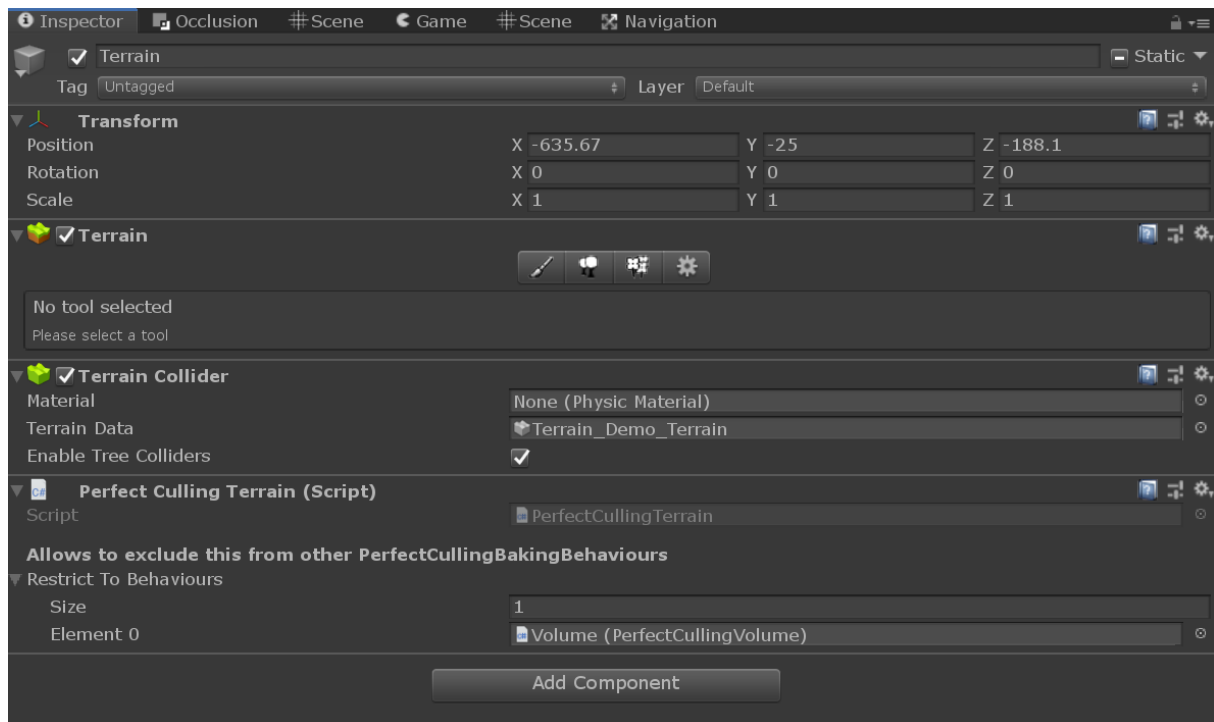


Terrains

Terrains are a very special type of geometry in Unity thus require an additional step to make them compatible with Perfect Culling.

Just find your Terrain in your scene and add the **PerfectCullingTerrain** script.

By default, the terrain will be picked up by all **PerfectCullingVolume**. If you want to restrict it to a specific volume, make sure to add the desired one to the array.

