu\_3DWidght，引入刚刚创建widght类

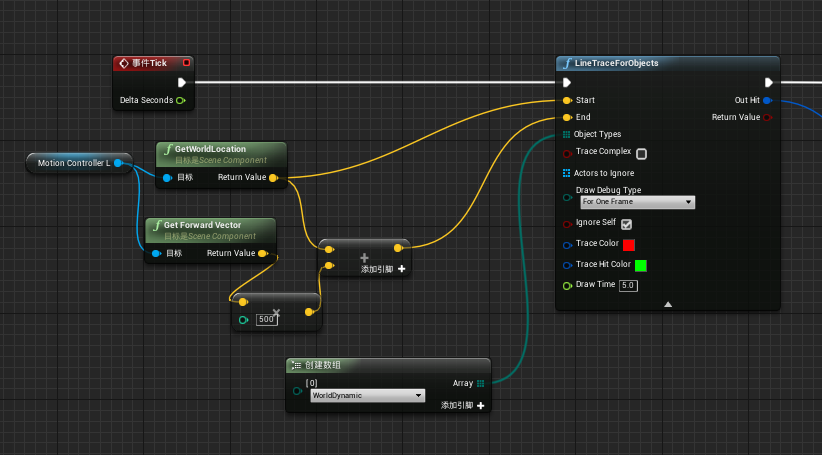


1. 在pawn中构建VR的功能，在左手上额外添加一个Mesh网格体PointerLeft用于制作时的标识，制作完后可以隐藏或者改为装饰物，子组件VR\_Menu\_Widght

