

- Developer
- / Documentation
- / Unreal Engine ▾
- / Unreal Engine 5.4 Documentation
- / Making Interactive Experiences
- / Artificial Intelligence
- / Navigation System
- / Navigation Components

Navigation Components

Briefly describes how to use Navigation Components to modify or extend pathfinding functionality.



Navigation Components are a type of component that modifies or extends the functionality of the **NavMesh** (Pathfinding) system in Unreal Engine.

Nav Modifier Component

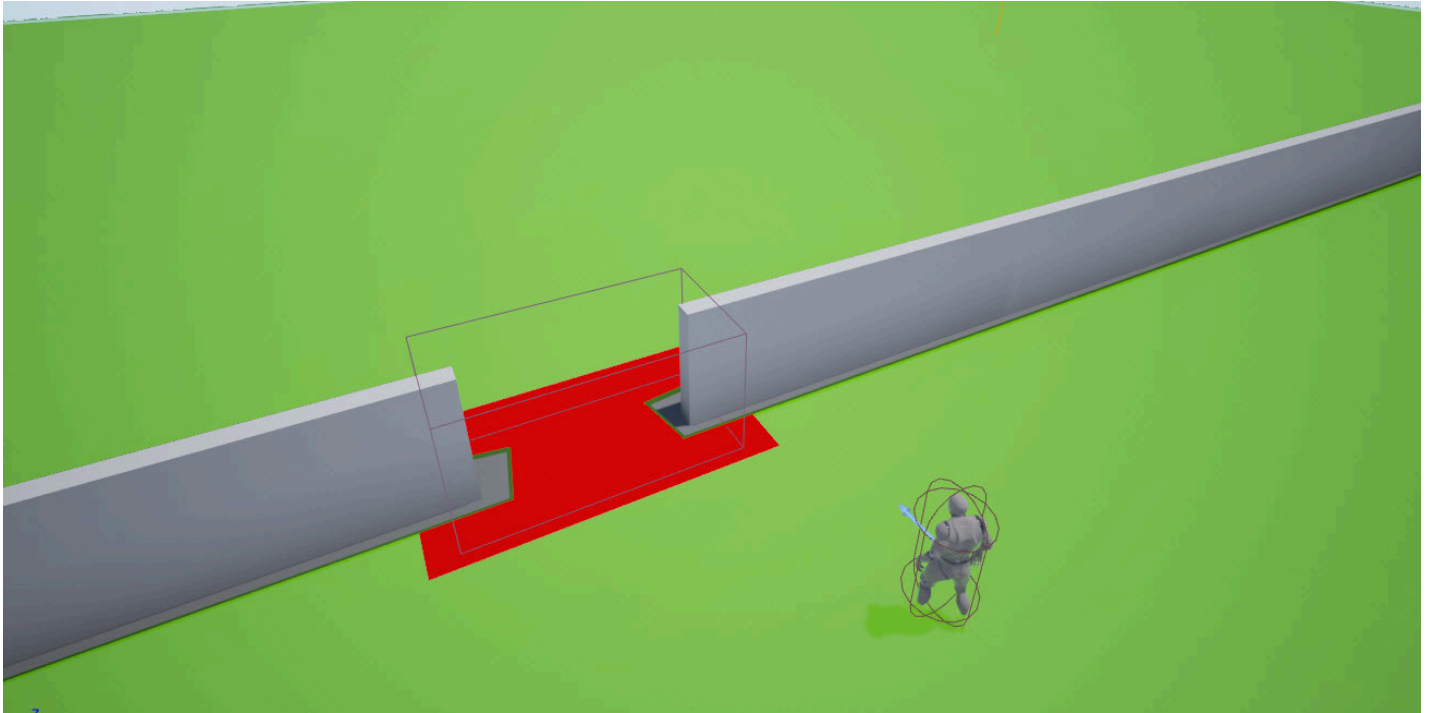
The **Nav Modifier Component** does nothing by itself, however, if you have a basic shape component as the root of an Actor, the volume of that Root Component will modify the NavMesh generation within it according to the Nav Modifier Component's **Area Class** property. You should only have one Nav Modifier Component per Actor as multiples of this Component have no effect. Also, these will appear outside the hierarchy in the Components Tab, and cannot be parented to another component or have anything parented to them.

