

STEPVR SDK UE4 插件使用教程

1 系统环境

1.1 硬件

Oculus, StepVr, PC

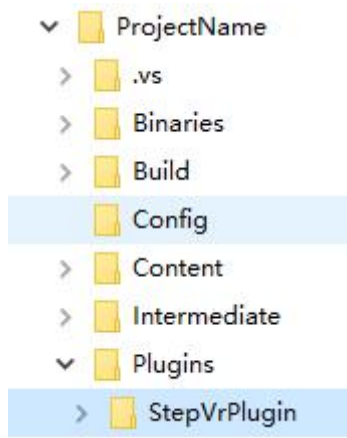
1.2 软件

Windows 10 64bits, Unreal 4.10 或以上

2 配置插件

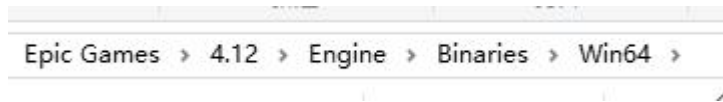
2.1 插件位置

将解压后的 **StepVrPlugin** 文件夹 放到目标工程的 Plugins 目录下 (如果没有 Plugins 文件夹 , 需要在 Content 同级目录下创建)

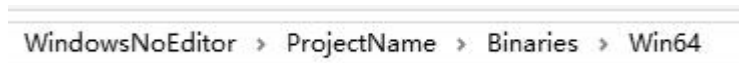


2.2 矩阵文件

开发时，将标定的**矩阵文件 TransMat.txt** 放到对应引擎版本路径下，所有工程都通用。如下图：



项目打包后，需将**矩阵文件**复制到相应的工程目录下，如下图：

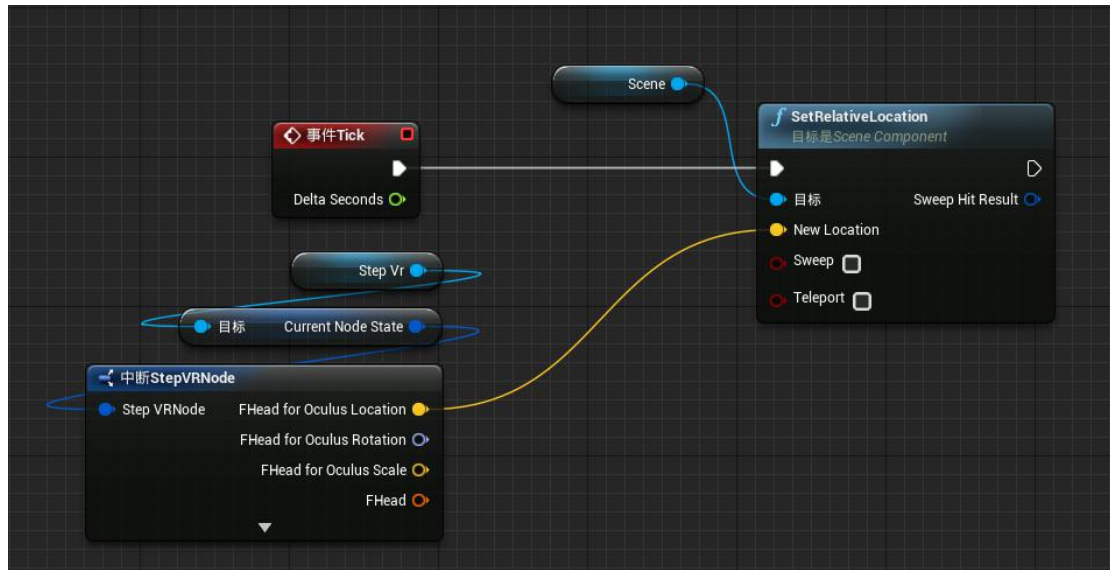


2.3 使用插件

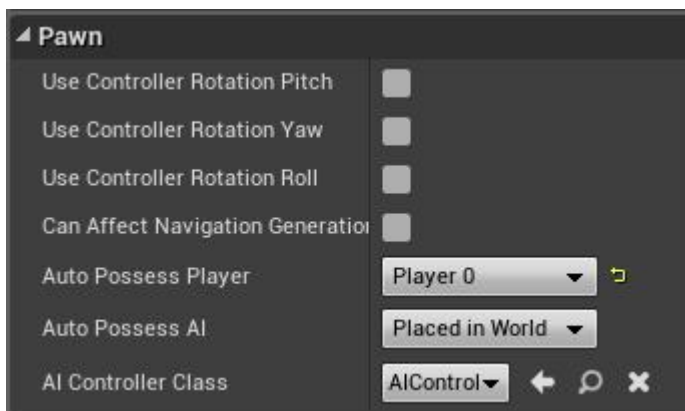
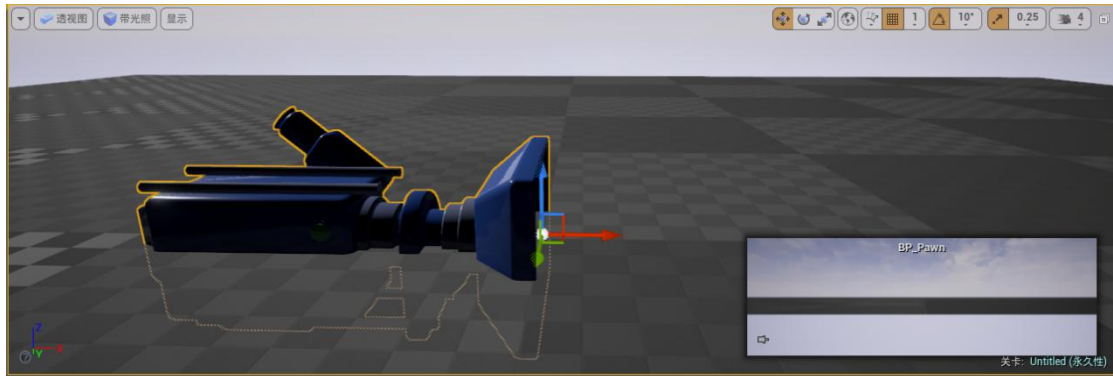
2.3.1 创建蓝图[BP_Pawn]继承 Pawn,打开蓝图，点击添加组件（左上角绿色按钮），添加 StepVr，Scene，Camera（**Camera 为 Scene 的子节点**）如下图：



2.3.2 添加 **EventTick** 事件，设置相机位置：



2.3.3 将蓝图[BP_Pawn]拖到场景中，**Z 轴**设置为 **0**（或者紧贴在地面），然后将 Pawn 的 **AutoPossessPlayer** 选择 **Player0**:

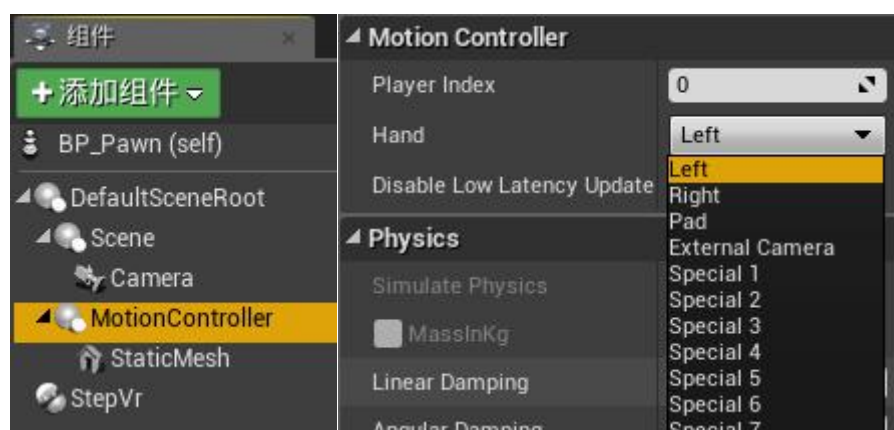


2.3.4 完成以上步骤，人物可以自由行走，选择虚拟现实预览并运行查看效果：



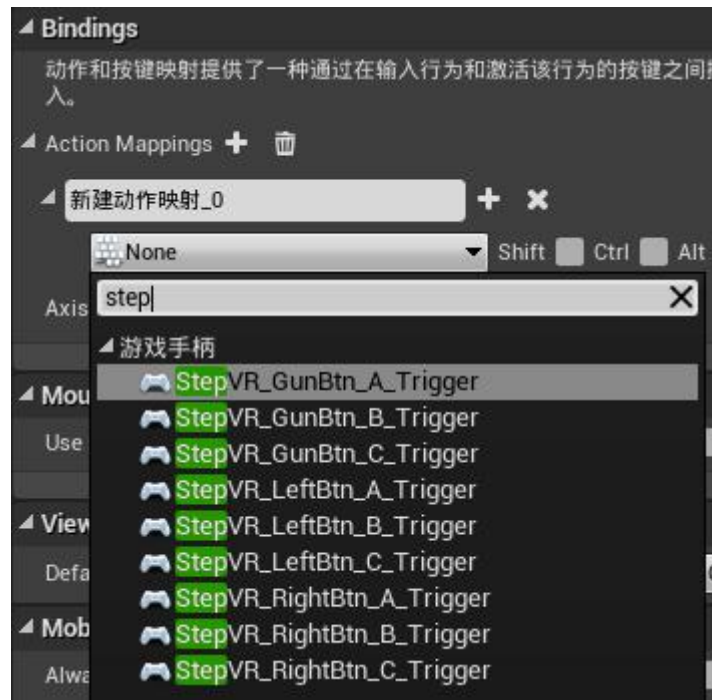
2.4 设置 MotionController

在 [BP_Pawn] 蓝图中添加组件 MotionController，StaticMesh(StaticMesh 为 MotionController 的子节点)，并设置运动控制器和 StepVr 设备的映射关系（左手柄：Left，右手柄：Right，枪：Pad），再次运行即可看到效果：



2.5 获取手柄按键事件

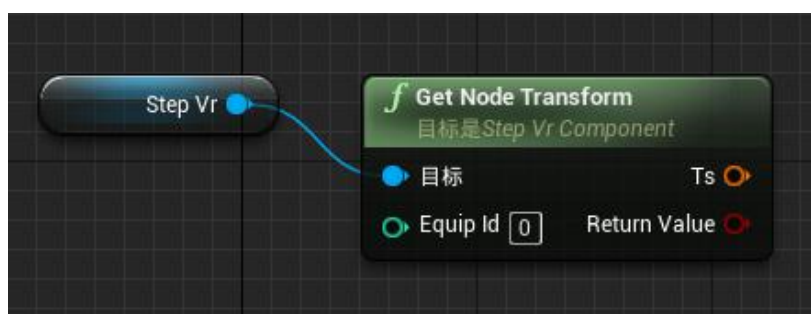
选择：设置→项目设置→引擎→输入→Bindings→Action Mappings，添加动作映射



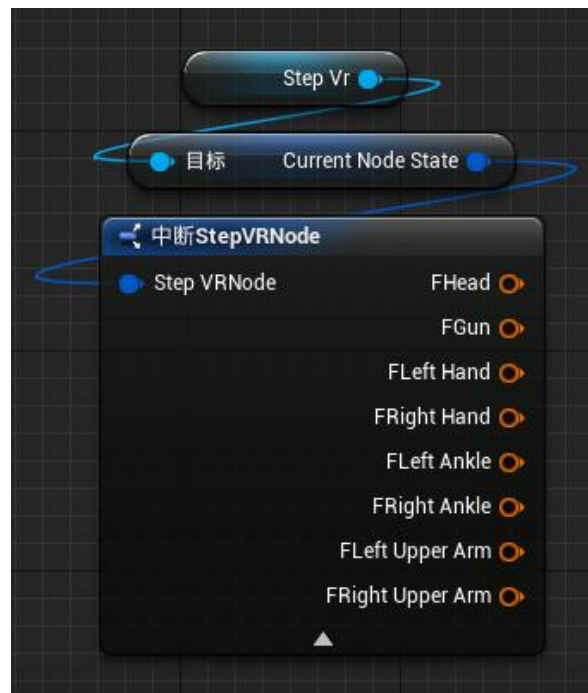
3 插件其他使用说明

3.1 获取其他设备 Transform 信息方法

通过设备 ID 号获取节点信息，GetNodeTranform()



