Yet Another Vanilla Upscaling Guide

By Lighty c:

Introduction

Hello there! I'm Lighty. You might recognise my name from all the mods I've made, but if you don't, hi! I specialise in vanilla upscales for Yet Another Body. I've had a few people message me to ask how I do it and for tips, and I've been intending to make a guide covering my process for a while, so here it is! Considering I was completely new to Blender when I started using the devkit, I wanted to go through my process in detail. It might end up being a little wordy for people who know this sort of thing already, or if you're interested in upscaling already modded gear, but hopefully it's sectioned out nicely so you can dig out just the bit you want to know about. I've also tried to **bold** all **keyboard shortcuts** and **instructions** to make it easier to skim.

Disclaimers

I keep my devkit and addon up to date (and I recommend you do the same), but be aware as things change, information in here may become outdated and my screenshots might be a little different to what you're seeing. I'll update as and when it's required (and I have time) though!

Latest update: 4th Oct 2025 **Devkit ver: 4.2.0r6.blend** **Addon ver: v1.0.2**

Don't forget, there are many different ways to approach upscaling. You don't have to follow this guide precisely if you find a different method works better for you, I simply wanted to document my process so someone with little to no blender knowledge could try their hand at upscaling. Keep in mind this process isn't perfectly linear and you will likely have to skip to the Exporting/Testing stage a few times during the process, so I've included links to other sections where required. The latest devkit is considerably different to the previous versions, so I'd recommend going over everything even if you've followed this guide before.

I am not a blender expert - I studied Game Art and 3D Modelling at university but was taught in 3DS Max - while a good chunk of the skills are transferable, I've never really used blender for anything except YAB+ upscales, so there absolutely will be settings, buttons or plugins I've missed that might make my process smoother, but we're all learning as we go. :D

Also fair warning, there's screenshots of the devkit in here with a naked body in it. Wouldn't recommend opening at work or in front of your grandma.

Any issues you have to report, questions to ask or just screenshots of your first vanilla attempt can be directed to me on Discord (@fightylighty or by a ping in the YAB/Rue server), ko-fi (https://ko-fi.com/lightyy) or BlueSky (@fightylighty.bsky.social). I've also included a link to a feedback form at the end of the guide if you'd like to give anonymous feedback, too.

Table of Contents

Where should I start?

If you're completely new to Blender and the YAB devkit, I'd recommend reading the whole document.

If you know your way around Blender and/or are happy with your layout and settings, feel free to skip Personal Settings in the Blender Setup section, but other sections are still important.

If you've already started on the upscale in a previous version of the devkit, go to Moving to a New Devkit.

If you're working with an already modded mesh, you will still want to check out Upscale Specific Setup, then you can skip most of Preparing the Exported Mesh (namely the Retopology sections), but I would still recommend skimming the rest of the section if you exported your modded mesh from TexTools, since it will split the model along UV (texture) seams.

If you have an already YAB-ed mod but want to change the size, you can skip straight ahead to Size Conversions, after following Upscale Specific Setup.

If you have an already YAB-ed mod and want to add YAS/Yiggle weights, you'll want to skip straight to Weighting, though you may still want to check settings from Upscale Specific Setup.

If you've followed this guide before and just want to know what's changed with the newest devkit and addon, quite a lot has changed! Import, Size Conversions (depending how recently you played with the devkit), Preparing for Export and Exporting, namely, but it may be worth skimming everything - though the process of the initial refit is much the same.

This guide covers the process of upscaling gear using the YAB devkit specifically which includes Rue, LavaBod and YaMasc bodies, as well as YAB Mini, UranusRE, and Yet Another Tiddy shapes. **If you're looking to upscale your gear to a different modded body**, the process for the initial refit will be more or less the same but your devkit and therefore the size conversions step likely won't be - feel free to read through to get a feel for the process, but bear in mind you'll have to find parallels to your body's devkit as you go.

What should I work on?

I think you're more likely to be invested in this process if it's a piece you like and you want to see upscaled, but in case you have several in mind or you're simply looking to follow along with something simple to get a feel for the workflow, here are some pointers on how to pick something a little easier. These definitely aren't hard and fast rules! I'd recommend going for something that obeys most of these criteria but still has a few challenges to push yourself.

Things that are harder:

- Shawls/ponchos/other draping fabrics that cross over the upper arm and chest are tough to weight.
- Gear with patches of skin showing in multiple areas are harder to snip out unnecessary skin from which may lead to clipping issues to touch up later.
- Joints are tough spots if you're dealing with both the skin and the gear at the same place. Try to avoid gear that stops too close to the underarms or *especially* the crotch - underwear is hard to get right!
- Tight skirts are really, really tricky with XIV's skeleton.

Things that are easier:

- More skintight gear will follow the weights of the body much better than looser fabrics, meaning they require less tweaking at the weighting stage. They are also less likely to need glove/boot compatibility.
- Generally speaking, trousers are far easier than tops.
- Gear without any skin showing is probably the easiest - but adding the body/skin textures is an important part of the process, so certainly don't shy away from it.
- Booty clearance for looser skirts/long shirts is a lot easier than you probably think it is - it's just a matter of pulling the skirt out far enough.

If you're just looking for something simple to follow along with, here are **some suggestions:**

- Skyworker's Singlet
- Leisurewear High-Cut Knit Shirt/Top (a good chunk of this guide follows me upscaling this one!)
- Valerian Brawler's Bottoms
- Wake Doctor's Bottoms

A Few Words of Advice...

Don't be scared! While this process is pretty complicated (even for the experienced modders!), none of this is game-breaking stuff. You're always able to start over if something goes wrong and try a different way. We have undo buttons and ways to repair our game files, nothing you do here is impossible to reverse.

Remember that this is complex stuff. Struggling is okay. Not understanding the guide is okay. I *never* would have managed to learn 3D modelling on my own - I was taught the basics over several years in college and uni. You're doing something amazing just by having the dedication to try to learn this by yourself.

There are no stupid questions. There are some that may seem obvious to people more experienced, but I promise you, all of us asked a few of those kinds of questions when we started. Don't be afraid to ask for help, *especially* if it seems like a dumb question - those are usually the ones that will hold you up the most and will be the easiest to solve. The sooner you get it clarified, the sooner you can move onto learning something else!

Don't expect to do this all in one sitting. You can (and should!) take this in manageable chunks. The save button is there so you can come back to it the next day if you're tired or frustrated. My first upscale took me several days to complete but the more I practise, the quicker and easier it gets (but I still make mistakes! That's just part of life).

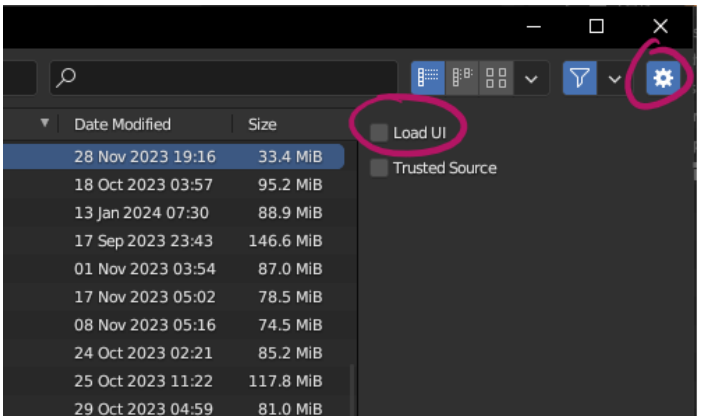
Your first upscale isn't going to be perfect. None of them are! But you aren't forced to say it's 100% finished either - I go back and update some of my older mods when there's something I think I can do better now that I'm more experienced. Try your best for now, and then you can come back to tweak it later as you learn more. Nothing is "finished" until you decide to put it down for good.

Blender Setup

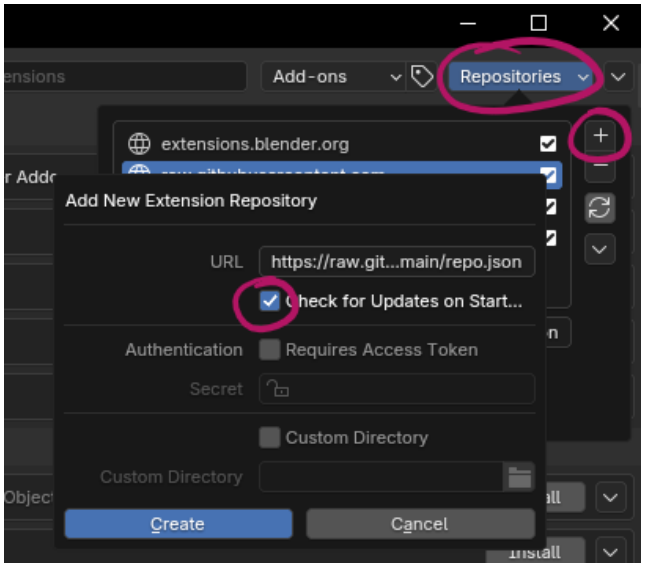
The Devkit and the Addon

The very first thing we need to do is **open Blende**r - ensure you have **version 4.2.1 or later**, as the devkit and the addon will not function with earlier versions. I personally use 4.2.11 now (4.2.2 previously), as that was what the most recent update was written in. Using a previous version *should* only give you a warning that scripts may not work, but I don't know if it'll produce any strange behaviour - it shouldn't! But just a warning.

Now, download the Devkit/Scriptkit from the **#wip** channel of the YAB server: https://discord.gg/bnuuybooty (it's a big server, I promise nobody will notice nor care if you choose to grab it and scoot and/or mute - but I would recommend sticking around for new versions if you want to keep upscaling). The latest as of this guide update is **"Yet Another Devkit 4.2.0r6.blend"**, but if you see a more recent version, grab that!

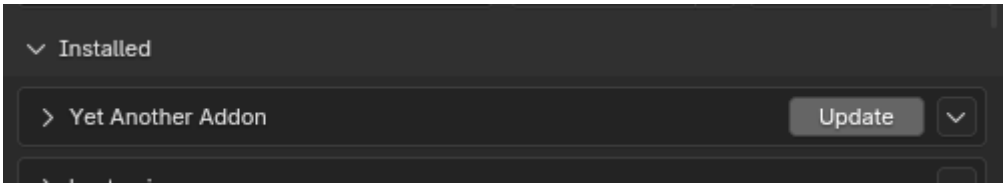


Go back into Blender, hit **File > Open** and find the **devkit.blend** file. If you have a custom layout or don't want your layout to change, go into the **additional settings** (the little cog in the top right) and **turn off "Load UI"**. If you get a pop-up about **scripts**, **allow them to run**. They are the foundation of the devkit functionality - won't be a whole lotta use without 'em.

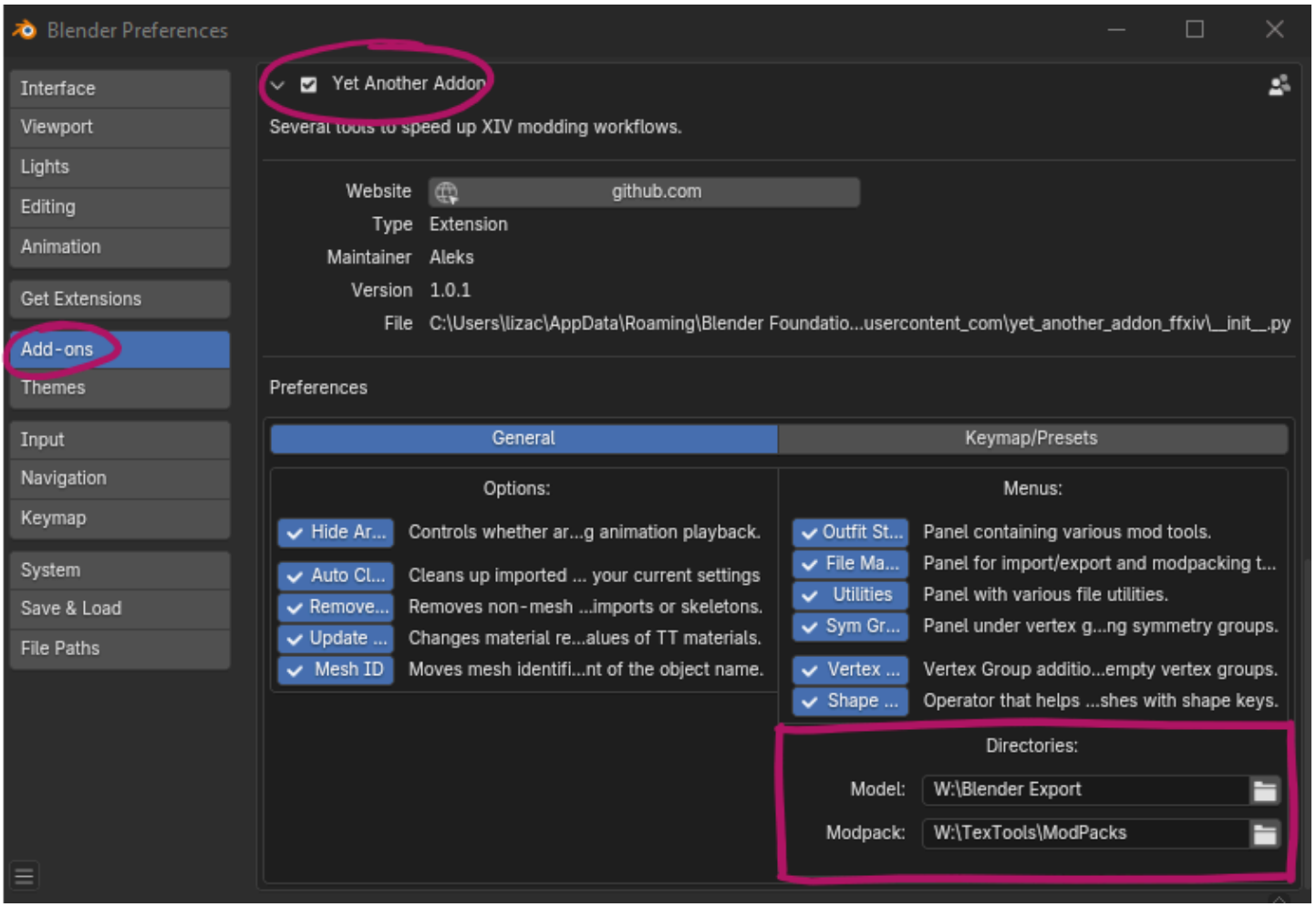


You can find the Addon here: https://github.com/Arrenval/Yet-Another-Addon. To add it, you'll want to hit **Edit > Preferences** and navigate to the **Get Extensions** menu. Click on the **Repositories** dropdown on the top right, click the **plus** to add, choose **remote**, enter "https://raw.githubusercontent.com/Arrenval/Yet-Another-Addon/main/repo.json" as **the URL** and turn on **Check for Updates on Startup**. Then hit create!

Once you have the Addon added, this **Get Extensions** menu will also be where you go to **update**. It'll look like this when there's a new version. It's good to get in the habit of checking it every time you open Blender!



If you intend to use the Devkit's Exporter and Modpack Creator to export/pack your models (which I recommend!), go to **Edit > Preferences > Add-ons** and scroll down to **Yet Another Add-on** and look at the **General Preferences**. On the bottom right is where you can set your default directories for **Model** exports and **Modpack** exports.

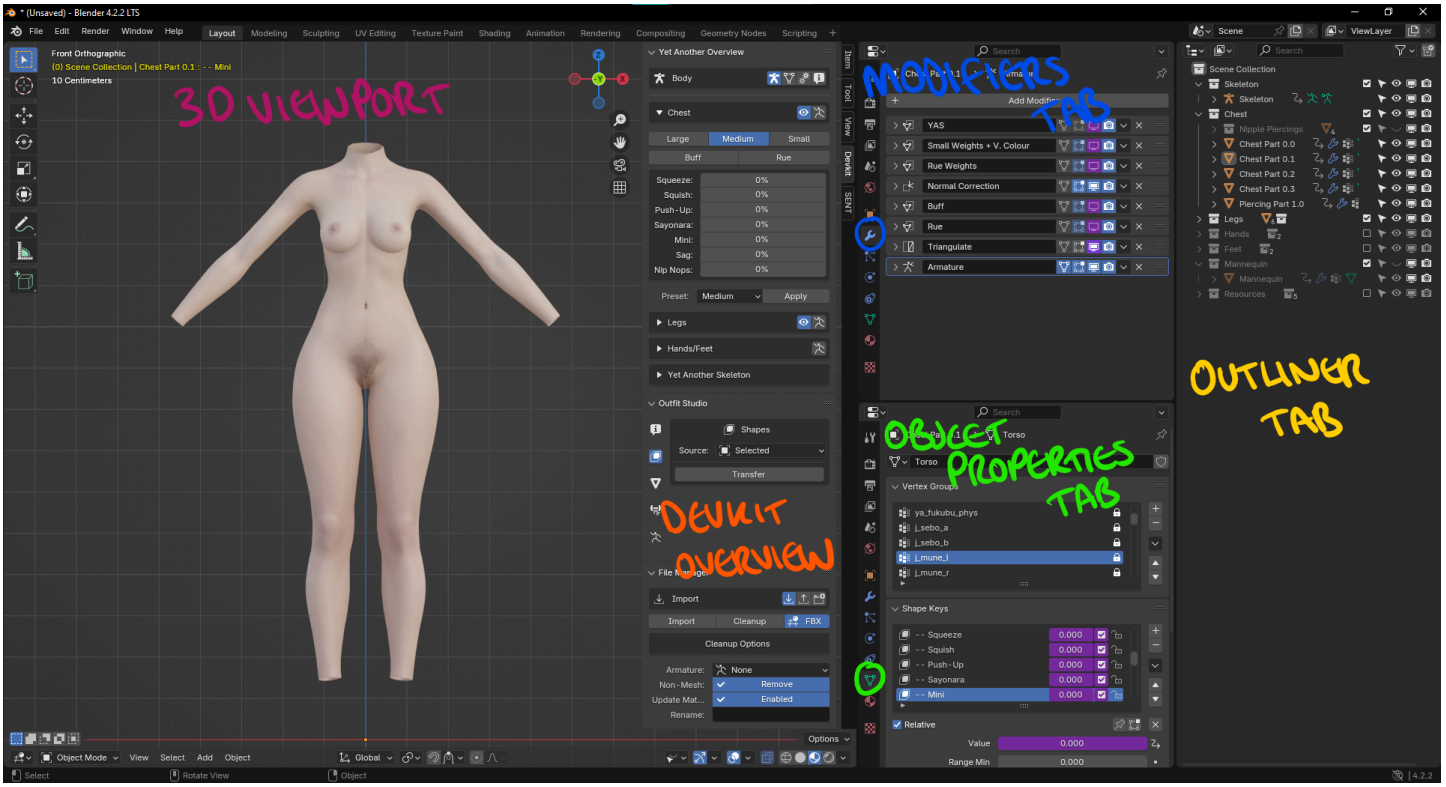


Personal Settings

This section is going to be me going over my layout and settings I find useful, so **everything here is completely optional** - I simply find customising it helps for my workflow. I would recommend playing with these settings yourself, even if you don't like my layout and want to make your own instead. If you already know your way around Blender and already have you set up how you like it, feel free to skip straight to Upscale Specific Setup.

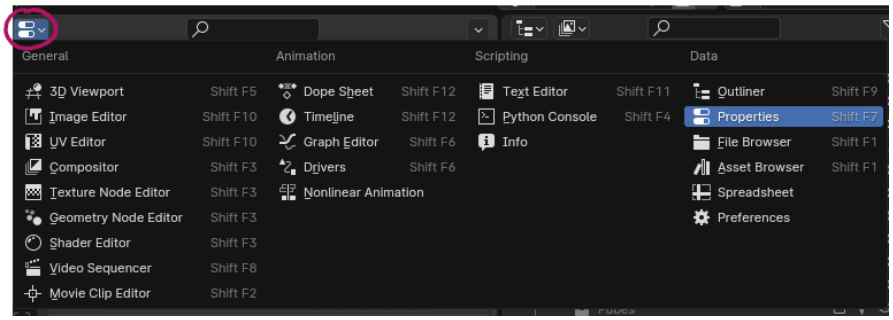
Since upscales is all I really use Blender for, every time there's a new devkit release I open that up, change these settings, and then **save that as my startup file** (this is simply what you see automatically open when you load Blender). It just means that it's open and ready to go automatically and there's no chance of me accidentally overwriting the devkit .blend file. You're welcome to work from the devkit .blend if you'd prefer, but I'd recommend at least keeping the .zip so you can replace the .blend later if you overwrite if you want to start over or upscale something else.

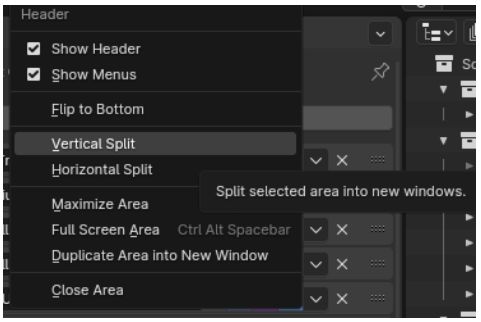
UI



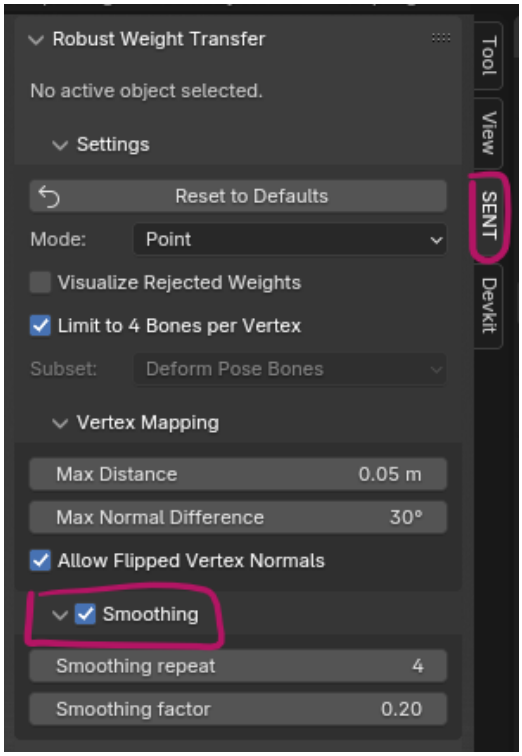
For my UI, I prefer to not have to scroll through the **outliner** so I have a full panel for that on the far right, and I don't like having to swap between the **modifiers** (top) and **object properties** (bottom) tabs, so I have two separate panels for those as well. On the left is my **3D viewport**, with the **Devkit overview** open on the side.

You can adjust the panels to your liking by hitting the little button on the top left to change what the panel does, right clicking the header and selecting vertical or horizontal split to make more panels in the same space (like my modifiers/object properties column!), and your traditional dragging to resize. (apparently Blender 4.3 has the ability to drag the docks around too!) You can also flip the header to the bottom, if you prefer to have your buttons there (which I have done for my 3D viewport). You're always able to save a new startup file later, so there's no reason not to tweak it on the fly and see if a new layout suits you better!





Other Add-ons

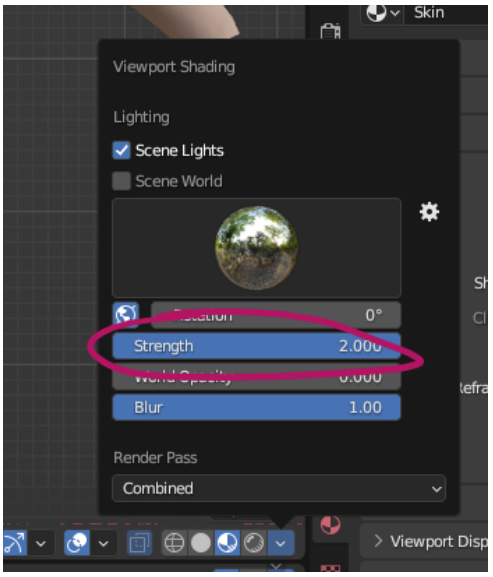


Additionally, there's one more Add-on I use alongside Yet Another Add-on; Sent From Space's [Robust Weight Transfer]. More on how it works when we get to that step, but it essentially does the work of a Data Transfer modifier used to transfer weights with an amount of smoothing built in. You can install it after you download (don't unzip!) with **Edit > Preferences > Add-ons > Install from Disk**. Open the "SENT" tab from the sidebar and **check Smoothing**. The default values otherwise work just fine.

I do go over how to weight without this, so if you don't want to download something else, don't worry!

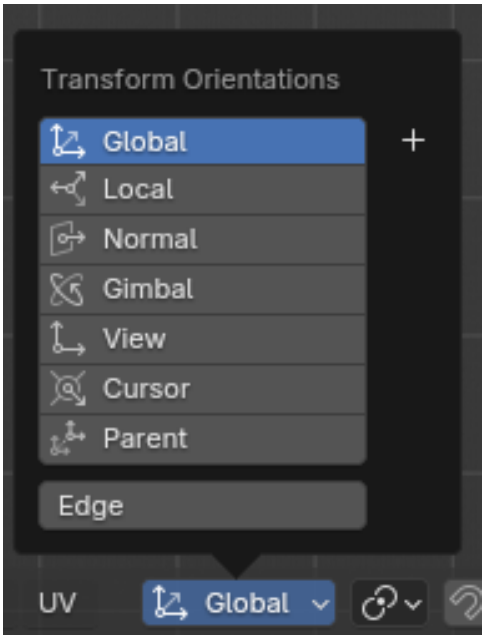
Miscellaneous Settings

I also like to **turn up the lighting** because it can be hard to see what you're working with (Update Material as part of the Cleanup Options in the Yet Another Addon will also help with this). On the far right of your 3D viewport header (or footer, if you've flipped it like I have) you'll have four balls with an arrow next to them - these are your **Viewport Shading Options**. Click on the arrow and **crank Strength up to 2**. You can also click on the orb to change the kind of scene lighting, which may make some details easier to see. Highly recommend this if the gear you're working with is very dark!

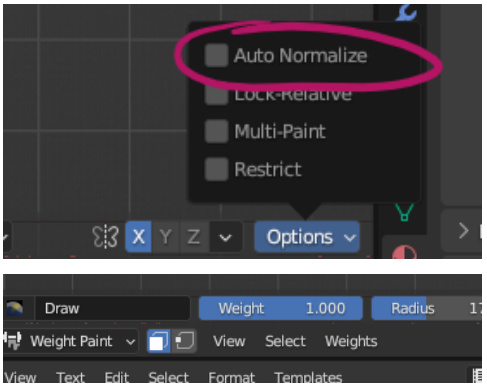


Side note, if you manage to accidentally turn off textures in your viewport, you'll want to make sure that third little ball is selected!

In **Edit mode** (hold **Tab** and move your mouse OR hold **Ctrl+Tab** to open the radial menu, then drag your cursor to the "**Edit mode**" option and let go), I **change both the gizmo orientation and transform orientation to Global** - both dropdowns are in the header/footer. View will move your selection perpendicular to your camera, so it can be very easy to drag things off at an angle if you aren't snapping your camera to specific viewpoints (which you can do by **holding Alt** while rotating the viewport with your **middle mouse button**, by the way).



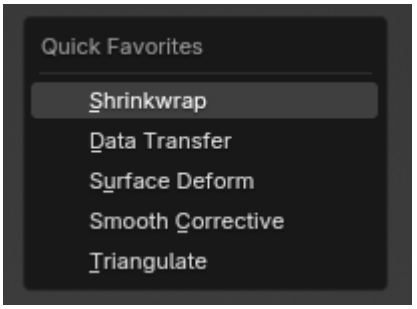
Then I go into **Weight Painting mode** using the radial menu (**tab or ctrl+tab**) again, **set the weight to 1.0** on the left of the bar at the bottom of the viewport and **turn off Auto-Normalise** from the Options dropdown on the right of the same bar. This is because I don't really paint any weights myself; I primarily use the weight painting mode to create vertex groups to use as saved selections to exclude parts from modifiers.

