

Mask Occluding Objects Asset Pack

Contents

ASSET PACK OVERVIEW	1
ACCESSING THE ASSET CONTENTS	3
SETUP INSTRUCTIONS	3
SETTING UP MATERIAL PARAMETER COLLECTION SETTINGS	5
SETTING UP CAMERA COLLISION RESPONSE	6
SHORTCUT FOR TESTING MATERIALS WITHOUT CHANGING CAMERA COLLISION SETTINGS PER MESH	7
SETTING UP VISIBILITY CHANNEL COLLISION RESPONSE	8
IMPORTANT NOTE FOR MOBILE PLATFORMS	12

ASSET PACK OVERVIEW

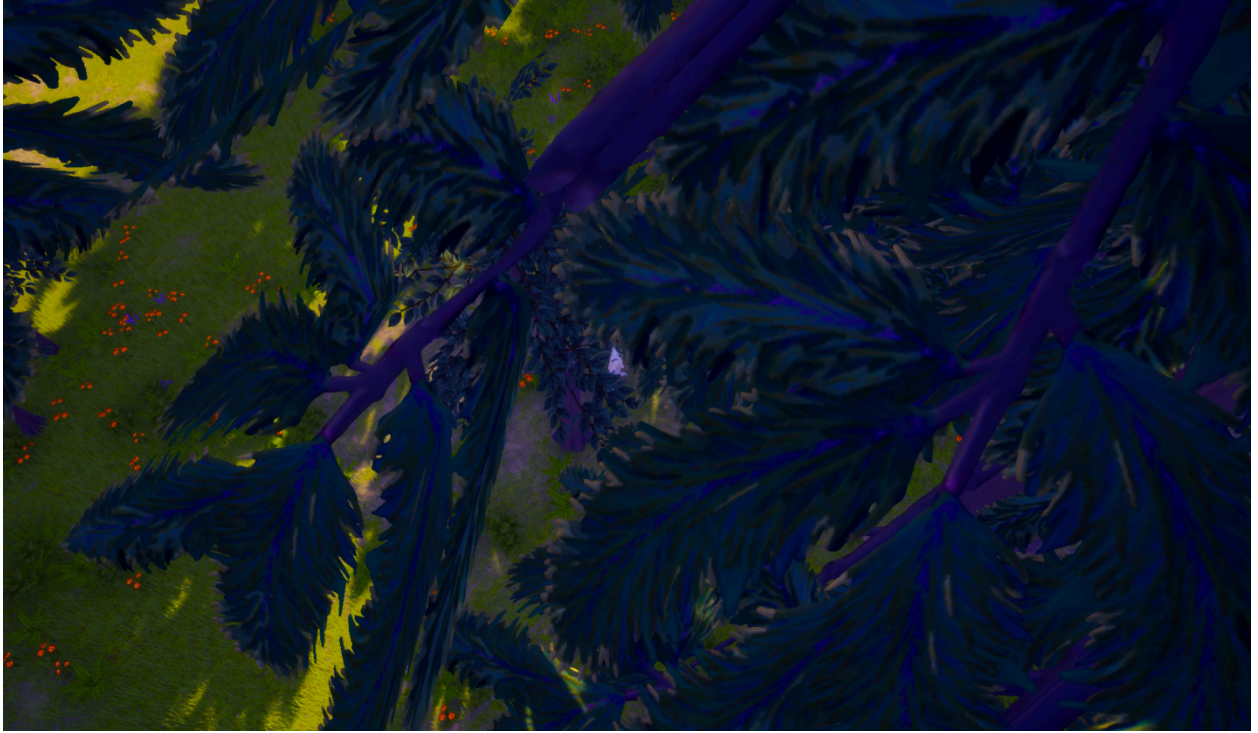
Camera work and occlusion play a big factor in many 3D games. What happens if the character moves behind an object, occluding the character from the camera's view?

Solutions to this problem include either using the camera boom and press the camera forward OR using an occlusion method such as an occlusion sphere or an opacity mask.

The occlusion sphere often clips nearby objects that are not actually between the character and the camera, which is an undesirable result. An alternative is a raytrace that will change the opacity of a mesh's material, or swap out the existing material for a translucent/transparent material. However, this is an all-or-nothing approach, as the entire mesh (e.g. the entire wall) will be affected.

Our solution is to add a simple material function to occluding mesh materials that computes a cylinder from the camera to the character and only culls the surfaces inside. By leaving the rest of the mesh unaffected, we believe this is a superior approach.

