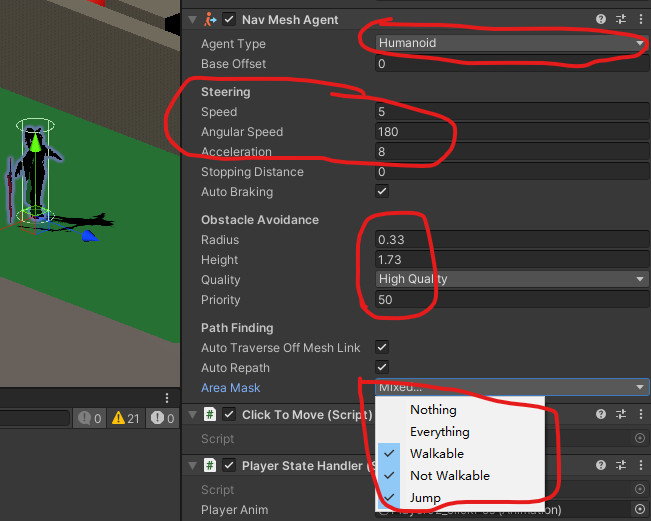
### NavMesh相关组件简介与使用

* NavMeshAgent



* 给场景中需要自动导航寻路的玩家或怪物添加NavMeshAgent组件，主要参数介绍：

AgentType：代理类型，不同类型的agent导航时对应不同的导航网。

Speed：在导航网上的移动速度

AngularSpeed：转向角速度

Acceleration：从静止到移动时的加速度

Radius：半径 （**与设置的Agent类型Radius保持一致**）

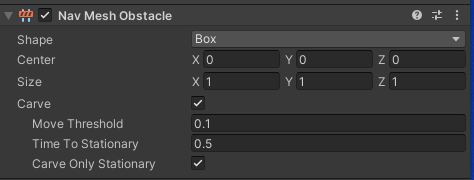
Height：高度（**与设置的Agent类型Height保持一致**）

Quality：避障质量。如果有大量代理，可以通过降低避障质量来节省 CPU 时间。设置为None时只会解决碰撞，但不会尝试主动避开其他代理和障碍物。

Priority：优先级，值越小优先级越高，优先级低的给优先级高的让路。

AreaMask：代理在查找路径时考虑的[区域类型](D:/%E8%B5%84%E6%96%99/UnityDoc2019.3/UnityDoc2019.3/Documentation/en/Manual/nav-AreasAndCosts.html)。可设置特殊区域禁止某种代理类型通行。

* NavMeshObstacle



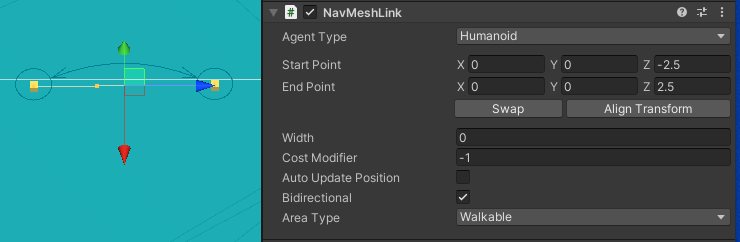
障碍物，**勾选Carve**（不勾选某些情况下得不到正确的导航路径），当物体静止时会重新生成导航网数据，有一定计算消耗，一般用于可动态销毁的物件，比如路障之类的。

MoveThreshold：当导航网格障碍物移动的距离超过移动阈值设置的距离时，Unity 将其视为移动。

TimeToStationary：物体停下来后多久后unity视其为静止。

Carve Only Stationary：**勾选**，障碍物仅在静止时重新计算一次导航网数据。

* OffMeshLink(略，使用新组件NavMeshLink件替代)
* NavMeshLink



链接两个点，用在两点之间跳跃。

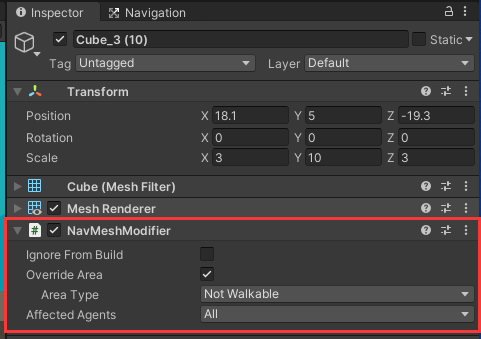
CostModifiter：参与导航计算的消耗值，（路径的总消耗值越小则为最优路径即导航结果）。