These Area Classes can define basic things like **Cost** to enter an area or more advanced things such as areas where a crouched character should be able to navigate.

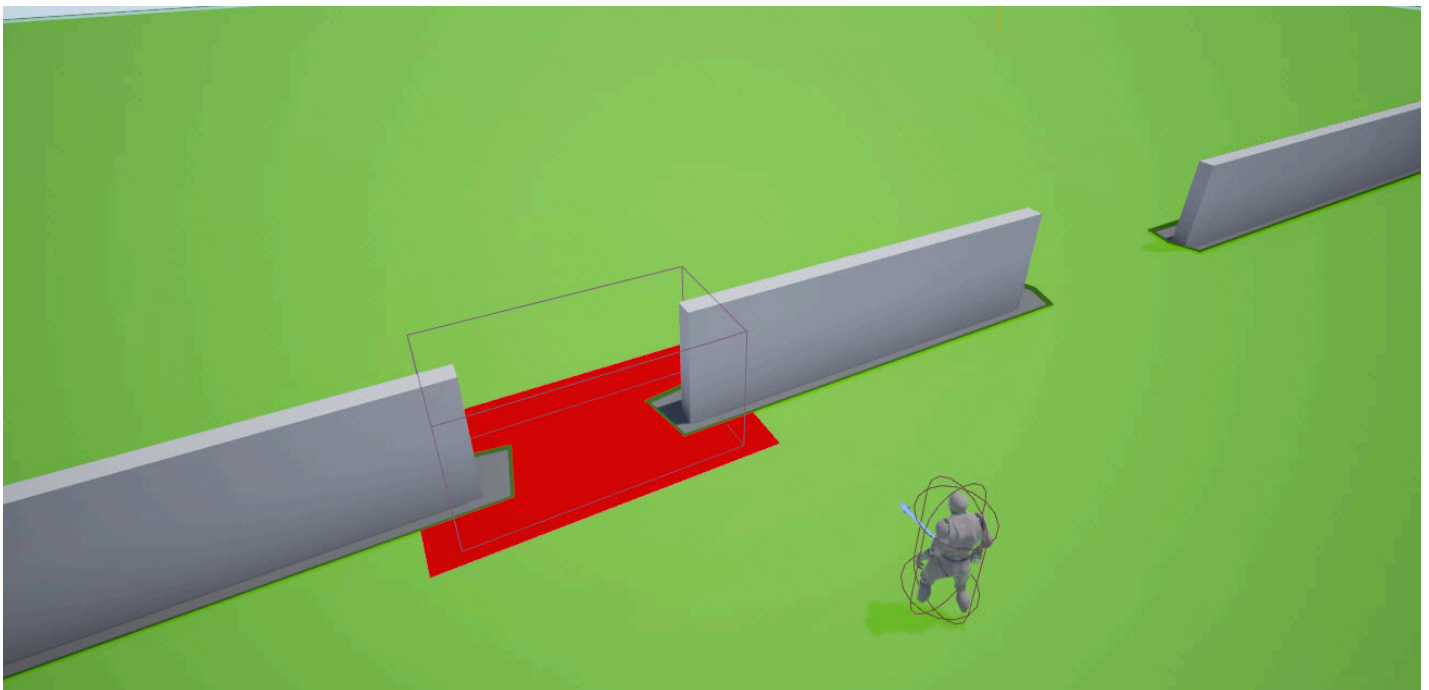
Cost is an important idea for the NavMesh system. Briefly, the total cost to move from one point to another using NavMesh is the sum of all the area costs the path moves through (a

single area's size is defined in the Project's Preferences). But, the solver will always try to find the cheapest path, so you can have it avoid areas (like slippery oil or rough terrain) by increasing the cost to pass through that area. Just remember the solver will still path through high cost areas if it is the cheapest path.

For example, the red area has a very high cost to move through, but the Pawn has no other choice but to move through it.:



But, if you remove a wall:



Then the pawn will avoid the red area even though it means taking a much longer route.