



polycam

Process for scanning your room with 3D LIDAR scanning and importing it to VaM

Use Polycam LIDAR scanning app on your Ipad pro to scan environment.

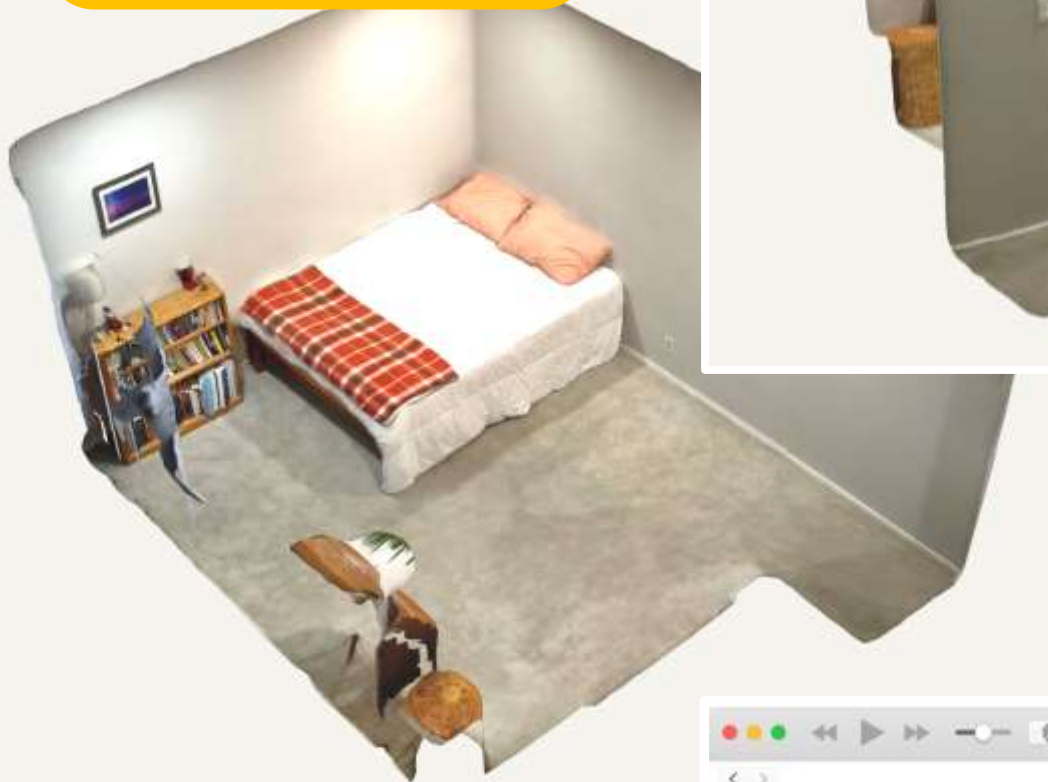
Some experimentation is required to learn the best movements to maintain tracking and avoid jagged textures.

Tips:

- Too dark ambient lighting will often make it lose tracking and produce large gaps or mismatches
- 1 strong light source might make it easier to imitate in VaM (the sun for instance)
- Open windows will produce bad results. Close curtains.
- Reflective surfaces such as mirrors will produce garbage or double geometry behind the mirror surface. This can be removed in Blender and fixed with reflective plane in VaM however, so don't be afraid to try
- Lighting must be stationary, no flashlight as you move



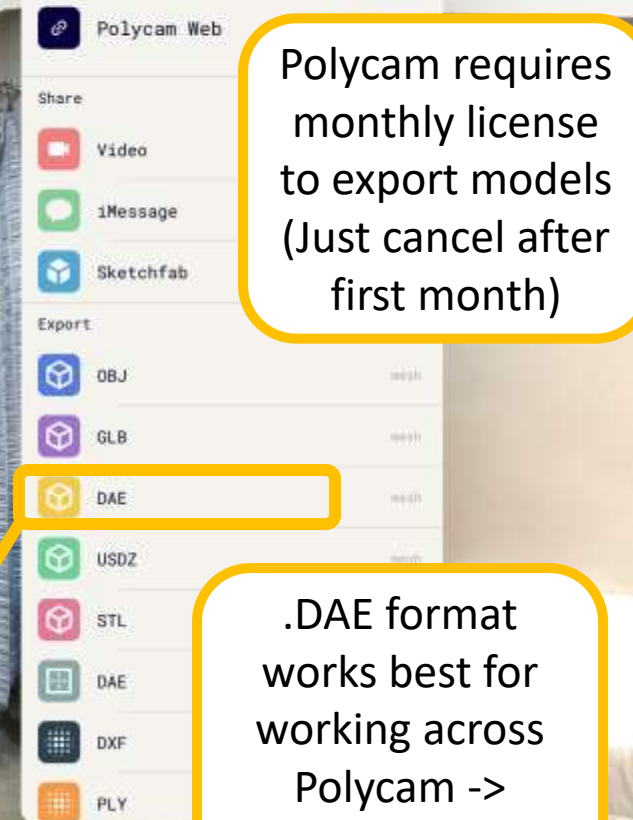
Polycam produces
lightweight textured
meshes with very little
surface noise



Export file to
.DAE format

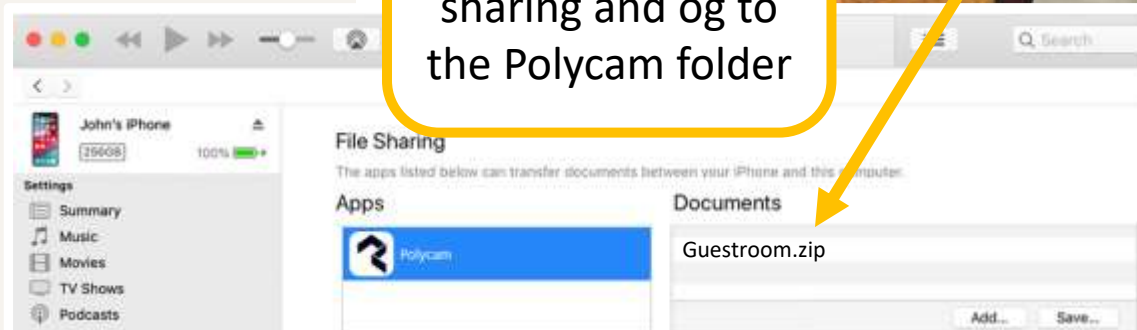


Polycam requires
monthly license
to export models
(Just cancel after
first month)



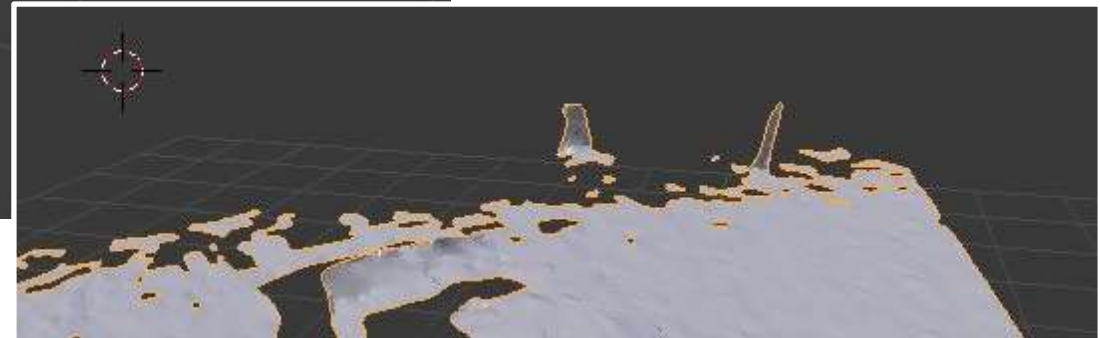
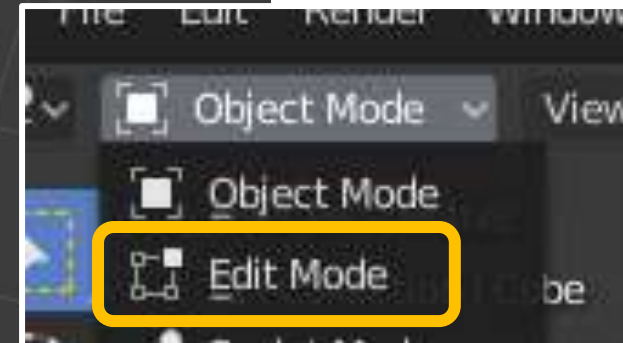
.DAE format
works best for
working across
Polycam ->
Blender / Unity

Transfer file to PC
using iTunes file
sharing and og to
the Polycam folder



If you have garbage polygons from open windows or reflective surfaces, the .DAE model can be imported into Blender and cleaned up before exporting again to .dae and continuing with Unity.

There's also options for closing openings if areas were missed in the scanning process



Read the instructions on how to use MacGruber's awesome

Unity AssetBundles for VaM 1.xx

<https://hub.virtamate.com/resources/unity-assetbundles-for-vam-1-xx.167/>

This tutorial gives you a quick introduction to how to export assets from Unity into VaM.