Without occlusion mask:

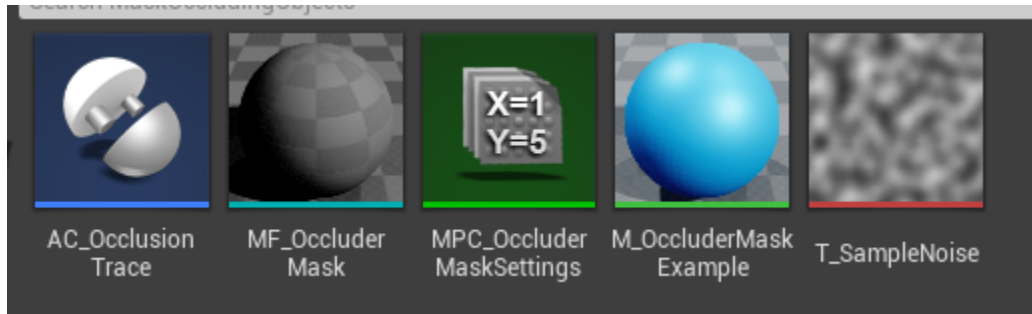


Using our material function:



ACCESSING THE ASSET CONTENTS

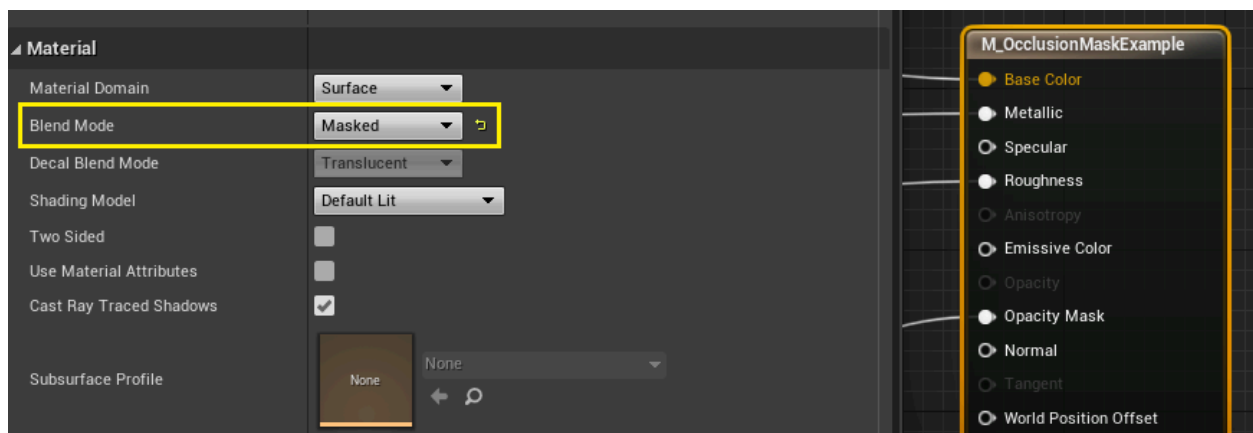
1. Import the asset into your project



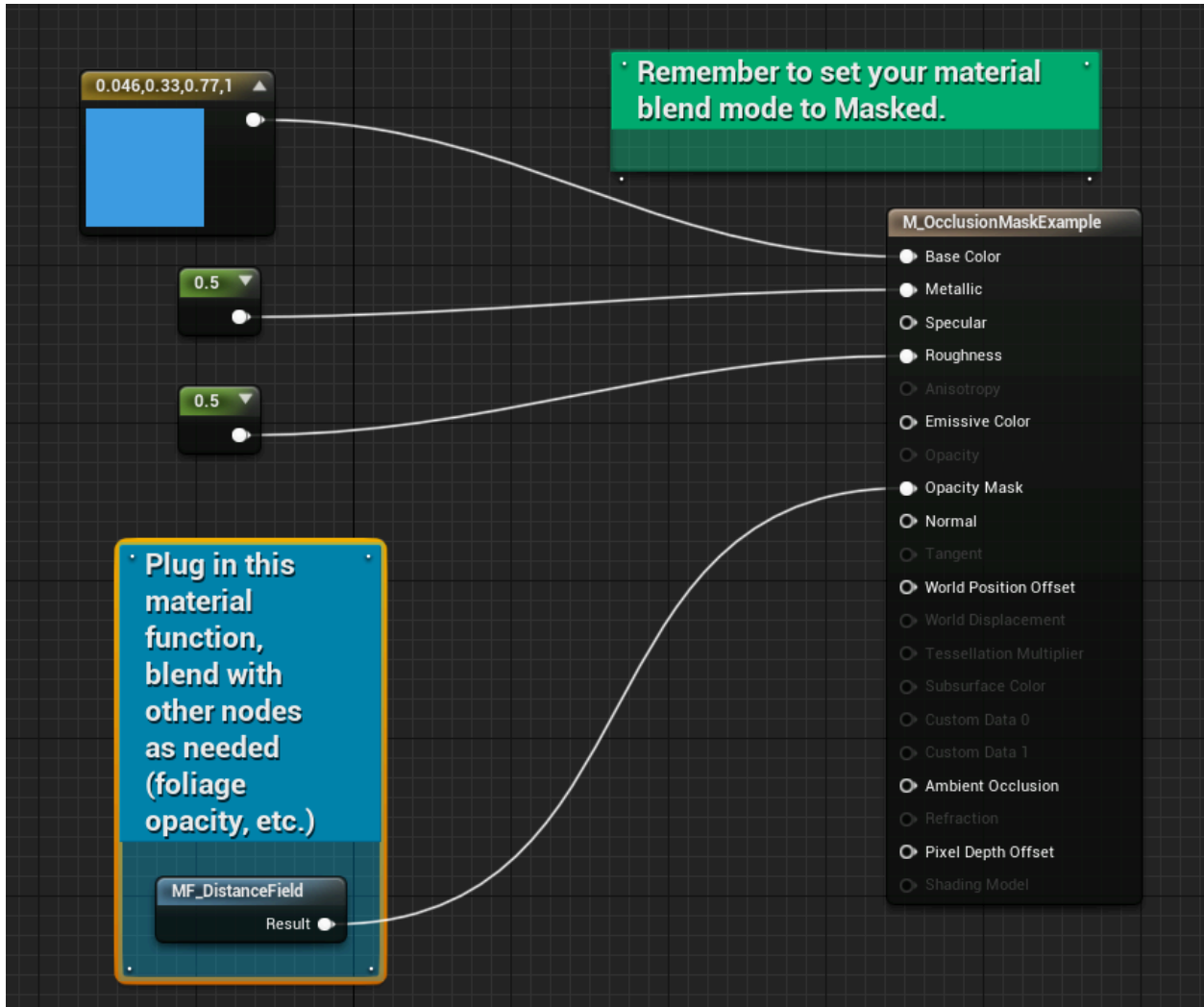
2. This is what each file does:
 - a. AC_OcclusionTrace: an actor component to add to your pawn that you want to show.
 - b. MF_OccluderMask: the material function that will be added to your materials to allow occlusion masking.
 - c. MPC_OcclusionMaskSettings: a material parameter collection used to gather data for the occlusion mask. Edit as needed to modify outputs.
 - d. M_OccluderMask: an example material you can simply add to a static mesh. It has no texture so it will appear as blue.
 - e. T_SampleNoise: a texture sample used for the borders of the occlusion mask. You can use any other texture if you'd like by replacing the texture reference in the MF_DistanceField function.

SETUP INSTRUCTIONS

1. Add AC_OcclusionTrace to your character
2. Modify your materials with the following:
 - a. Click on the material output node and change the blend mode to masked.



- b. Plug the MF_OccluderMask material function into the opacity mask.