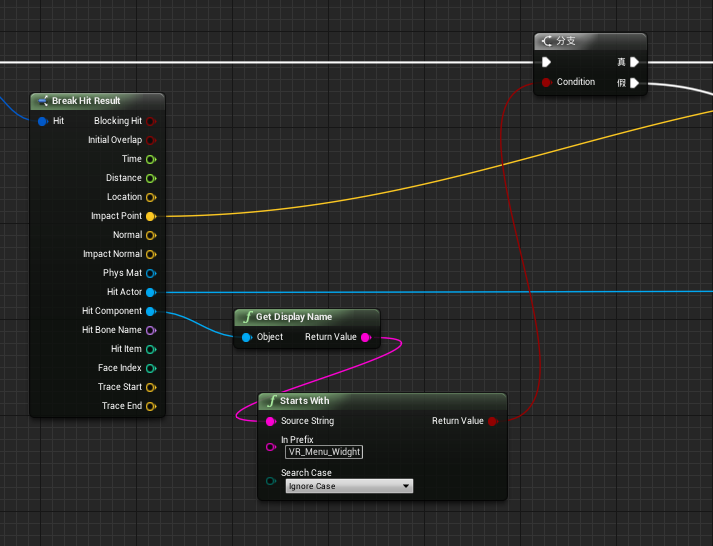




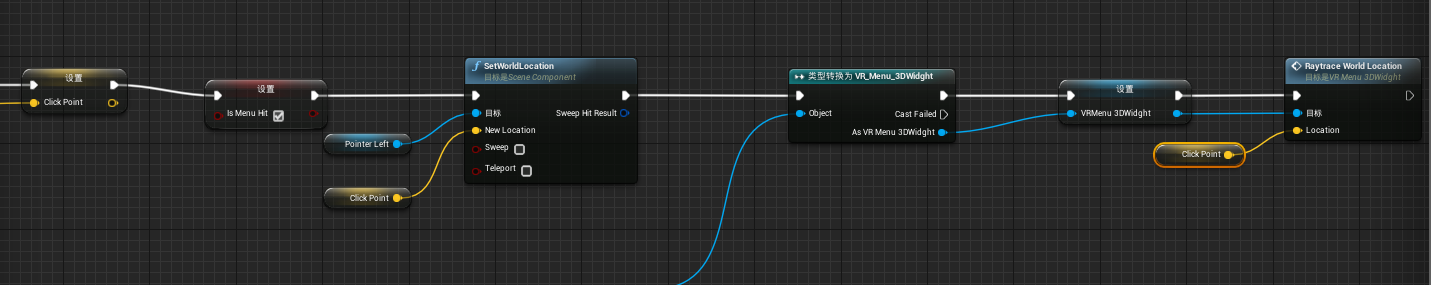
1. 在左手绘制出一条射线



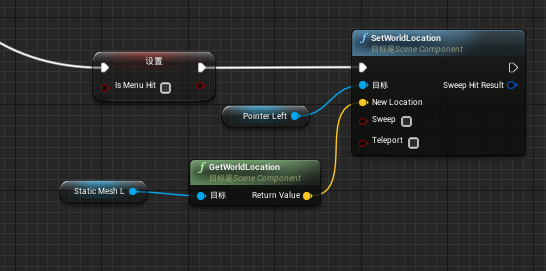
1. 将碰撞信息展开，判断射线是否指向这个子组件



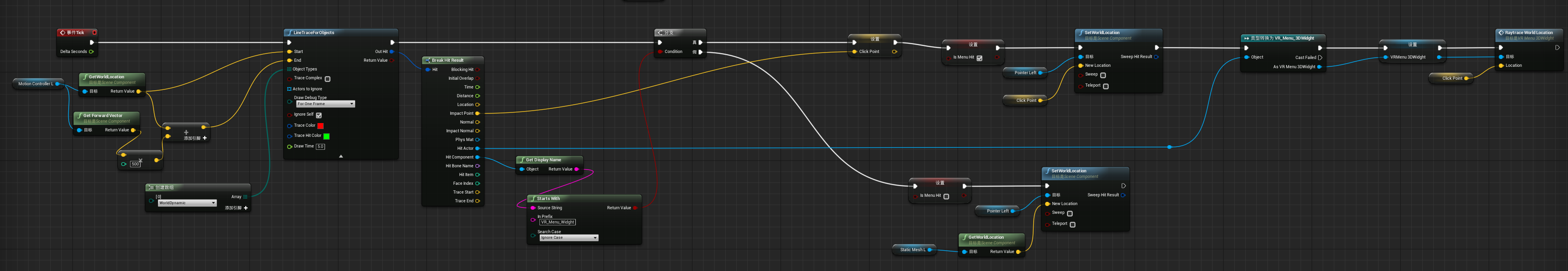
1. 如果指向了这个子组件，那么就设置碰撞布尔值为真，保存点的位置，设置左手上的网格体的位置，进行类型转换，转换成子组件的类型



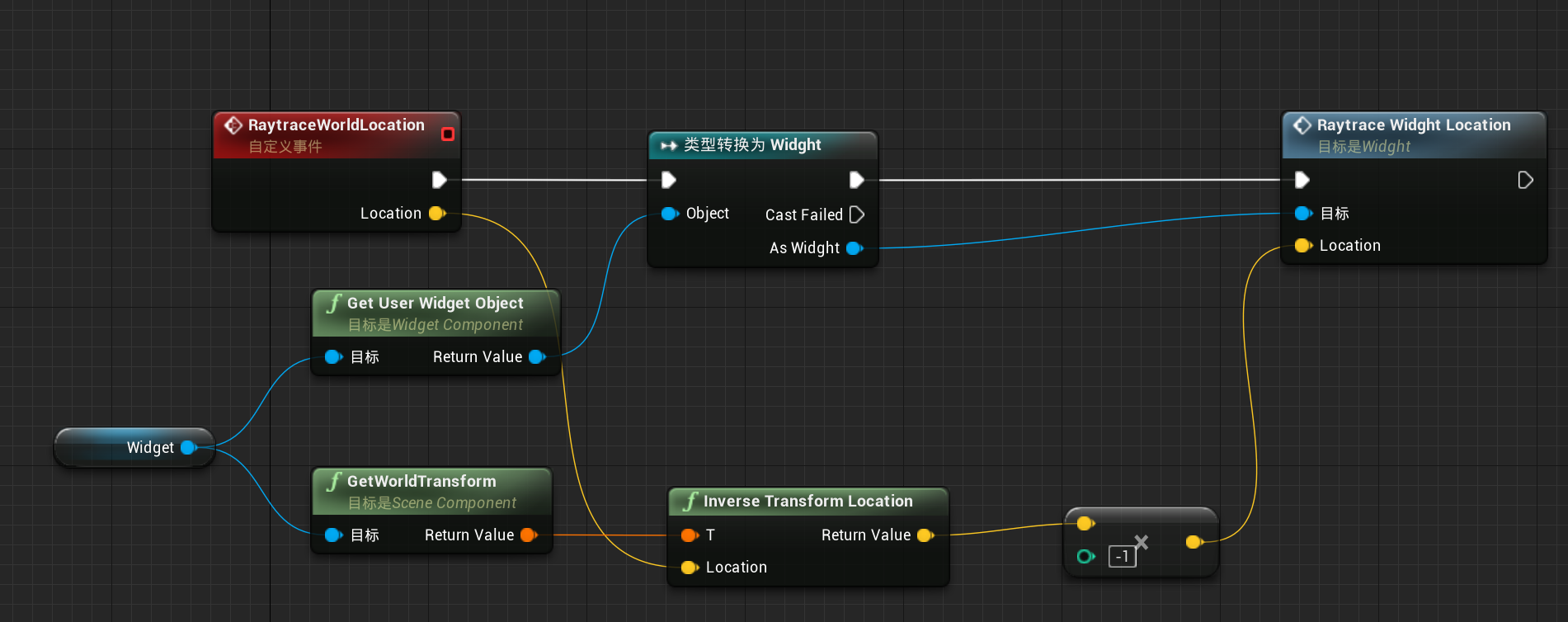
1. 如果为假，那么就设置碰撞布尔值为假，将网格体的位置还原



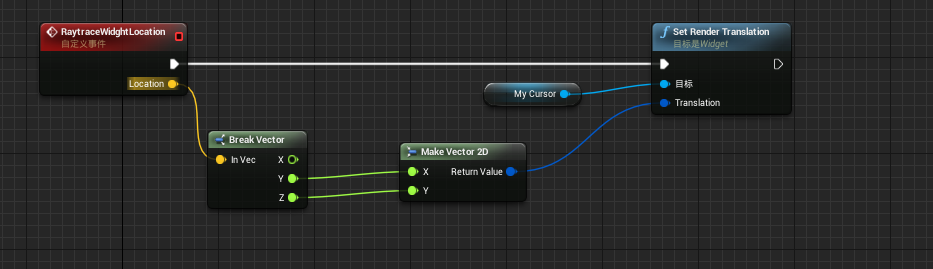
1. 总览



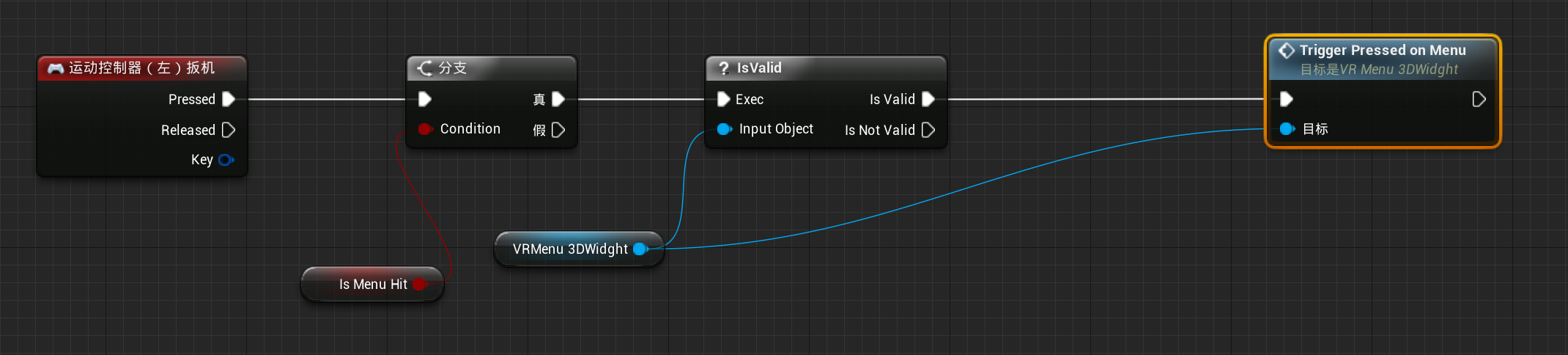
1. 在VR\_Menu\_3DWidght中编写自定义事件，将指向的世界位置转换为widght中的相对位置，以及类型转换为Widght



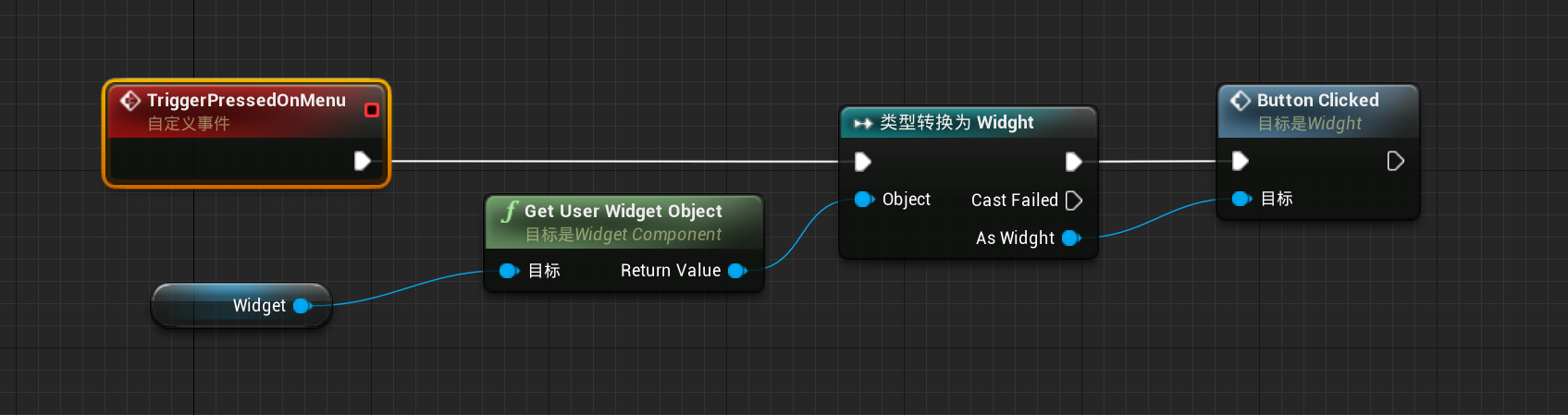
1. 将位置传递给MyCursor，因为Widght是一个平面，它的相对位置与X轴无关，所以舍弃，现在我们已经可以获取到射线指向widght的位置了



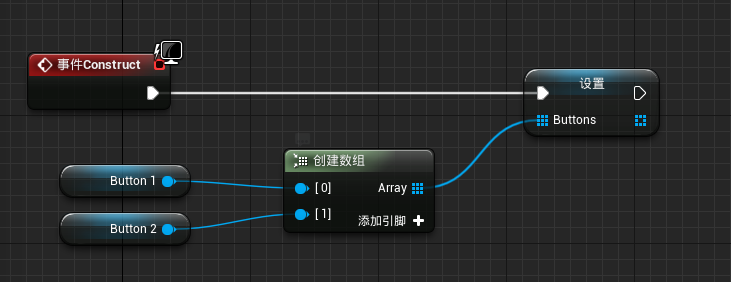
1. 扣下扳机时先检测有没有指向这个子组件，然后判断类型转换是否正确，调用VR\_Menu\_3DWidght中的TriggerPressedOnMenu自定义事件



1. VR\_Menu\_3DWidght中的TriggerPressedOnMenu自定义事件，进行类型转换，然后调用Widght中的ButtonClicked自定义事件



1. Widght中的Construct事件会将所有的按钮保存在一个数组中



1. ButtonClicked自定义事件，将每一个按钮的位置同射线指向的位置依次比对，符合位置信息的将会执行对应程序，不符合的会重置按钮颜色

