Game models into detailed printables

a brief guide by Kaucukovnik VI.

Even the most graphically advanced games don't have all the details modeled in 3D, instead using textures and normal maps to convey small features and surface irregularities. By simply making the 3D geometry printable you will lose some amount of detail, more so with older games.

Let's see how to get the most out of game models for the purpose of 3D printing.

This guide assumes some basic knowledge of 3D modeling and texturing. I'm using the terminology of Blender, but other software like 3dsmax should have all the required functionality as well. I share my findings, and point out possible pitfalls that I had to learn the hard way.

I don't claim to be an expert, this is just a method that has worked well for me to make hundreds of figures. Hopefully this information will be useful for other game art enthusiasts.

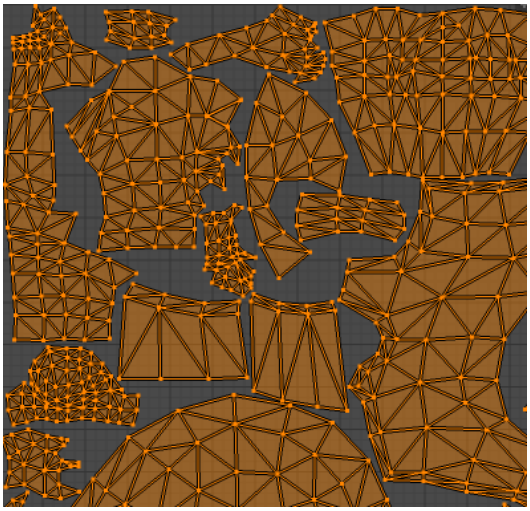

Here we have all parts of a Requiem shaman armor set imported. I have mapped all the textures onto a single sheet, merged all the geometry and deleted excesive skeletons left by individual body parts. One skeleton for the entire model is all we need here. Luckily Requiem stores the whole skeleton with each body part, so it doesn't matter which one you keep.

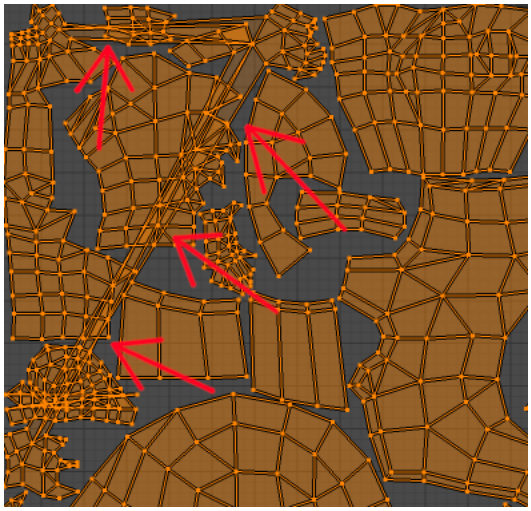
Textured the model looks nice and detailed, but the geometry itself is very low-poly and plain. That's okay, let's clean up the geometry first.

Before you move on it's a good idea to try importing animations to make sure they work right. A brand new animation issue can surprise you with any model, even if the same game gave you no trouble with a dozen models before.

If the model gets messed up when using its correct animation file on it, you can try moving the gometry in relation to the skeleton. Sometimes they are just misaligned. Look for the "sweet spot", where the model looks the least deformed, along one axis first, then do the next axis. Rotation or scaling might work as well.
If a game has multiple variations on the same creature, you can also try using the skeleton and/or animations from the other versions. For example Requiem's undead warrior twin blade animations are busted, but animations of the two-handed weapon variants work fine for all the models.

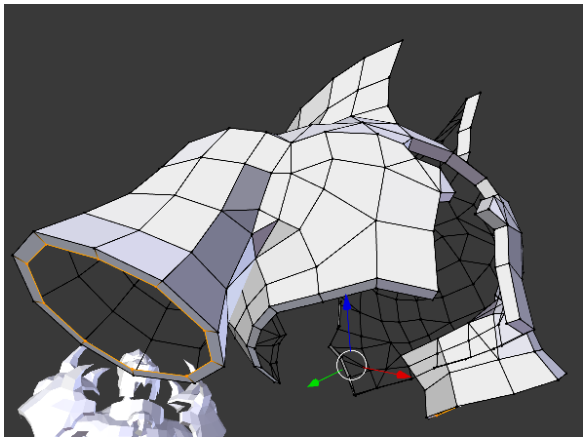
First convert **triangles to quads**. You can skip this step, but quads usually provide cleaner surfaces when subdivided later on. You need to do this first. If you do it after merging duplicate vertices, some triangles belonging to different UV islands are almost guarranteed to be merged and textures on them will get stretched and messed up like this:

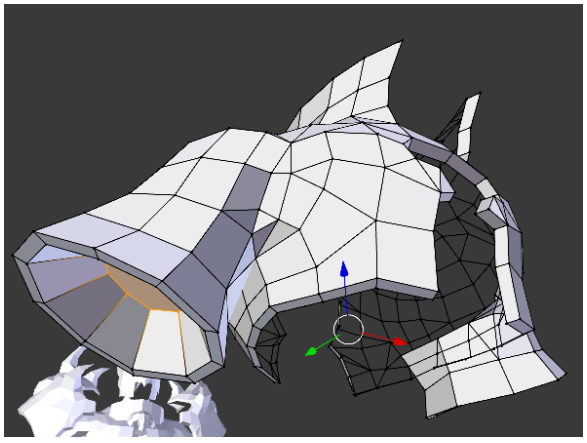


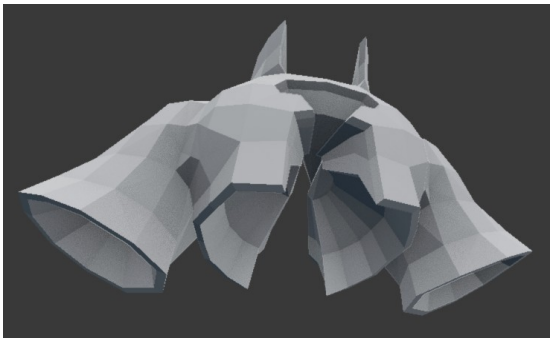


The next steps depends on the model. Sometimes parts of geometry are already separated in a way that is advantageous for us, other times **merging duplicate vertices** makes selection of individual parts easier. In this case I'm opting to leave them as they are for now.

Now I select the tunic, separate it into a new mesh and move it aside. Snapping to grid helps with alignment. Then I merge duplicates, if I haven't done so before, and close all openings to get a fully **enclosed (manifold) mesh**.







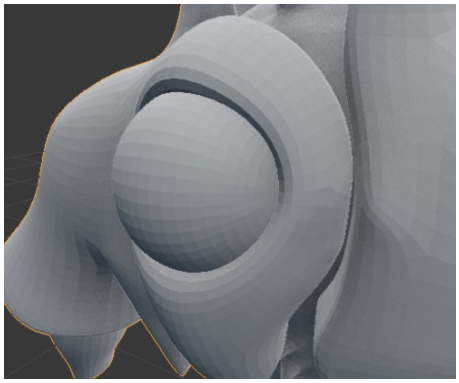
I do the same for all distinct parts of the geometry. The finished ones get joined into one output model.

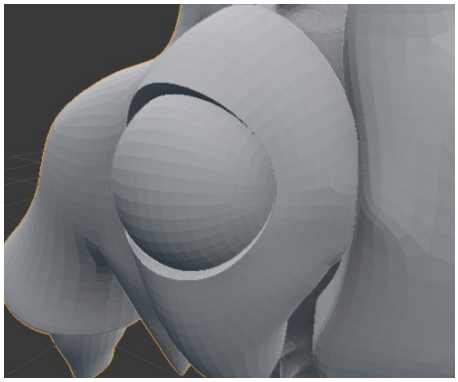
You can add some modifiers from later steps now to see how things are shaping up. Sometimes I do multiple steps in one go for every part of the model. Depends on defining features of the model.

Zero thickness bits like cloaks or skirts have to be solidified or extruded to actually have some volume. Bracelets, belts and such are best turned into solid cylinders to completely avoid gaps between them and the main body. I like to make dangly bits very thick and lean them against larger surfaces to make them easier to print and more durable.

To check whether your model consists of enclosed shells, I recommend using the "select non-manifold" function. It also detects inner walls that need to be removed. It's easy to overlook those issues and they are more cumbersome to deal with later on.

When you have the entire model cleaned up, it's high time to add the **Subdivision Surface** modifier. It has two purposes: one is to smooth out the geometry, the other is providing dense enough mesh for the following modifier to work with.
If you now get holes that were not visible before, you either forgot merging duplicate vertices earlier, or the model has some "dirty" geometry that needs manual cleanup. It wasn't an issue in a game, but it would be for a printable model.
Now crease edges (Shift+E) you want to keep sharp. If you want to get a sharp corner or point, all the edges connecting to the point have to be creased.





Now let's add a **Displace** modifier. Create a grayscale version of the texture and use it as the displacement map (aka height map) for now.
If a game uses **normal maps** you can instead convert the normal map into a height map (I use the program **NJob** for this).

Adjust Subsurf modifier levels so that all the detail is clearly visible. Usually 4-5 works well. Also depends on your PC performance.

Adjust the displacement modifier's strength to make the details as pronounced as needed.

Set displacement base level to a value slightly below the darkest color in your heightmap. Zero for a heightmap with black surfaces, 0,5 for 50% gray for baseline, etc. If you set it too high, sharp spikes and other thin bits can start turning inside out as the heightmap commands them to displace inward.

We have the details now, but they are somewhat crude and noisy. Let's do something about that.

In case of super clean normal maps the model might look great already. Usually there will be a bunch of seams. If you are using a diffuse map, the surface is most likely a mess.

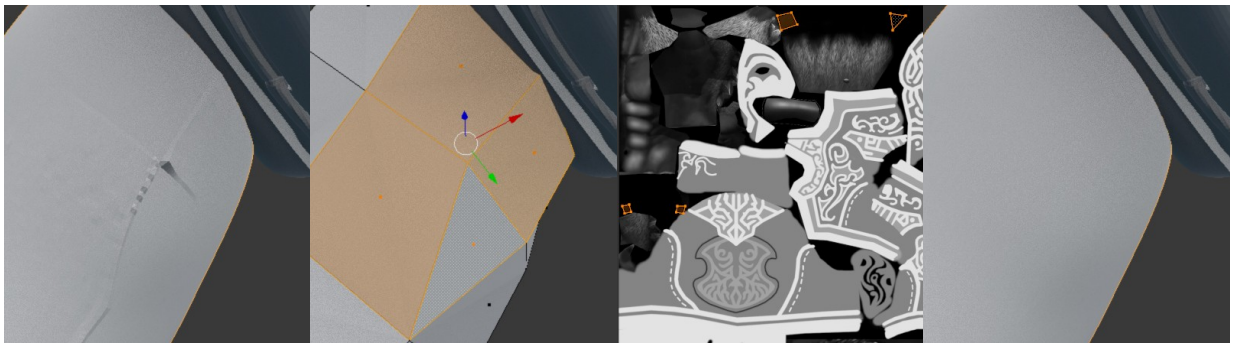
Here are some ways to improve your heightmap:

- high pass filter - only preserves small detail and discards large shapes. For quality normal maps can be all that is needed to get a nice, clean result.

- manually repaint offending parts - this tends to be time consuming, but in case you only have a diffuse map to work with, it's the only way to achieve clean surfaces. Not that important for organics like monster skin, scales or bone, but crucial for armor, tech stuff etc.

-to get rid of seams (usually on the inner sides of arms and legs, sides of the torso...) you can export the UV map from Blender and paint over opposing edges of the affected UV islands with the same color. This will make the height on all sides of the seams even, eliminating them.

On this set of images we can see a 4-way seam. I filled most of the base body surface with black, and now these few polygons (highlighted on images 2 and 3) remain to be painted black to the very edge to get rid of the seam.



- fill areas with textures of corresponding materials like fabric or chain mail. Easier than cleaning up textures already present. By adjusting contrast of the texture you can control the depth of detail on these surfaces. Scale textures so that the detail doesn't get lost in your intended print size.

The main deal with height maps is to make raised parts brighter. Straight slopes are achieved with smooth gradients, roughness can be easily done with a noise filter, etc. Beware of too high contrast, it's very easy to accidentally turn small rivets into long spikes. Sometimes it is necessary to simplify things to keep the surface clean and tidy.
You can use raster or vector graphics to create height maps. Both have their pros and cons. Go for whichever you are more comfortable with. Either way try to keep your texture editable (be it via separate vector objects or raster layers) so that you can adjust different elements as you go on.

As far as I'm aware there is no magical automatic way to process textures for neat displacement maps. AI upscaling can improve low resolution textures, but can't help with proper depth in different spots or remove painted-on details that aren't supposed to represent plastic details.

It's up to you how many seams and other issues you remove at the displacement level and what you're going to deal with via manual sculpting later on. The more poses or variants you plan to make, the more attention should be paid now.

Here we have the entire texture, before and after manual repainting into a height map:

And the model with the height map applied:

I have also made a variant of the tunic, based on an official piece of artwork:

When you have the base figure processed, I recommend saving a master file you can return to at any point. It's easy to accidentally save after applying a modifier so that you can't go back anymore.

I personally store all compatible sets from a game in a single .blend file so that I can pick and choose their parts comfortably.

Now it's time to **pose** the figure. Disable subsurface modifier to get good performance and **import an animation file**. Find a good keyframe and enable subsurface again. Check if the model looks right and nothing deforms or clips badly.

If some parts of the model are fixed to one spot instead of animating with the rest of the model, some vertices are probably not assigned to the correct **vertex groups**. Selecting suspect vertex groups should eventually reveal one or more with holes that corresponds to the non-moving bits. Select those vertices and assign them. Continue until all errors are fixed.

If there is **clipping or deformation**, you can apply the Armature modifier (on top of the stack, should be present by default as soon as you imported the model). Now you can manually edit the mesh in your chosen pose. Reshape or remove any offending bits.
This is also the time to edit hands and fingers to hold items properly and make other adjustments to the general shapes.

Now apply the Subsurface and Displace modifiers. Go to Sculpt mode to get rid of **seams**, inflate overly thin bits, etc. Any issues you missed earlier will now have to be fixed for every exported pose individually. If there are no such issues and you don't want to do any other sculpting, you can skip applying the two above mentioned modifiers altogether.

Now that your model looks just like you want it, it's time to **decimate** it to get rid of excessive polygons.

Decimation leaves out vertices that are the least important to keep the model's shape. Ratio of 0.5 halves the polycount, 0.1 cuts it down to actual one tenth. Polygons on flat surfaces are the first to go, so if your model has a lot of those, it can take a lot heavier decimation than one fully covered in fine detail.
Naturally the modifier doesn't discriminate between important details and noise.

The ratio depends on how many subdivisions you chose earlier. Try to find the best middle ground between details and file size. Even if you end up with the same final polycount, more subdivisions followed by heavier decimation yields sharper details. I suggest going as high as your hardware comfortably allows for. If you are about to export an army of fifty 1/72 figures, you probably don't want to wait five minutes to decimate each of them. For a single big display piece it may be very well worth the wait.
To save on performance I usually export the full detail models first and decimate them separately in a new Blender scene. I can then line them up there for taking a picture and exporting the final models.

If you have a potato computer, you can do sculpting after decimation. You will have less detail available (doesn't even matter for seams and artifacts on flat surfaces), but performance will be better.

SPECIFIC GAMES & BLENDER

In general these are some sites to search when it comes to dissecting game data:
https://forum.xentax.com - general game data research
https://www.gildor.org - Unreal engine exporter with a lot of game-specific support available

Requiem Online
Blender 2.78 - this version's nif plugin works with all Requiem character models and most monsters
Blender 2.9 - try if 2.78 fails, often the combination of both is needed - 2.78 for the model and 2.9 for the skeleton and animations.

Diablo II
Noesis with the granny2 plugin: export model as .fbx, including animations
In Blender 3.2.2:
Multi-part models only contain the corresponding bones with each part. Torso part usually contains everything except for hands and feet. Hands can be cheated by assigning all hand vertices to forearm vertex group. Fingers won't move though. This leaves us with two skeletons per animation - one for torso, one for legs.

- Sometimes you can pose the entire model with one part's skeleton and then merge another one into it. Assign the entire model to the torso skeleton, pick your keyframe, feet will be off. Join the same animation's legs skeleton into the torso skeleton, feet will be fixed, but they will be frozen on this particular keyframe. You will have to undo the join operation or return to earlier saved file to get them moving again for another pose.
- Other times you have to cut the model into pieces and animate each with its own skeleton. Keeping the original separate parts isn't enough, for example an upper portion of the "legs" piece may animate wih the torso's skeleton instead. It's a mess, really.
- Light, medium and heavy armor can each have its own skeleton, in which case you have to import animations for each version.

Worst case scenario you import one animation per each body part / armor tier combination.

Otherland
umodel, game specific build, available here (8th post)
Blender 2.9 with the psk/psa plugin confirmed to work