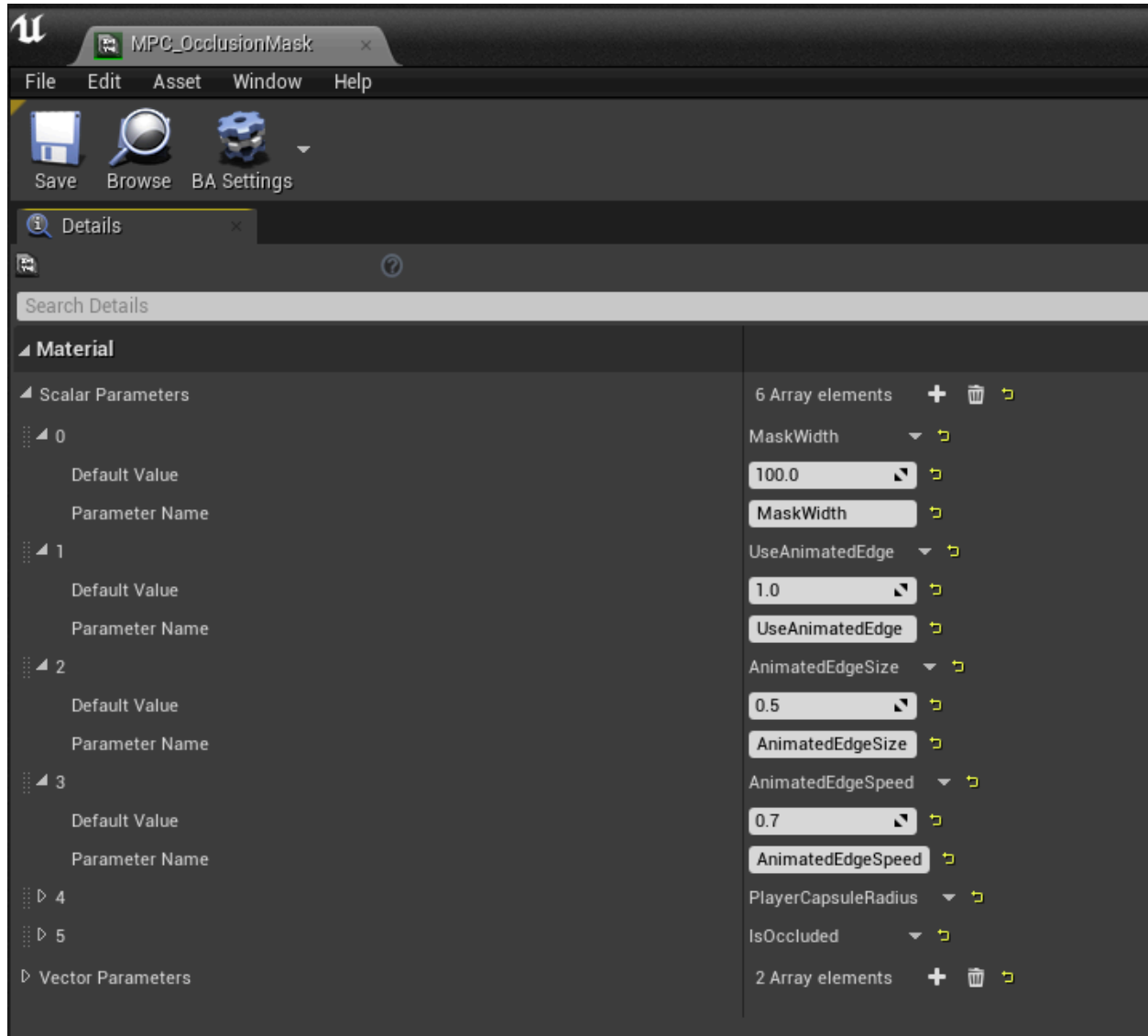
For already masked materials (e.g. Foliage), you can multiply the existing opacity mask input with our material function.



You will need to do this modification with each material of each mesh that the occlusion mask should apply to.

SETTING UP MATERIAL PARAMETER COLLECTION SETTINGS

Here is a list of all material parameters that modify the shape of the occlusion mask:



Mask Width: The radius of the occlusion mask around the character.

Use Animated Edge: Determines whether the mask will apply a panning noise texture around the occlusion perimeter. Set to 0 for a static circle (animation off). Set to 1 for a wavy effect. The animation is based on the noise texture, replace as needed.

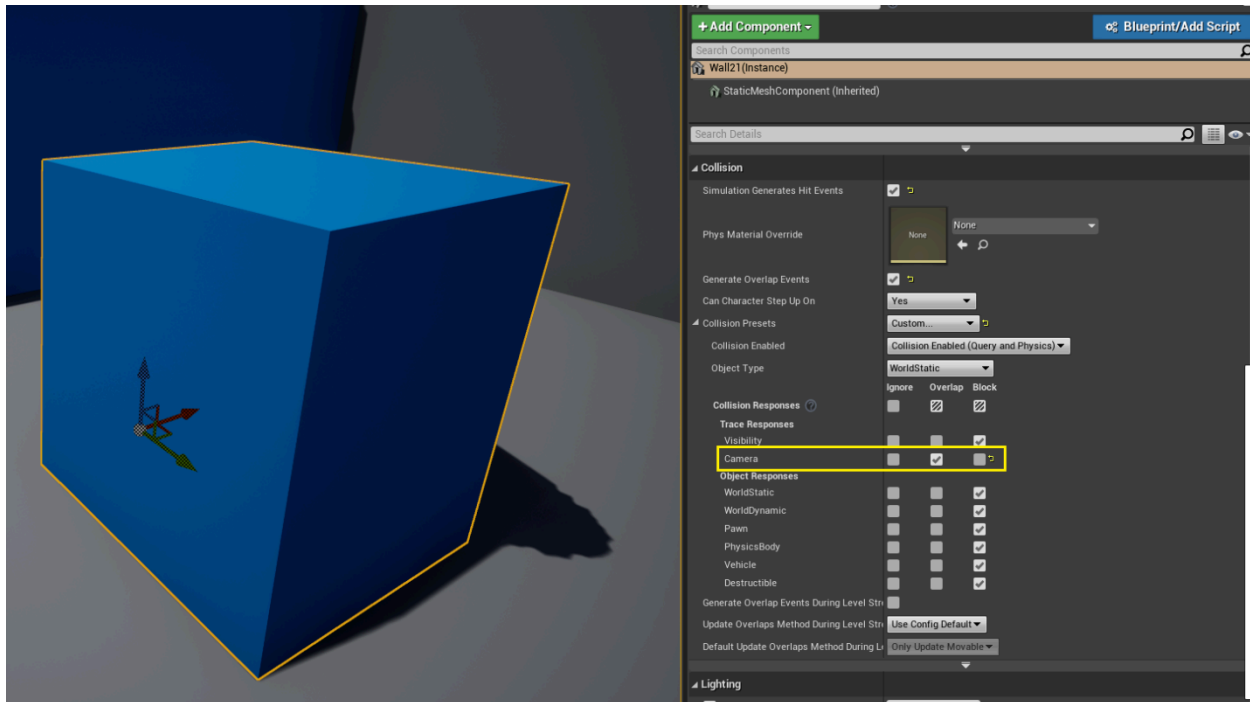
Animated Edge Size: The amplitude of the edge animation.