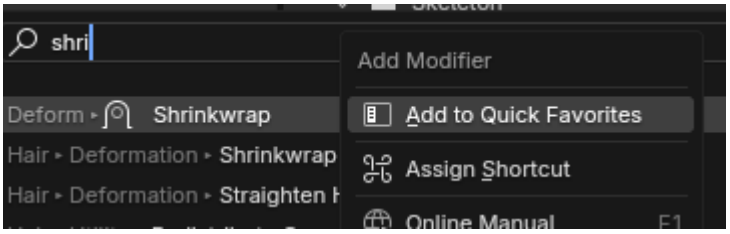


Once these things are finalised and set up, I hide the body meshes (just to make the vanilla imports easier to sort as soon as they're in) then I go into **File > Defaults > Save Startup File**, and hit it again to confirm.

Quick Favourites

Additionally, I recommend returning to **Object mode**, going into the **Modifier Properties** menu and click the **Add Modifier** button and searching several modifiers to **right click > Add to Quick Favourites** menu (which can be brought up with **Q**) to keep them handy - **Shrinkwrap**, **Data Transfer**, **Surface Deform**, **Smooth Corrective**, **Triangulate, Mirror** and **Mask**. These are saved in your preferences rather than the startup file, so you can edit these anytime. The tools you like to use may be different, so don't be afraid to stray from my selections!



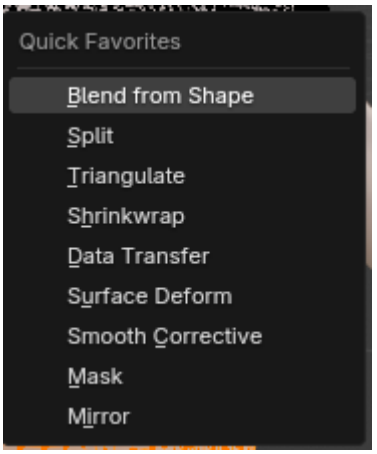


The Quick Favourites menu is contextual; it is different depending on which mode you're in. So, I will also go into **Edit mode** and add those same modifiers to my Quick Favourites, as well as two other tools; **Vertex** (from the 3D viewport header/footer bar) **> Blend From Shape** and **Alt+N > Split**, since they're tools that I use frequently. Then once more in **Weight Painting mode**, this time adding **Weights > Smooth**, **Weights > Limit Total** and **Weights > Normalise All**. Any other tools you find yourself using frequently in any mode can be added as and when you like.

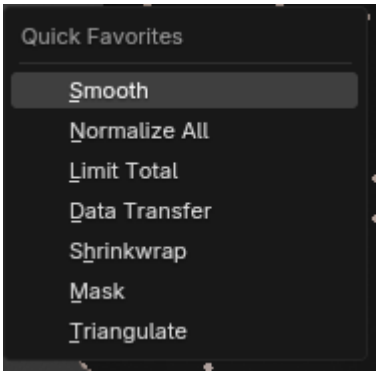
Here are my quick favourites in each mode to give you an idea of what I find handy (and which ones get duplicates elsewhere!):

Object Mode:

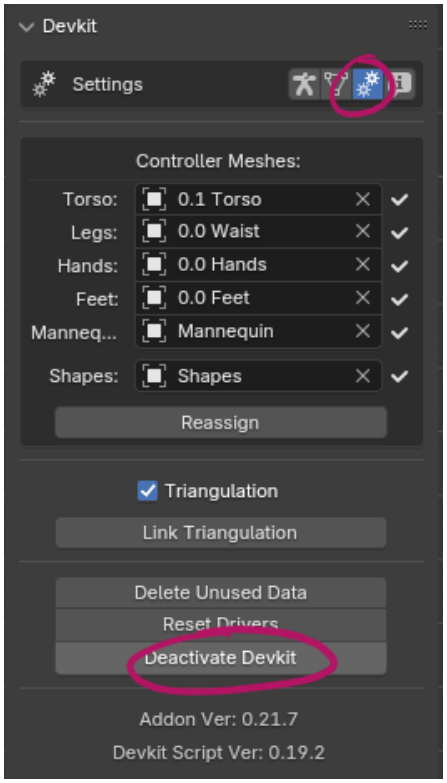
Edit Mode:



Weight Painting Mode:



Moving to a New Devkit

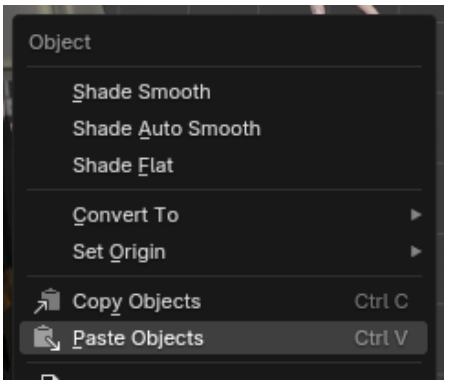


If you take what I said about keeping the devkit updated as well as trying not to do a whole upscale in a single sitting, you'll quite likely encounter a time where a new version of the devkit is released while you have a work in progress. Unfortunately unlike the Add-on, there isn't a one-click update, but I promise moving to a newer devkit isn't too complicated.

First, you'll want to open your WIP as well as a second instance of Blender with the new devkit loaded up (or your startup file, if you've already set that up with the new devkit file). In your **WIP file**, go into the **Settings tab** of the **Devkit Overview** and hit **Deactivate Devkit**. Doing so will remove the Devkit overlay tab, leaving you only with the Add-on tabs (Mesh Studio and File Manager).

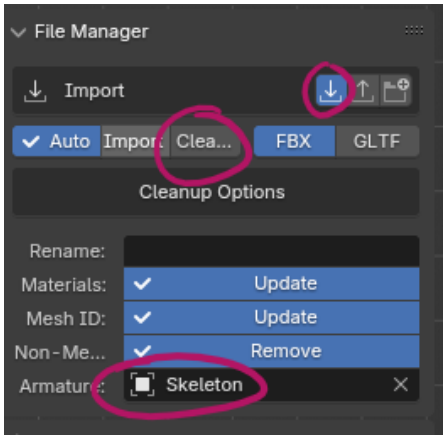
The purpose of this is to remove all the drivers and modifier sources to limit what you're copying over; if you try to do this without deactivating first, Blender will attempt to take the *entire* scene and every collection, since they're all linked - and as of more recent versions, this will make blender crash and immediately close.

Unhide all the parts you want to move to the new file, **select**, and either **Ctrl+C** or **Right Click > Copy Objects**.



Now go to your second instance of Blender with the **newer Devkit** and **Ctrl+V** or **Right Click > Paste Objects**. You'll end up with a few excess pieces from the old devkit which end in .001. **Delete these**.

Now all you have to do is go through each object and set the sources to the new Devkit meshes. It should maintain all your modifier settings from the old file, you'll just need to re-target the Mannequin! A little tedious, but it's the price you pay for keeping up with bug fixes.



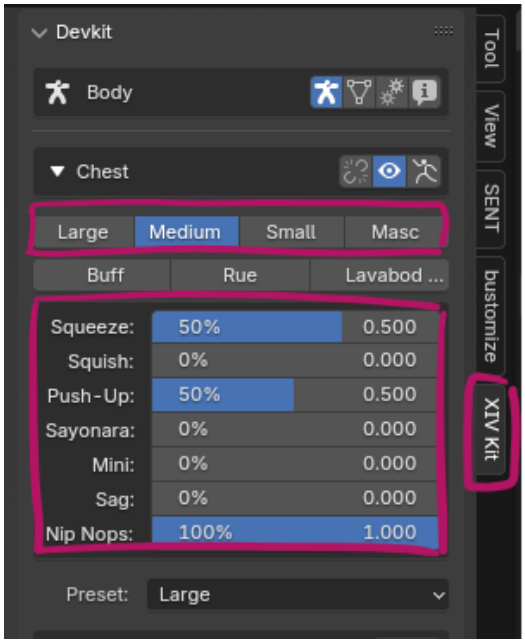
For the Armature modifiers we have a little shortcut - if you go into the **File Manager > Import** tab, you can set the **Skeleton** (of the newer Devkit) as the **Armature** at the bottom, then hit Cleanup and it will target and parent them all for you.

Then, use the **Mesh Studio > Shapes Menu** with Method: Selection to **re-link your Shape Keys**, if you have any.

Upscale Specific Setup

Setting Your Sizes

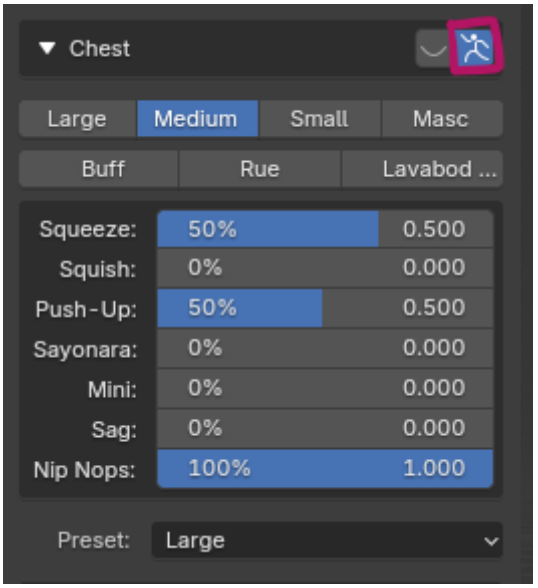
Part of making sure my startup file is ready for me to jump into upscaling with is **setting up the models in the size I prefer to work with** for the initial refit. My personal preference is to refit onto medium, then adjust to adjust to large, and finally small. And sometimes I mix it up and start with a different size! I know a lot of YAB modders like to work with Large first since it's the most demanding on the topology and shrink it twice, and honestly small is pretty close to the vanilla chest, so starting there is a viable option too! Pick whatever order suits you, or simply start with the chest size you use. Don't worry about non-Buff/Buff or Rue for now - just **start with not Buff/not Rue** for now, unless you only intend to make the one size (but making additional sizes, *especially Buff,* is very easy, give it a go!).

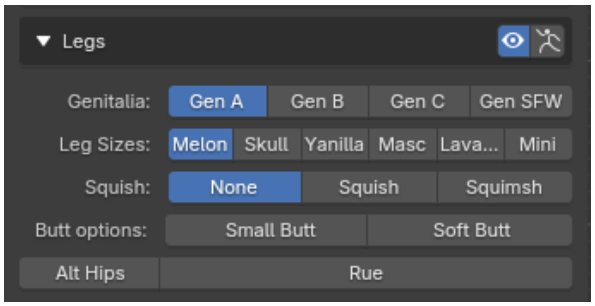


To set this up, **open up your XIV KIt tab** on the side of your 3D Viewport, enter the Body panel, toggle visibility of the Chest with the eye icon if it's hidden and open the rollout. Now you can **select your size and edit the shape**, either manually or using the presets at the bottom. Since most of what I make is clothed, I typically set the shapes to 50% on both Squeeze and Push-Up and 100% on Nip Nops to flatten them down, as this gives a supportive bra look - set them by clicking and dragging, or by clicking once and typing your chosen decimal. You can always tweak these later when you have your gear in the scene if it looks too pushed up (or not enough)!

These shaping keys are unique to each chest size, so toggle through all sizes to set them how you wish. Squeeze can even be set to -50% on Medium and Large!

For the initial refitting process, you're mostly going to use the **Mannequin** as the target for most of your modifiers, so **switch to Mannequin mode** with the little armature icon next to the eye, and **set these up the same way** as the size you're refitting to.





The process is the same for Legs, simply use the Legs rollout instead! Those are all toggles, so it's even easier.

Penumbra Sync

> **Note!**
>
> As of the latest Addon update, you can do this process almost entirely without TexTools and thus this section is mostly redundant, but I still think it can be very useful to know!
>
> Even with these updates, I still use TexTools to extract models, examine existing attributes and shape keys, and to grab XIV model paths so you likely will still want it to follow this guide.
>
> There are still a few circumstances where this Penumbra Sync function comes in handy - specifically installing mods to extract modded models (though you can import directly from a .pmp) or adding Hyur Midlander Female models if one does not already exist, which I'll cover in greater detail here.

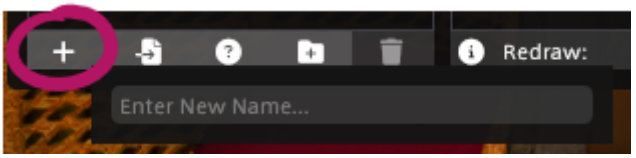
Since TexTools typically functions by directly editing your game files, we can't import anything while the game is open. This *used to* make it a real pain to test things.

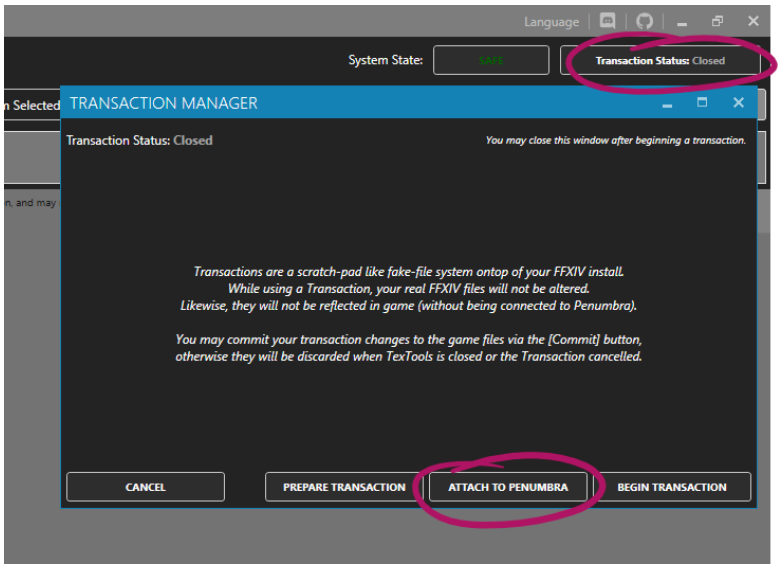
However, as of TexTools version 3.0 and later, we have access to a really neat feature called **Transactions**. This allows TT to look at your game files, but write all of your changes and imports to a modpack in Penumbra like a scratch disc (temporary storage) *while the game is running*, which can either be **committed** (saved) or **cancelled**. It is also able to "see" all of the edits made in this modpack to the game files even if they were made *before* the transaction started, so you can save your progress for later, too.

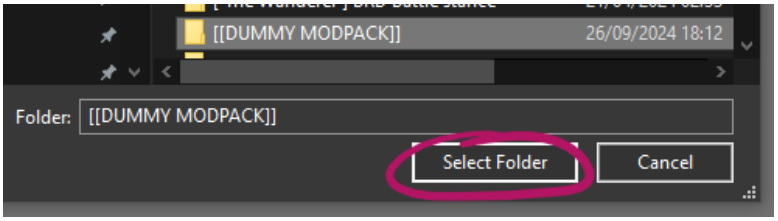
Now the Addon supports MDL export and has a modpacker, I don't end up using TT for much aside from extracting the initial model, but this is still a useful thing to know!

The Penumbra Sync function can only be used on a single option modpack, so my preferred method is to create a 'dummy modpack' for TT to hook into and save files to for testing.

To set this up, have the game running and open your **Penumbra** window. In the **Mods** tab, click the **+** at the bottom of your mod list and name your dummy modpack. Mine is quite literally named [[DUMMY MODPACK]] to make it nice and easy to find in my Penumbra folder, since square brackets are at the top.



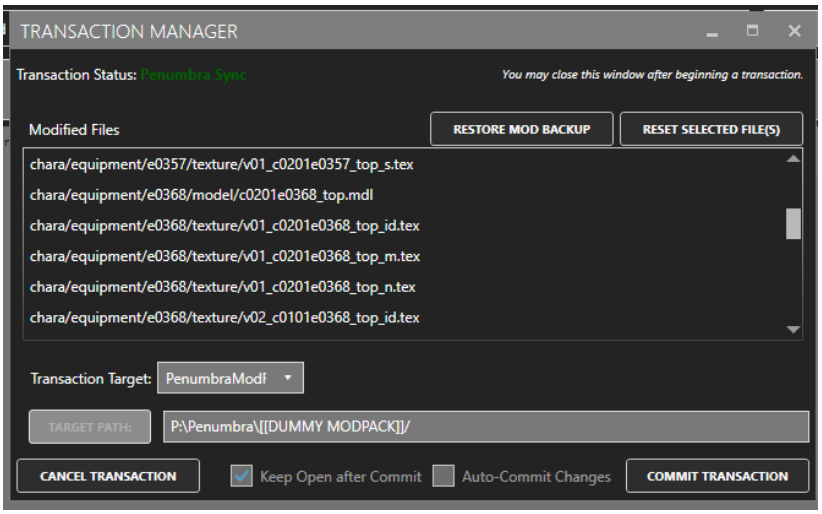




Open TexTools. In the top right, you'll see a button that says "Transaction Status: Closed", click on it to open your **Transaction Manager**. Click on **Attach To Penumbra** at the bottom of this window. This will open a file explorer window where you'll need to navigate to your **Penumbra root directory** (visible in the Settings tab in Penumbra if you don't remember where it is) and **locate the folder** for this newly created modpack - click on it once, then **press Select Folder.**

Let it think for a moment, then the top of your Transaction Manager window and the Transaction Status button at the top right of your TT window should say **Transaction Status: Penumbra Sync** in green and your Transaction Manager will list modified files - you should have none if you just created the modpack, but it will list some if you're reattaching after a previous session. You can close the Transaction Manager window without losing the sync, and you can simply click the button at the top of the TT window to re-open it.

The **Cancel Transaction** button on the bottom left of the Transaction Manager window will remove all of the modifications you have yet to commit/save to penumbra in that session - good to use if something is broken and TT is throwing an error, refusing to let you view the item. **Commit Transaction** on the bottom right will save your current unsaved files to the dummy modpack, allowing you to come back another day, reattach to the same modpack, and continue where you left off.



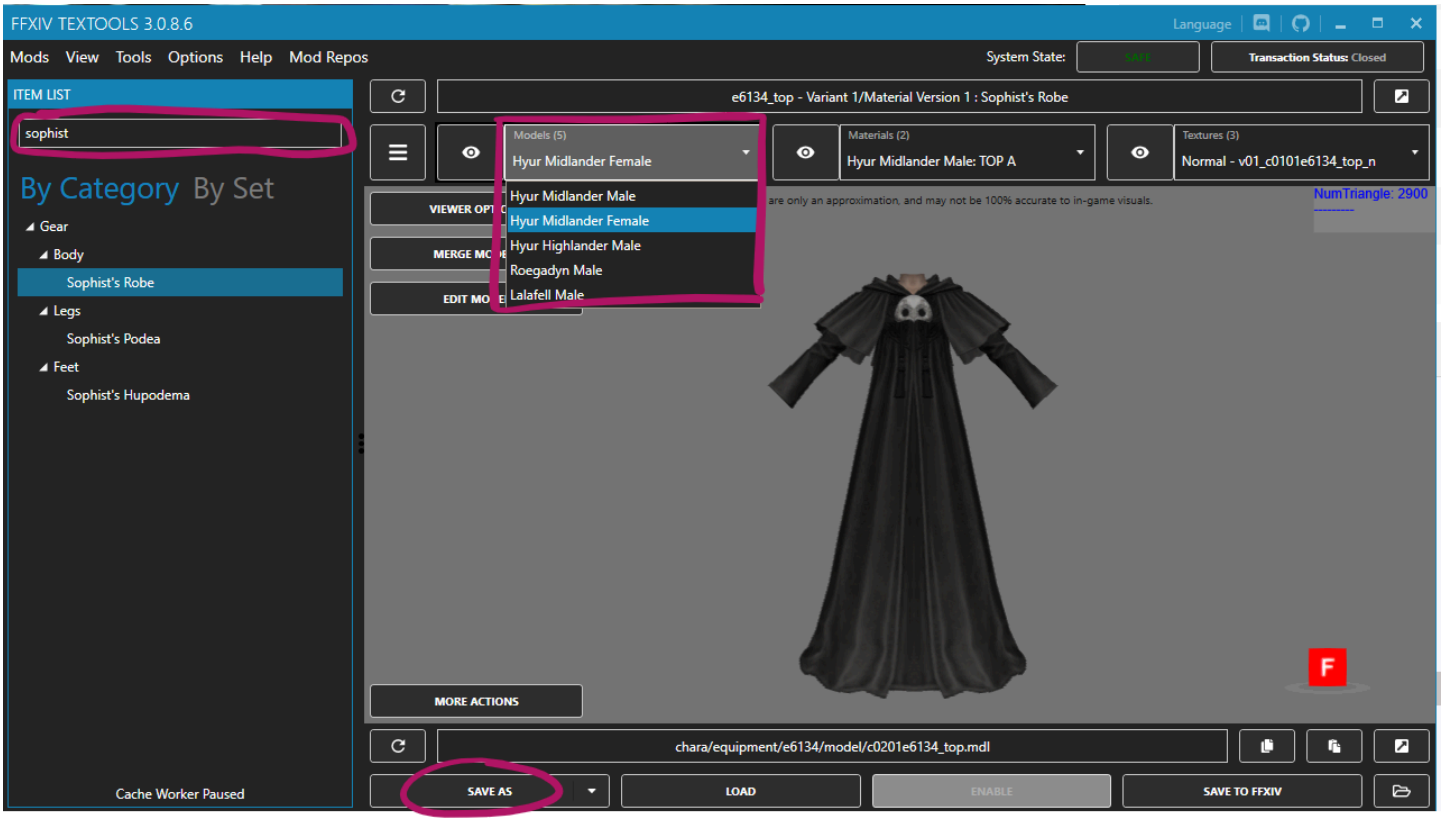
I haven't added anything to [[DUMMY MODPACK]] today - these are all files from previous sessions that are reappearing because I re-attached to the same modpack folder! Handy :)

Extracting the Vanilla (or not!) Model

If you're already looking into upscaling on your own you probably know how this part works, but just in case you don't, here's a quick little rundown.

Open up TexTools from wherever you have it installed (check here https://www.ffxiv-textools.net/ and follow the setup procedure if you don't have it installed yet) and search for the gear piece you'd like to work on from the box at the top of the Item List. Click on the item, then **select Hyur Midlander Female** from the first dropdown at the top, then click **Save As** at the bottom (which will save an .fbx), or click the arrow next to it and select either **FBX** or **MDL**. Choosing FBX will maintain the weights and also pull the textures into Blender, which while not necessary for your export can make it easier to see what you're working with, while MDL import has the added bonus of being pre-set with attributes, so it's entirely up to you which you'd prefer - I'm a fan of FBX, since I almost always re-order the attributes myself! Rename and choose somewhere else to save if you wish.

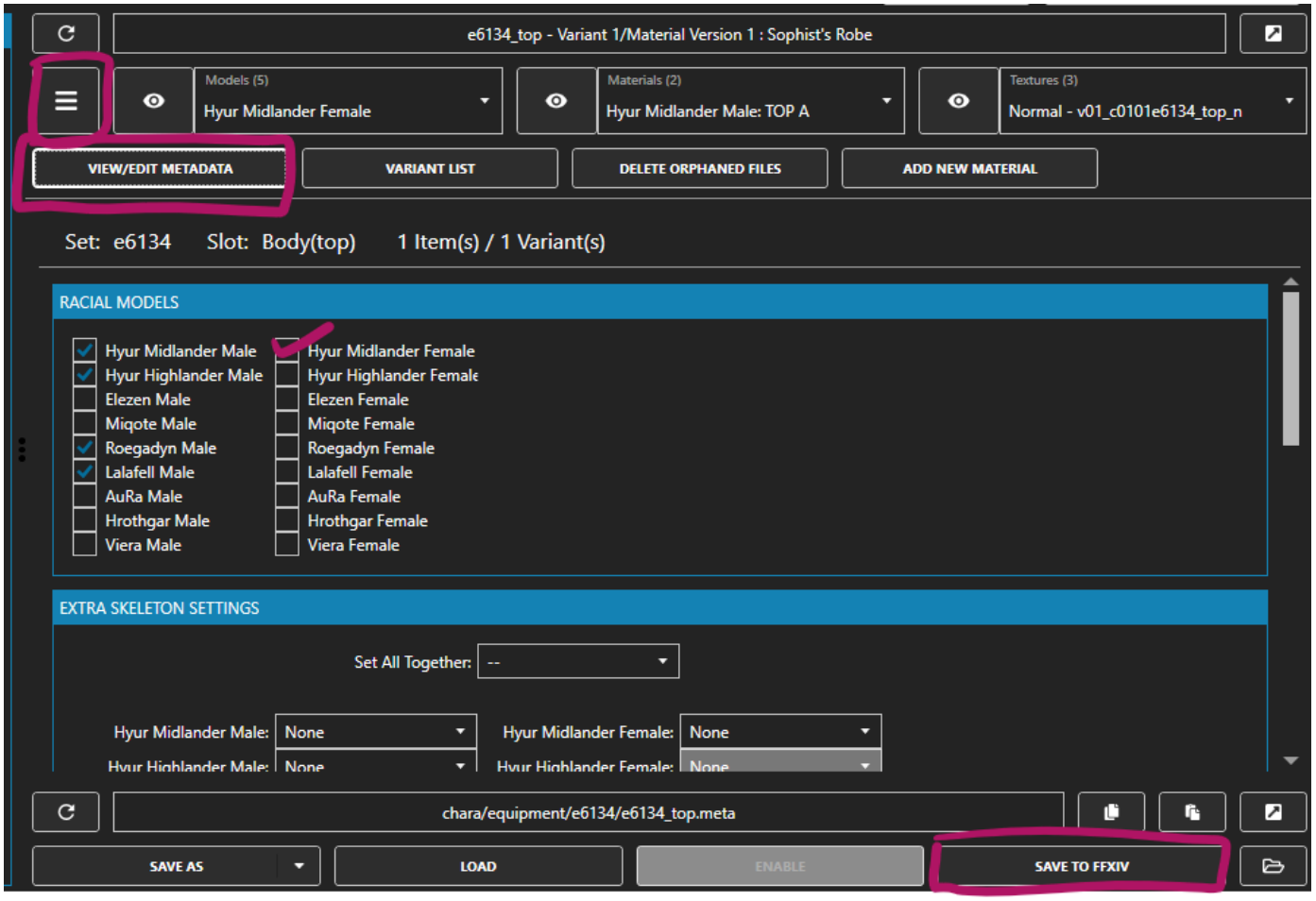
If you're upscaling a modded piece, make sure you **install the mod** into TT (you'll need to use either **Unsafe Mode** in the top right or open a Penumbra Transaction to do so) before trying to grab the model!



What if there isn't a Hyur Midlander Female model?

There's some gear in the game that either cannot be equipped by Midlander Females (racial gear, for example) and some where the Midlander Female inherits from the Midlander Male model (often hand or feet slots). If you were to export this gear, it will be incorrectly scaled compared to the body in the devkit and may be a challenge to align.

The workaround for this is to force a Midlander Female model by opening the **additional options menu** (the three lines) above the top left corner of your TT viewport and clicking **View/Edit Metadata**. **Check Hyur Midlander Female** for the racial models option at the top and **save**. (I recommend using Penumbra Sync for this, so you can save this without directly editing your game files and needing to close your game). You can now return to the models tab and export the Midlander Female model.