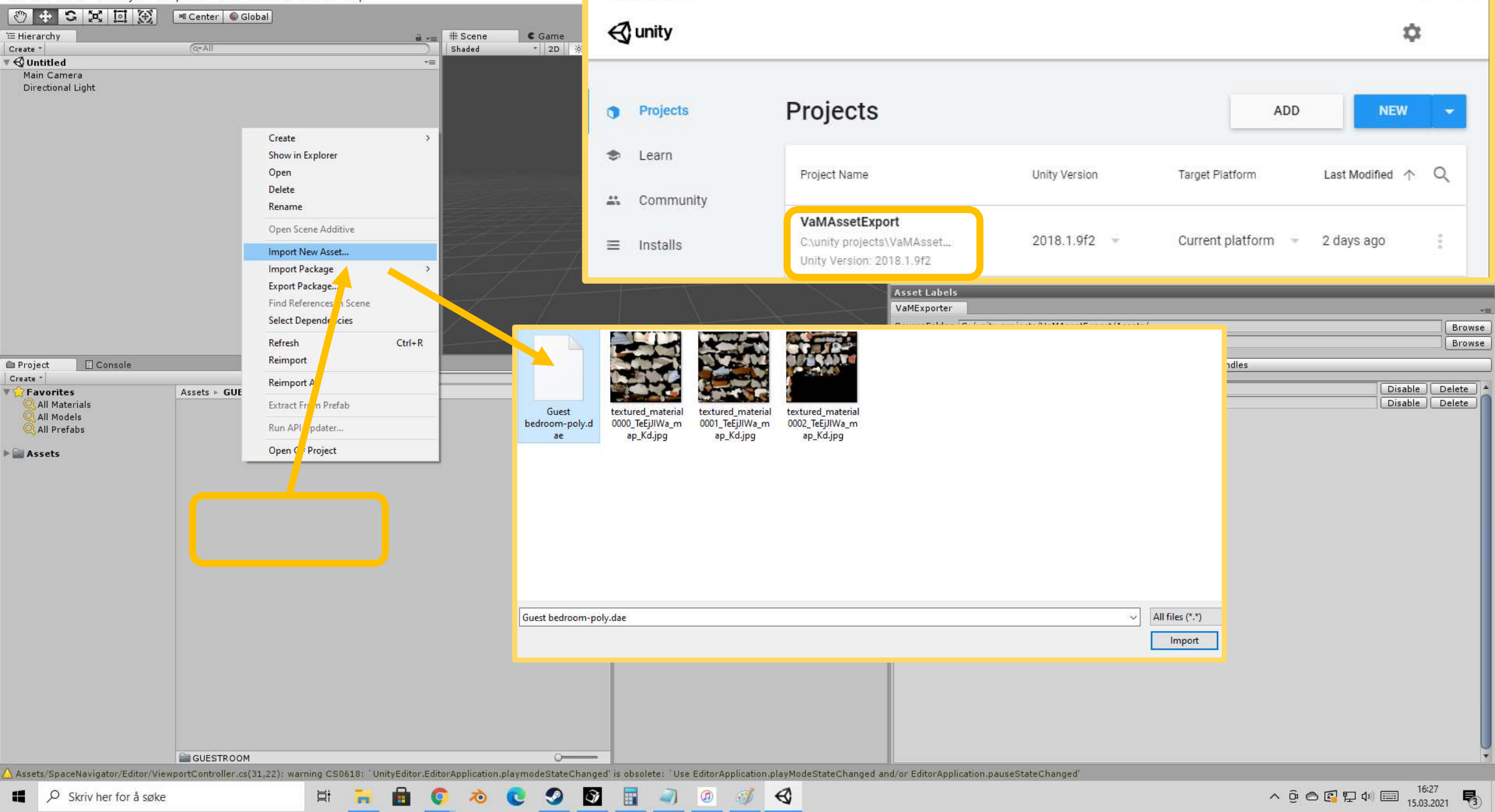
Overview

Install the correct Unity version.....	1
Setup your Unity project.....	2
Getting some Assets	5
Adding collision	6
Creating a material variation	7
Creating Prefabs.....	8
Creating the AssetBundle.....	9
Using VaMExporter	10
Loading in VaM	11
Extracting assets from AssetBundles	12
License considerations	12
Included Tools: EditorOnly, ProBuilder and ProGrid	13

Install the correct Unity version

Note that **you can't just install the newest version of Unity**, you will run into trouble. The reason is that VaM 1.xx is still being build with an old Unity version back from the year 2018. It can't load anything produced with newer versions of Unity. Assuming you want to make AssetBundles **for VaM 1.xx versions, you will need Unity 2018.1.9** or a point VaM will likely switch to a newer Unity version, though, likely for VaM 2.xx.

which allows you to have multiple versions of Unity installed in parallel. You might find a nice tool to manage your Unity projects. Once you got Unity Hub, you



- Create
- Show in Explorer
- Open
- Delete
- Rename
- Open Scene Additive
- Import New Asset...**
- Import Package
- Export Package
- Find References in Scene
- Select Dependencies
- Refresh Ctrl+R
- Reimport
- Reimport All
- Extract From Prefab
- Run API Updater...
- Open C# Project

Guest bedroom-poly.dae

textured_material_0000_TeEjJIWa_m_ap_Kd.jpg

textured_material_0001_TeEjJIWa_m_ap_Kd.jpg

textured_material_0002_TeEjJIWa_m_ap_Kd.jpg

Unity Hub 2.4.2

unity

Projects

Learn

Community

Installs

Projects

Project Name

Unity Version

Target Platform

Last Modified

VaMAssetExport

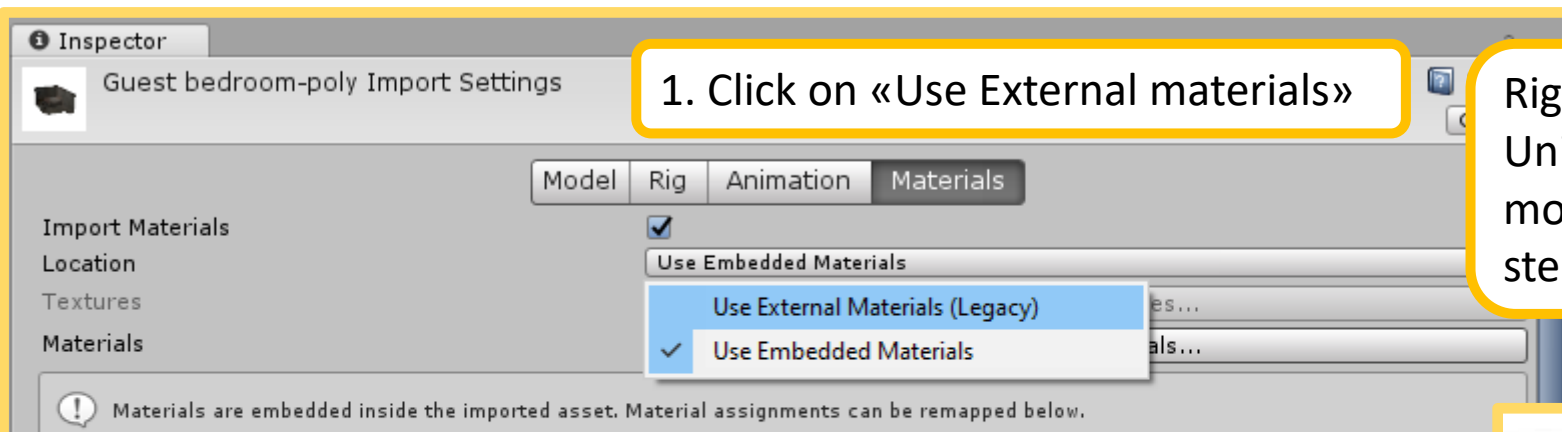
C:\unity projects\VaMAsset...

Unity Version: 2018.1.9f2

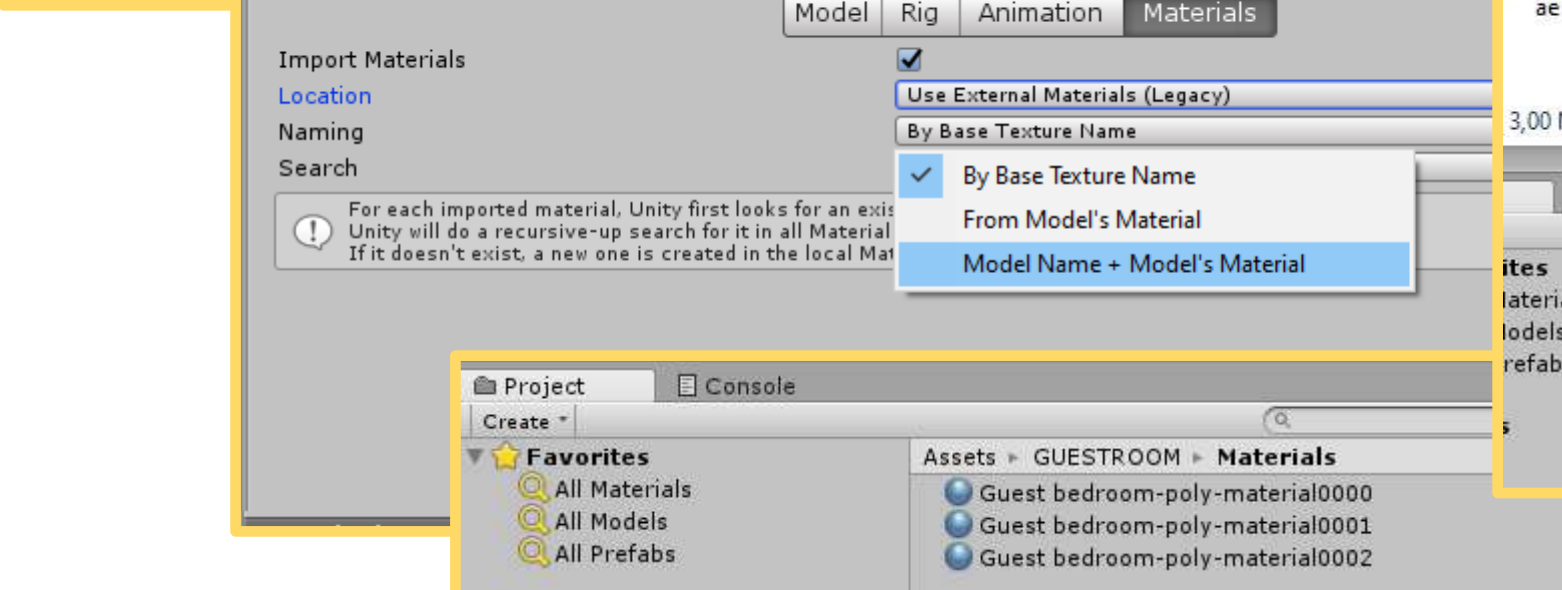
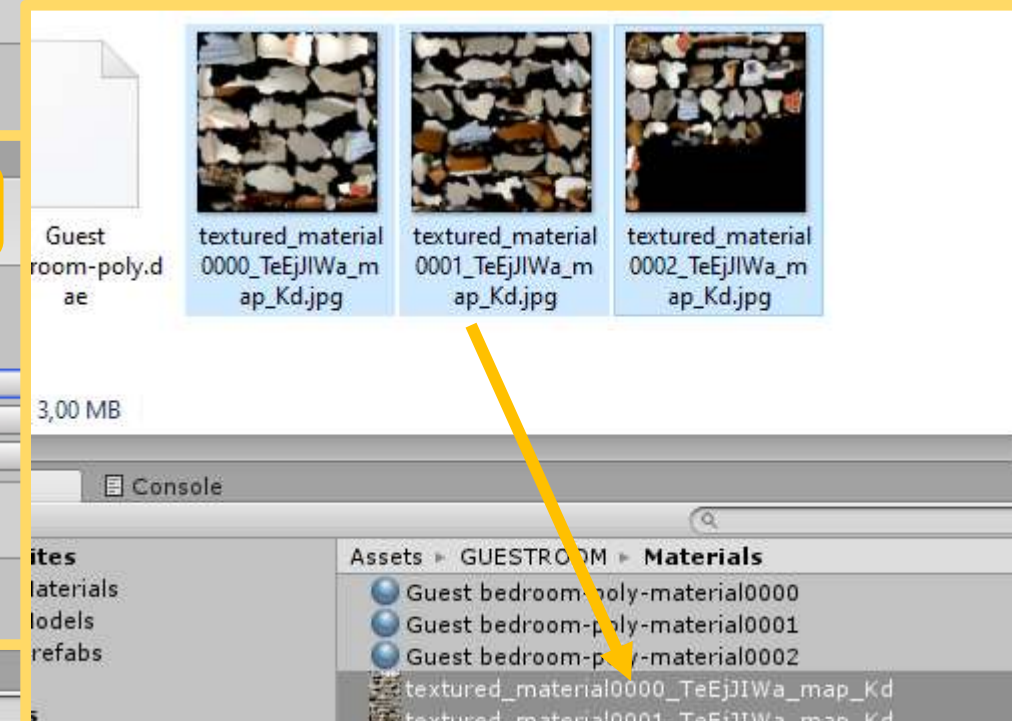
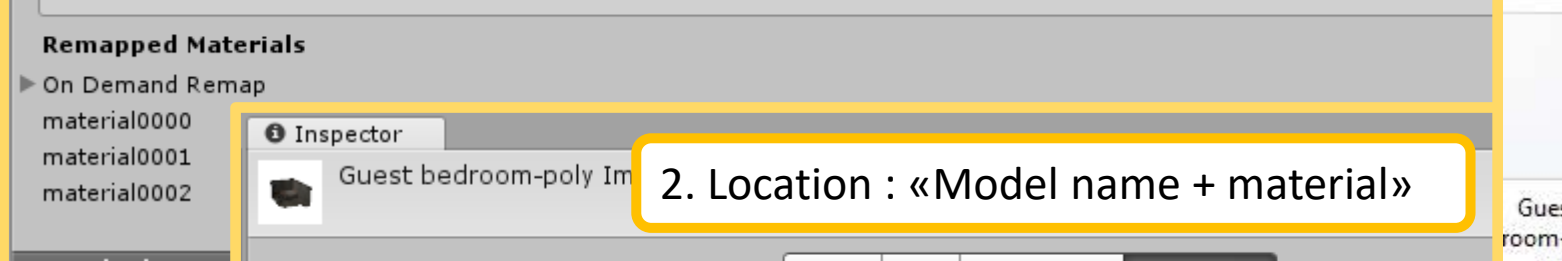
2018.1.9f2