Animated Edge Speed: The speed of the edge animation.

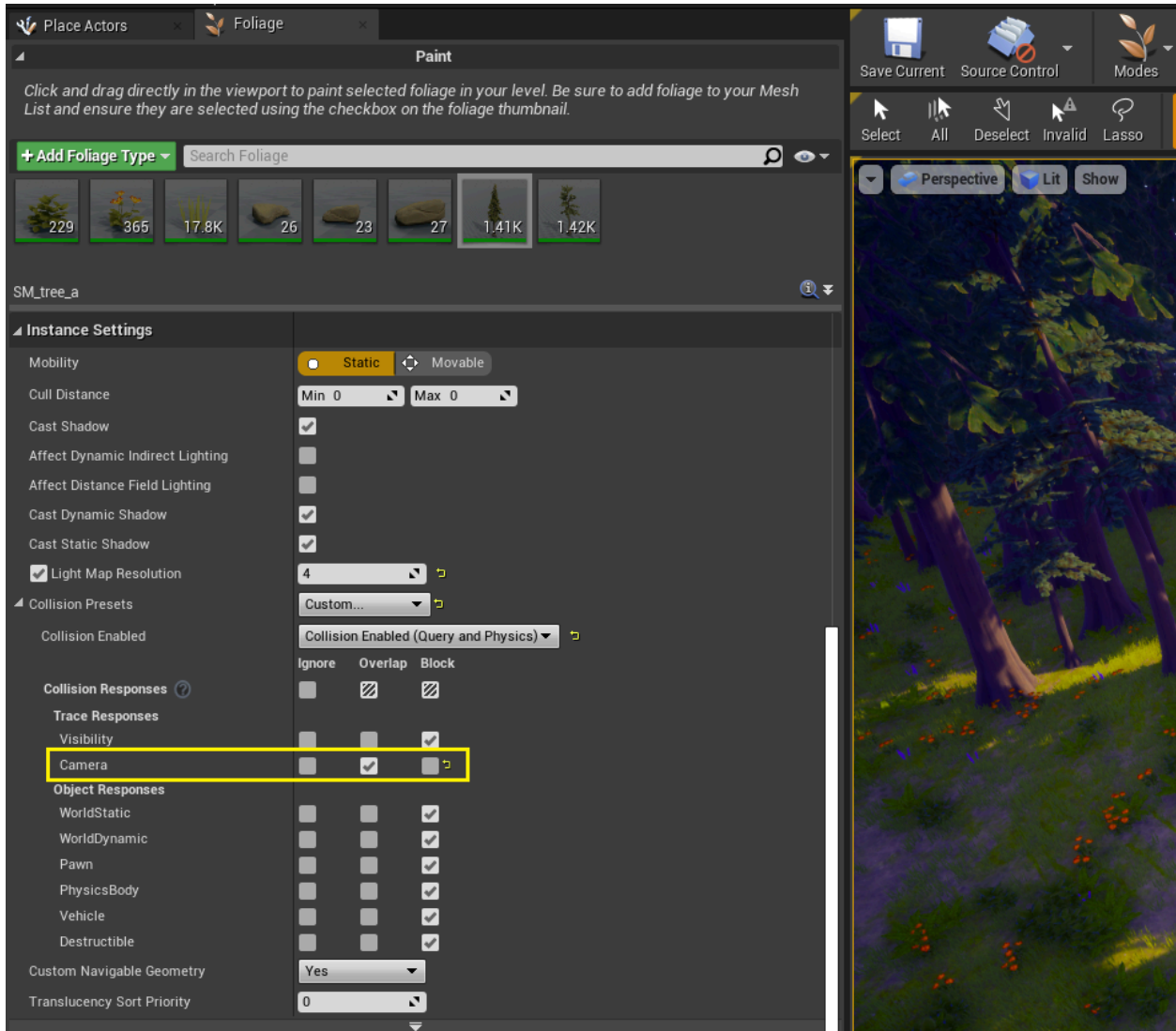
The other settings are internal use only. DO NOT EDIT.