AutoUpdatePostion：不勾选。

Bidirectional: 勾选后可以双向跳跃，不勾选单向跳跃

With：设置宽度，不再局限与两个点，变成两条线上可以跳跃，比如峡谷两边可以用一个Link链接，设置宽度后，在两边的任意一点均可跳跃。

* NavMeshModifier



# 修改器，调整特定游戏对象以及它的所有子节点参与导航网烘焙的参数。

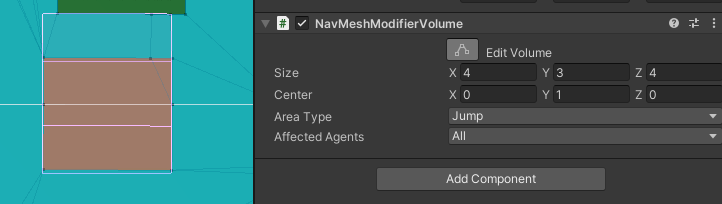
IgnoreFromBuild：构建导航网时忽略该节及其子节点物件。

Override Area Type：修改区域类型。

AreaType：选择新的区域类型，勾选OverrideAreaType后出现。

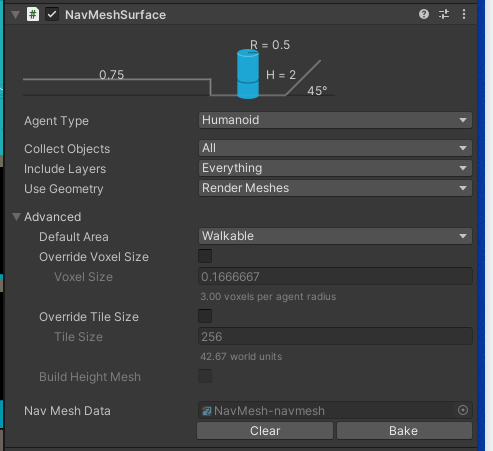
AffectedAgents：影响的Agent类型，这个组件修改的参数只会影响该类型的agent的导航网。

* NavMeshModifierVolume



功能和NavMeshModifier相似，修改的范围变成图中包围盒内的区域。只能修改AreaType。

* NavMeshSurface



为某一类型agent的烘焙导航网数据。

AgentType：代理类型，烘培的导航网对给该类型的agent使用。

DefaultArea：**选Walkable**，设置构建导航网格时生成的区域类型，除了有使用NavMeshModifier修改的地方外，其他地方的区域类型为此类型。

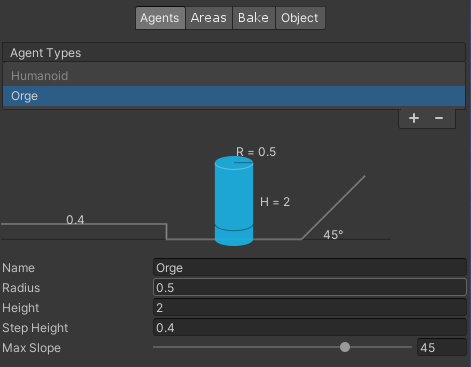
CollectObjects：烘焙时需要计算的物体，**选All**。

IncludeLayers：烘焙时需要计算的layer，选场景物体所包含的layer，一般**除了UI都勾选**

UseGeometry：**选RenderMeshes。**

**代理创建与删除**

Window->AI->Navigation->Agents



**导航网区域创建与删除**

Window->AI->Navigation->Areas