Current platform

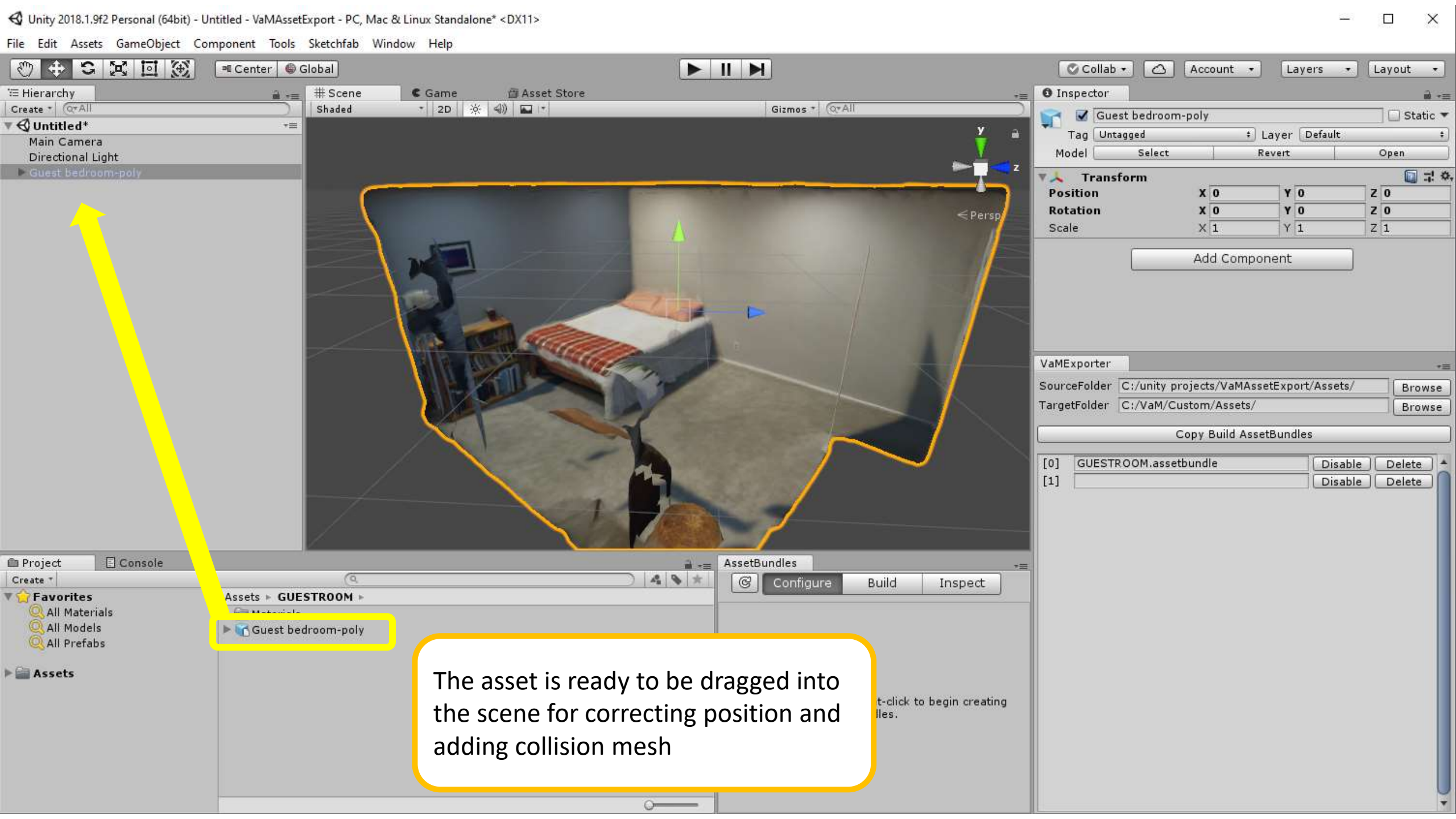
2 days ago



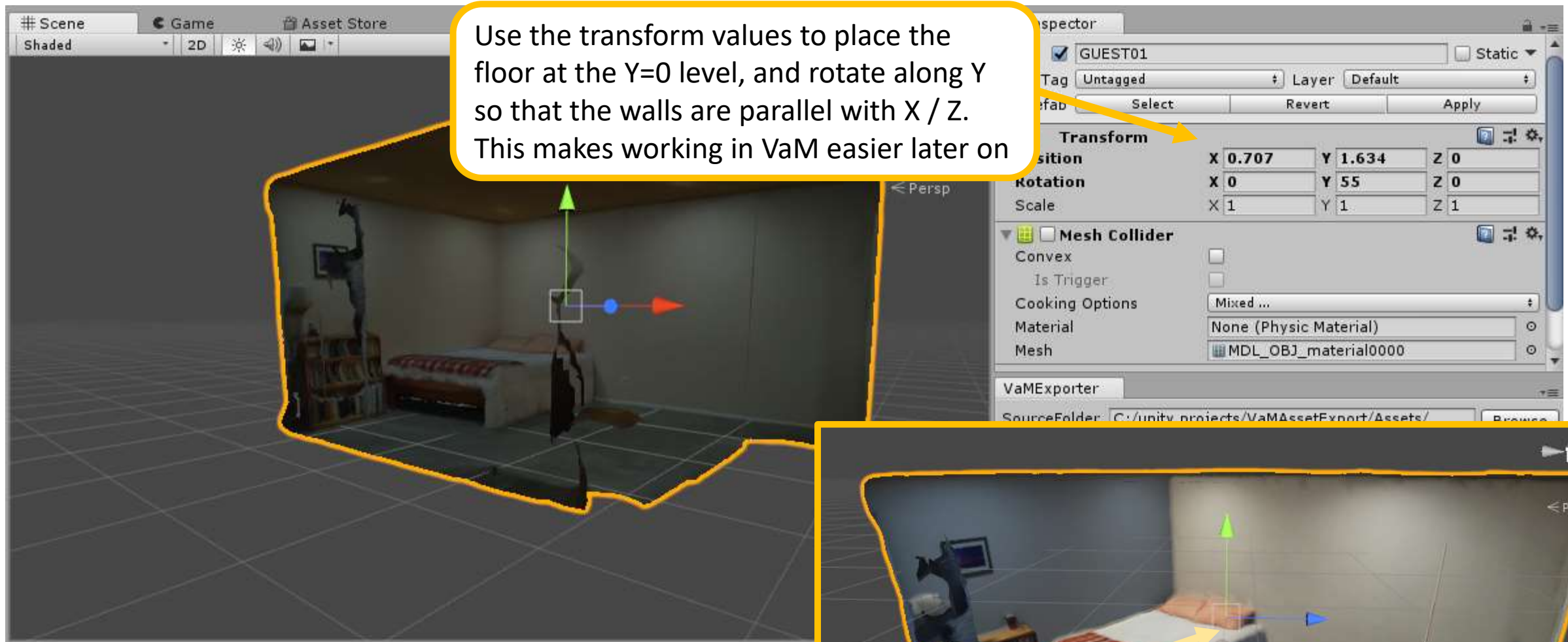
Right after importing your .dae file, make sure Unity opens the materials also in the 3D scan model. For me the only way has been these steps:



4. Drop the JPG textures from the Polyscan file next to the materials



The asset is ready to be dragged into the scene for correcting position and adding collision mesh



Polycam usually puts the origin in the centroid of the model, and it's usually rotated along the vertical. It's always level, however and in correct 1:1 scale (which is very time saving)