SETTING UP CAMERA COLLISION RESPONSE

For individual static meshes, Camera collision response should be changed to overlapping rather than blocking. If you block, the spring arm will move the camera instead.

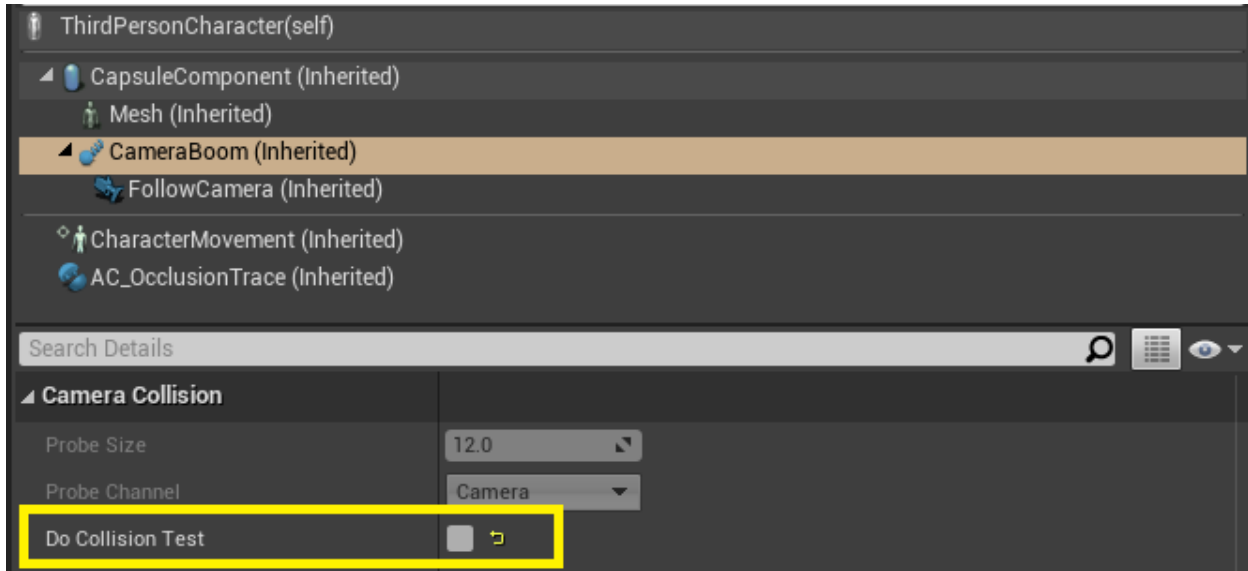


For foliage actors, change the collision in the instance settings in Foliage tool



SHORTCUT FOR TESTING MATERIALS WITHOUT CHANGING CAMERA COLLISION SETTINGS PER MESH

For testing purposes, if you don't want to change the collision of all the meshes at once, you can turn off the camera spring arm collision. Not recommended for final release but still useful. Will let the camera freely move without being blocked by static meshes, walls, landscape, etc.