通过 StepVr 组件，获取 CurrentNodeState 成员变量，查看 FStepVRNode 结构体，获取相应的节点数据，如下图：



3.2 手动重置头显偏移（正常启动无需设置）



4 使用注意事项

4.1 注意事项

启动游戏时会自动矫正头显正方向，此时如果设备处于被遮挡状态时启动，可能造成矫正有误差，最好在无遮挡情况下启动游戏，如有误差，可以使用 3.2 方法手动校正

插件纠正 Oculus 偏移会重置 ControllPawn 朝向（朝游戏的 X 轴方向），

使用前，需先插上 AP 等设备，等设备指示灯闪烁后再启动 UE 编辑器或开始独立游戏（如拔插 AP 需重新启动），否则可能获取不到设备数据

4.2 游戏无数据情况处理

1 首先关闭 UE4 编辑器来关闭插件，打开 XCOM 串口工具，看是否有数据变化，如无变化为设备连接问题

2 打开编辑器会启动插件，保证串口不被占用，否则插件无法正常启动获取数据（打开 2 个带有插件的工程，第二个无法打开串口，导致无法正常获取数据）

3 查看设备 ID 号与插件的 ID 号是否对应

4 检查矩阵文件 TransMat.txt 是否放在正确的位置（查看步骤 2.2）