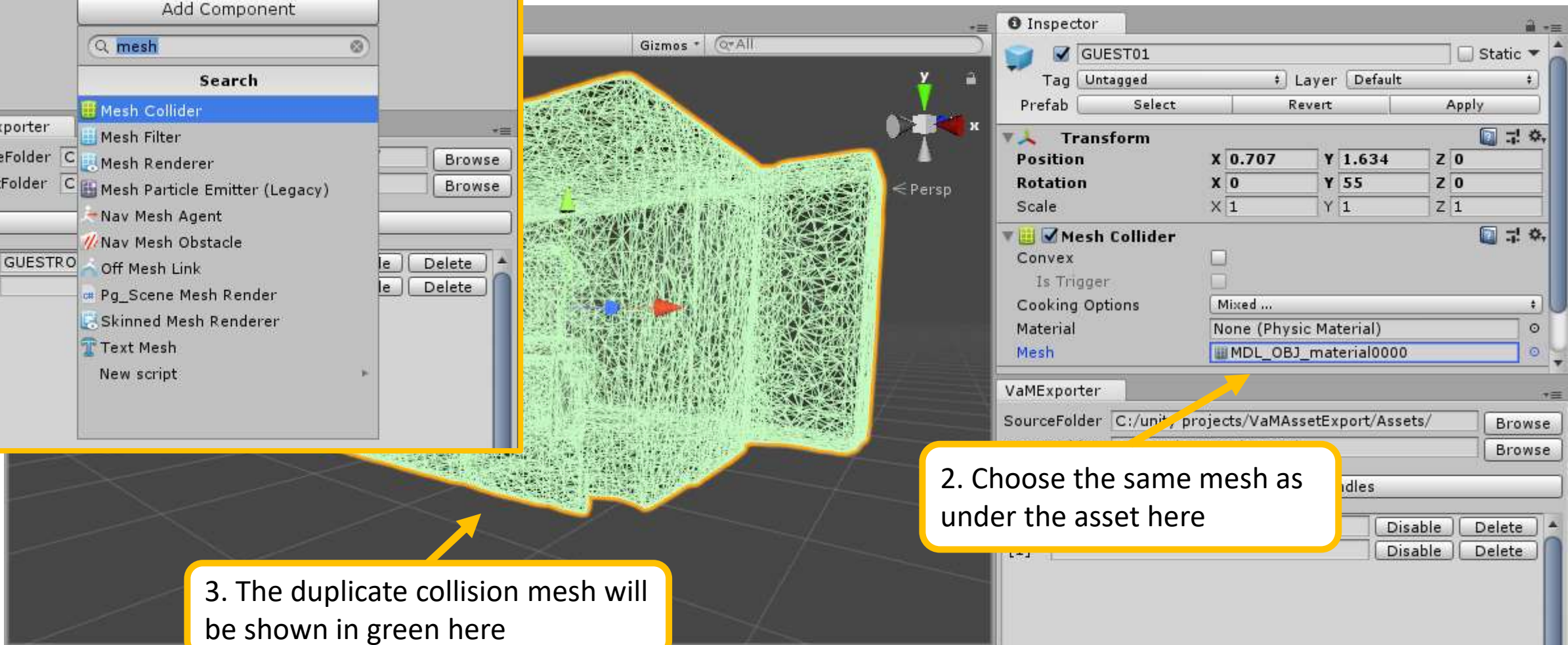


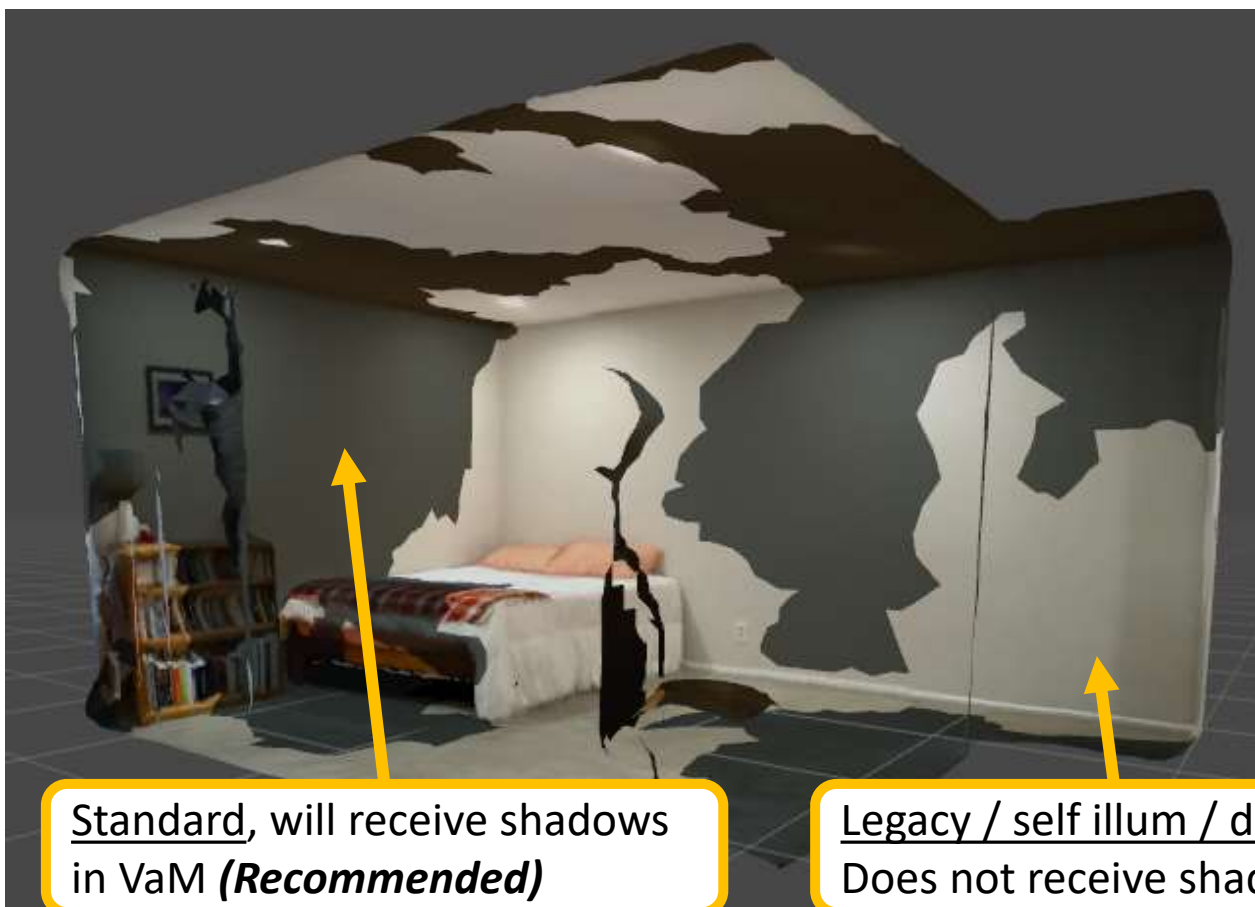
1. Choose «add component»
and «mesh collider»

The next important step is to add the collision mesh.
This is just an invisible duplicate of the scanned mesh
that VaM uses for physics. This is required for all the
physical interactions that will happen in VaM.

3. The duplicate collision mesh will
be shown in green here

2. Choose the same mesh as
under the asset here

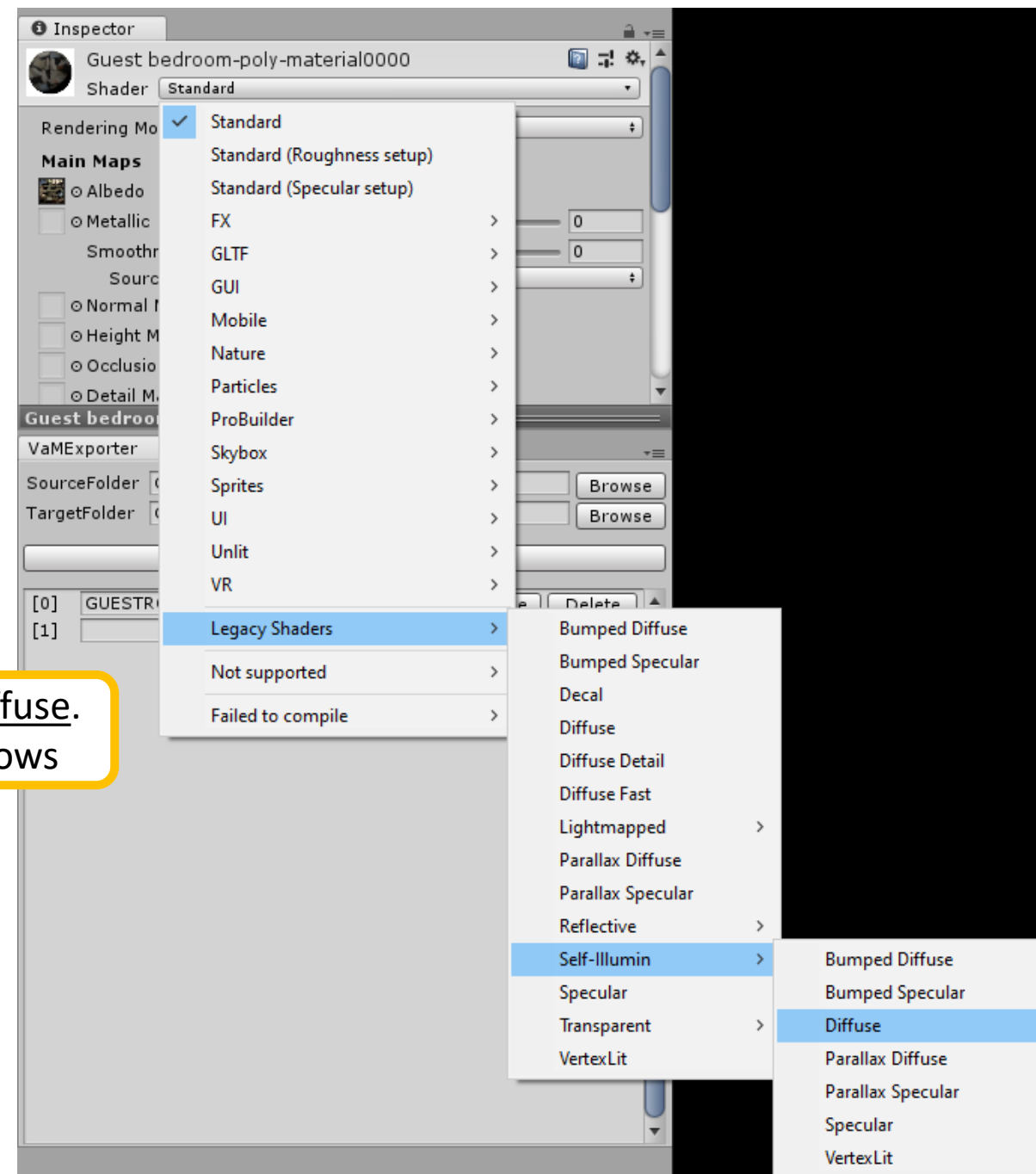




Standard, will receive shadows
in VaM (**Recommended**)

Legacy / self illum / diffuse.
Does not receive shadows

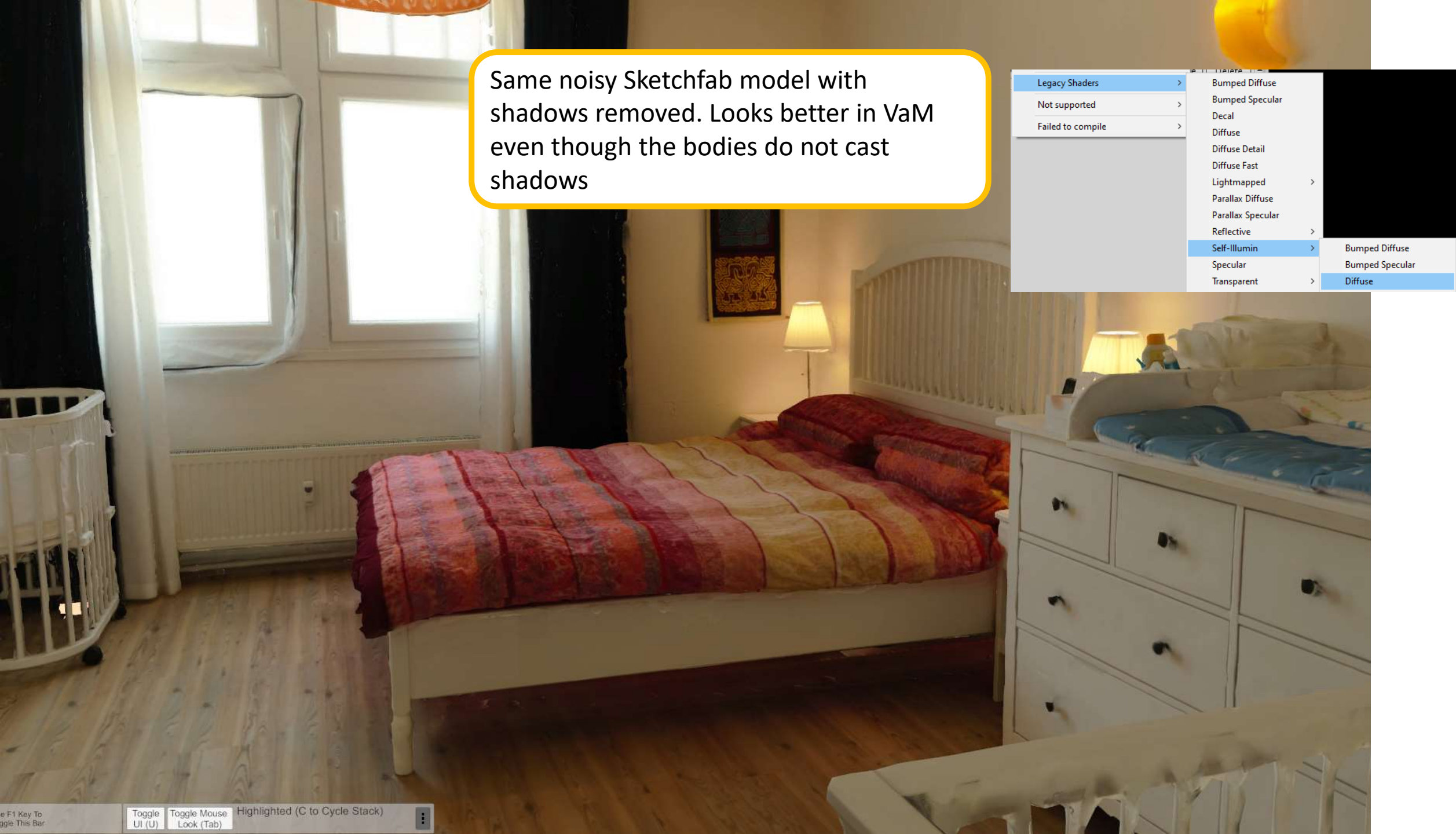
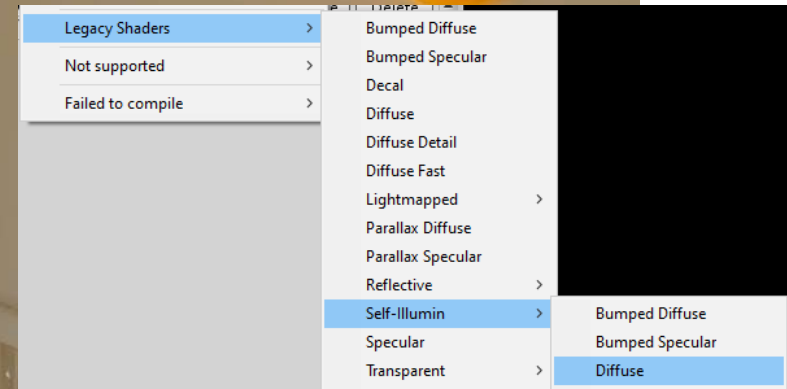
At this point you can choose to only use
the baked lighting in the 3D scan textures.
This is the correct option if you have good
textures but a noisy model.

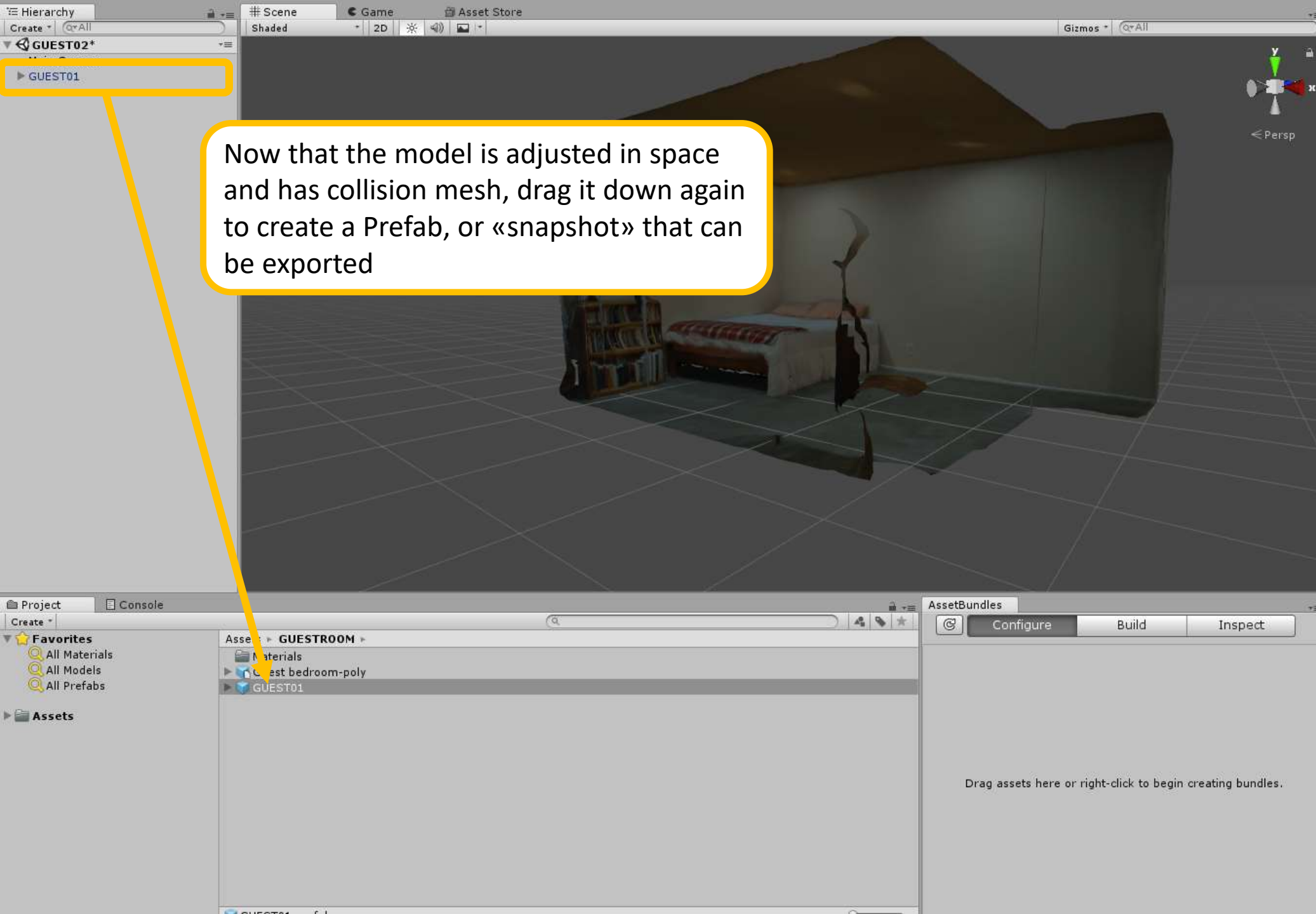


Noisy photogrammetry model from Sketchfab as example. The «melted» look comes from received shadows in VaM on noisy polygons.

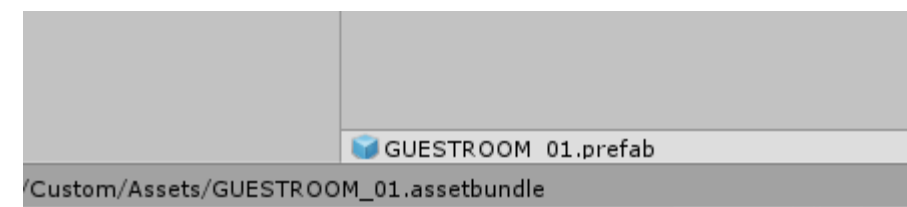
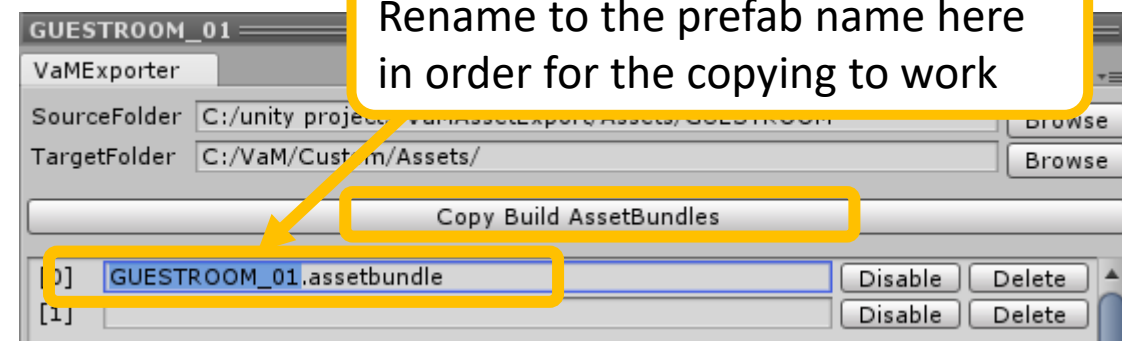
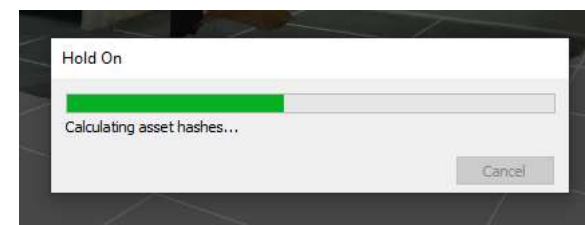
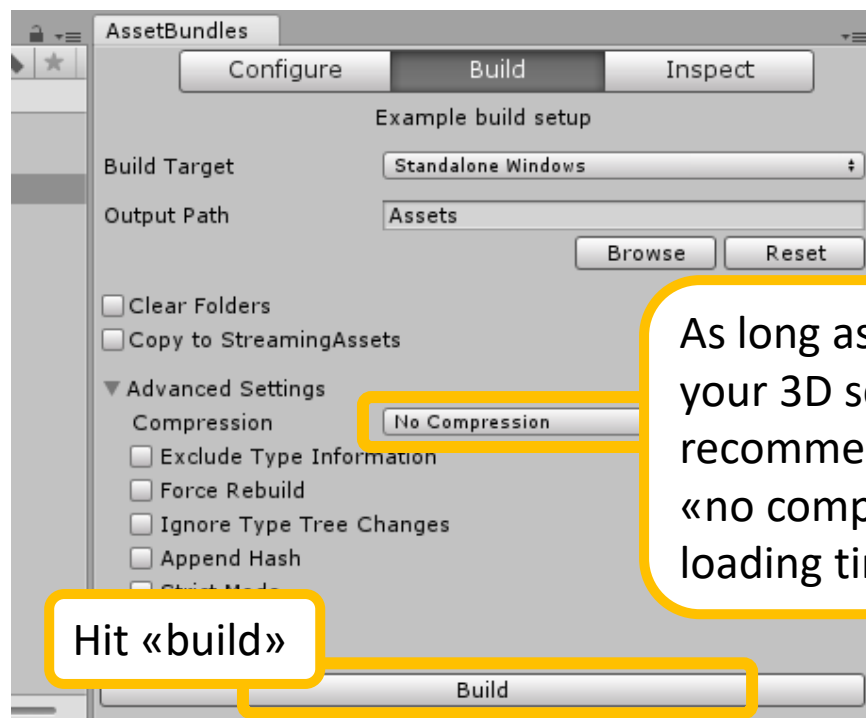
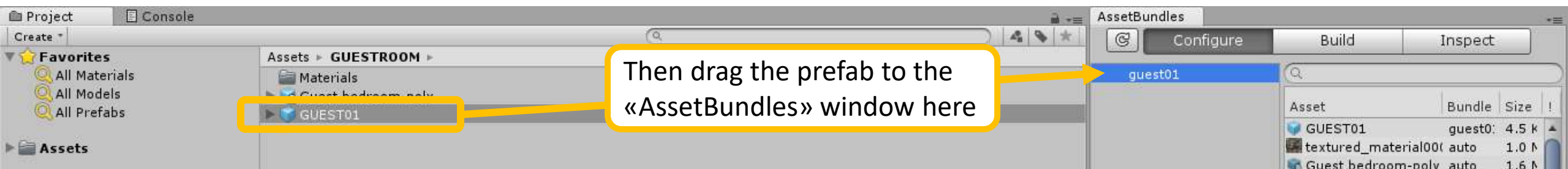


Same noisy Sketchfab model with shadows removed. Looks better in VaM even though the bodies do not cast shadows

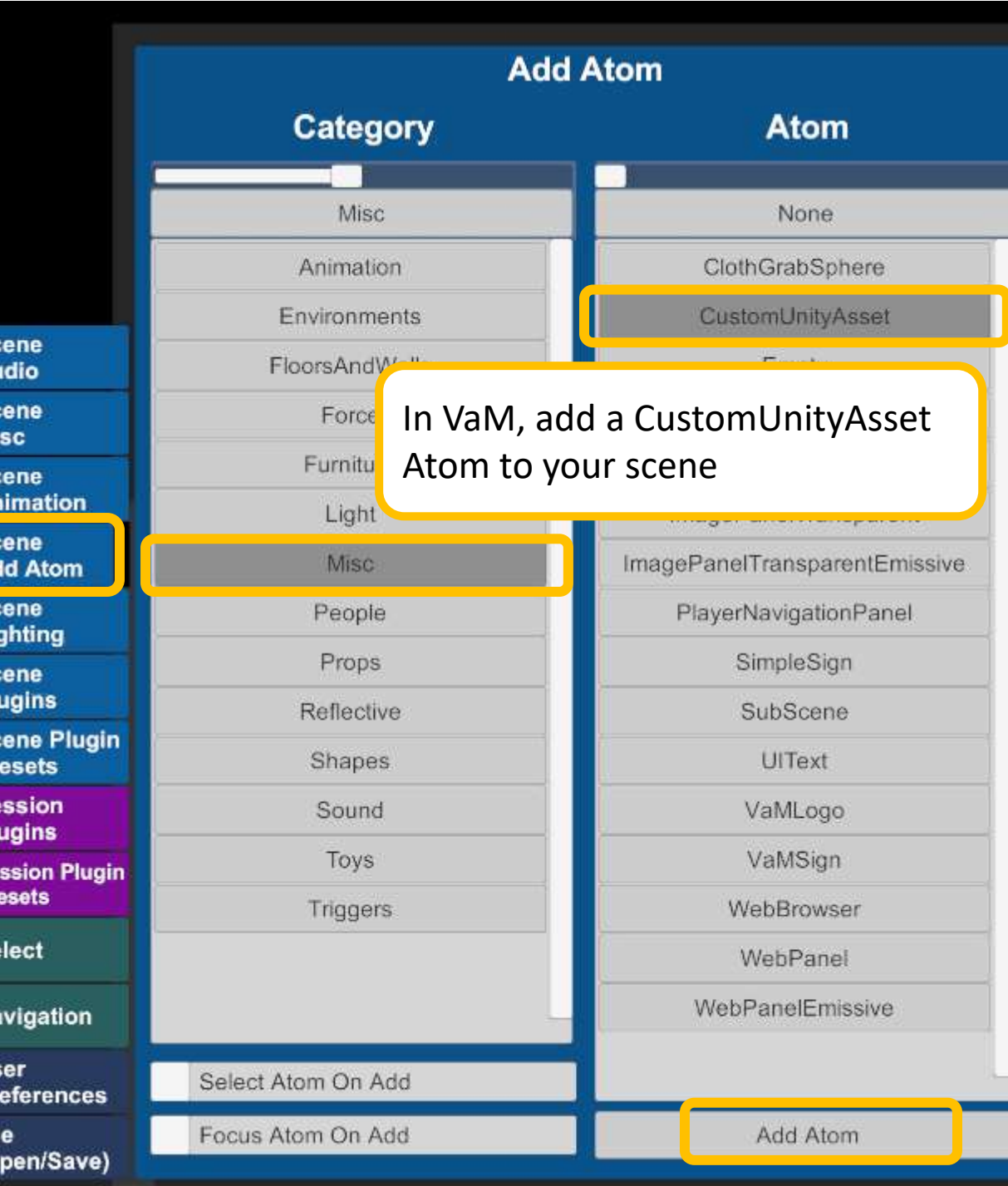




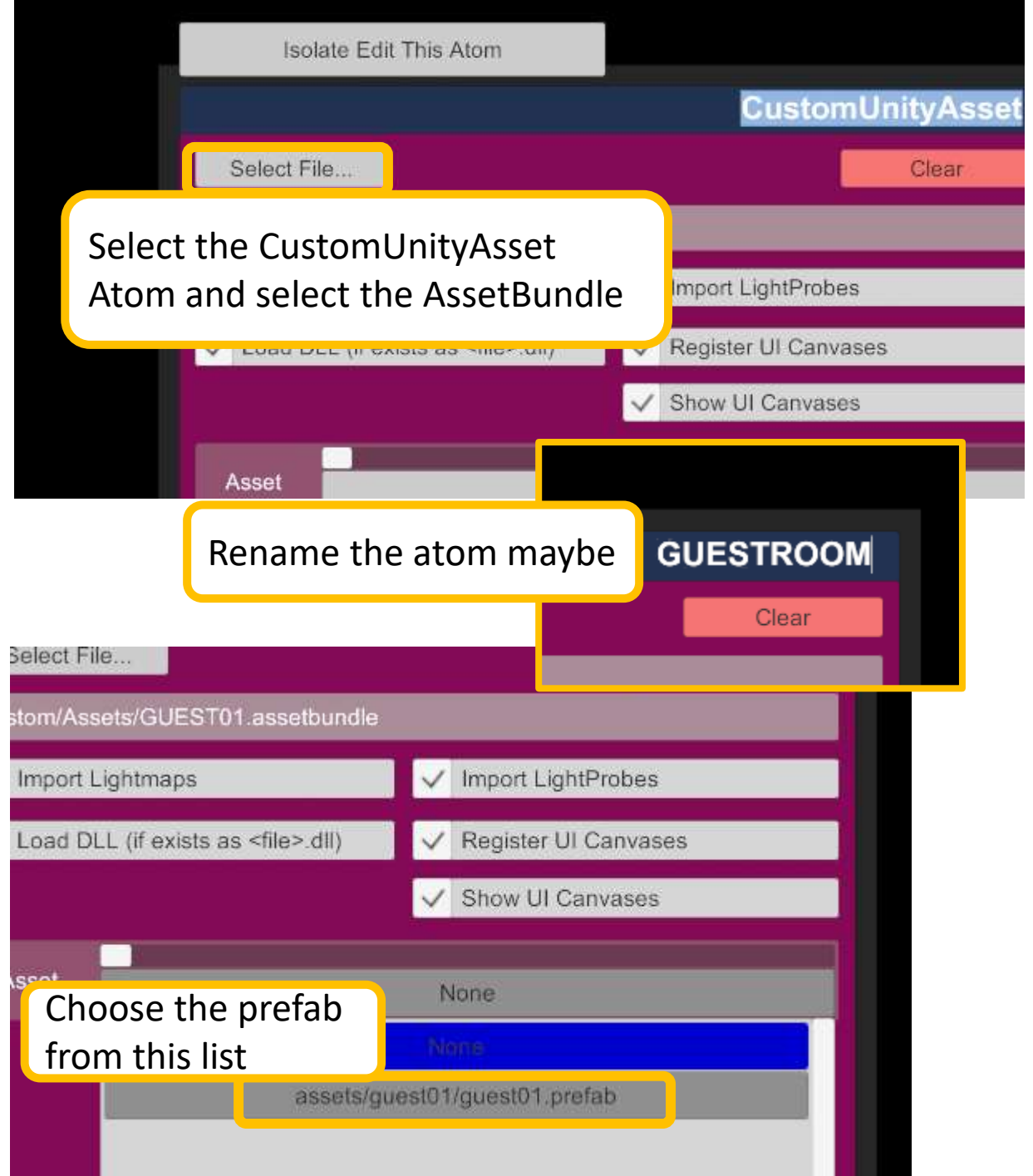
Now that the model is adjusted in space and has collision mesh, drag it down again to create a Prefab, or «snapshot» that can be exported



Check this message in the corner for confirmation that the AssetBundle has been copied



In VaM, add a CustomUnityAsset Atom to your scene



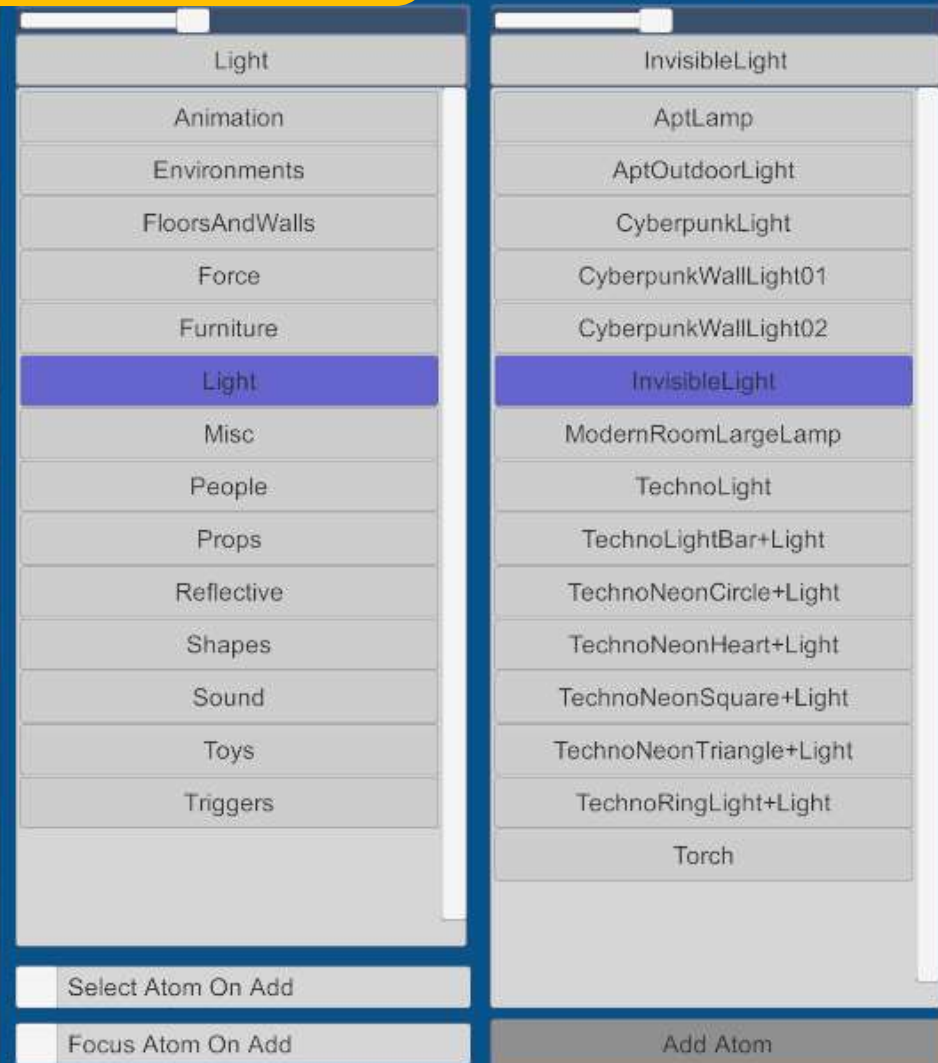
Select the CustomUnityAsset Atom and select the AssetBundle

Rename the atom maybe

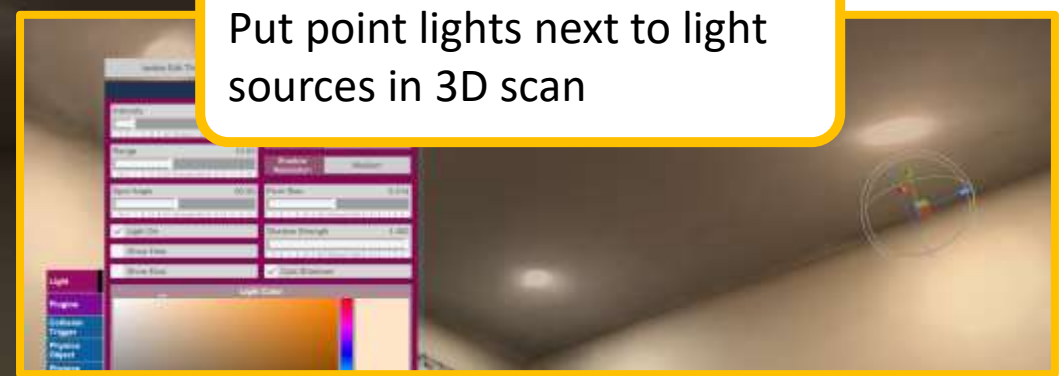
Choose the prefab from this list

The scene will be dark until you add a light. Do it now

Scene
Audio
Scene
Misc
Scene
Animation
Scene
Add Atom
Scene
Lighting
Scene
Plugins
Scene Plugin
Presets
Session
Plugins
Session Plugin
Presets
Select
Navigation
User
Preferences
File
(Open/Save)