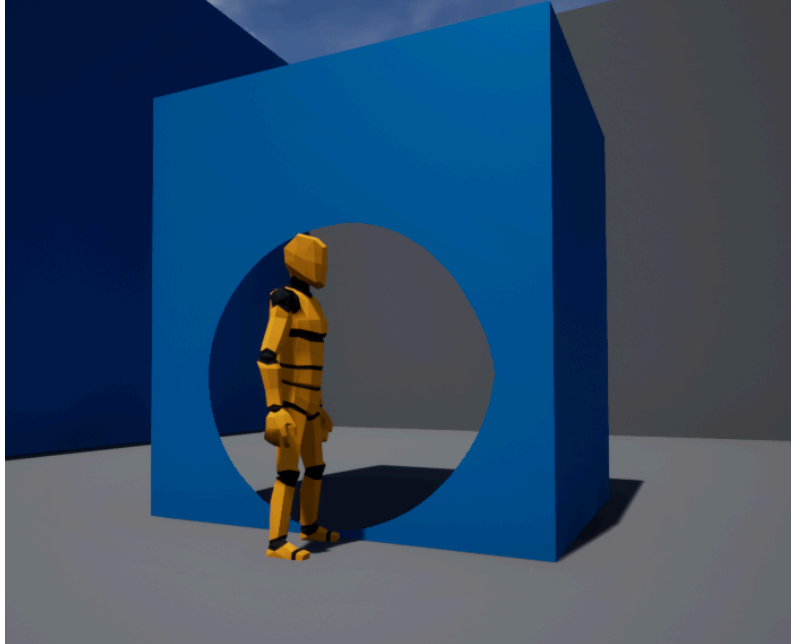


SETTING UP VISIBILITY CHANNEL COLLISION RESPONSE

There are three factors that determine whether a given point on a static mesh surface will be masked and made transparent.

1. Whether the point falls within a cylinder of a specified radius between the character and the camera.
2. Whether the point is closer to the camera than the outer edge of the character's collision capsule component.
3. Whether the character is actually occluded at all from the camera's point of view.

#1 is the basis of this plugin. #2 and #3 are more nuanced but they add a level of polish, preventing objects from getting unintentionally masked. For example, with #2 and #3 both off, this will happen:



As we get close to a surface, the mask is extending past the character and unnecessarily affecting the surface behind it.

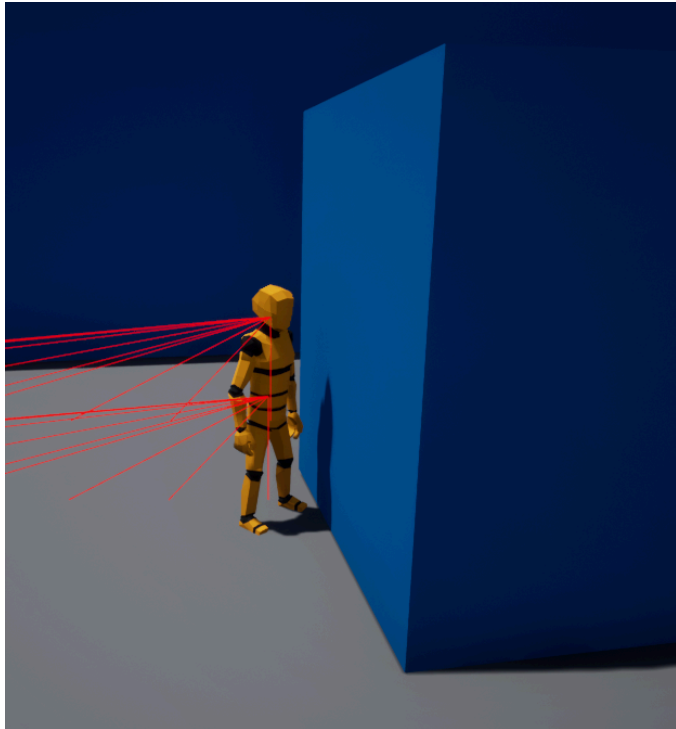
#2 (mostly) solves this issue and is not designed to be changed, as this check is built into the material function. By measuring to the closest edge of the collision capsule rather than the character mesh, we see the following:



However, it is only “mostly” because at certain angles the mask can still inadvertently act on this surface.