If you're working with racial gear, you also want to uncheck the race it belongs to before you save, then repeat the process in reverse by re-enabling the race's model (and uncheck Hyur Midlander Female, if you want to) again *after* you import your upscaled gear over the Midlander model - this will force the racial gear to inherit from the upscaled Hyur Midlander model rather than its own unique (and not upscaled)

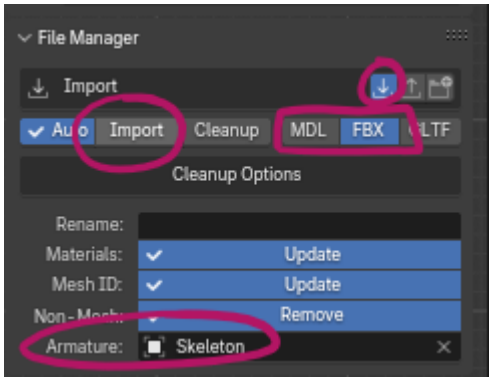
mesh. It's easier to do it this way and rely on TT's automatic racial scaling than to try to match it up yourself.

Preparing the Exported Mesh

The difficulty that comes from upscaling vanilla gear is from this part - preparing the mesh to fit to our modded bodies. Vanilla (and even some modded) meshes that we extract through TexTools wont work for our purposes out of the box for two main reasons, those being that the models are very low-poly (meaning they are blocky and don't have a lot of detail, so they won't conform to curves very well) and are split into many pieces (meaning parts will separate when modifiers are applied to change the shape). While a modded mesh will likely be high poly enough to suit your purposes, if it was pulled out via TT it's still going to be split in some silly places, so you likely may still need to do some prep work.

This step is a little tedious and probably not very fun, but I promise it's important in saving you some headache later on!

Import



Let's start by importing the mesh we want to work on by opening the **Import** tab in the **File Manager** at the bottom of the Add-on's "**XIV Kit" tab** from the right sidebar of the viewport. Be sure to **select** whether you're importing **MDL** or **FBX** - whichever you happened to export earlier.

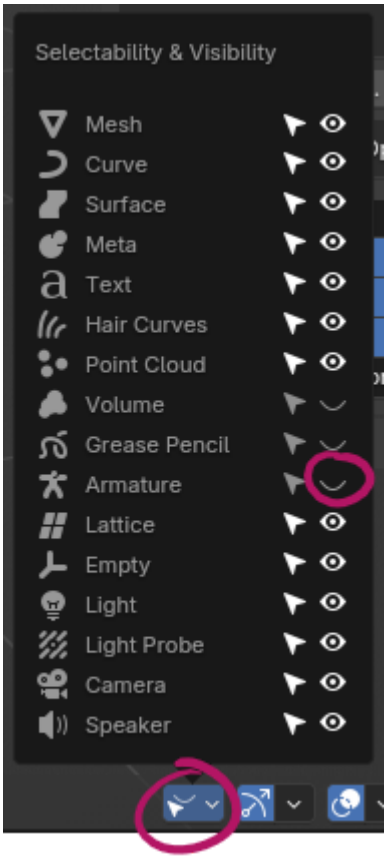
I personally recommend leaving **Auto on** and leaving **all the options checked**; **Materials** will correct the Blender display modes to more closely resemble the game, **Mesh ID** will move the 0.0 number to the beginning of the name to make them appear in numerical order in your scene, and **Non-Mesh** will delete a few extraneous pieces TT will put into the file. All very useful options! You can use the first field to **Rename** all parts you import, which can be handy if you intend to import multiple gear pieces into the same scene but isn't necessary.

Make sure the **Armature** is pointed to the **Skeleton** and you've swapped to FBX import if that's what you exported from TT, then you can hit **Import** and find your exported model!

Your model will typically be saved in **Documents\TexTools\Saved** unless you picked somewhere else. When it comes to exporting later, the Batch Export function works by exporting everything in the Chest/Legs/Hands/Feet collection(s) in various sizes, so we want our mesh to end up in the existing folders, but whether you import it into there or import it outside of the collection and move them as you work is up to you; I'm partial to moving them as I sort.

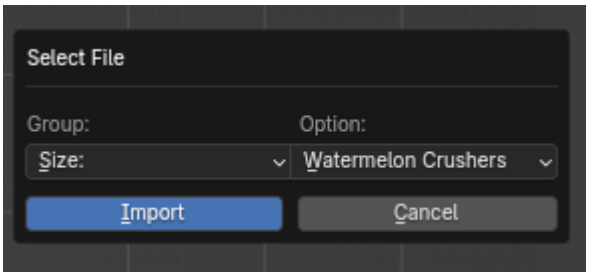
If Auto was turned off, you can also select your imported models and hit the **Cleanup** button next to Import to initiate the process manually.

If you have the **Armature hidden** in the **Selectability and Visibility menu** (meaning it is still in the scene for Blender, but hidden from you to clean up the viewport to work in) on the header/footer bar of your 3D viewport, this process may fail to delete the n_root (the TT exported skeleton) as part of the Non-Mesh removal procedure, but you can now safely delete it yourself.



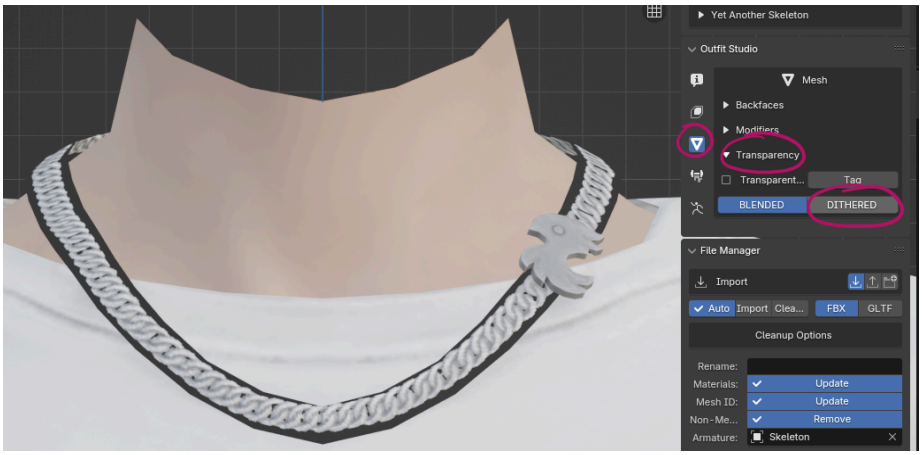
Importing From a PMP

The addon can now import MDLs directly from a .pmp! To do this, select MDL import, then when your window explorer opens to find your model, select your .pmp. This will cause a new window to appear that prompts you to select which option to import.

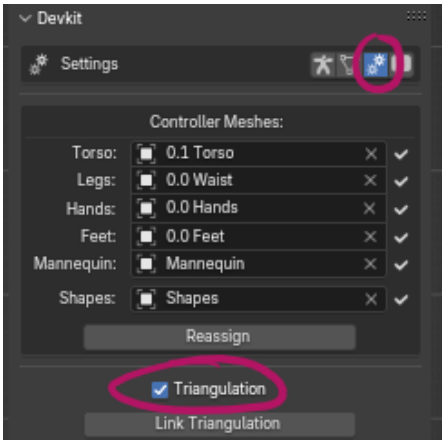


Do note that this has the same pros and cons as direct MDL import; no textures, but pre-set attributes.

If your model has transparent parts, you may notice they display a little strangely; places that should be transparent seem to fill with the background or sometimes the body beneath. This is because the Add-on importer sets the blending mode to Blended for performance. Go into **Mesh Studio > Mesh > Transparency** and swap this to **Dithered** if you want transparency to display correctly, but keep in mind it may be worth swapping back if your computer struggles.



Additionally, if your PC still isn't running too well, enter the **Settings** tab of the **Devkit Overview** and **turn off Triangulation** with this checkbox here. Triangulation alters the geometry of all of the devkit meshes, which is a little resource intensive!



Sorting Parts

What's With the Names?

Your mesh will be in many parts, and if you **didn't** use the **Rename** and **Mesh ID** functions, they will be named things like "*c0201e6134_top Part 1.0*". Let's decipher it really quickly, because I'll refer to parts of it later that may not make much sense otherwise.

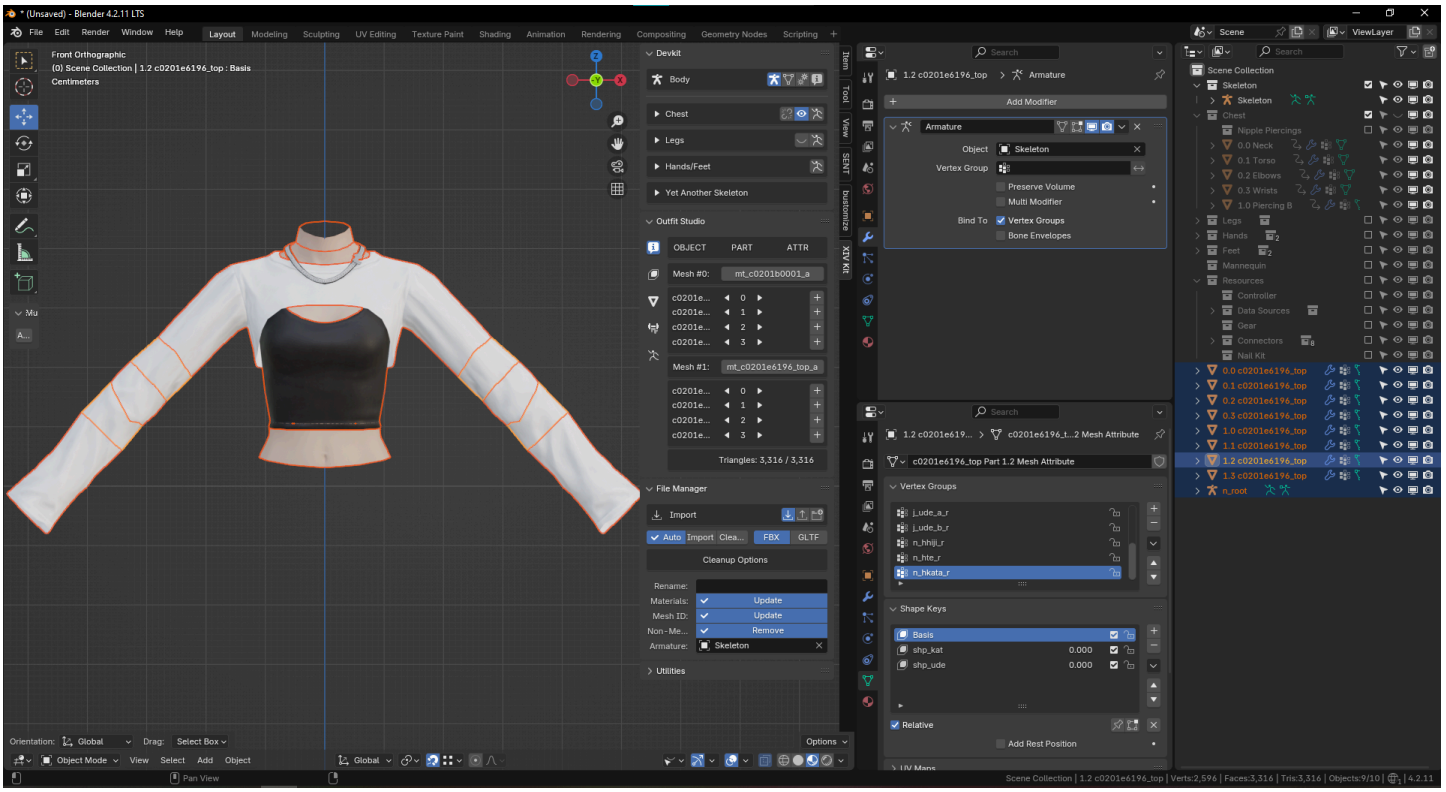
- "*c0201*" refers to your **racial code** - 0201 is Hyur Midlander Female (if yours isn't, you might want to check What if there isn't a Hyur Midlander Female model? - the body in the YAB devkit is Midlander Female, anything else will be scaled incorrectly).
- "*e6134*" is the **equipment** or **set code** - in this example, the Sophist's set.
- "*_top*" refers to the **gear slot** the piece occupies - top is chest, dwn is legs, glv is hands and so on.
- "*Part 1*" is the **mesh number**. All models with the same mesh number will have the **same texture** assigned in game, so you will have one (or more) for your gear meshes, and a separate one for your skin meshes. They can be in any order (the skin can be 0 and gear is 1, or gear is 0 and skin is 1).
- "*.0*" is the **submesh number**, denoting separate models within the same mesh. Separate submeshes can be given **different attributes** such as those that hide the wrist when longer gloves are equipped.

So, consider it to be c***[race]***e***[set]_[slot]*** Part ***[mesh].[submesh]***

The first three parts "*c0201e6134_top*" are simply identifying what the model is for your reference and aren't used by the game engine; these can safely be deleted, allowing you to replace the codes with an easier to understand name if you prefer. This is what the **Rename** function does for you.

However, the "Part *1.0*" *is* used by the engine (specifically the *1.0* bit) - this is how it assigns textures and attributes to different parts of your mesh. Any meshes you export must have a name ending in "x.y", or it will be ignored upon import. You also want to ensure all of the combinations are unique or the duplicate mesh will be ignored, and it's generally good practice to not skip numbers either. So, you can have a 0.0, but you shouldn't have a 0.2 unless you already have a 0.1. All materials and attributes can be reassigned to new parts, so you can reorder them if you want to.

If you **did** use the **Mesh ID** function, the *Mesh.Submesh* numbers will simply be moved to the front of the object name to get them sorted numerically in your scene, but they do still need to follow the same rules concerning no duplicates and separate mesh numbers for separate textures. The Add-on's exporter will handle adding the "x.y" at the end of the name for you when you export, but you still want to keep the initial number to things with matching textures and keep all the number combinations unique.



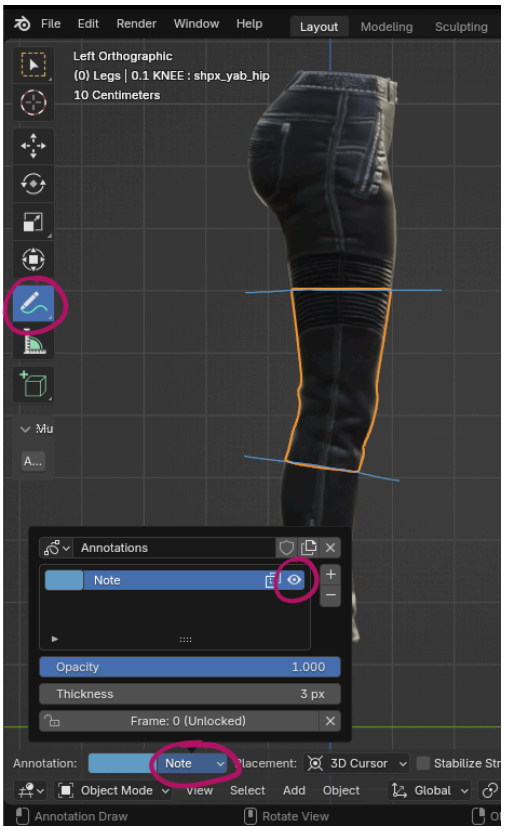
We're going to be replacing the body in the piece entirely, so we can **delete all of the imported skin meshes** sticking out of the vanilla gear, if you have any. (Only the imported body! Don't delete any of the devkit meshes!)

As you can see when I first import my model, the elbow is split into two pieces - you'll find this happens with just about every gear piece you export from TexTools (elbows or skirts in particular). **Click** on one piece (either through the viewport or the outliner) you want to join, **hold Ctrl** and **click** as many others as you wish to merge and press **Ctrl+J** to **Join** (they will join into the last selected mesh in case you want to keep a name). Your cursor *must* be over the viewport to do this - just like the Quick Favourites Menu is contextual, a lot of keyboard shortcuts are too!

Interesting Technical Stuff!

The reason some parts are split into many pieces is actually for LOD (Level Of Distance) Culling. Basically, some of the mesh is hidden if your model is super far away from the camera (be it by zooming out or on another character's model further away) to relieve the rendering load on your computer.

But, honestly, if you're modding your game, you likely don't need to worry about chopping out 20 or so tris... so just join them! :)



As I mentioned earlier, your mesh being in separate pieces can cause it to split when you apply modifiers, so the fewer seams we have, the better. Additionally, the devkit workflow is designed to **merge wrist, elbow and torso** pieces to split later, as well as **shins, knees and hips for the legs**.

Before you merge all the parts, I recommend using the **Annotate** tool from the left sidebar of the viewport to mark where the seams are between the separate parts to help when it comes to re-splitting them at the end of the upscaling process - saves a few minutes picking through the edges to find where the clean line is afterwards!

If the lines get in the way, you can open the note menu on the header/footer with the Annotate tool active and toggle the little eye icon to hide them until you need them again.

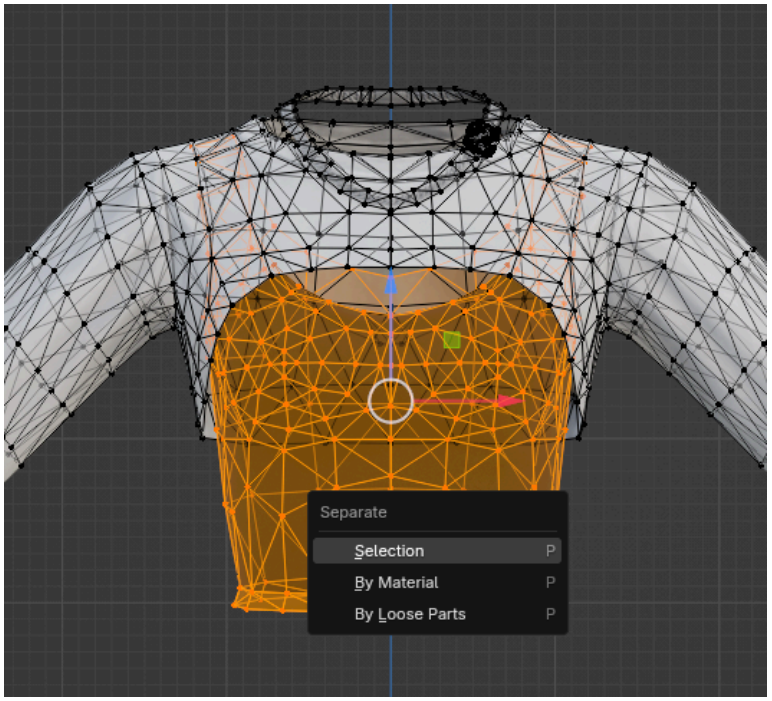
Once you've made the notes you want to, merge the full sleeve/leg.

Any pieces you intend to toggle as variant parts such as pauldrons or belts will need to be separate meshes at the end so you *can* leave them separate from the get-go, but do consider that each unique mesh in your scene will need to have modifiers and shape keys applied individually; if your part is in a place that will be affected by a lot of shape changes between sizes (a badge over the breast for example), multiple modifiers or several shpx_ keys (custom Penumbra shape keys), consider leaving it connected and detach later.

Separate belt and bags!

For example, on the Fieldfiend's Shirt (pictured), I had the belts and three bags attached to them toggle separately, but they are all affected by the booty clearance shpx_ keys I made - it's easier to leave them as one object, make the shape keys, then separate for toggles at the end.

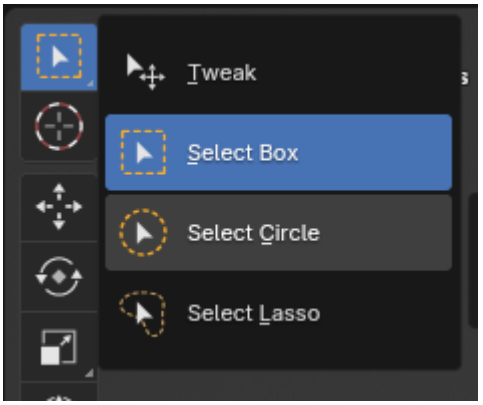
Also worth noting, modifiers like Shrinkwrap *cannot* reference part of the same mesh, only a unique object. I use this for buttons or other shirt details a lot, separate them to Shrinkwrap to the shirt with a small offset to keep them on top.



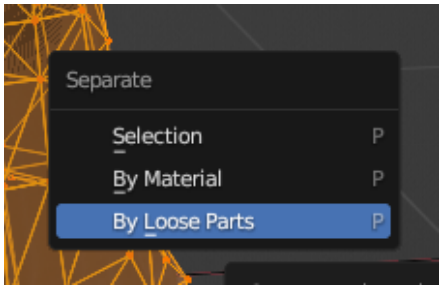
So, going back to my Leisurewear High-Cut Knit Top, I've deleted the vanilla skin, combined the sleeves into one piece, making it one single mesh. Now I want to separate the tank top from the High-Cut Shirt part, as well as detaching the necklace and wristbands to allow the shirt, necklace and wristbands to be togglable as variant parts

To do this, go into **Edit Mode** (via the **Tab+drag** or **Ctrl+Tab** menu) and select your preferred selection method by pressing **1** (vertex), **2** (edge) or **3** (face) with your cursor over the viewport or these buttons in the viewport header/footer. I personally mainly use vertex selection, but they all work!

Ensure **X-Ray Mode** is on with **Alt+Z** or the toggle button on the header/footer (you'll be able to see greyed-out topology through the mesh), **Select** part of the mesh you want to separate and use **Ctrl+L** to **Select Linked** until you have all of it. Next, press **P > Selection**, to separate what you've just selected.

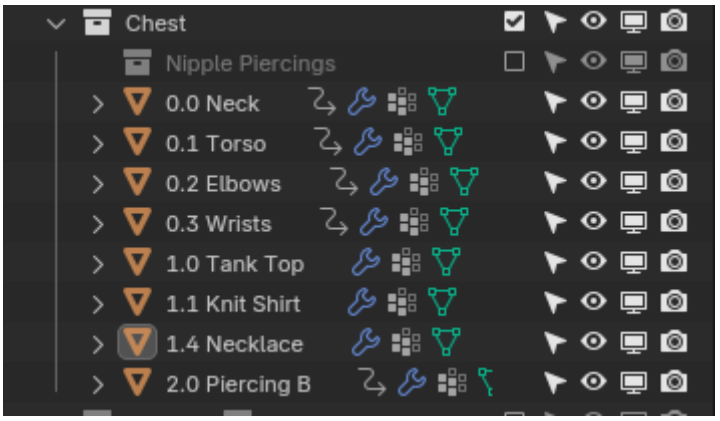


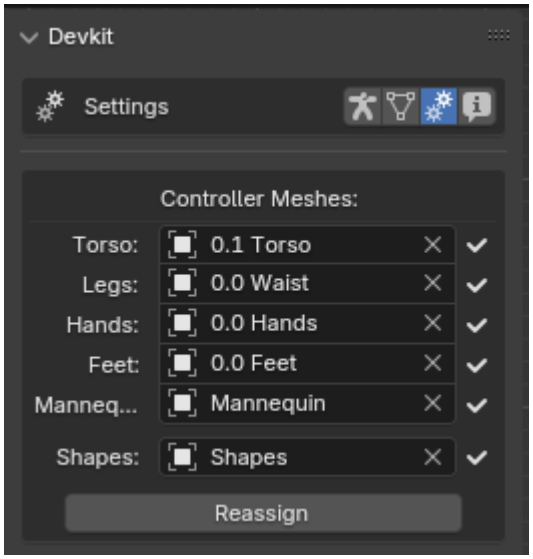
You can use any selection method you like from this menu - I'm partial to select box and circle (which can also be toggled with **C**)!



If a piece is in many parts and is particularly difficult to navigate around, I will sometimes **P > By Loose Parts** to split every separate piece into a unique mesh and piece them together with **Ctrl+J** to **Join** piece by piece. It's a little messy, but sometimes it's easier to hide all the small parts to find the piece you actually want to join. Find a way that works for you!

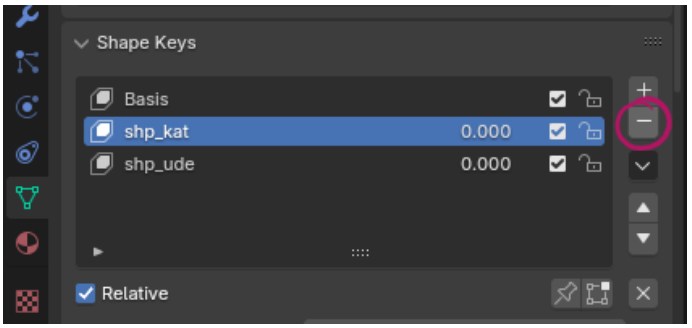
Once you have them split and joined appropriately, it's time for a little housekeeping. **Rename** your parts with a little description (to make it easier to identify) and give them unique part numbers, starting at 0. I've left 1.2 and 1.3 for the Elbow and Wrist parts of the sleeve when I detach those later.

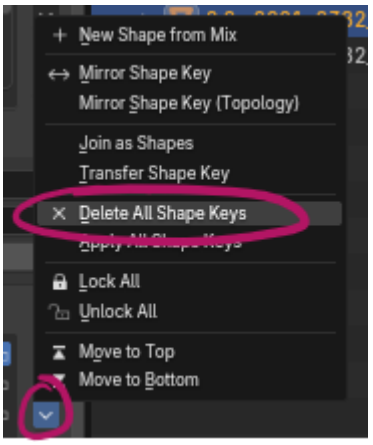




Note how I've also renamed the Piercing from 1.0 to 2.0 - remember these number combinations must be unique and everything with the same *first* number will have the same texture applied to it. You can rename the devkit meshes to your heart's content, but be sure you **never delete the controller meshes** which you can find in the Settings panel of the devkit overview. Ideally, you can make liberal use of the hide buttons and not delete a thing!

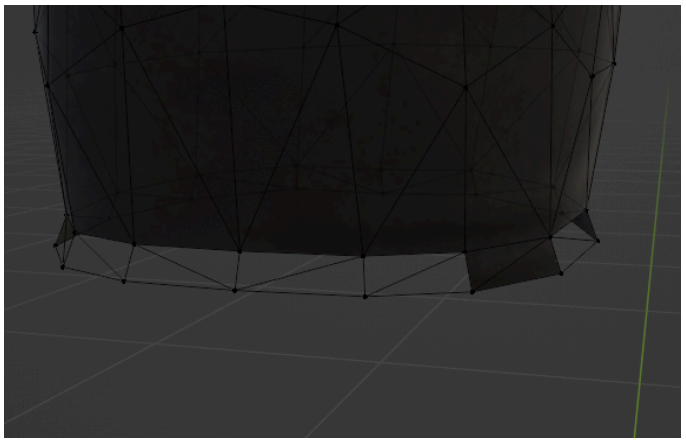
You can either move them into the Chest collection now, or wait until you're actively refitting so you can keep toggling the Chest meshes separately from your gear meshes. This is also a good time to **remove any shape keys** left on the gear in the **Object Data Properties tab** with either the **-** button or the **Delete All Shape Keys** option in the little arrow menu - a lot of modifiers won't apply with them on, and we'll be recreating them later anyway.





Merging

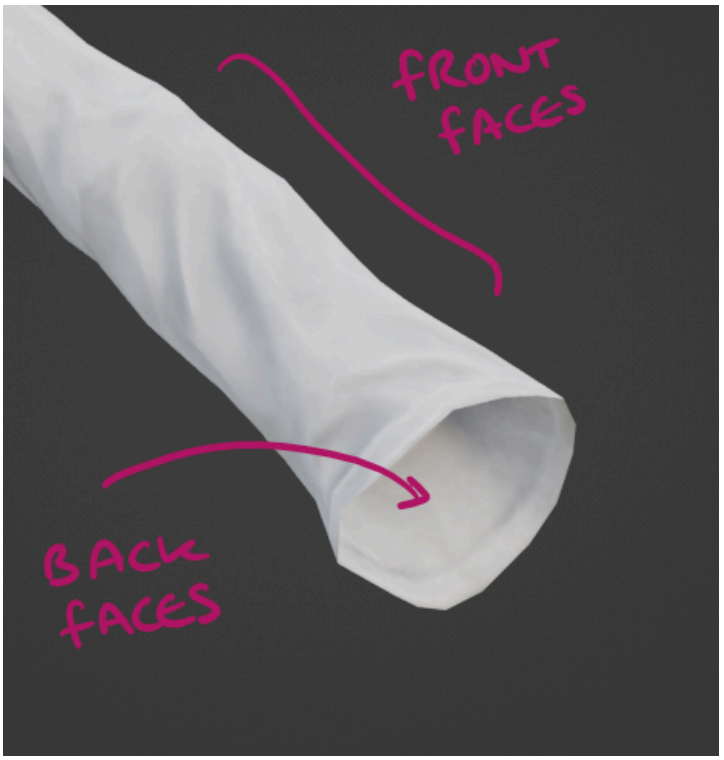
Notice how even these individual objects are split into several parts, making it a little tricky to select it all in one go? We don't want that - these sections are liable to separate when we apply modifiers, or sometimes the verts in the same spot don't get the same weights and they'll split in motion, too. So once we have our mesh in the parts we want, we want to join all the verts in the same spots together. But! There's a catch; we (probably) have backfaces. If I merge by distance as it is, Blender will smoosh all the faces together and not know which way they should point, leaving our model patchy, as shown here.



Interesting Technical Stuff!

The reason the mesh is split into so many pieces is because the game engine can't handle a single vert having multiple UV (texture) co-ordinates, so instead it splits any vert on a texture seam into two (or more!) verts with all the other data (like weights and normals) completely in tact. Neat!

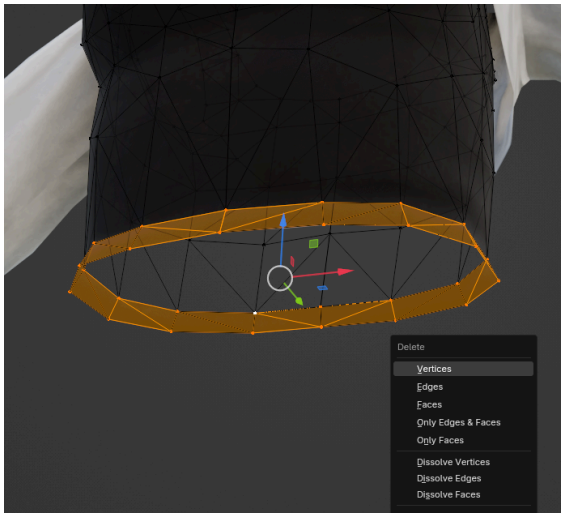
What's a Backface?



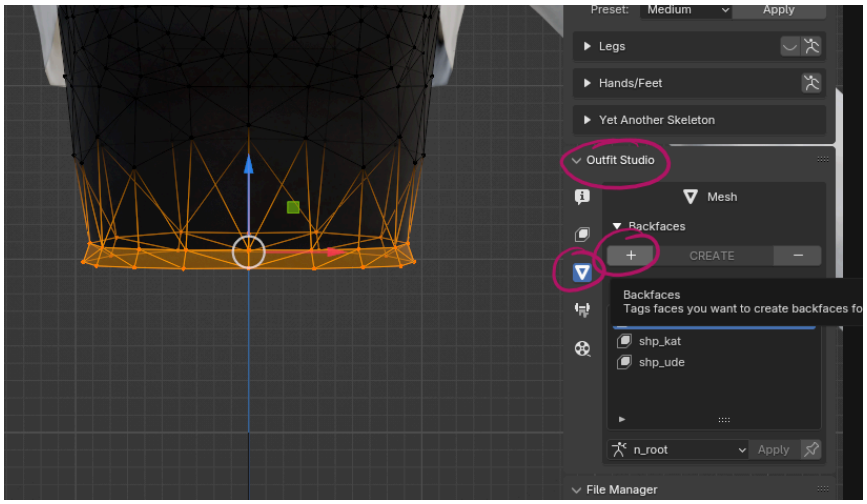
Backfaces are, simply put, the *backs* of *faces*. A face is a single poly or plane on a model, be it a tri (3 edges), quad (4 edges) or n-gon (more than 4 edges).

XIV's engine doesn't typically render the back of faces, making them only visible from the one side. Have you ever moved your camera in such a way it clipped inside your character's head, and their face turned invisible but (parts of) the eyes and teeth were still visible? That's because the face part doesn't have backfaces, since you aren't supposed to see the inside, while the mouth and eyes do. What XIV does instead for parts that *can* be seen from the backside is - it adds another face in exactly the same spot but facing the other direction.

Simply think of front faces as the faces pointed outwards, and backfaces as those pointing inwards, like the inside of sleeves or collars.



The Addon gives us another fantastic tool for this part of the process, and that's the ability to automatically recreate backfaces upon export by tagging parts with a special vertex group to duplicate and flip later. Check that your front and back faces share textures (95% of the time they do, but some things like capes may not) and if they do, **select** the pieces facing inwards (with **Ctrl+L** to ensure you get the whole piece) and delete with either **X** or simply the **delete** key - any of the first three options should do the trick, Vertices, Edges or Faces.



Next, **select the front faces** from the same location as the faces you just deleted, open the **Mesh Studio** rollout in the Add-On Overview, open the **Mesh menu** and click the **+** under **Backfaces**.

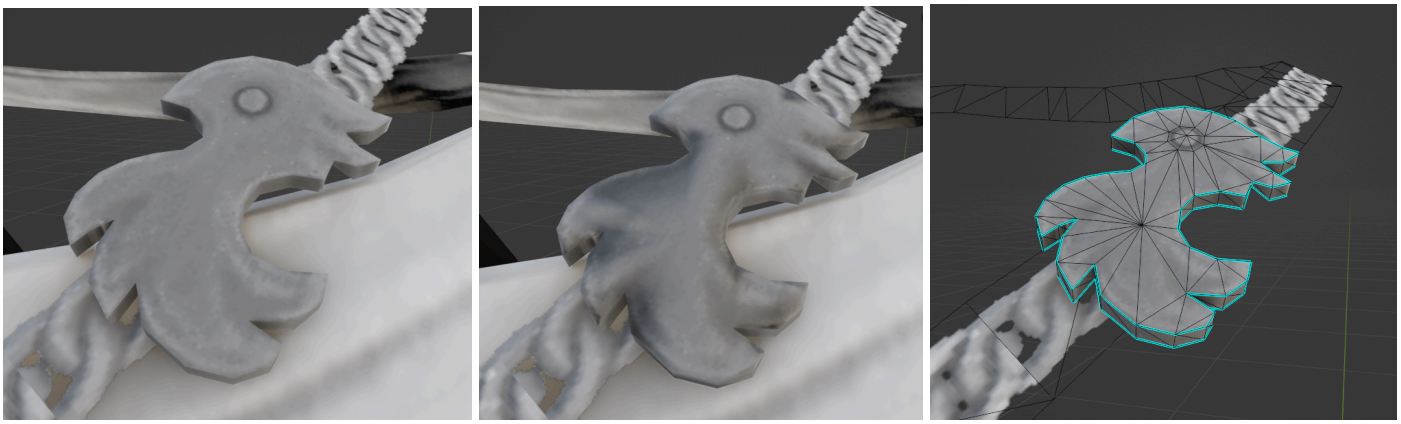
Once you've sorted your backfaces, it's safe to merge the remainder of the model. Hit **A** in **Edit mode** to **Select All**, then hit **M > Merge by Distance**, which will merge all vertices occupying the same space and making it one single easy-to-work-with mesh.

What if your mesh has backfaces with unique textures? Use **Ctrl+L** to **Select Linked** as before to select all front faces first, **M > Merge by Distance**, then repeat by selecting just the backfaces with **Ctrl+I** to **Invert Selection**. Unfortunately you'll just have to work around the separate pieces! It is an option to separate the backfaces into a separate object with **P > Separate By Selection**, then apply a Data Transfer modifier to the backfaces mesh with the **Vertex Data > Vertex Groups** option selected and mapping set to **Nearest Vertex** to keep the weights the same, but you will need to be extra careful that you select both meshes before going into Edit Mode to alter the shape or fit at all, and I'd recommend leaving them connected when it comes to Size Conversions so the backfaces don't separate from the frontfaces.

Repeat this process for all meshes as required.

Splitting Normals

You may find that now on the merged seams the lighting looks weird - Blender is trying to smooth the *vertex normals* across both faces that share an edge, despite them facing in very different directions. To fix this, we split the normals (data telling the engine which direction a face/vert is pointing) with **Alt+N** (Normals Menu)(or **Q** for Quick Favourites if you added it) **> Split** , creating a hard normals seam but maintaining a single vertex. This is very useful for any sharp edges and corners such as this Chocobo pendant. I will add seams like this to any part of the mesh with over a 90° angle (and sometimes less - again, use your best judgement!). Going into **Edge Selection** (**3**), holding **Ctrl** and **clicking on an edge** further along the seam (**Pick Shortest Path**) is very helpful here. **Shift+click** can add a single edge and also resets where it's trying to path from, and Shift+clicking on a selected edge will unselect, too.



The pendant unmerged > Merged > The normals seams added

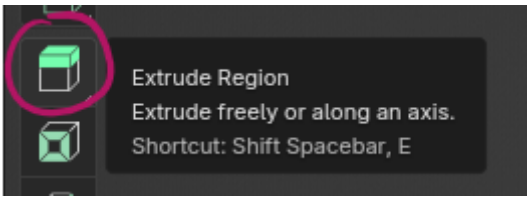
Repeat this step for every edge and corner with strange lighting. I will also add a split like this across much smaller seams, like the seam between shirt and sleeve if the textures align with mesh edges - just to enhance the visual effect of the normal texture. Totally optional! I just like to do it. :)

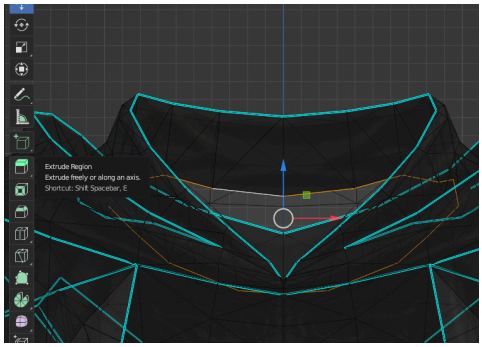
These seams are also very useful when it comes to Retopologising, as it can stop Blender trying to *quadrify* across spaces with different textures.

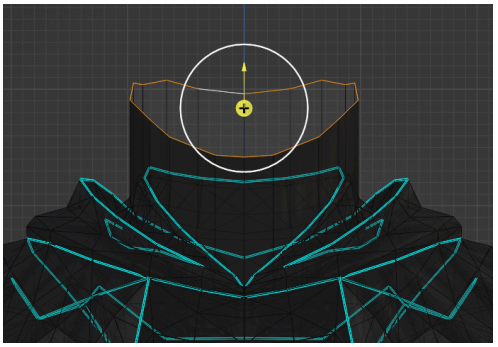
Once you have all of your seams in place, it can be a good habit to get into to press **A** (**Select All**) **> Alt+N > Reset Vectors** - this will get the normals to re-calculate themselves with these seams in mind, which will likely make them look a lot neater.

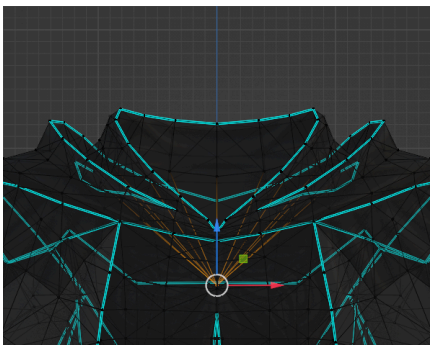
Capping Holes

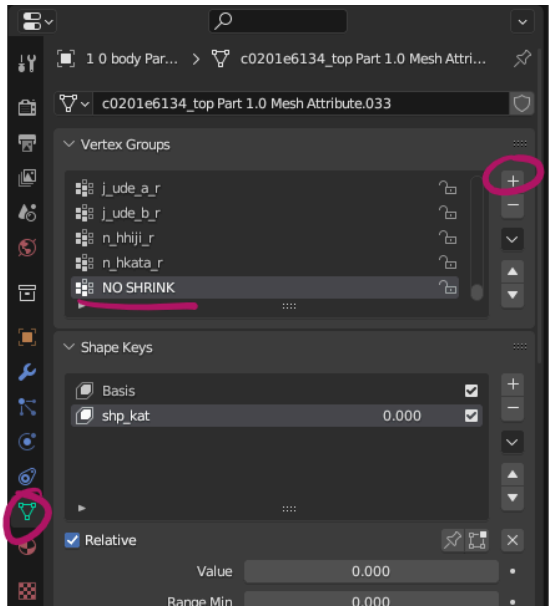
This is also where I would cap off things like sleeve or neck holes if a part doesn't use the Mesh Studio's Backfaces options - while the vanilla meshes tend to perfectly align with the polys of the body so anything on the inside can be deleted, making a vanilla mesh as high poly as the modded body would make it a nightmare to load and work with, so instead I extrude the inside edges and merge into a single vert and tuck that inside the body so you don't end up with little holes if you look at a certain angle. Either use **Ctrl+click** to **Pick Shortest Path** around the edge, or **double click** an edge to select the whole loop (unfortunately doesn't always work with triangulated meshes), then click on the **Extrude Region** button on the left sidebar. Drag out, **M > Merge At Center** to make it a single vert and tuck it away.



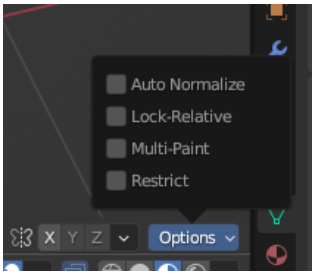








This is often where I would **create a new vertex group** (**Object Data Properties tab**, the only green icon, first dropdown is **Vertex Groups**, click the **+** next to the list), call it "NO SHRINK" - I put it in caps so it's easy to tell apart - enter **Weight Paint Mode**, press **2** with your cursor over the 3D viewport to go into vertex selection mode (or by using the toggle in the header/footer). Set your Weight on the slider in the header/footer or the right click menu to 1.0 and turn off Auto Normalise if you haven't already from the setup section, then press **Ctrl+X** to set the weight of your selected vertices to 1.0. I will later exclude this vertex group from the shrinkwrap modifier, so if there's anything else you know you want to exclude, you can add it here, but you can add things later too.



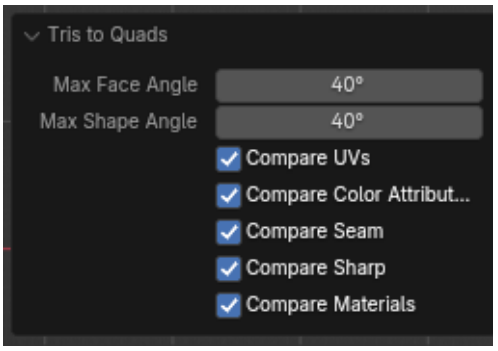
As before, repeat with all parts that need it.

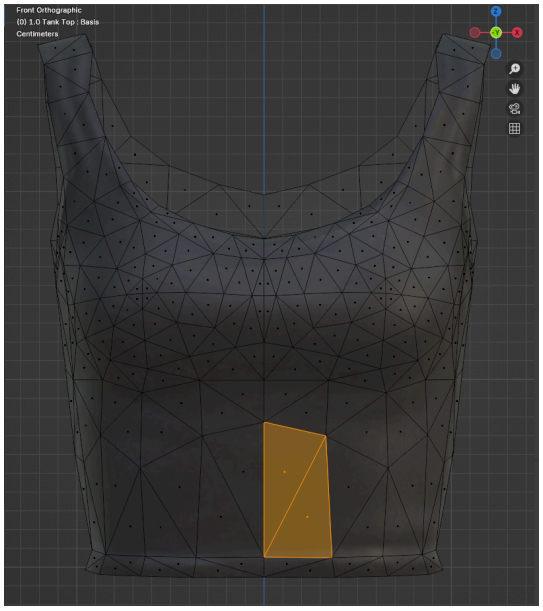
Retopology

Vanilla models are so low poly that if you try to upscale them to the shape of a modded body they will look blocky and jagged and not at all shapely, if not having chunks of the body clip through. *Retopology* is the process of changing the geometry of a mesh to better suit your purposes ("topology" being the term for the structure of the geometry making up the surfaces).

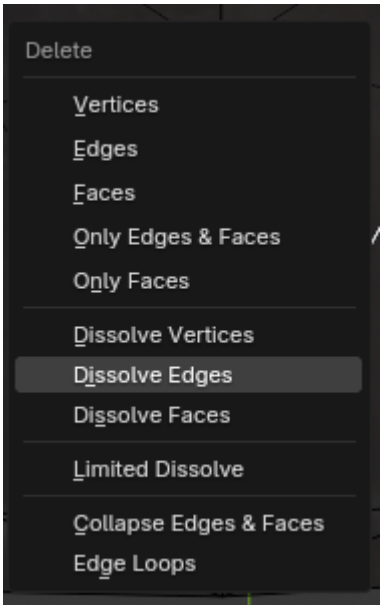
First of all, we want to be rid of those pesky tris; models are much easier to work with in quads. Game engines can only handle tris so every quad gets sliced in half when it's imported, but we need to undo that. Reverting it is harder than the initial triangulating, but these days Blender can do *most* of the work for you.

Select the whole mesh in **Edit mode** with **A**, then press **Alt+J** to convert **Tris to Quads.** I would recommend checking all of these options, but most importantly **Compare UVs** to prevent the textures stretching strangely where the quads don't align with seams and **Compare Sharp** to respect the seams you marked in [Splitting Normals].





Sometimes Blender doesn't quite get it right because of how highly optimised XIV meshes are, so you can undo and fix some quads manually by selecting just the two tris and using **Alt+J** (giving it no other choice) or selecting the diagonal edge in the middle (remember, press **2** over the viewport to enter edge selection mode) and removing it with **X** or the **delete** button and selecting **Dissolve Edges**. If you pick out all of the tough parts (or sometimes just give it one to use as a jumping off point), you should be able to give **Select All** (**A**) and **Tris to Quads** (**Alt+J**) another try with a better result.



Recreating Loops

Warning!

This section contains pretty advanced modelling stuff. I don't go particularly in depth so feel free to give it a go, but if you're having trouble you can skip this part and go straight to Subdivision. Subdivision can do all the things this section aims to do, just in a slightly less ideal way - but a *far* easier way too!

I never did any of this on my first upscales, and even now there are some pieces where I *still* don't do any of this. It's absolutely not required, and you can make upscales just fine without, it's simply tidier!

Don't forget to save, this part can very easily go wrong!

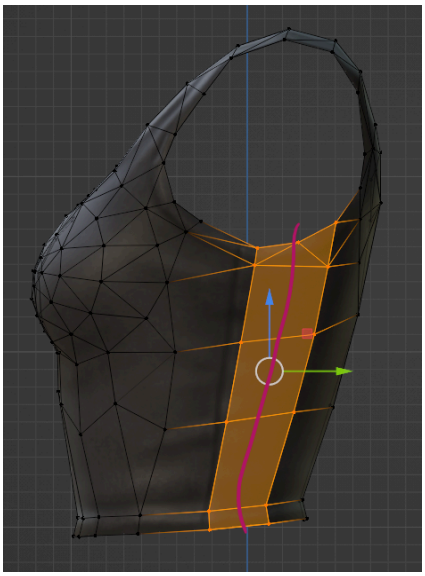
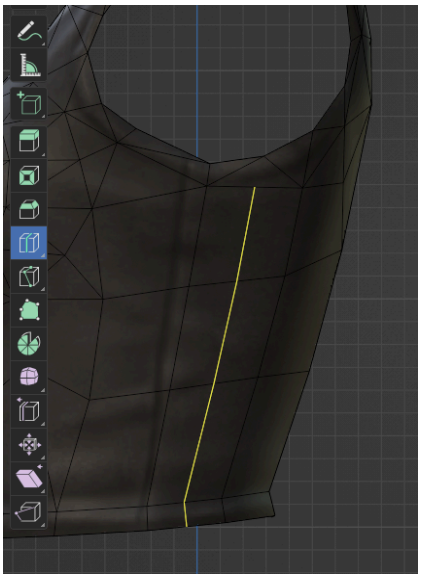
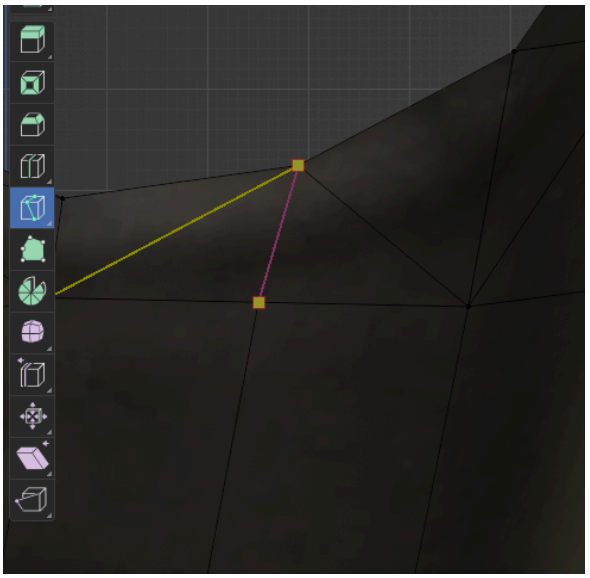
Retopo is very much unique to each mesh, so instructing you on how to do it for yours will be a little difficult. Additionally, the most common kind of retopo is creating a *lower* poly mesh after a sculpt, so a lot of guides and tools don't quite show what we're looking for.

I'm only going to cover really simple tweaking of the topology here, because this is a *little* too complex for a guide aimed at beginners. I just want to add some tools that are useful for fixing little wonky bits that aren't quite acting how you want them to, not completely retopologising the mesh - though if you're an overachiever, you may find value in learning how to do that properly! You can read more about the theory of the subject here, if you want to.

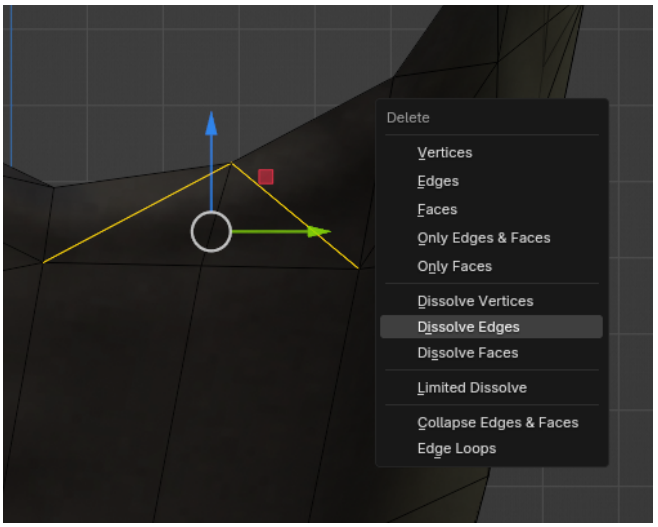
The main things you want to use here are **Loop Cut** and **Knife** from the left sidebar, as well as some more **X/Delete > Dissolve Edges** action. **Loop Cut** will slice Quads in half along a loop, good for adding a bit of extra smoothness once you have loops established. **Knife** will let you cut faces freeform; it can snap to verts or the middle point of edges by holding **Shift**, so it's very good for recreating loops where they're broken by Tris.

Manually retopologising has the primary benefit of allowing you to use **Loop Cut** to add topology only where you need it, rather than having to subdivide the whole mesh (and making it more taxing on your computer to render in game) or alter the fit.

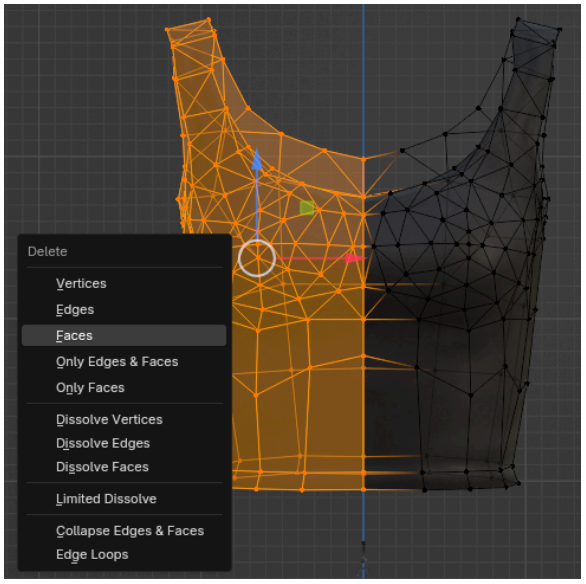
For what it's worth, the main areas you'd want to add extra loops to is this line down the back and behind the shoulder (for the Buff shape), around the chest (for the Large shape and especially Uranus), the butt, and perhaps around sleeves, legs or skirts to achieve a smoother and more rounded shape.

Where I want to add a loop > **Loop Cut** on the quads > **Knife tool** on the tris at the top



As an example of these tools in use, there's a strip on the sides of my Tank Top here where it's almost all nicely quaded, but the top is three tris instead of two. Since I want to add more topology for a smoother curve anyway, what I can do is cut all these quads in half using **Loop Cut** and make that edge loop meet the vert at the top with either the **Knife tool** snapping it to the vertices by holding **Shift** and then pressing enter to apply the cuts (press Esc to exit cutting mode without applying if you need to), or by selecting the **Vertices** in **Vertex Selection mode** (**1**) and pressing **J** to **Join** them. I can then either select the faces and **Alt+J** to use **Tris to Quads**, or I can select these middle edges and press **X** or **delete** and select **Dissolve Edges**.

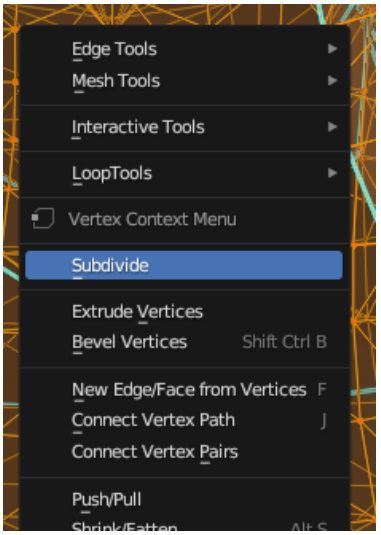


But wait, this is only affecting one side! Easy solution if your piece is symmetrical like mine - cut it in half and apply a **Mirror modifier**. By default it will share the same UVs (textures) and *will* flip the weights (which is very handy! though we'll be adding weights later, so we needn't stress over that). Be sure to apply the Mirror once you're finished, though, as it doesn't always play nice with the other modifiers we use for refitting, nor does it like being applied once you make shape keys.

If your piece isn't symmetrical, you would have to repeat these cuts on the other side where needed.

Subdivision

If you did a lot of work in the previous section (Recreating Loops) you may be able to skip this with liberal use of Loop Cut, but you may still want to use this tool in some places so I would recommend reading through.



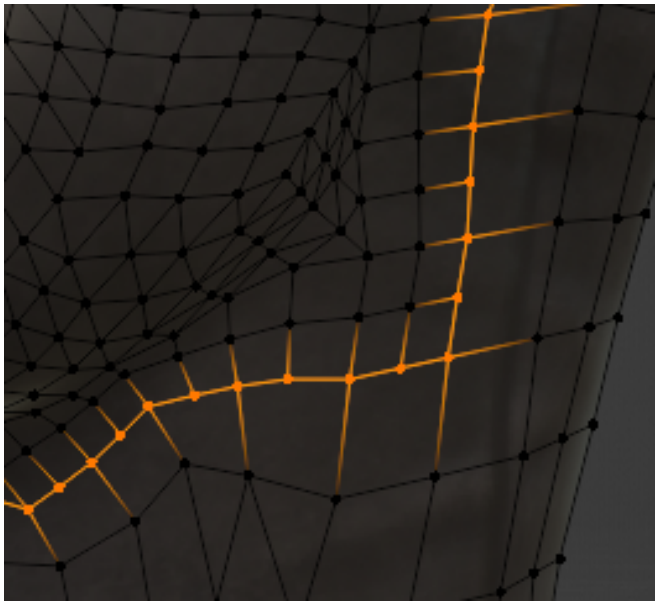
Subdivision is simply splitting each face into 4 smaller faces to make the mesh smoother. Given that this ups your tri-count every time you use it, you want to use it sparingly. Vanilla meshes are so low poly that you could easily subdivide your mesh twice with no noticeable effect on how your game runs, but it's still worth being cautious not to overdo it where it isn't needed. Keep in mind, the more subdivided your mesh is, the harder it is to smooth areas out that aren't in direct contact with the skin, particularly between the breasts!

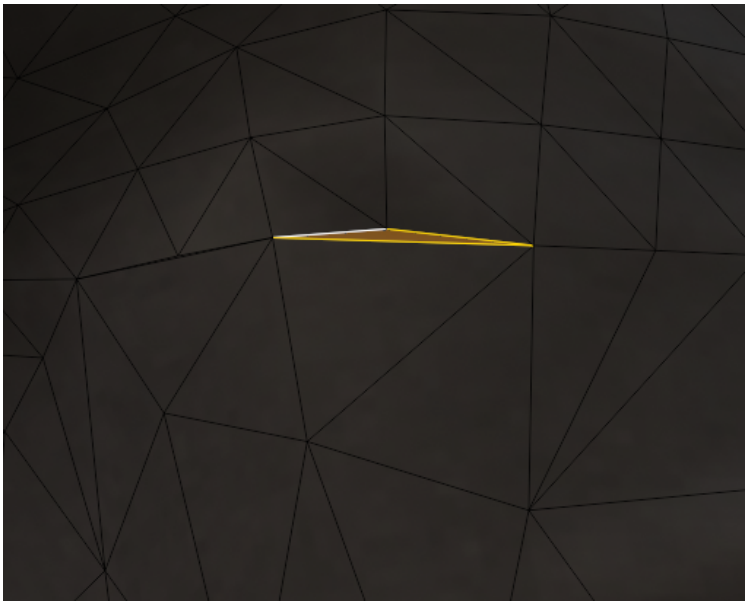
I would initially add one level of subdivision to all of the mesh that conforms to the body or should be smooth, essentially only excluding pauldrons, buttons, bags or other accessories unless it looks like they need it to deform properly later on. Blender will handily average weights and normals on these new verts, so it can be used on skirts if you want them a little rounder. I can comfortably select all of my tank top mesh (**A** in **Edit mode**) and **right click > Subdivide**.

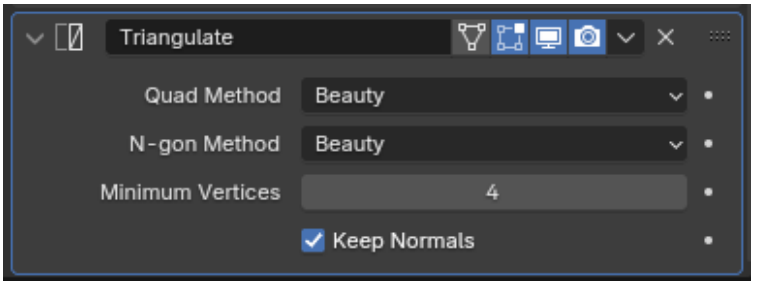
Shrinkwrap applied with no edits to demonstrate >

One level of subdivision added

Some areas will need some extra subdivision to be smooth, namely the chest (especially if you're including Uranus sizes) and the butt, since they're pretty round. Considering we don't want to subdivide too many times where it isn't necessary, if you were to select only these areas and subdivide you can end up with a seam like this between different levels of subdivision. This isn't good for several reasons, but without delving too far into dry theory lessons, the main reason we don't like this is when the mesh is triangulated for the game engine, it can cause puckering where it creates incredibly squished triangles.







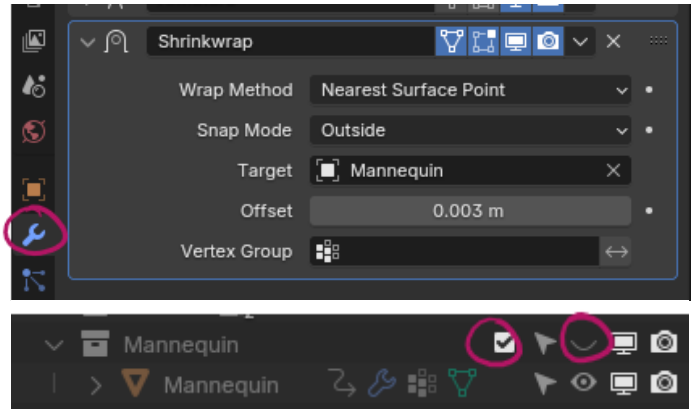
In an ideal situation (and if you did look further into retopology), you would use edge loop reduction flows (some examples here), but if you don't want to spend hours playing connect the dots, we can alleviate this issue with one little "good enough" fix; a **Triangulate modifier**. You may need to fiddle with the settings to see what works best for your mesh, but **Beauty** is usually a safe bet. Check **Keep Normals**. So long as these seams aren't at areas with a lot of deforming (joints like the shoulder, elbow, groin, etc), you're unlikely to be able to tell it was done with a quick triangulate modifier.

> Tip!
>
> Don't be afraid to start over! Once you're elbow deep into the refitting part, if you find a piece you didn't correctly prep in the merging stage, it can be quicker to start from the vanilla mesh again than to try to play damage control and fix how the issue has compounded later on down the line. It's a lot less stressful, too! I've lost count of how many upscales I've started over (sometimes the whole thing, sometimes just a single part) that ended up looking a lot better for the really specific experience of where things go wrong. Don't consider it wasted time, it's lessons learned!
>
> Oh, and **don't forget to save again**! :)

Refitting

Shrinkwraps

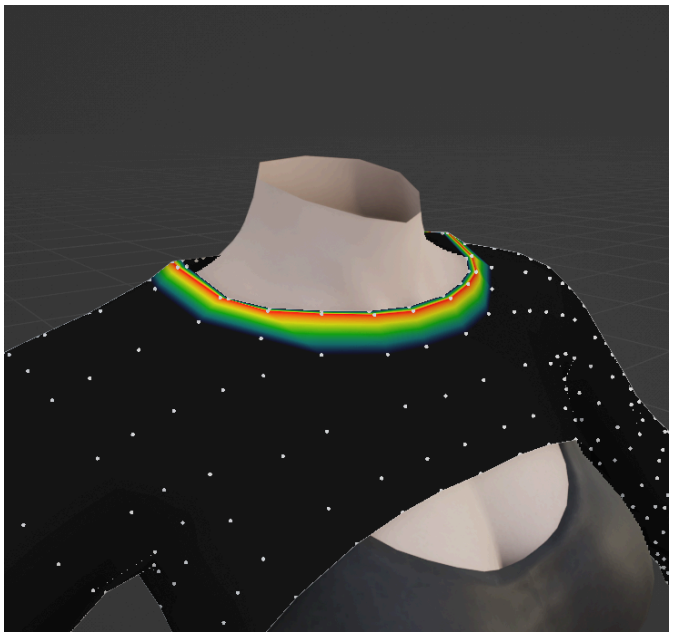
Finally, we can move onto the actual upscaling part!

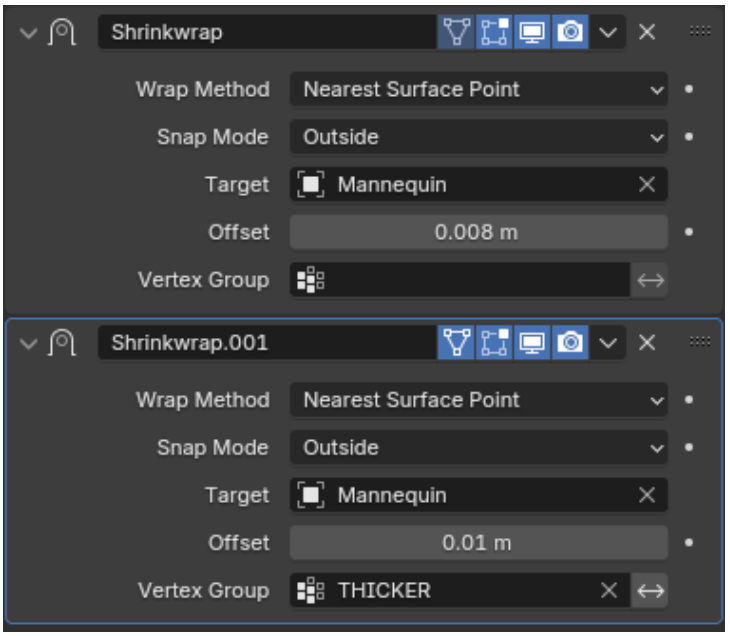


First of all, the primary modifier I use to refit the gear is called the **Shrinkwrap** - apply one from the **Modifier Properties tab** (little spanner/wrench) or your **Quick Favourites** (**Q**) to the mesh. Set the **Snap Mode** to **Outside**, set the **Target** as the **Mannequin** (you may need to enable the Mannequin collection with the checkbox, then hide it with the eye icon to keep it out of the way) and set the **Offset** to float a bit. I typically default to 0.005m and use 0.003m for tight fabrics or 0.008m for thicker pieces. I've gone as high as 0.015m for some details, to give you an idea of the range you can play around with.

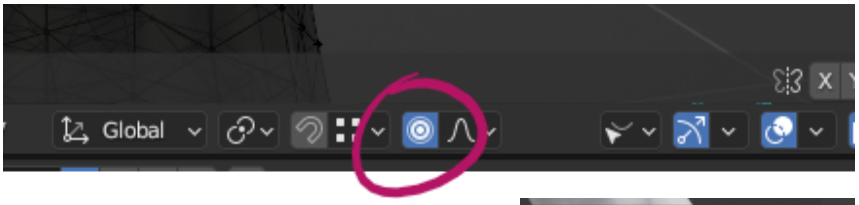
I find some parts like skirts can freak out if I try to shrink the whole mesh, so to rectify this I hide the modifier with the monitor icon, go into **Weight Paint mode**, select the parts I want to exclude, either add a new vertex group called "NO SHRINK" (or add it to the group I created previously), ensure **Auto-Normalise** is turned **off** and your weight slider in the header/footer or the right click menu is set to 1.0**,** then **press Ctrl+X** to set the weight to 1.0 for the selected vertices. Tell the **Shrinkwrap** to exclude this part by selecting the **Vertex Group** in the modifier, then use the arrows toggle next to it to exclude.

If a mesh has multiple thicknesses, set this shrink for the thinnest height, **duplicate** the **Shrinkwrap** modifier with the v arrow next to the camera icon, create a **new vertex group**, name it something like "SHRINK THICKER" and limit the new Shrinkwrap to this group and set it a tad thicker than the previous shrinkwrap. I want to use this on the collar of the knit top for some extra dimension.

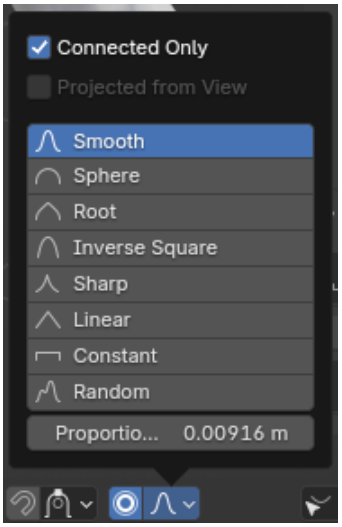


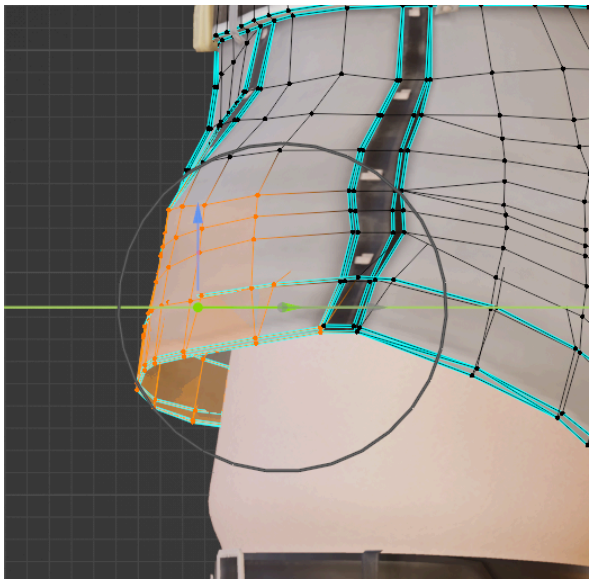


Select the parts you want thicker either in **Edit** or **Weight Paint mode**, then in **Weight Paint mode** enter vertex selection mode with **2** and **set the weights to 1.0** in this new vertex group. I personally prefer to do the selecting in **Edit mode** because I find it a lot easier to see what I'm grabbing. Also means you can make use of **Ctrl+L** to select all linked verts (everything on the same element) or **Ctrl+Num +/-** to grow/shrink selection (which also works in Weight Painting mode), as well as **double clicking** for loop selection.



You may notice that the Shrinkwraps aren't quite doing the whole job for you; some places will be bunched up and you'll probably still have some skin clipping through. So now it's time to move onto the main *tool* we want to use - **Proportional Editing**. Go into **Edit mode** and turn on **Proportional Editing** with the bullseye icon in the header/footer or **O**. You can toggle **Connected Only** in the dropdown (or **Alt+O**) to move by connected geometry rather than distance, which can be useful for things like moving a cape draped over a skirt without moving the skirt itself - though be warned if you didn't merge things properly earlier this can rip parts of your mesh away from one another.





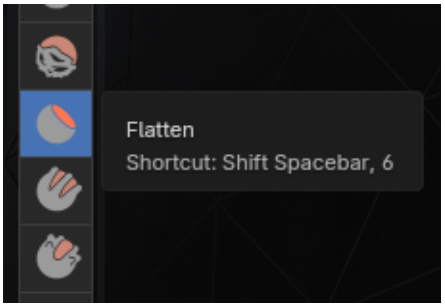
Use camera snapping (**hold Alt** while moving the camera with the **middle mouse button**) for Left or Right view and turn on **X-Ray mode** (**Alt+Z**) to keep your selections symmetrical, then use the **Move** and **Scale** tools to shift the mesh about how you'd like. Scroll your mouse wheel while moving parts with either of these gizmos to alter the influence of your proportional editing, indicated by the size of this grey circle. Use this Proportional Editing alongside your Shrinkwraps to nudge the gear into shape, piece by piece.

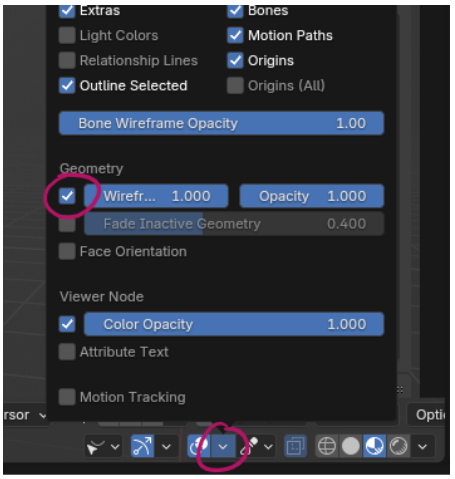
It can be worth seeing how the wireframe is behaving post-Shrinkwrap; you may have noticed that some parts are **without** the mesh overlay visible in **Edit mode** - these are the parts being affected by the modifier. You can view the mesh how it will appear post-modifier by toggling **On Cage** next to the shrinkwrap name.

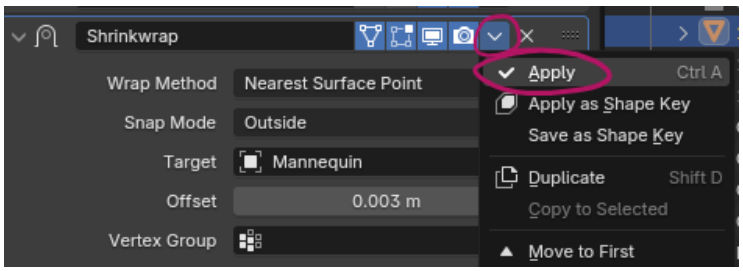
This step usually involves pulling out and widening the chest area a little so the curve is uniform between faces and the texture doesn't stretch too unevenly (though stretching is unavoidable at larger sizes), pulling out the cleavage or the butt so the clothing lies naturally between the breasts/glutes, positioning the underbust so textures line up with the shape and there's no pinching/folding as well as a natural fall to the fabric, and tucking in parts you want to be more form fitting such as the waist.

Before > After

The Shrinkwrap should take care of smoothing out most of the shape for you, but if you have any particularly stubborn scrunched up parts (the cleavage is a common culprit), you can go into **Sculpt mode** and use the **Flatten tool** to gently smooth them out. A tool like this would be ideal, but these are a functional free alternative. You may want to turn on **Wireframe Overlay** to see it working and enable **Symmetrical editing**.







If **Move** and **Scale** aren't working how they should because the mesh underneath the Shrinkwrap is a vastly different shape, you can duplicate your Shrinkwrap and apply it, then continue working with a better base shape: click the arrow on the **Shrinkwrap Modifier > Duplicate**, then hit the arrow on the new **Shrinkwrap modifier > Apply**.

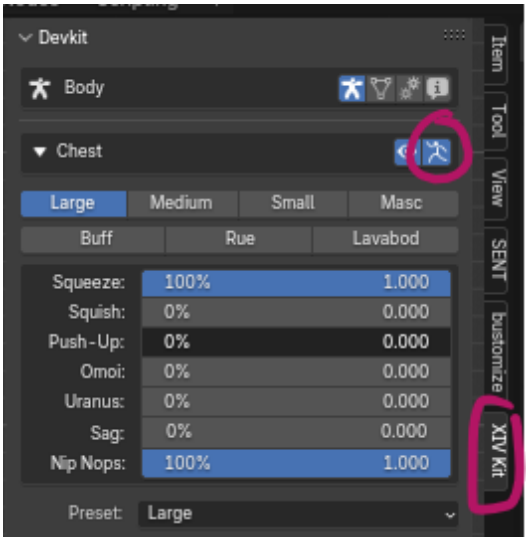
Don't forget to play around with the **shape keys** on the **Mannequin** while fitting too - primarily *Squeeze* to add more cleavage, *Squish* to flatten, *Push-Up* to lift. Play around with them and see how the gear changes to decide how you'd like to set yours! Don't forget to change the Chest's keys to match.

Once you're content with it (you can always add another Shrinkwrap and tweak it some more later), return to **Object Mode**, click the arrow on the **Shrinkwrap modifier > Apply**. Or, if you're making multiple sizes, **Duplicate**, **Apply** the new Shrinkwrap(s), then **toggle this one off** with the monitor icon - we'll use these to touch up the fit on other sizes later.

What if I Change My Mind?

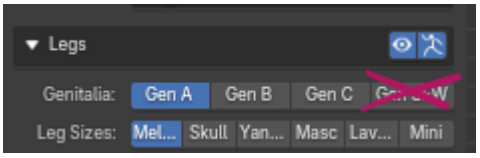
If you've already applied your Shrinkwraps and refit your gear to one shape - say, 1.0 on the Push-Up key - but later decide you'd prefer it to look braless without any Push-Up, you can **edit the shape** with another modifier called **Surface Deform**.

Set your Mannequin to match the Chest shape you upscaled to (1.0 Push-Up in this example). **Add a Surface Deform modifier** to your gear from the Modifier Properties Menu (or **Q**, Quick Favourites). **Target** the **Mannequin**, then hit **Bind**.

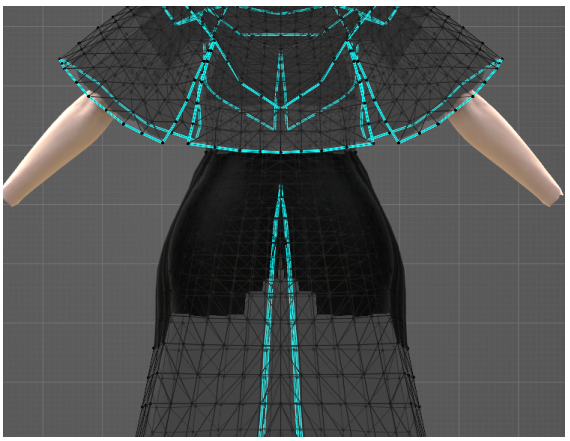


Now you can edit the Mannequin shape keys with the overview to alter your fit as you like, and once you're happy, hit **Apply** on the Surface Deform - you may need to tweak with more Shrinkwraps and Proportional Editing after.

Tip: If you get the error "Target has edges with more than two polygons" when trying to Bind your Surface Deform, open the Legs rollout and toggle to any **non-Gen SFW** shape on your Mannequin.



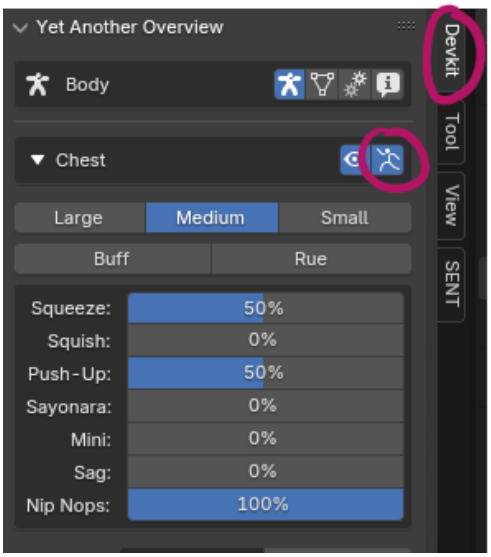
If the top has a skirt or is long enough to cover the booty, I do most of the resizing and reshaping for this part manually. Trying to minimise skirt clipping with how XIV's skeleton is built is a bit of a faff- however, I have found that if I add a **Shrinkwrap**, **Target** the **Mannequin** and give it an **Offset of 0.035m** and then use this as a reference to how far/wide to pull the booty, it often leaves minimal clipping to fix later. Much like with tweaking the chest and waist, I go into **Edit mode** and start pulling verts around with **Proportional Editing** until they match this size/shape particularly towards the lower half of the booty - the primary difference being that I **delete the modifier** after, rather than applying it. Make use of the **show/hide** modifier button (the little monitor) to see how you're doing and if the shape is looking okay.



Repeat this process with all of the parts as needed.

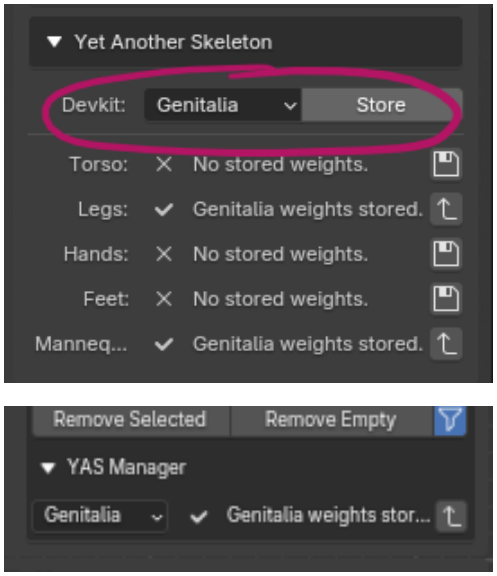
Weighting

When you're happy with the fit, there's one more primary part to the actual refitting - weights! If we were to leave it as it is, the actual body's chest will clip through the shirt because the vanilla meshes don't have as much *mune* (breast) weights as the modded body does. You'll also likely see some clipping in the shoulders, elbows, back... everywhere, really, since the weights are vastly different.



This is what the **Robust Weight Transfer Add-on** is for, it essentially condenses the Data Transfer, Smoothing and cleanup process into just a few clicks. It can be a little finicky though, so I'll go over how to do it normally first - there's some tips in here that will be useful no matter which way you choose.

To begin with, ensure your **Mannequin** is set to the same size as your Chest. **Toggle Rue on** the Chest and Mannequin if you intend to include Rue sizes with YAS, else it wont include the *fukubu* (belly) weights - the Add-on gives us a neat little shortcut to remove them for non-Rue weights later, making it easier than adding them later.

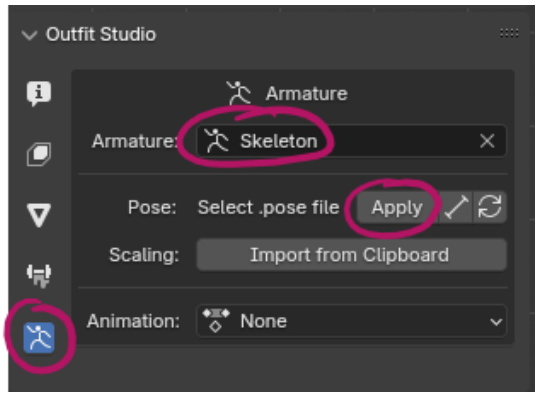


If you're doing SFW bottoms, you aren't going to want any genitalia weights on your legs. Instead of having to remove them later we can store these weights; Go into the **Yet Another Skeleton** rollout, **select Genitalia** from the dropdown and hit **Store**. Doing so from this menu will remove the weights from both the 0.0 Waist mesh and the Mannequin in one go! This function hides the weights without deleting them, so even if you decide you want to add them in later, you simply need to return to this menu and hit the arrow button to **Restore** the weights. You can also do this individually for each devkit mesh via the Weights menu.

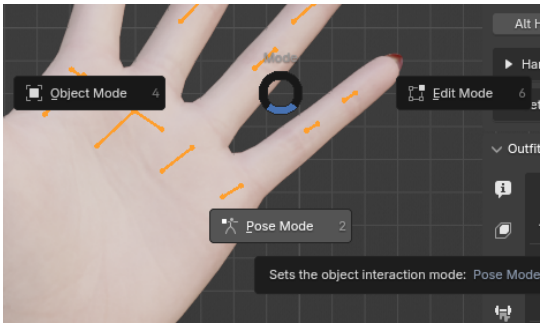
The A-pose the body is already in will work fine for weighting the upper body, but places where two body parts are close together can get very muddy like this - namely between the legs and fingers. To help with this, we can pose the skeleton before we transfer weights. If you've ever seen long boots where a bit seems stuck to the other leg, this is what pre-posing aims to eliminate!

Interesting Technical Stuff!

A-pose is the name for a T-pose with the arms at 45° angles rather than 90° - this is more preferable when modelling, rigging and animating because it causes less stretching over the top of the shoulder joint, especially considering a neutral stance is arms down!

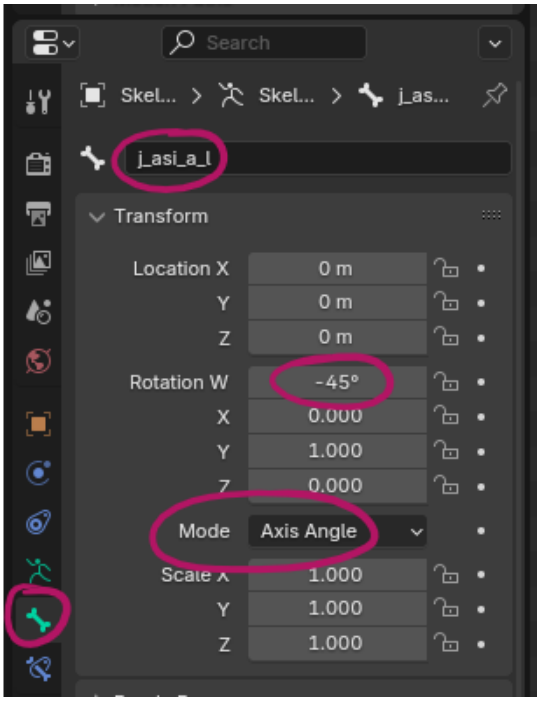


Go to the **Mesh Studio > Armature** menu of your Add-on panel**.** Ensure **Skeleton is selected as your** Armature, then click the **Apply** button to pull up an explorer where you can locate a .pose. I simply went into /gpose in game, did /moogledance, paused at the jump and saved that as a pose. You can grab it here if you're feeling lazy - it works very well for the legs, and decently for the hands!



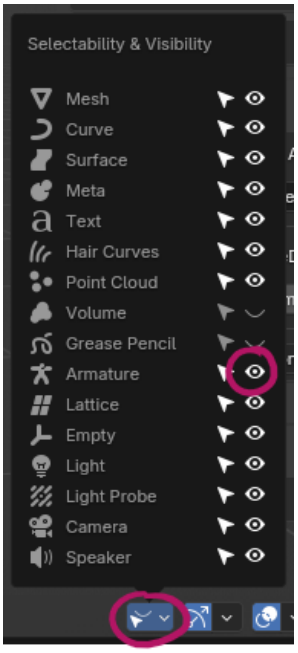
You can revert the pose by hitting the **Reset** button next to Apply on the **Mesh Studio** in the **Overlay**.

Alternatively, you can pose manually by **selecting the Skeleton** in the Outliner and using the **Tab menu** to enter **Pose Mode**. Select the bone you wish to pose (I find it easier to grab from the Outliner for some areas; here is a list of all the names translated) and either use the **Rotate** tool or enter the **Bone Properties menu** (a good place to check you've grabbed the right bone!), swap **mode** to **Axis Angle** and change the **Rotation W value** by dragging or typing the desired value - I prefer to do it this way, so I can do perfect 45°/90° angles.

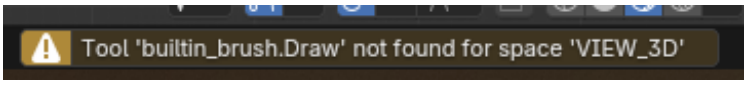


I can't see the skeleton in the viewport!

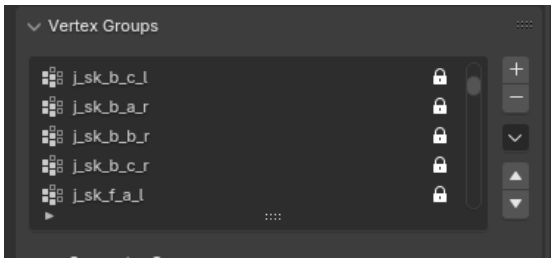
The solution to this is simple; locate the **Selectability and Visibility menu** in your 3D viewport header/footer and **toggle the** eye icon for the **Armature**.



What does "Tool 'builtin_brush.Draw' not found for space 'VIEW_3D'" mean?



This basically just means no tool or object is selected when you're swapping to another view; it's nothing concerning! Just select a tool in your 3D viewport (such as Move, Rotate, Scale, Annotate) and try again.

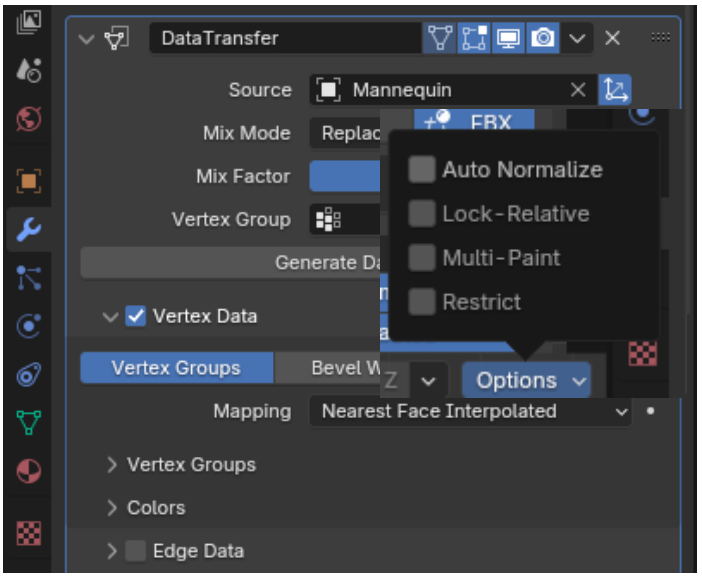


If you have an area you wish to exclude from this weighting (usually skirts), go into **Weight Painting mode**, enter vertex selection mode with **2** and select the verts you want to exclude, make a new vertex group called something like “NO WEIGHTS”, ensure **Auto-Normalise is off** and use **Ctrl+X** to **set the weight to 1.0**.

Additionally, I would go through the **Vertex Group list** and **Lock every weight you want to leave untouched** - *j_sk_* for skirts. This will *not* stop those areas from having weights transferred to them - if your locked skirt weight is 0.5 and *j_kosi* (waist), left unlocked, is the other 0.5, the weight transfer will replace the 0.5 *j_kosi* weight with *1.0 j_kosi* (and likely some *j_asi* (leg) as well) and *leave the 0.5 skirt weight on top*, meaning your weights now total 1.5. But, once we tidy up the weights that are transferred (removing the j_asi (leg) weights, for example), using the **Normalise All** function will normalise only the transferred weights *relative to the locked groups*, leaving them unchanged. It’s a little roundabout, but it works!

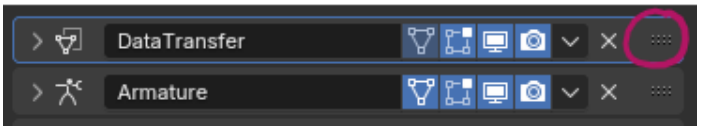
(Here is where you can swap to Weighting with the Robust Weight Transfer Add-on if you want to use that.)

Once you’re all posed and stored, add a **Data Transfer** Modifier to your mesh(es). Set the **Mannequin** as your **Source**, **check Vertex Data** and **select Vertex Groups**. Set **Mapping** to **Nearest Face Interpolated** or **Projected Face Interpolated** - Projected *can* do better, but it doesn’t handle backfaces on the target *or* source mesh very well so it typically only manages things like sleeves or socks cleanly, while Nearest is rather consistent.



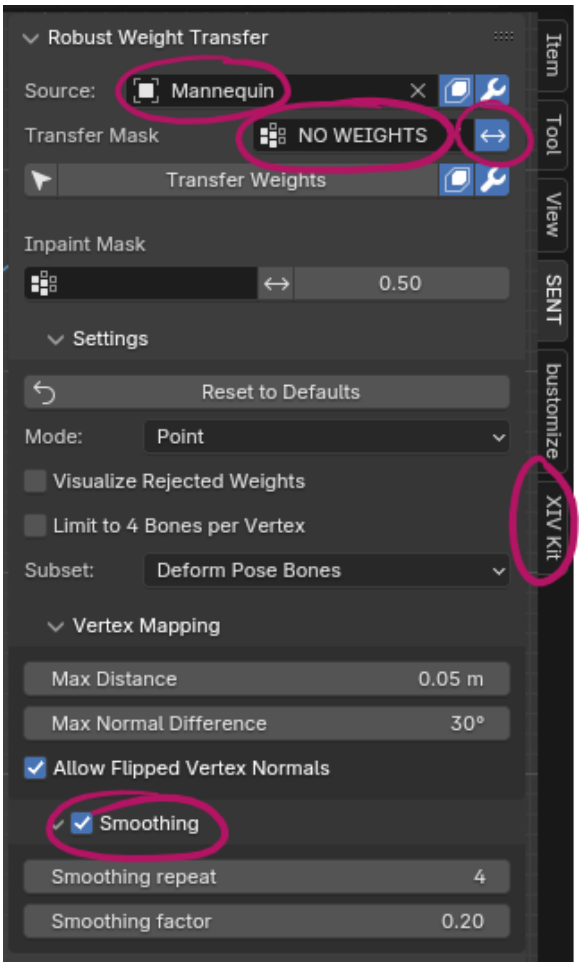
Select the “NO WEIGHTS” Vertex Group here if you made one, and press the arrows next to it to exclude.

Hit **Generate Data Layers** to populate your Vertex Groups list with all the bones onto your mesh - it can’t transfer weights for bones that it doesn’t have.



To see the new weights in action, **drag your Data Transfer above your Armature modifier** with the grey dots on the far side and, if you *did* pose your armature, you should see it move in your viewport. If not, you can go into Weight Paint mode, select a bone and toggle it on and off to see the weights changing.

Weighting with the Robust Weight Transfer Add-on



I absolutely adore the Robust Weight Transfer Add-on and it saves me a ton of time when upscaling! That's not to say it doesn't need any cleanup after, but it needs far less than a Data Transfer modifier.

To use it, open the **SENT** tab located on the side of your 3D Viewport. Select the **Mannequin** as your **Source** (ensure it's **enabled** with the checkbox in the Outliner, or it can behave a little funky!). Make sure **Smoothing** is turned on.

If you made a "NO WEIGHTS" group to exclude, select it in the **Transfer Mask** and hit the arrows to invert the mask.

Then, **hit Transfer Weights** below your Source and Transfer Mask!

This plugin will handle generating the Data Layers for you, and unlike a regular Data Transfer it doesn't add the ones that aren't affecting the area - no more random finger weights on your trousers to remove later.

If you're doing this posed, much like Data Transfer, how the weights transfer is based on how the mesh is posed already so it may be worth hitting Transfer Weights a few times to ensure they're correctly aligned. Judge it by how much the mesh moves as you change the weight, and undo if it starts acting strangely.

You may still need to do some Cleanup and Testing afterwards, but there will likely be far less to tweak with this method.

```
Failed weight inpainting on 1.3 A Skirt: This usually happens on loose parts, where vertices are not findin
g a match on the source mesh. Use Select Rejected Loose Parts to solve the issue.
```

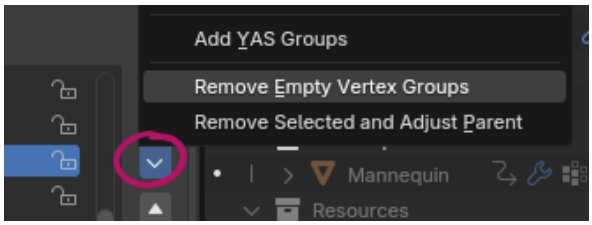
Unfortunately, the aforementioned jank can give you an error like this if the mesh doesn't line up well with the Mannequin, and selecting loose parts often shows nothing. :(If I see this, I weight manually with a Data Transfer instead.

Cleanup

Removing unwanted weights

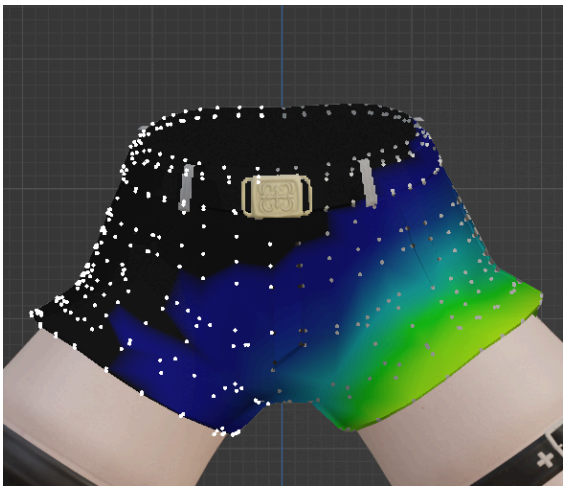
Vanilla Weights > Data Transfer Weights

You may notice... much like the Shrinkwrap Modifiers we used for the initial refit, they don't quite work magic. There's a few steps of cleanup needed. But before we try to fix things, let's **Apply** any **Data Transfer modifiers**, else we aren't tweaking the new weights.

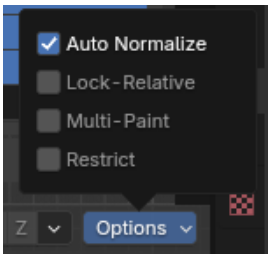


Once your new weights are applied, go into **Object Data Properties** and find the arrow next to your Vertex Groups list. Click this, then hit **Remove Empty Vertex Groups**. This will remove any bones Generate Data Layers added earlier that you don't need (since it just adds EVERYTHING), such as fingers on a leg piece and make the list more manageable.

First of all, we want to remove any weights where they shouldn't be - you'll usually find this in the aforementioned legs and fingers. Using legs as an example, go into **Weight Paint mode**, press **2** to enter vertex selection mode, click on one of your leg bones in the Vertex Group list (*j_asi_a_l* shown) and select the *opposite* leg on the mesh - I often swap back to **Edit Mode** and **toggle the Armature modifier off** in the Modifiers tab to un-pose quickly to make it easier to see what I'm selecting here.



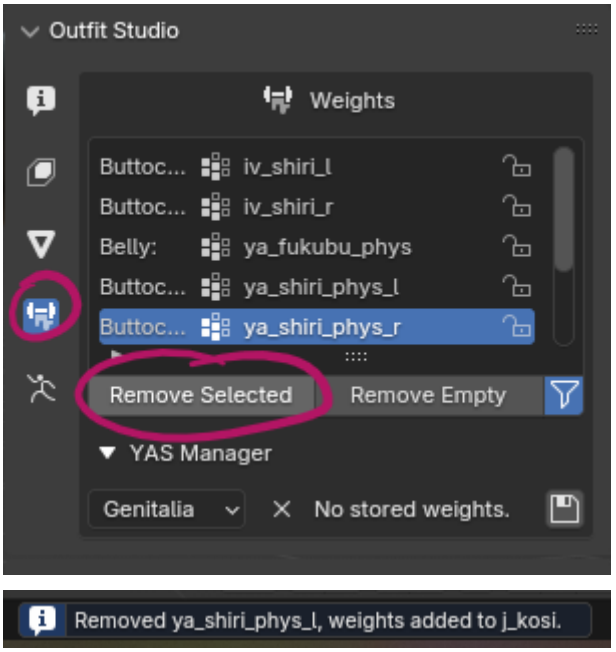
Turn *on* Auto-Normalise and **set your Weight** in the header/footer or right click menu **to 0**. Press **Ctrl+X** to set the weight of the selected verts for the selected bone to 0. If your mesh is still posed, you'll see it move as you do so.



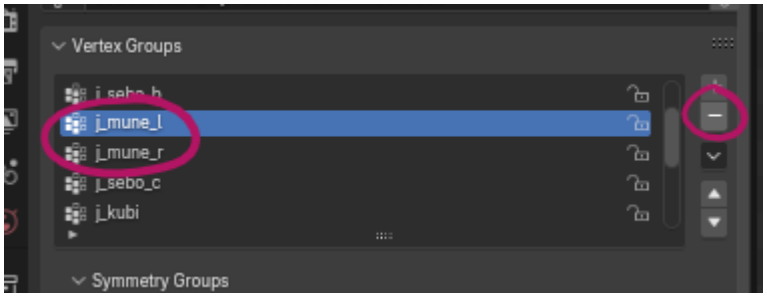
Use **Ctrl+I** to **Invert Selection** (I also make use of **Ctrl+Num +/- to Grow/Shrink Selection** to exclude the middle line of verts from both selections for legs) to **swap to the other leg**, **select the opposing bone** (*j_asi_a_r* for example) and repeat **Ctrl+X** to **set the weight to 0**. Repeat this for any child YAS bones, such as the thigh (*ya_daitai*) and buttocks (*iv/ya_shiris*).

Considering that Data Transfers while posed are influenced by how the mesh lined up with the vanilla weights it had, it can be helpful to repeat the process a few times to get it a touch closer, but keep in mind some areas it will always mess up - I often limit consecutive Data Transfers to a new vertex group to touch up one area, such as the side of the hips.

How about YAS weights you don't want, such as Buttock or Thigh weights on a skirt? Even easier. Go to the **Mesh Studio**'s **Weights menu** and select the weight you want to remove (Use the **filter** button on the bottom right to show only YAS bones to make it easier to find the one you need) and hit the **Remove Selected** button. This will take all of the weights on the bone you're removing and add them to the parent bone instead of deleting them completely (which removing it from the Object Data Properties > Vertex Groups list will do), ensuring your weights remain Normalised (all weights on a vertex total 1.0), which will prevent TT from automatically Normalising them for you and making them a little janky.

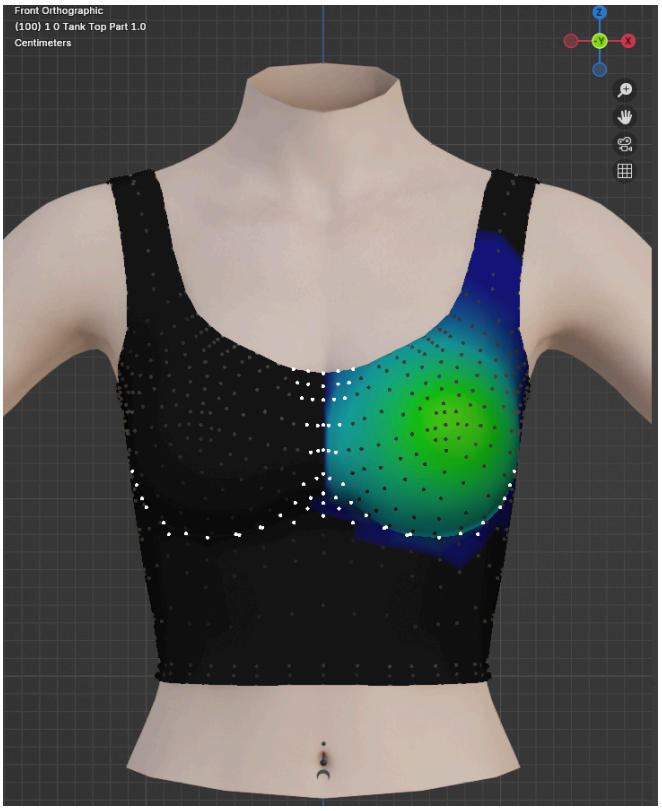


This works with non-YAS bones such as vanilla *j_asi_* (leg) weights on a skirt, too! Though you'll have to turn off the filter and find it in the whole list. Sorry!



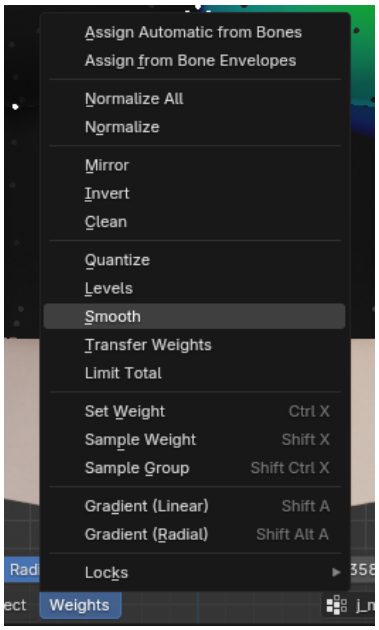
Additionally, the devkit is set up to work in YAS weights and convert them to vanilla on export if you don't want them. While most YAS weights are in *addition* to the vanilla skeleton, there is one exception - *iv_c_mune_l* and *_r* completely replace *j_mune_l* and *_r* - breast weights. When you transfer weights from the Torso or Mannequin, because these have no *j_mune* weights, the weights you transfer are *on top* of these *j_mune* weights. To prevent normalisation issues later we want to remove these, and to prevent normalisation issues *now* we want to **remove these two via Data Properties > Vertex Groups** rather than the Mesh Studio.

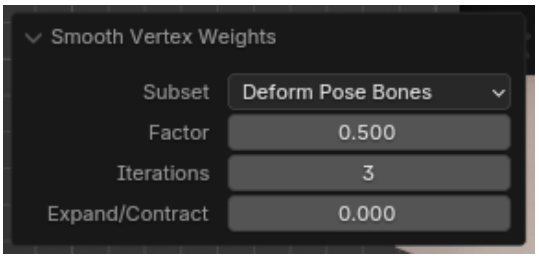
Smoothing



Next up, smoothing. You may see your Data Transfer leaves some harsh lines between weights where the clothing isn't flush to the skin or where there's a fold in the reference mesh (Mannequin), like the cleavage, underbust or armpits, especially when using Nearest Face Interpolated. These can look quite rough in game, so we want to smooth them.

Now in **Weight Paint mode**, press **2** to enter vertex selection mode if you haven't already and select verts where these harsh lines occur, similar to the selection shown here. Be careful not to select any verts that actually touch the skin, as editing the weights there may cause the underlying body to clip through. We will be snipping out excess body mesh later, so if a bit of underboob sticks out of the shirt and you wouldn't be able to see it from any other angle (such as looking down the shirt into the cleavage), so there's no need to stress about a few pixels poking through.



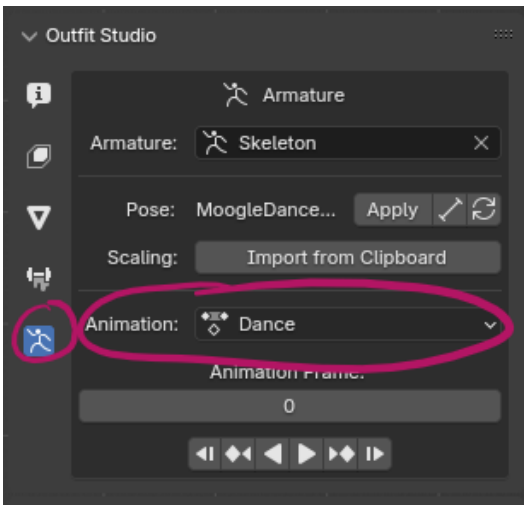


Now click **Weights > Smooth**, set the **Subset** to **Deform Pose Bones** to affect all weights in one go and up the **Iterations** until the line is less harsh, I typically do around 3. The other settings can be left default.

For good housekeeping, follow this up with **A** to **Select All**, then **Weights > Limit Total** with a limit of **8**, then **Weights > Normalise All**, also with the **Subset** set to **Deform Pose Bones**. This limits the number of bones affecting a single vertex to 8 (XIV's engine's limit) and ensures all weights on a vertex total 1.0 to avoid errors upon export/import to TT.

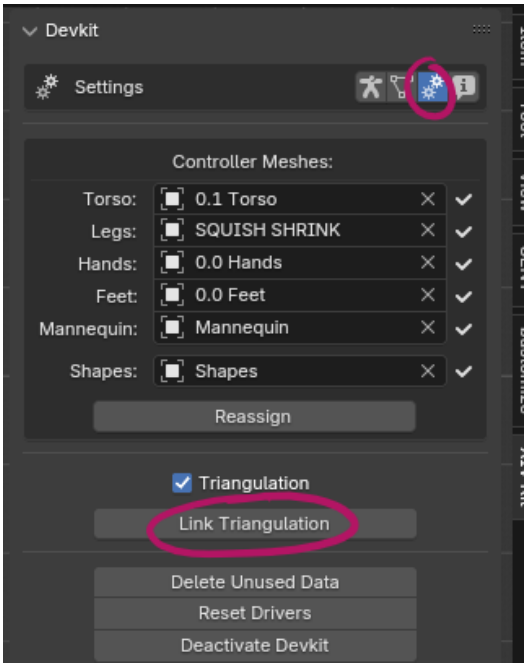
Testing (In Blender)

The mesh in blender and in XIV are never going to act exactly the same, but to save some time exporting your model, importing into TT, getting it in game, then posing in game to check for clipping, you can use the **Mesh Studio**'s **Armature menu** to do some cursory tests.



You can use the pose options like we used for weighting with some different poses (here is Roe Sit 1 and Elezen's Idle 0 if you're still feeling lazy - they're great for testing weights on shorts and undies). But! You can also use this menu to test a few animations, too.

Select the Animation you want to see from this dropdown (Base Pose will set it back to your A-pose) and you can play, stop and skip with the buttons below.



If it's running badly, you can try **disabling the Mannequin** collection from the Outliner, since it's very resource intensive (and you should have applied any modifiers using it as a Source right now) and going into the **Settings menu** of **the Devkit Overview**, clicking **Link Triangulation** to add all Triangulate modifiers to this toggle, then toggling **off Triangulation** here, as Triangulate is a modifier that alters the geometry and is therefore also rather resource intensive. Don't forget to turn Triangulation back on before you export!

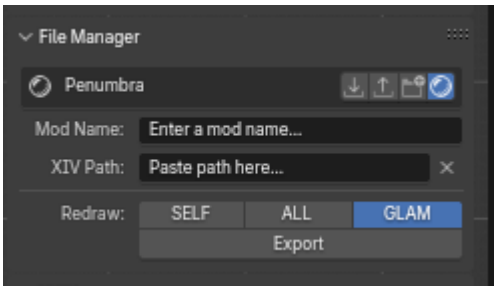
If your PC still can't run them very well, you can also just scrub through to a few frames and take a peek at it in specific points.

Keep in mind you will very likely see some clipping here that isn't visible in game! I would use this to check for large spots of clipping from missing YAS bones or something similar, but I would always take the mesh into game to check over small areas of clipping like this.

If anything clips majorly, return to the start of Weighting and have another go or revisit the Cleanup stage.

Testing (In Game)

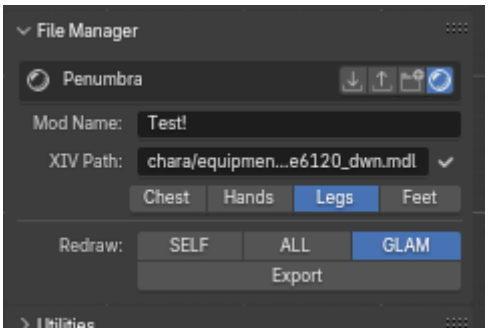
The Addon has a nice new feature that allows us to send a model directly into the game for testing purposes (**currently requires Penumbra testing**)! Have your game and Penumbra up and running.



To do this, open the **File Manager** tab of your **Overview** and enter the last tab - the little ball. This will create a new mod in Penumbra for you - in the first field, enter the **Mod Name** you want to give this new mod.

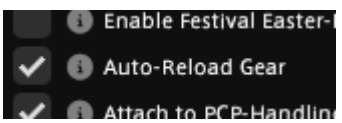
Next, it needs to know which model to replace with the **XIV Path**. It's structured like so; "**chara/equipment/e[SET CODE]/model/c0201e[SET CODE]_[BODY SLOT].mdl**". It uses the same codes I mentioned in [What's With the Names?], if you remember that.

You can either type this in yourself if you remember the codes or want to search them up, or you can also look up the item in **TexTools** and **copy the text** that appears under the model viewer window.



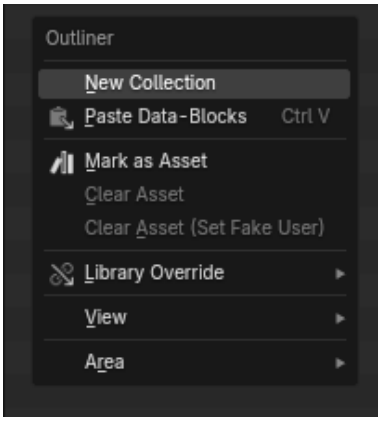
It should look something like this! The Chest/Hands/Legs/Feet selection is there to swap between parts of a set, it should auto-fill from the XIV path you entered.

Then, the last thing to check is how you want the game to **Redraw**. You can ask it to redraw self or all (like the Penumbra commands) or the GLAM button uses Glamourer's Auto-reload gear setting, allowing you to change the model without your character vanishing for a second - I highly recommend this one!



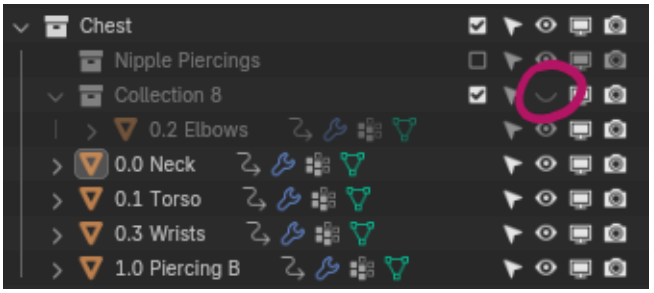
Check your mesh is **Triangulated** (or add a **Triangulate modifier**). The **Export** button will now export your model as it is visible in your scene (so make sure to show/hide parts appropriately). Click it, let it think, then you can examine your model in game, too!

Trimming the Body



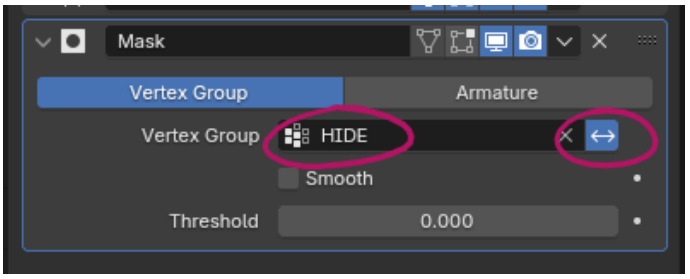
Unfortunately, grabbing the weights from the Mannequin may not remove *all* of the clipping unless you Subdivide the clothing a whole ton (which you shouldn't, be kind to those tri counts). The easiest way to avoid clipping issues is to not have anything that *can* clip, and by this I mean no skin under the clothes if you can't see it (which is *also* kinder to the tri counts).

First of all, check which body parts are visible under the gear. If the elbow (0.2 Elbows) is completely hidden under the sleeves, there's no need for me to export that part. You can use the **eye icon** in the **Outliner tab** to hide all of the parts you don't need, though be aware if you toggle the collection (with the checkbox or the Devkit Overview toggles) they will be unhidden again. My workaround for this is to make a **new collection** by right clicking on the outliner tab, then putting the bodypart in that and toggling the **visibility** for the *collection* instead - it'll stay hidden this way!



Some body pieces may have portions visible, but others hidden, usually on the torso. So, we can go into **Edit mode**, select the visible parts and use **Ctrl+I** to **Invert Selection** (**turning off X-ray mode** with **Alt+Z** here can help to check if you've selected a face that is actually visible). Be sure to check at all angles, even down the cleavage!

Once you're happy that your selection is accurate, go into **Weight Paint mode** and add a **new Vertex Group**. I call mine "**HIDE**" and **set weight to 1.0** with **Ctrl+X**. Add a **Mask modifier** to your torso mesh, select your **HIDE** as the **Vertex Group** and **Invert** selection with the two arrows. You don't want to activate Smooth, as it will alter the normals near the edges.



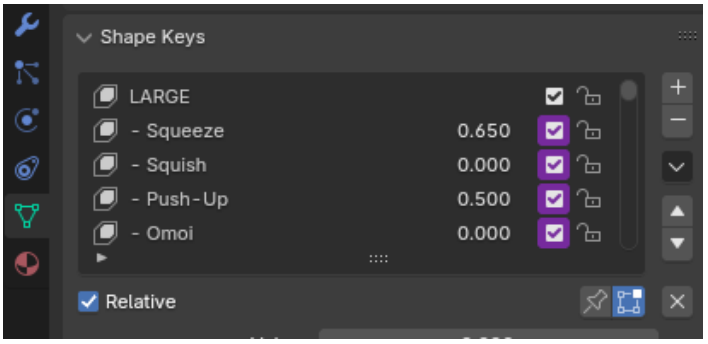
It's essentially a non destructive delete! You can tweak the Vertex Group as often as you need to while you work, or leave this until the last step to include.

If you end up with more (or less) skin visible between sizes, because this is a modifier we can also have it toggle between sizes using **drivers**. I will cover how to add those in Meshes with Differing Weights if you end up needing them, but often if something is clipping between sizes you want to work on adding more topology or tweaking weights rather than snipping out more of the body.

Size Conversions

If you only intend to make the one size, feel free to skip to Preparing for Export. However I do recommend giving size conversions a go - they're a lot easier than you think!

If you have your layout anything like mine and have seen the **Object Properties tab** for the 0.1 Torso or Mannequin at all, you'll see the **Shape Key** list is *full* of sizes, shapes, toggles and morphs. If you take a moment to look closer, you may also notice this list and its values directly reflect the size and shape you've input into the **Devkit Overlay** - the size of the body (/Mannequin) that you're currently seeing.



So... What is a Shape Key? It's a morph of the model to a different shape without affecting the topology (how many faces/edges/verts the model has). By default, a shape key can be both toggled on and off, but can also have blended values - mixing two shapes together. That means you can blend the *Rue* key on top of the *Medium* key instead of also needing an additional "Medium Rue" key (or even a whole separate mesh). It's convenient! It's neat and tidy!

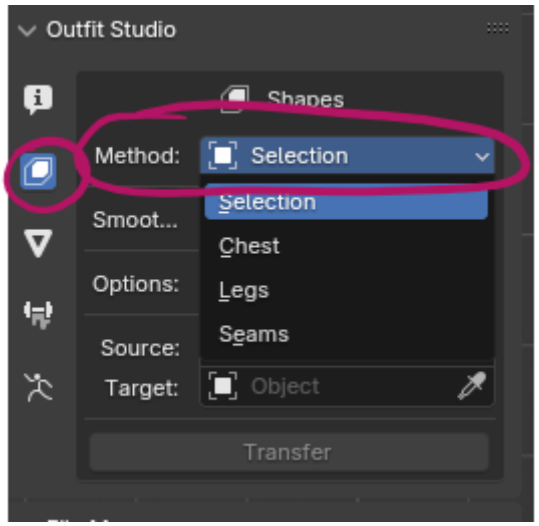
The purple colouring on them indicates they are controlled by a **driver** - an automation process that can be used to link the shape key of one mesh to the shape key (or modifier!) of another to make them activate at the same time. More on that later.

Wouldn't it be so handy if we could link the *gear*'s shape to change with the *body*'s shape...?

Well, fantastic news! **That's exactly the process we want to employ for converting sizes**. And the **Mesh Studio** will do the setup part and a rough size conversion for you!

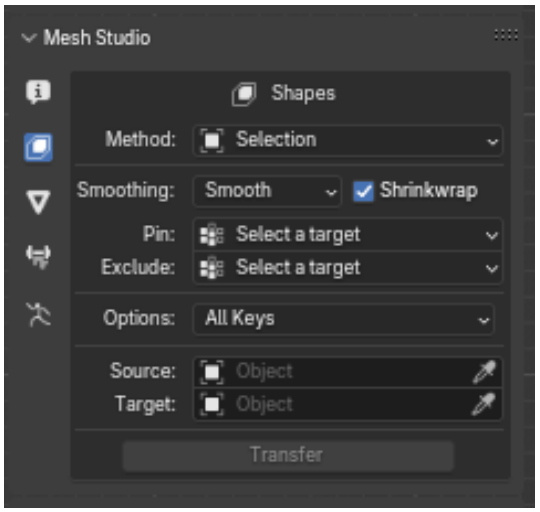
Transferring Shape Keys

This part of the process is where the Mesh Studio *really* shines. Not only can it create the shape keys and drivers to link your gear to the body, but it can also do a rough upscale for you - it's not perfect, much like any of the modifiers we've used previously, but it does save some time.



You'll want to go into the **Mesh Studio > Shapes menu** and pick your **Method** from the first dropdown. I'll go over how they work in order, so you can skip to the section you need. *Chest* and *Legs* are fairly self-explanatory, *Selection* can be used for any other body parts (gloves or shoes for example) as well as relinking your Shape Keys after you Move to a New Devkit. And *Seams* is how we create *shpx_* (custom Penumbra shape keys) for smoother waist, wrist and ankle seams.

Selection

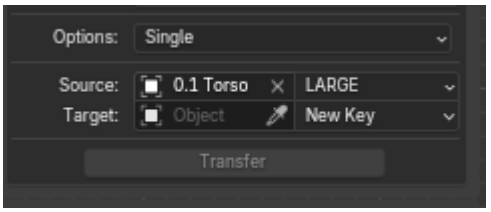


Method: Selection allows you to select any mesh in your scene as the target, making it useful for hand or feet gear that don't have a dedicated Method option, as well as re-linking Shape Keys if they get detached.

Smoothing controls whether you want the mesh smoothed out after it roughly converts the sizes for you, and you can choose between **None**, **Smooth** and **Aggressive**. Selecting Smooth or Aggressive will cause a **Vertex Group field** to appear below, entitled **Pin**. Here you can select a vertex group to "pin" in place - something to remain unaffected by the deforms and smoothing. It's highly recommended you use this, as it gives the addon a good idea of the working area to deform. Consider it for areas like body seams or anywhere that shouldn't be moving!

Checking **Shrinkwrap** will apply a basic **Shrinkwrap modifier** to your mesh when it does your cursory size conversion. It doesn't include a particularly large offset (0.001m), so you will have parts of the underlying body visible after that you'll have to tweak, but this setting aims to prevent the gear from vanishing into the body to make it easier to edit. Checking this box will also cause another **Vertex Group field** to appear below, this one named **Exclude**. This will be for areas you want to exclude from Shrinkwrapping, especially areas of the clothes that cross over body part seams - a long glove or boot that extends beyond the hand or foot, for example - or vertices that need to remain inside the body, like your caps for holes.

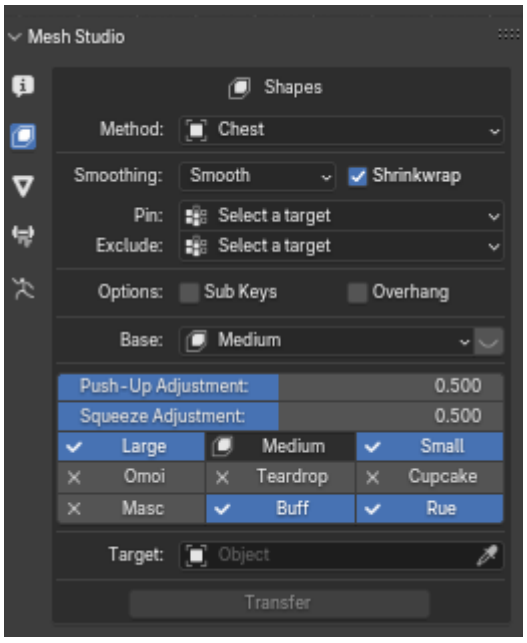
Options lets you select whether you want to transfer **Existing**, **All Keys**, or **Single**. **Existing** will only transfer shape keys that already exist on your target mesh. This is most useful for reattaching drivers after Moving to a New Devkit. **All Keys** will add all shape keys and drivers on the source mesh - this is what I would use for converting hand or foot gear. And finally **Single** will let you pick only one shape key and driver to transfer, if you want to be careful about what you add. It will cause two new dropdowns to appear next to the **Source** and **Target** fields, allowing you to select the shape key you wish to copy, and if you'd like to make a new key or which existing key to replace on your Target.



And lastly, your aforementioned **Source** and **Target** fields. Your **Source** will be the mesh that has the shape keys already - so the Devkit 0.0 Hands or 0.0 Feet mesh, or 0.1 Torso if you want to add *Uranus* and/or *Mini*. And finally your **Target** - the mesh you want to put the shape keys and drivers onto - your gear mesh that you've just upscaled.

Once these options are all filled out, the **Transfer button** will become clickable and after a few seconds, you'll find your gear's **Shape Key list populated** with keys and drivers. You can toggle sizes you included with the Overview to check the drivers are working as intended.

Chest



Method: Chest is, rather aptly, designed for transferring shape keys for Chest meshes specifically.

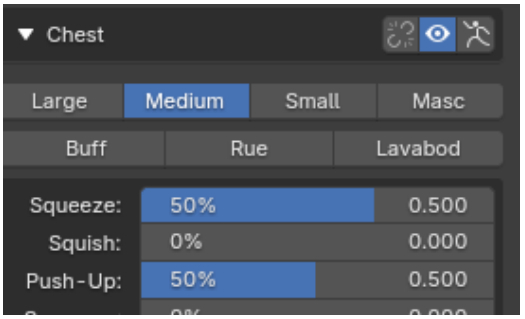
Smoothing controls whether you want the mesh smoothed out after it roughly converts the sizes for you; you can choose between **None**, **Smooth** and **Aggressive**. Selecting Smooth or Aggressive will cause a **Vertex Group field** to appear below, entitled **Pin**. Here you can select a vertex group to "pin" in place - something to remain unaffected by the deforms (or smoothing!). This is used for any area that shouldn't deform and gives the addon a good 'work area', so to speak, so it's recommended you make use of this. Consider it for things like the band of bras, lower back areas that shouldn't deform with any sizes you have selected, and wrist seams.

Checking **Shrinkwrap** will apply a basic **Shrinkwrap modifier** to your mesh when it does your cursory size conversion. It doesn't include a particularly large offset (0.001m), so you will have parts of the underlying body visible after that you'll have to tweak, but this setting aims to prevent the gear from vanishing into the body to make it easier to edit. Checking this box will also cause another **Vertex Group field** to appear below, this one named **Exclude**. This will be for areas you want to exclude from Shrinkwrapping, especially areas of the clothes that cross over body part seams - a long sleeve that extends over the hand or parts of a skirt that extend over the waist seam, for example - or vertices that need to remain inside the body, like your caps for holes.

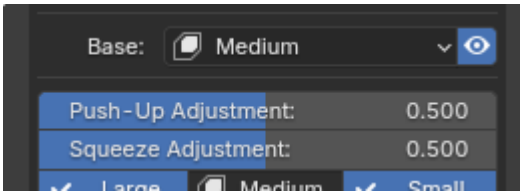
The **Sub Keys** option will also add the additional *Squish*, *Squeeze* and *Push-Up* keys to your mesh, though it won't deform them for you - **you are very unlikely to need this**. You'd only want to include these if you want to tweak the shape later (which you can do with a Surface Deform modifier) or add YAT (Yet Another Tiddy) sizes. Keep in mind, *Mini* is a sub key of *Medium*, *Uranus* a sub key of *Large* and *Sugar* a subkey of *Cupcake* so you will either want this checked (and probably trim excess shape keys after) or the recommended option: use Method: Selection to add those later.

Overhang is to attempt correction for pieces that hang directly down from the bust, such as a scarf or an unfitted shirt.

Next, you'll want to **select your Base**: which size you're converting from - the one you initially upscaled to. Additionally, you'll want to change the **Push-Up and Squeeze Adjustment** to match the values your body is currently set to. Remember that Large's Squeeze starts at 30%, this is for adjustments made *on top* of that percentage. If you used 65% like I mentioned waaaay back, that's only 50% on this Adjustment scale.



Toggle the overlay with the eye icon to check the shape is the same. The overlay is completely flat across the boobs, so don't worry if the cleavage of your top disappears behind it somewhat.



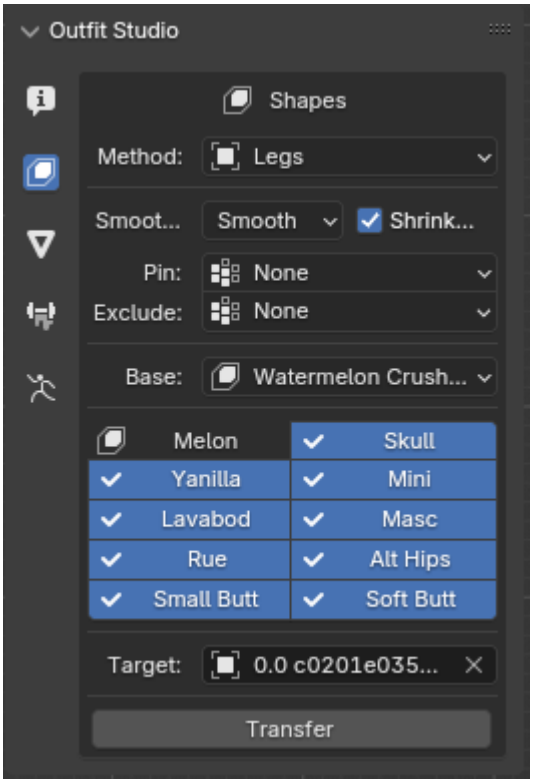
And finally, you can **toggle all the sizes you want to convert to**. *Large*, *Medium* and *Small* (YAB sizes), *Omoi*, *Teardrop* and *Cupcake* (LavaBod sizes) and *Masc* (YAMasc) are all individual chest sizes, while *Buff* and *Rue* are modifiers to be blended with the chest sizes. Don't forget, *Mini* is a subkey of *Medium* and *Uranus* is a subkey of *Large* - if you want to include one (or both!) of these you will need to select Medium/Large also.

Now press **Transfer**! It should only take a few seconds, but Blender may become unresponsive for a moment.

Now if I look at the **Object Properties tab**, I'll see that my tank top has shape keys of its own, all with the drivers required!

Return to the **Devkit Overview** and change the chest shape to ensure the gear is deforming with it correctly.

Legs



Method: Legs is tailored for rescaling bottoms and is probably the most consistent one of them all. If you've skimmed the previous two sections you already know what all of this is about.

Smoothing controls whether you want the mesh smoothed out after it roughly converts the sizes for you; you can choose between **None**, **Smooth** and **Aggressive**. Selecting Smooth or Aggressive will cause a **Vertex Group field** to appear below, entitled **Pin**. Here you can select a vertex group to "pin" in place - something to remain unaffected by the deforms (or smoothing!). Any area that shouldn't move between sizes should be included in this - think the waistband if you aren't adding Rue sizes or ankle seams. This gives the addon a good idea of its work area so to speak, so it's recommended you make sure of it!

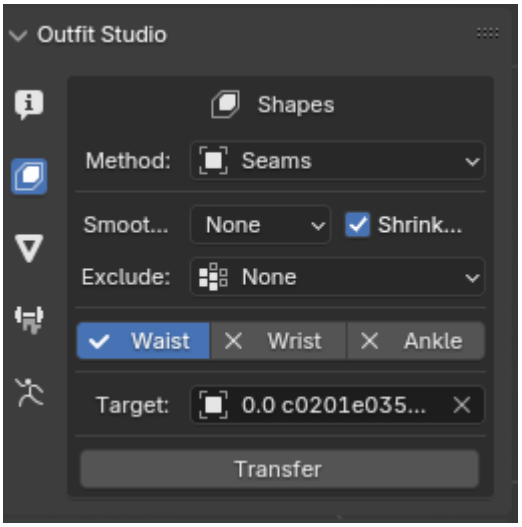
Checking **Shrinkwrap** will apply a basic **Shrinkwrap modifier** to your mesh when it does your cursory size conversion. It doesn't include a particularly large offset (0.001m), so you will have parts of the underlying body visible after that you'll have to tweak, but this setting aims to prevent the gear from vanishing into the body to make it easier to tweak. Checking this box will also cause another **Vertex Group field** to appear below, this one named **Exclude**. This will be for areas you want to exclude from Shrinkwrapping, especially areas of the clothes that cross over body part seams - high waisted trousers or long ones that cover a good portion of the feet, as an example - or vertices that need to remain inside the body, like your caps for holes.

Next, you'll want to **select your Base**: which size you're converting from - the one you initially upscaled to.

And finally, you can **toggle all the sizes you want to convert to**. *Melon*, *Skull*, *Yanilla* and *Mini* (YAB sizes), *LavaBod* and *Masc* (YAMasc) are all individual leg sizes, while *Rue* and *Small Butt* are modifiers that can be blended with any of the above leg sizes. Additionally, *Alt Hips* and *Soft Butt* are some fancy new *shpx_* shape keys (Penumbra custom shape keys) that can be used to add hip dips on YAB sizes, remove them from Rue sizes and make the booty a tad less perky. These don't take up additional "sizes" when it comes to packing your modpack, they can be toggled on any size (at least, any YAB/Rue size in the case of *Alt Hips*). Also they cute as hell and I think you should include them. :3

Now press **Transfer**! Give it some time to think - Blender will likely stop responding for a few seconds while it processes. Once it's done, return to the Overview and toggle through your sizes.

Seams



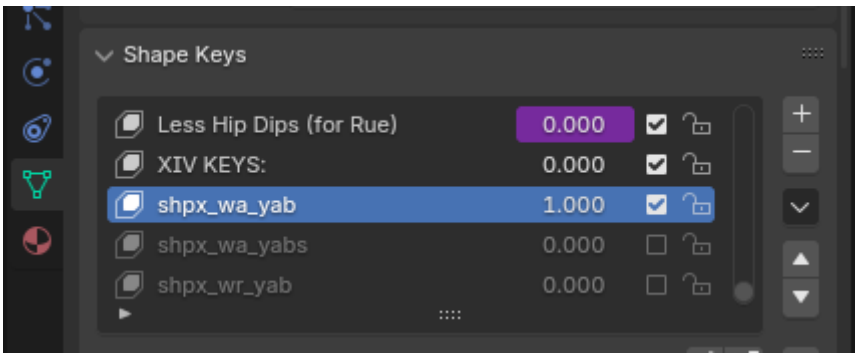
One of the newest features Penumbra has been updated to include is custom shape keys, denoted by "*shpx_*" at the front of the name as opposed to simply "*shp_*", the vanilla shape key prefix. One of the main uses for this is allowing models to deviate from the vanilla seams between body parts so long as the model on the other side of the seam has the matching shape key. This means we can have smoother seams and not have to worry about compatibility with vanilla/*shpx_*less models! **Method: Seams** is designed specifically to add these seam keys to your model. There aren't many options to sort through here.

Smoothing controls whether you want the mesh smoothed out after it roughly converts the seam for you; you can choose between **None**, **Smooth** and **Aggressive**. Selecting Smooth or Aggressive will cause a **Vertex Group field** to appear below, entitled **Pin**. Here you can select a vertex group to "pin" in place - something to remain unaffected by the smoothing. Consider this for buckles and buttons that need to retain their shape. I find this one does best without smoothing!

Checking **Shrinkwrap** will apply a basic **Shrinkwrap modifier** to your mesh when it does your cursory seam-shape conversion. It doesn't include an offset, so you will have parts of the underlying body visible after that you'll have to tweak, but this setting aims to prevent the gear from vanishing into the body to make it easier to edit. Checking this box will also cause another **Vertex Group field** to appear below, this one named **Exclude**. This will be for areas you want to exclude from Shrinkwrapping, namely vertices that need to remain inside the body, so the flat top of my trousers.

Select the seams you want to add on your piece, **Target** the mesh you're adding the keys onto, and hit **Transfer**.

Unlike the other shape conversions, these keys are not controlled by the Overview. To test them, you will have to go into the Shape Keys list in Object Data Properties for both your edited mesh and another mesh to test the seam with - I often choose the Mannequin, since it will show both sides of the seam - and enable and set the value to 1 for both.



Tweaking Shape Keys

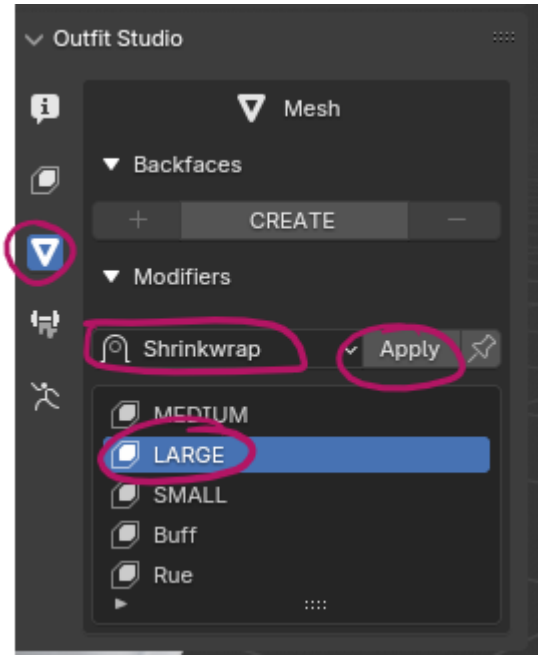
Now, much like the modifiers we used to initially upscale, this rough process is incredibly helpful and saves a ton of time but it certainly isn't magic and we *will* need to do some tweaking.

Looking at LARGE, the shirt is vanishing into the chest in several places - it's completely lost the 0.003m offset we originally gave it with our Shrinkwrap.

Correct it by adding another **Shrinkwrap modifier** with the **same settings** you used previously (If you're targeting the Mannequin again, be sure to change the Mannequin's size). However this time we don't want to simply apply it as that will alter our base mesh instead of the LARGE shape key.

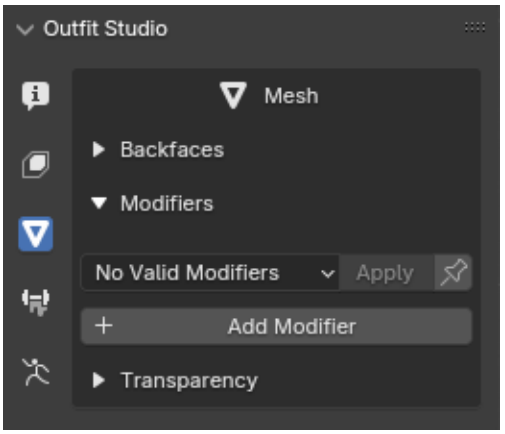
We could apply this as a Shape Key but it doesn't have the driver attached, and if we were to attach it, the list of shape keys could get out of control... So instead, we blend them!

Open the **Mesh Studio**, go into the **Mesh menu** and open the **Modifiers tab**. Select the Shape Key you want to edit from the list, select the modifier from the dropdown (in this case, the **Shrinkwrap**) you wish to add from the dropdown, then hit **Apply**! The pin button will keep the modifier after application, meaning you can use it for different sizes as well. Useful!



No Valid Modifiers?

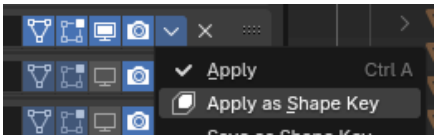
Sometimes when you open the Modifiers rollout it will say "No Valid Modifiers" due to... caching jank or something. Simply click onto another object in your scene and back again with the menu open and it should behave.