Put point lights next to light sources in 3D scan

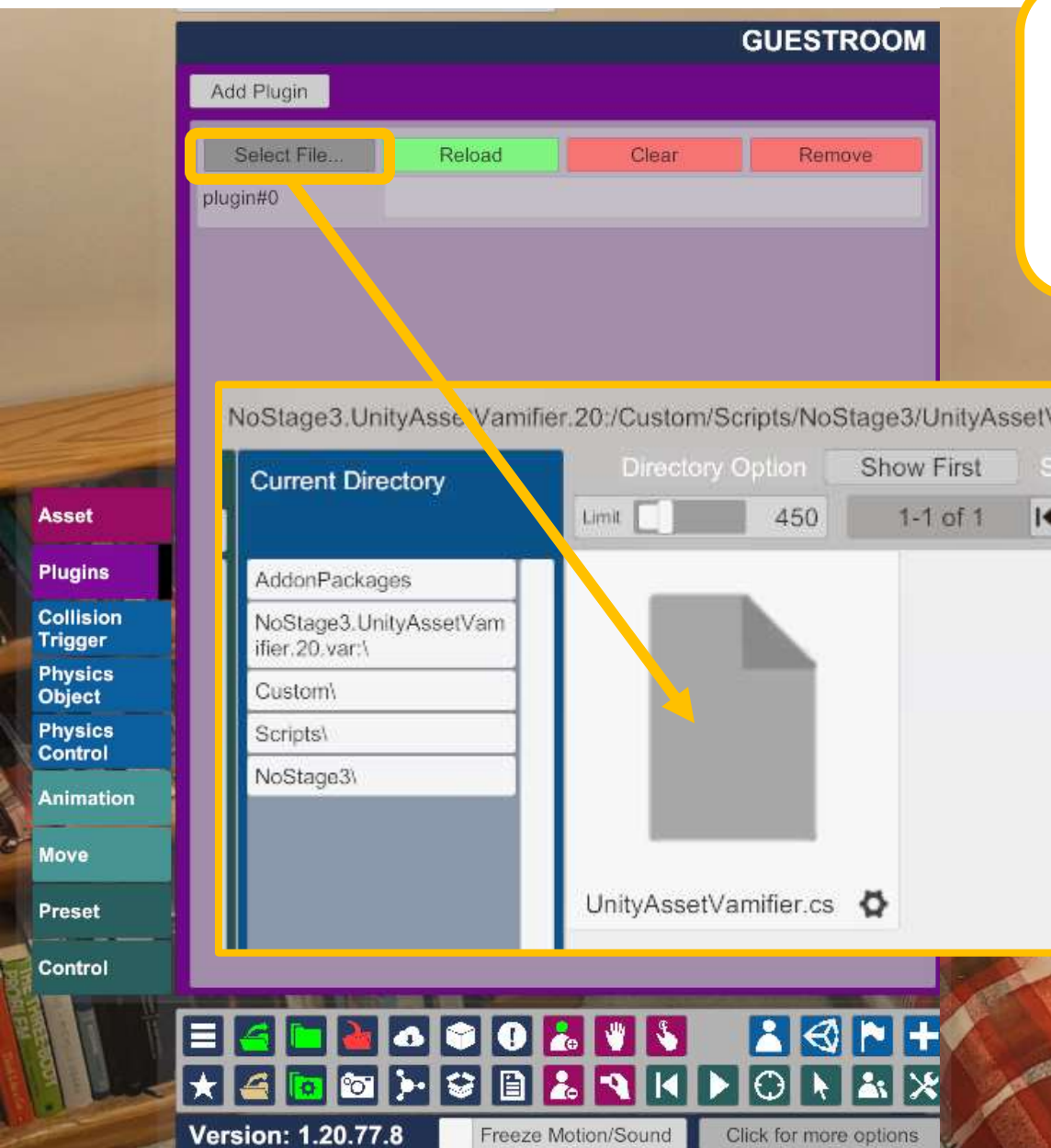


This room has 4 spot lights, so we create 4 point lights slightly below



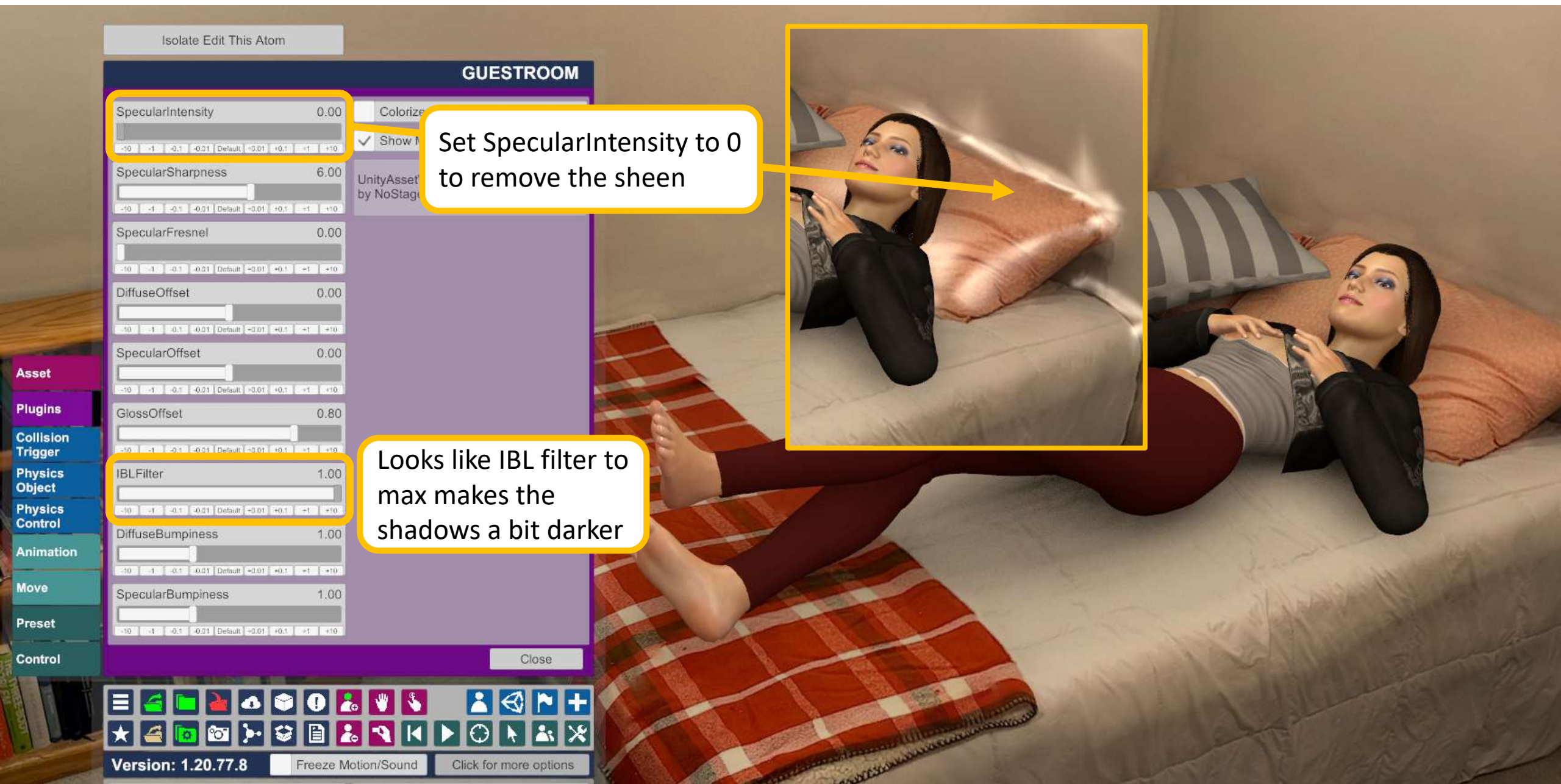
Light params

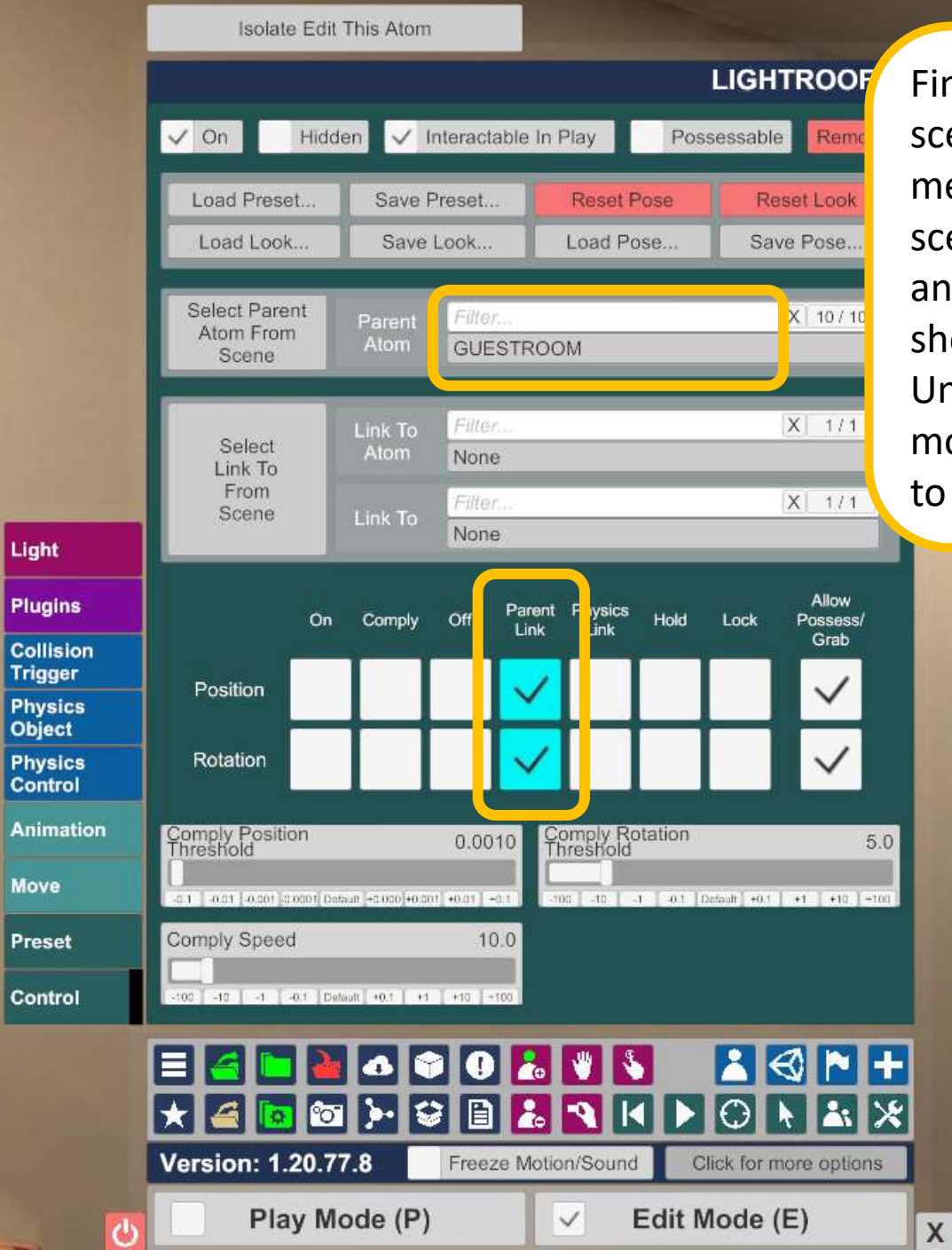




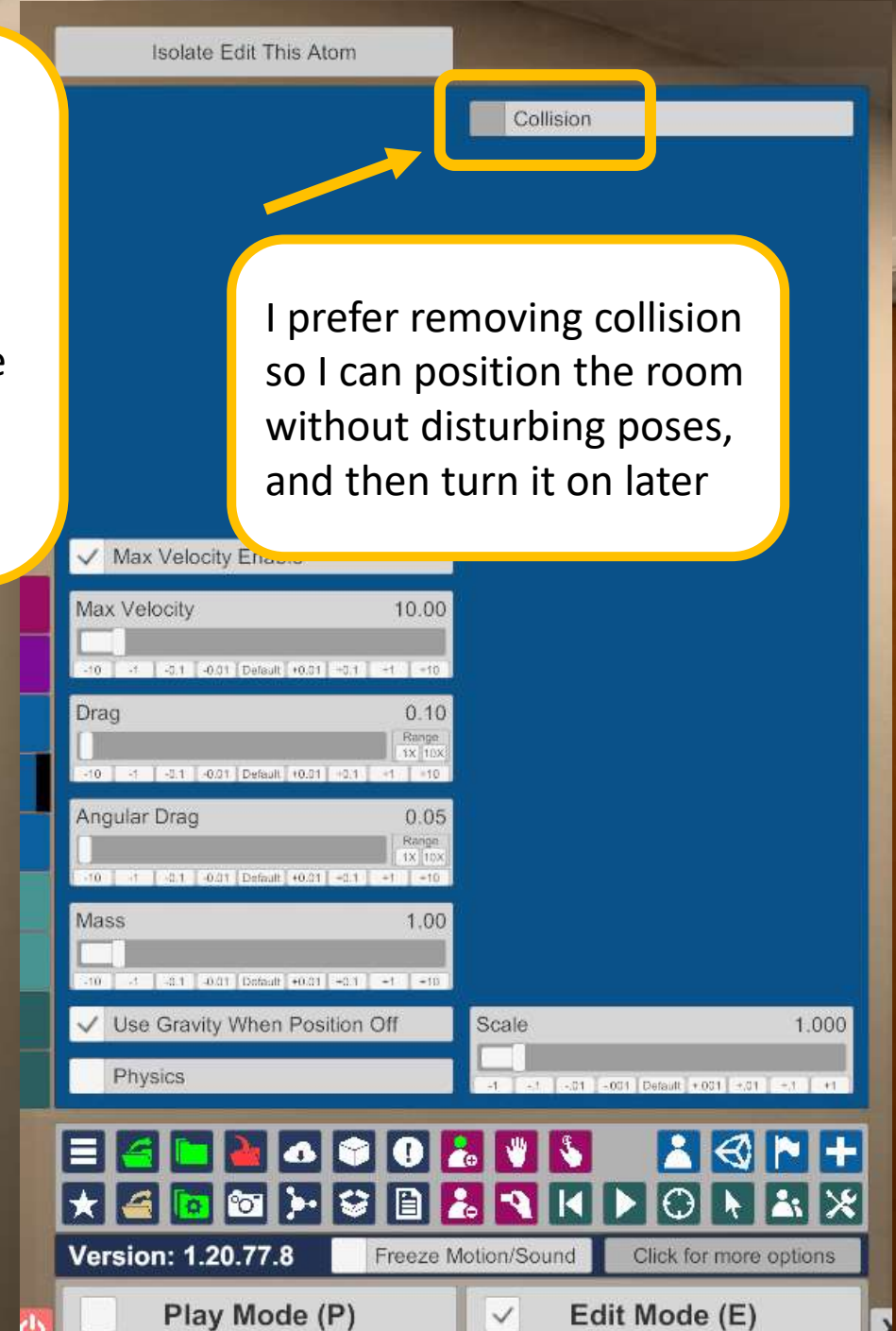
When adding bodies to the scene, the shadows will not look realistic. We need NoStage3's excellent UnityAssetVamifier to soften the shadows







Finally, to make the scene ready for merging with other scenes with animations, the lights should be linked to the UnityAsset so you can move the room freely to fit other animations



I prefer removing collision so I can position the room without disturbing poses, and then turn it on later



Have the 3D scan as the starting file and merge load the scene with the animations / poses. Remove room atoms and lights from merged file