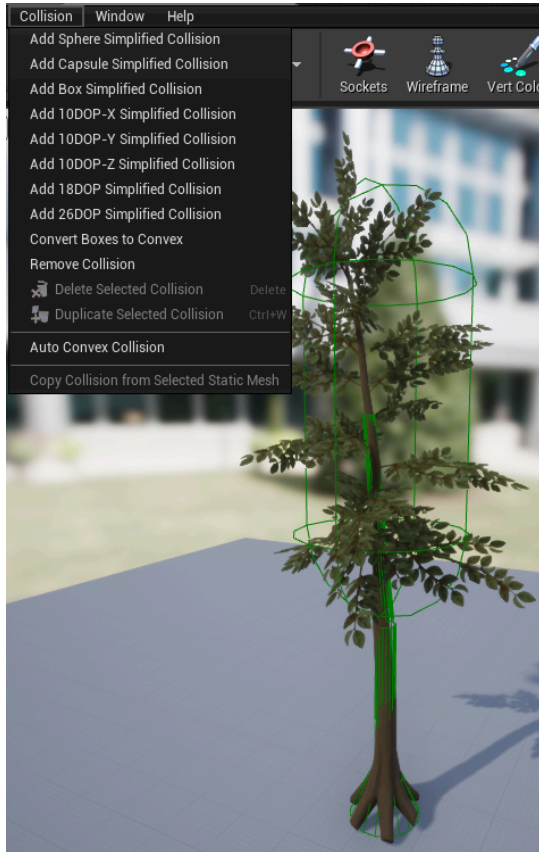


At this acute angle, for instance, the culled points *are* technically closer to the camera than the character itself. Hence check #3. For check #3, we implement a line trace on Tick that determines whether an object actually stands between the character and the camera.



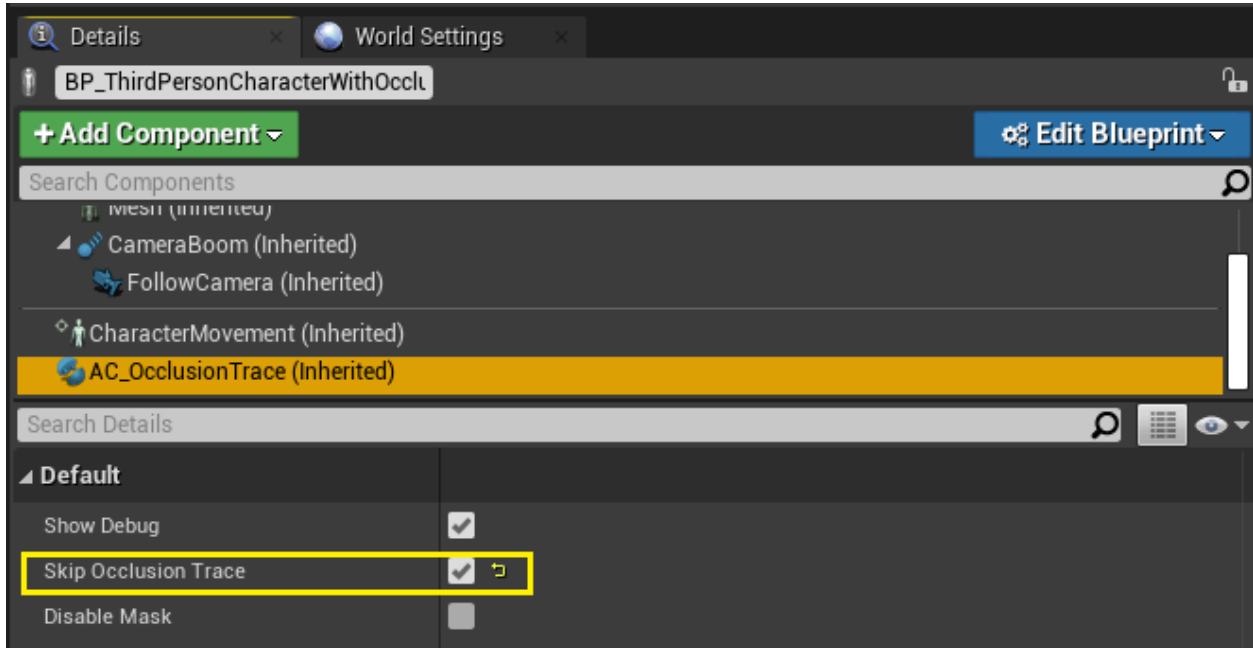
For better results, there are two line traces, starting at the center of the character collision capsule and at the head mesh socket.

Implementing #3 may require some extra work, especially if you are using foliage. For the ray traces to work, we need reasonable collision shapes for each mesh that the character could potentially see through. A tree mesh, however, might only have a collision shape at the trunk, so you must add an additional collision shape to account for the leaves (because you want to register a successful line trace hit when looking through leaves, not just the trunk).



For example, this tree asset only had collision at the tree trunk. A capsule collision component had to be added manually.

The amount of manual collision setup work depends entirely on the assets you use. You may only need to do this for a few assets, or none at all. But if the amount of work is overwhelming, and the visual results using only check #2 are acceptable to you, check the “Skip Occlusion Trace” option in the AC_OcclusionTrace component settings.



You can also select the “Disable Mask” option to disable the occlusion mask entirely.

IMPORTANT NOTE FOR MOBILE PLATFORMS

On mobile platforms, shaders are not calculated with the same precision as on desktop, and this can result in errors when doing distance calculations. If you are targeting Android or iOS, please enable the following setting (Use Full Precision) in each material you are using the mask on:

