



LODs

The asset automatically handles LODs for you. Under the hood it makes sure to add them into a shared bake group (similar to how Custom Scene Groups work). This reduces the memory footprint and also makes sure to give back control to your LOD system if a renderer was found to be visible.

Important to know:

- The native renderer will always use LOD0 for generating the occlusion data
- The Unity Renderer will always use the active LOD for generating the occlusion data

What does that mean?

Fortunately, it usually doesn't cause issues because the culling becomes more conservative by using the Merge-Downsample step. However, if you are using the Unity Renderer and your LOD settings vary a lot you might see Perfect Culling cull objects just because the LOD system culled them during the baking process. That means they are not visible at baking time and thus will be culled.

Feel free to contact me if you run into issues like that and I'm sure we can figure something out.