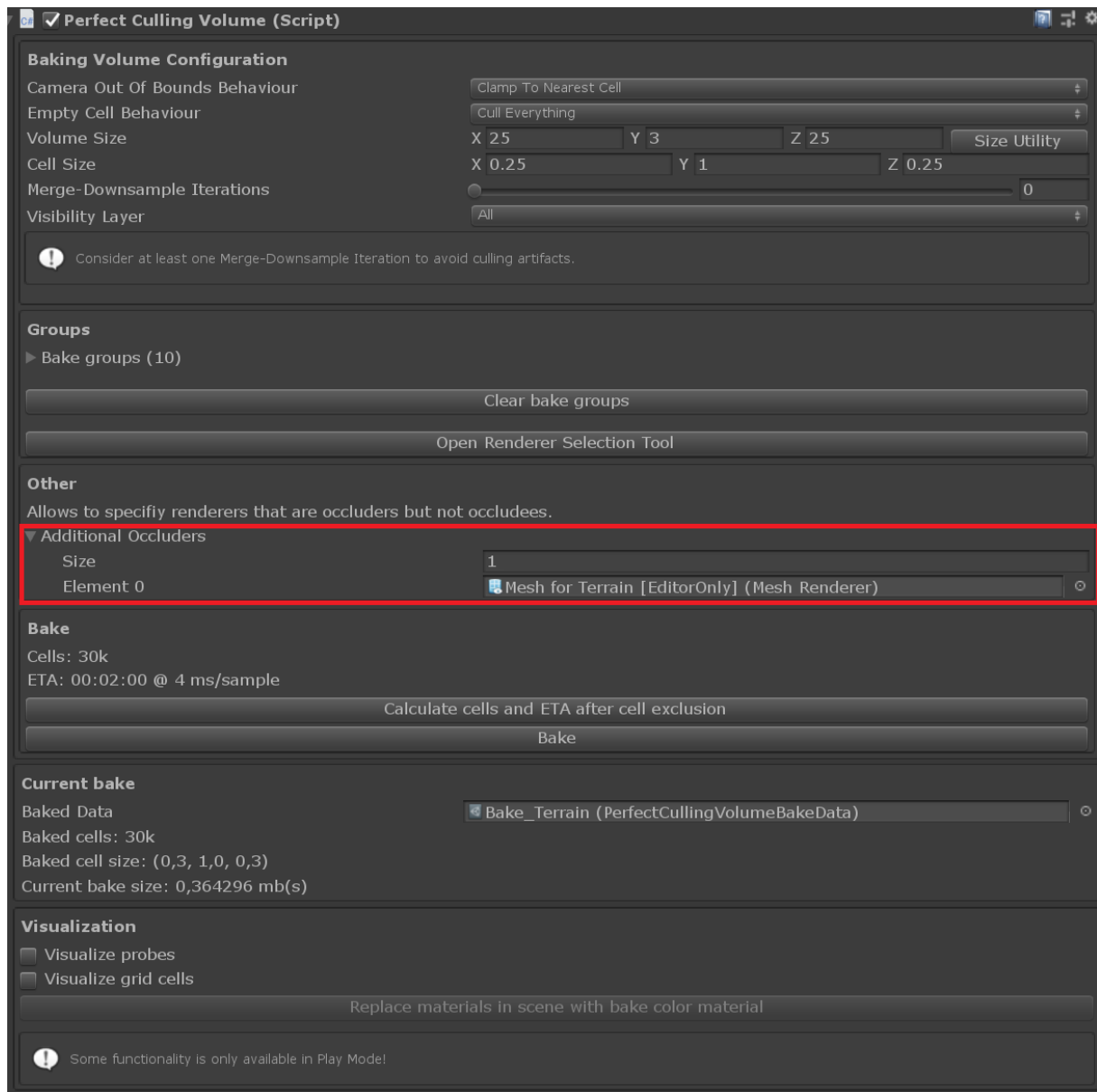


Example setup to make Perfect Culling take into account the Terrain. This terrain is restricted to be controlled by a single volume (recommended).

Because the Unity Terrain is a different type of geometry it is temporarily converted into a mesh during the baking process. This only applies to the baking process and does not change your Terrain in any way!

However, it is also important to understand that Perfect Culling can only cull the Terrain as a whole whereas Umbra is capable of culling individual terrain chunks. This is engine internal functionality that is not available to Perfect Culling and cannot be replicated at this time.

If your terrain is very complex and you absolutely need to chunk the terrain into multiple pieces you can of course do that and just pass the individual meshes to Perfect Culling like any other renderer.



Example of an additional occluder. This works for any kind of mesh and does not cause any run-time overhead at all.

Instead, if your terrain is always in view and all you want is to allow the terrain to occlude other objects you might want to consider to just add it as an “Additional Occluder”.

The way how you can achieve that is by adding the **TerrainToMeshUtility** script to the Terrain and create a mesh. Now you can just drag and drop the mesh to the “Additional Occluders” array on your Perfect Culling Volume.