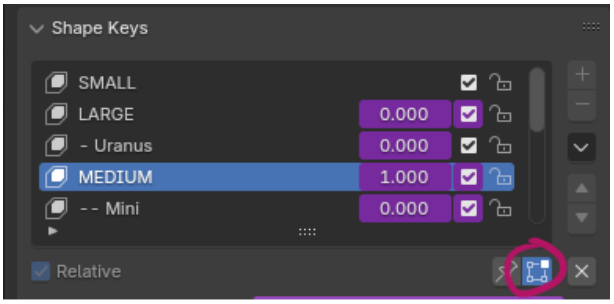


Without the Mesh Studio

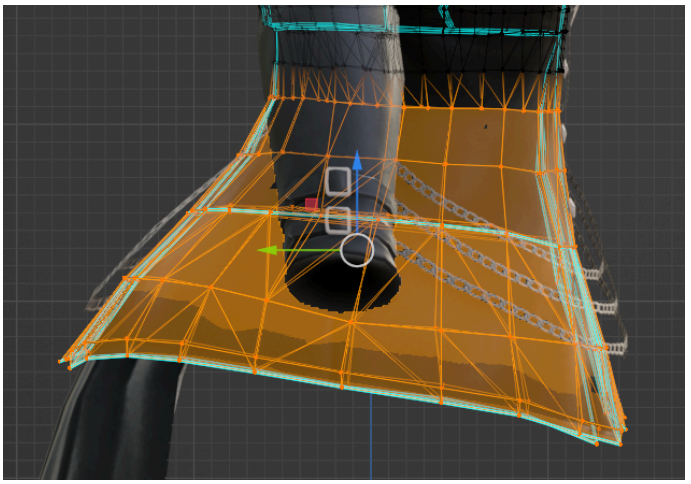
This is functionally the same as applying the modifier as a shape key and using the **Blend from Shape** tool to add it in then deleting the excess Shape Key, condensed into one click. In case it doesn't work right or you want to use this process elsewhere, here's how it goes:

Set the Shrinkwrap (or any other modifier you're using) as you like, then click the arrow and select **Apply as Shape Key**.



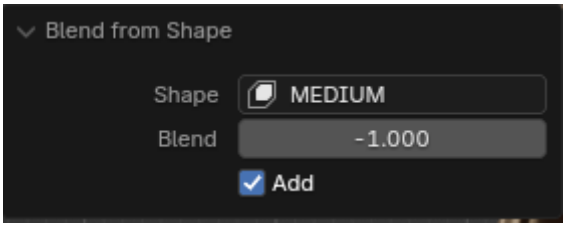


Select the shape key you want to change from the **Object Data Properties > Shape Keys** menu and click the **Edit Shape Key** button below the list.



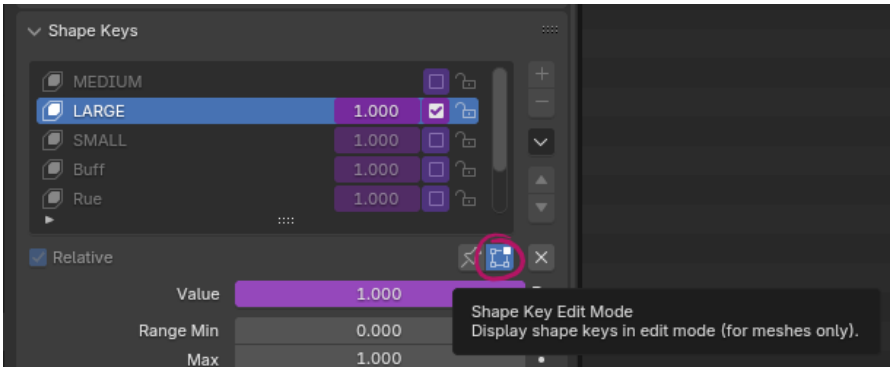
Go into **Edit Mode** and select the part of the mesh you want to edit the shape of - it doesn't have to be the whole mesh! Click the **Vertex** menu **> Blend From Shape**.

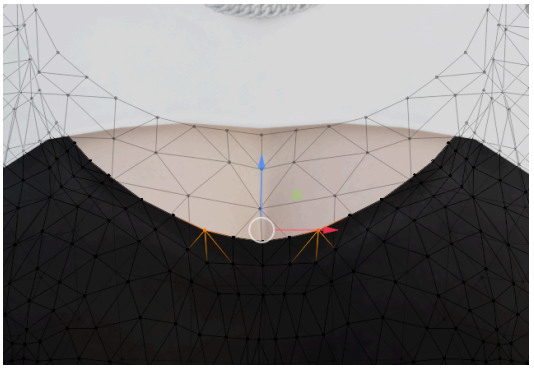
In the settings that pop up, select your newly applied Shrinkwrap (or other modifier) from the bottom of the list. Leave the Blend at 1.0 and **untick Add**, and you should see your shape key move into place.



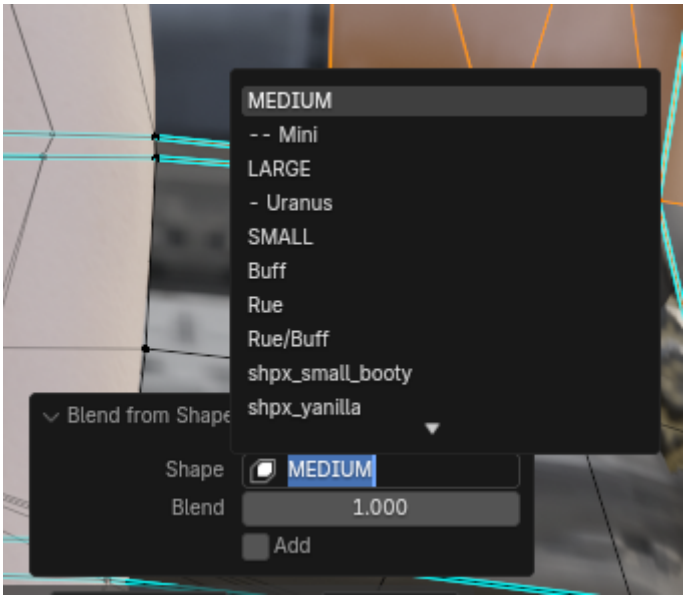
For sub keys like *Mini* and *Uranus* sizes, you may find when you apply a modifier to the shape it can blow up or shrink unexpectedly. This is because it's adding the Medium/Large deform it's a sub key of at the same time (assuming they are not your basis). But not to worry, there's a quick fix! In **Shape Key Edit Mode**, select the whole mesh with **A** and **Vertex > Blend from Shape**. Select MEDIUM (for Mini) or LARGE (for Uranus) (or the parent key to whichever subkey you'd just edited), set the Blend value to -1 and check Add. This will remove the parent key edits, leaving only the subkey alterations intact.

If something still isn't looking right and it can't quite be corrected by modifiers, you can directly edit the Shape Key by toggling **Shape Key Edit mode** below the Shape Keys list in the **Object Data Properties tab**. This will allow you to edit just the shape key in **Edit mode** as if it were the base mesh (barring altering the topology). I used it here to round out the neckline again.





But sometimes this conversion doesn't work quite as well as we'd like - I find Rue shape tends to scrunch up the sides of the hips and Buff can make the arms on baggier shirts look kinda puffy, especially if we don't make liberal use of the **Pin** function when transferring our shapes. This is because the Mesh Studio's process for creating these shapes is the same as using a *Surface Deform* modifier. It imitates shape changes on a reference mesh onto your target mesh, which is well and good for most things... But imitating the Buff shape key from the body on my Knit Top's baggy sleeves here will puff up the *whole* sleeve, while in reality the only part that would actually be affected would be a portion around the shoulders and the bicep, where the larger muscles would push the fabric outwards. If you read this guide a long while back, you may remember I liked to do these size conversions with a Shrinkwrap modifier, rather than a Surface Deform.



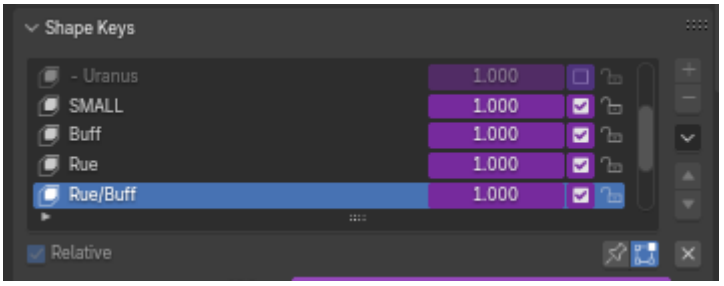
So for any shape keys I'm not so fond of the rough upscale I've been given, I will use **Blend From Shape** as described above to revert it to my basis shape - Select the **Buff shape key** with **Shape Key Edit Mode** enabled, **Select All** with **A**, click **Vertex** (**or Q/Quick Favourites**) **> Blend From Shape**, then set the **Shape** to blend from as your basis - the shape at the very top of the list. **Uncheck Add** to tell it to completely overwrite the shape, and the Buff shape key is now empty.

Then, as I did for the initial upscale, I'll add a **Shrinkwrap modifier** with the same settings I used for the refit (0.008m in this case), set the Mannequin to **Buff** (but keep it as my basis chest size, else Buff will also edit the breasts), use **Mesh Studio** > **Mesh menu** > **Modifiers** to **Apply** the **Shrinkwrap** on the empty Buff Shape Key and voila! My Buff shape key is no longer empty, nor is it strangely puffy!

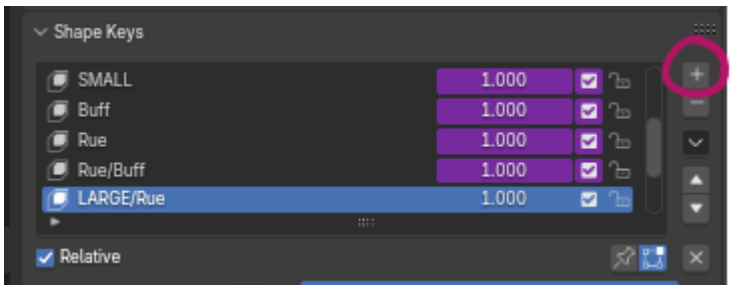
No more peek-a-boo bicep either. :)

Simply repeat this process with each Shape Key until they're all to your satisfaction, and be sure to test how they blend together - especially look at the underbust of Large and Small with Rue enabled, as that's a spot where they can interact strangely.

Adding Corrective Shape Keys and Drivers

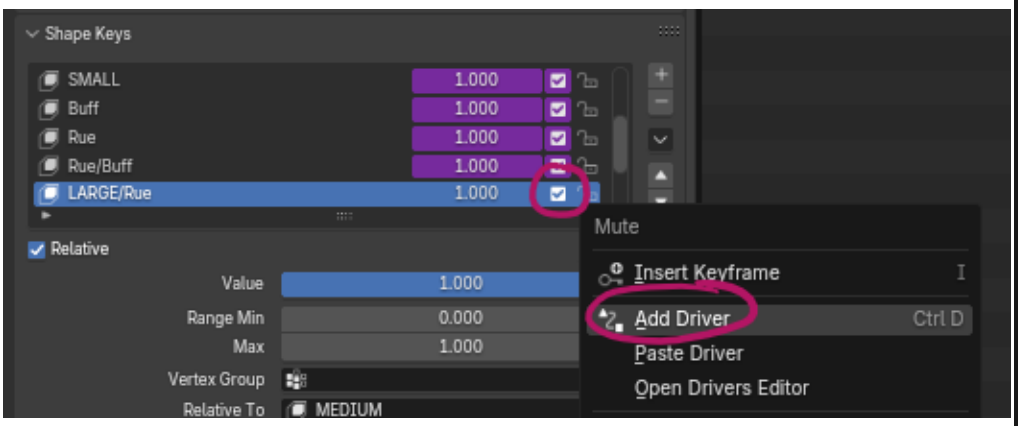


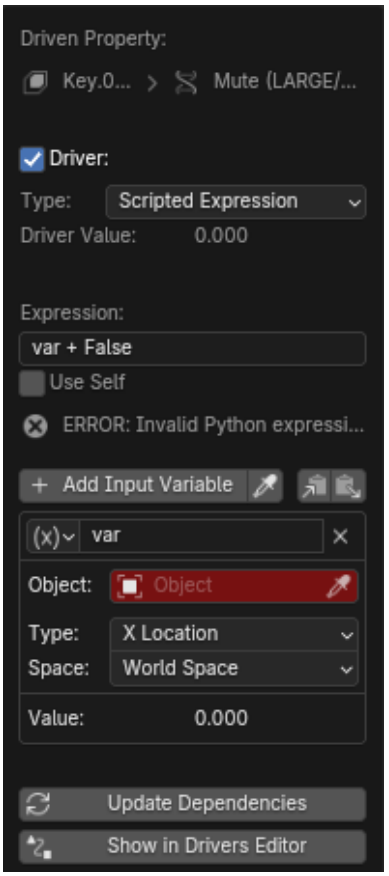
What if they do interact strangely, but you like the shapes you have for the keys alone? If you transferred the keys for both Buff and Rue, you'll notice there's also a Rue/Buff key in your list that only triggers when both are active. This is a corrective key - the Buff key edits the abs on the body, but Rue doesn't have any abs, so this is the key that handles that. This is the same process we want to employ to make a correction!



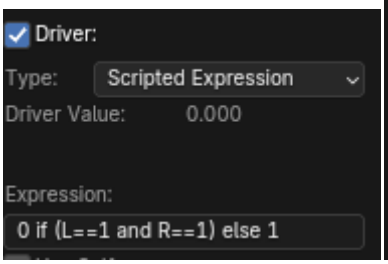
Add a new shape key with the **+** button and name it based on the keys you want to add correction for. I'm going to go over LARGE/Rue as an example. Set the value to 1.0, enter **Shape Key Edit Mode** and make your corrections.

Now, driver coding can get a little wordy, so bear with me here.



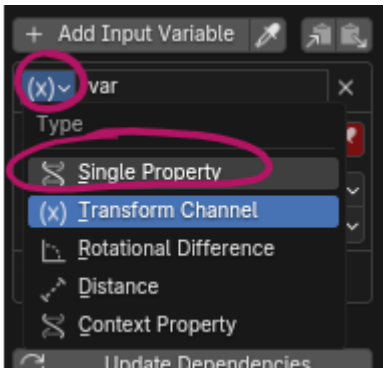


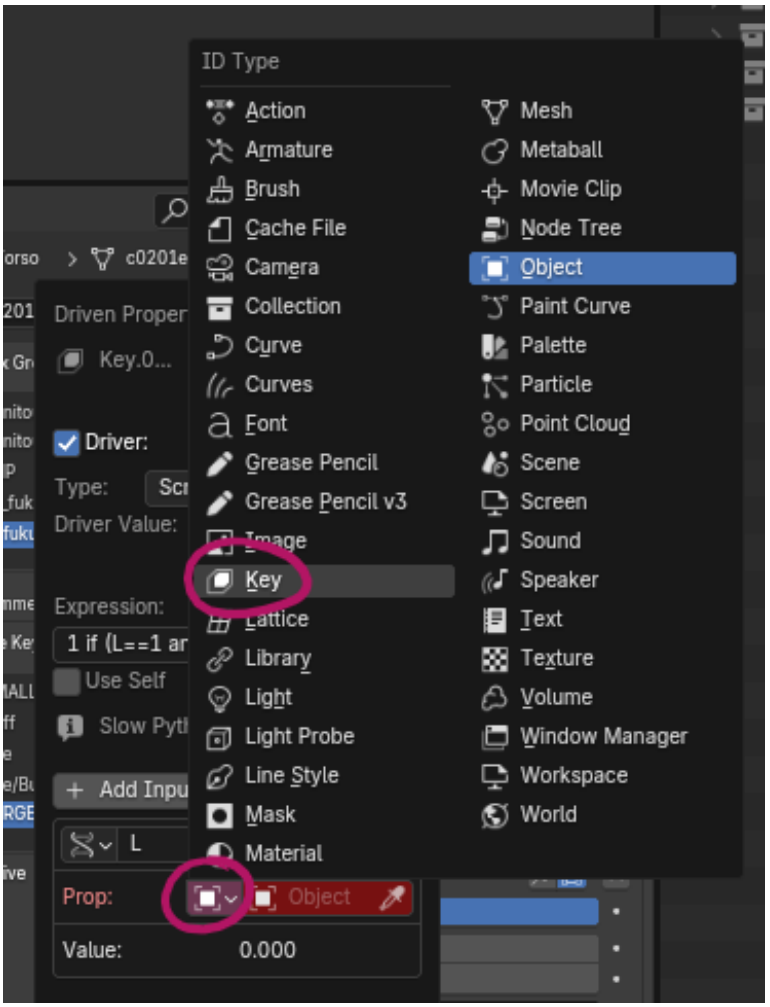
Right click the **mute button** on your new key and select **Add Driver**. This will bring up a small Driver Editor window, which you can now re-open at any time with **Right Click > Edit Driver**.



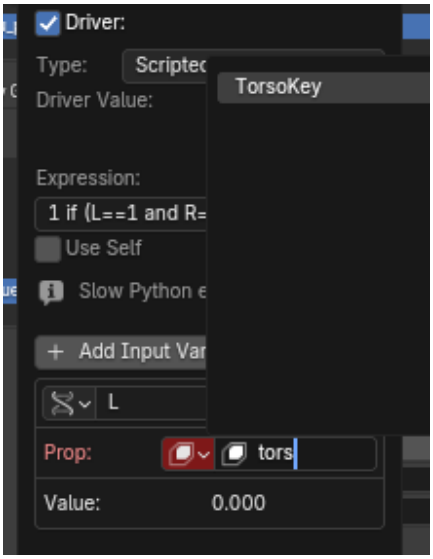
First, ensure **Driver Type** is **Scripted Expression**. In the **Expression field** below, type **"0 if (L==1 and R==1) else 1"**. Essentially *'ON if LARGE and Rue are both on, else OFF'*. (If you're keen you may be thinking "but surely 1 is on and 0 is off!" but annoyingly the logic for mute buttons is reversed...) Ignore the error below for the moment: it's because we haven't set "L" and "R" as our variables yet.

Next, set the existing **Variable** below to use **Single Property** with the dropdown before the name. **Rename it to L** (it doesn't have to be a single letter or anything, I just like to keep it short and sweet).





This will change the fields below to look for a **prop**. Select the **dropdown** to swap this to a **Key** instead of an Object.



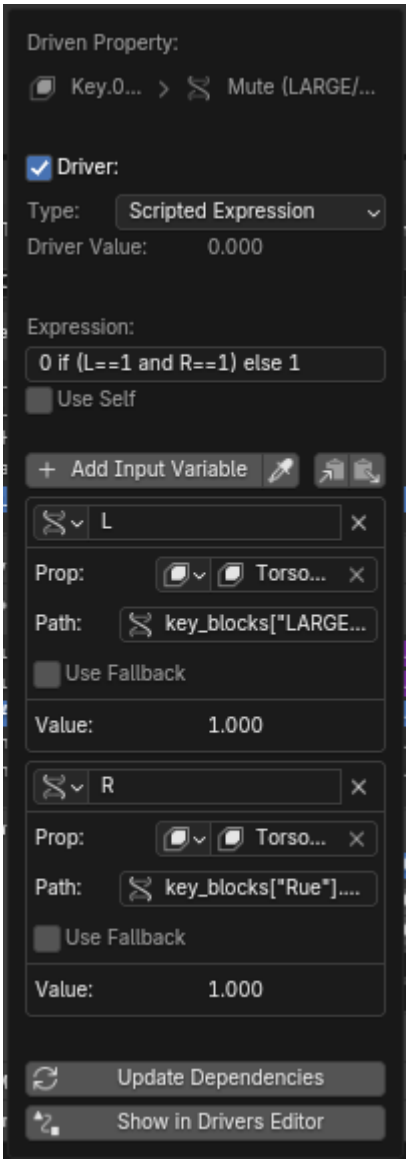
Once this is looking for a Key, search in the field next to it for **TorsoKey**.

In the new **Path** field that appears, type or paste in "**key_blocks["LARGE"].value**". If you have the Large shape enabled in your overview now, you should be able to see the value below change to 1 to show you the value the path is referencing.

Now we want to hit **Add Input Variable** and repeat for the Rue key - just name it "**R**" and set the path as "**key_blocks["Rue"].value**" instead.

Once you're finished, your Driver Property menu should look like this! >

To do this with different keys, for example Small, I would name it "S" instead of "L" (don't forget to replace the name(s) in the expression as well) and replace the "LARGE" text in the path with "SMALL" - ensure it matches the name of the shape key you want to reference, which *is* case sensitive. Chest sizes are all caps, and sub keys require the dashes and spaces, e.g. "-- Mini" or "- Uranus" - just copy paste it from the name of the shape key if you have trouble!



Splitting the Sleeves/Legs

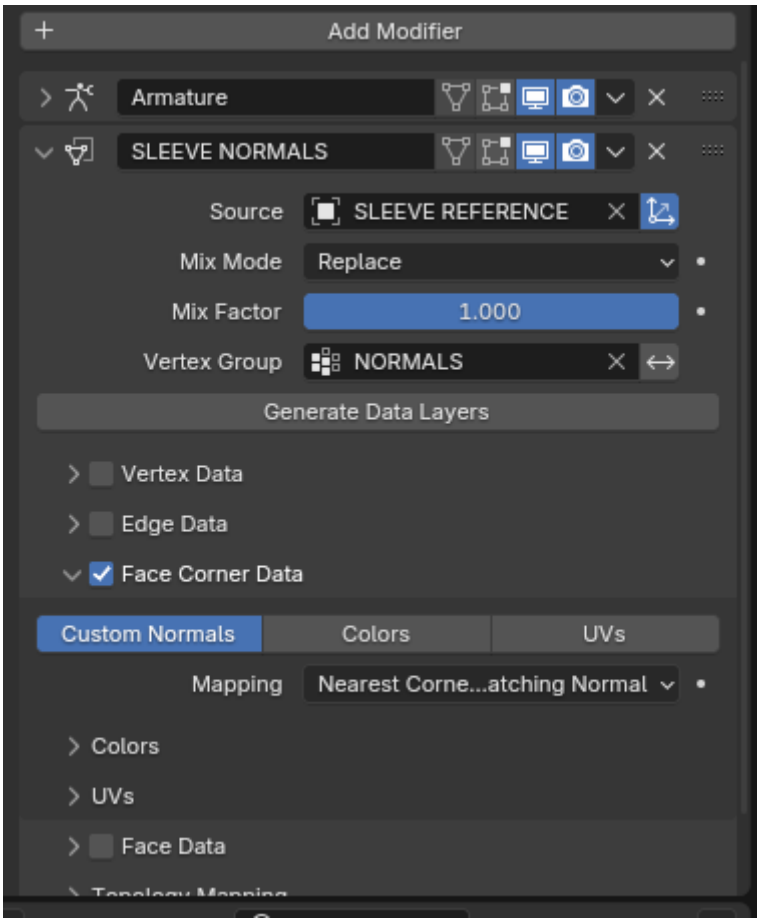
If your sleeve covers the whole arm like mine does, you'll remember we needed to join it for it to work nicely with the Mesh Studio's shape transfers. However, it needs to be split into parts to hide with various lengths of gloves (or boots, if you're working with leg gear) in game, exactly like the Chest and Legs meshes are split into several parts.

> Tip!
>
> I'd recommend taking a quick detour back to Testing (In Game) to get your un-split model(s) into the game to ensure there are no weights or shape mishaps, as these will be a tad more work to fix after your mesh is split up.

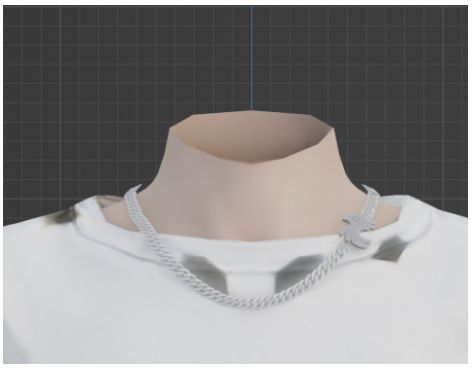
Before you start splitting parts though, you need to keep in mind that because shape keys like Buff and Skull Crushers shapes change the shape over the seams between mesh parts, the *vertex normals* will be different for those sizes - the direction a vertex or face is pointing. This will create tiny little bumpy seams when the shape of the gear is edited with these shape keys.

To remedy this, we want to create a **new collection inside your Chest collection** (or Legs collection if you're working on leg gear), name it something like **"REFERENCES"** and hide it. **Duplicate** your whole sleeve/leg mesh with **Shift+D** and rename it to something like **"SLEEVE REFERENCE"**. Drag and drop it into your REFERENCES collection.

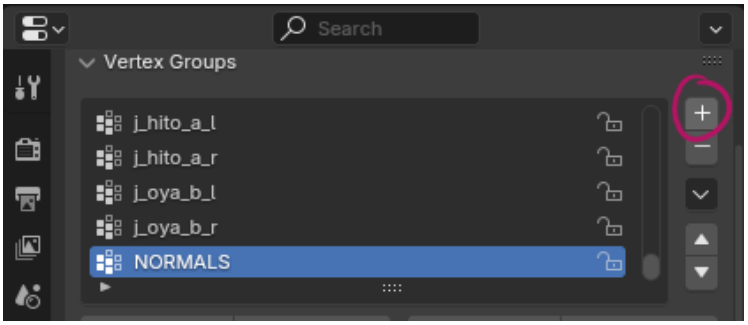
Return to your Chest collection's sleeve part and add a **Data Transfer** modifier. Rename it to something like "SLEEVE NORMALS" (you'll likely end up with several Data Transfer modifiers here, so it's best to keep them identifiable) and select your **SLEEVE REFERENCE** as the **Source**. Check **Face Corner Data** and **Custom Normals**, and it should be on **Nearest Corner and Best Matching Normal** for **Mapping**.

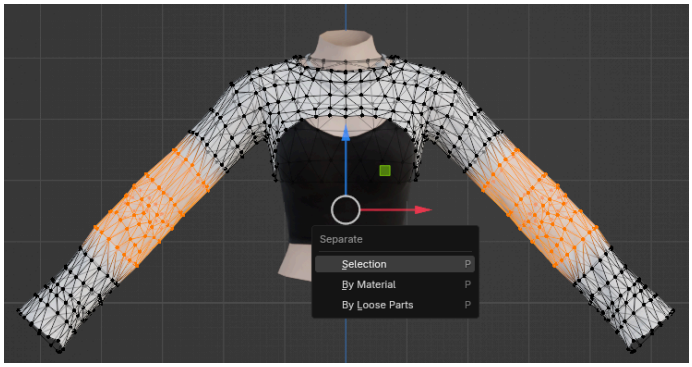


You may notice parts with backfaces like the collar here freak out a little because they're grabbing the normals from the wrong places, so we want to create a vertex group to limit this modifier to only the spots where you need it. If you used the annotate tool to make a note of where the seams are, this is a good time to unhide your notes! If you aren't sure where the seams are going to be, feel free to just select the whole sleeve (or select the problematic parts and exclude those specifically!).

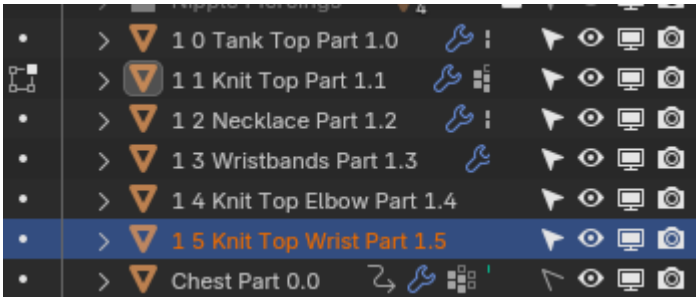


Click the **+** in the **Vertex Groups** list of the **Object Data Properties** tab, rename it to something like **"NORMALS"**, enter **Weight Painting mode**, **2** to enter **Vertex Selection mode**, grab your verts and **Ctrl+X** to **set weights to 1.0**. Add this **Vertex Group** to your **Data Transfer modifier**.





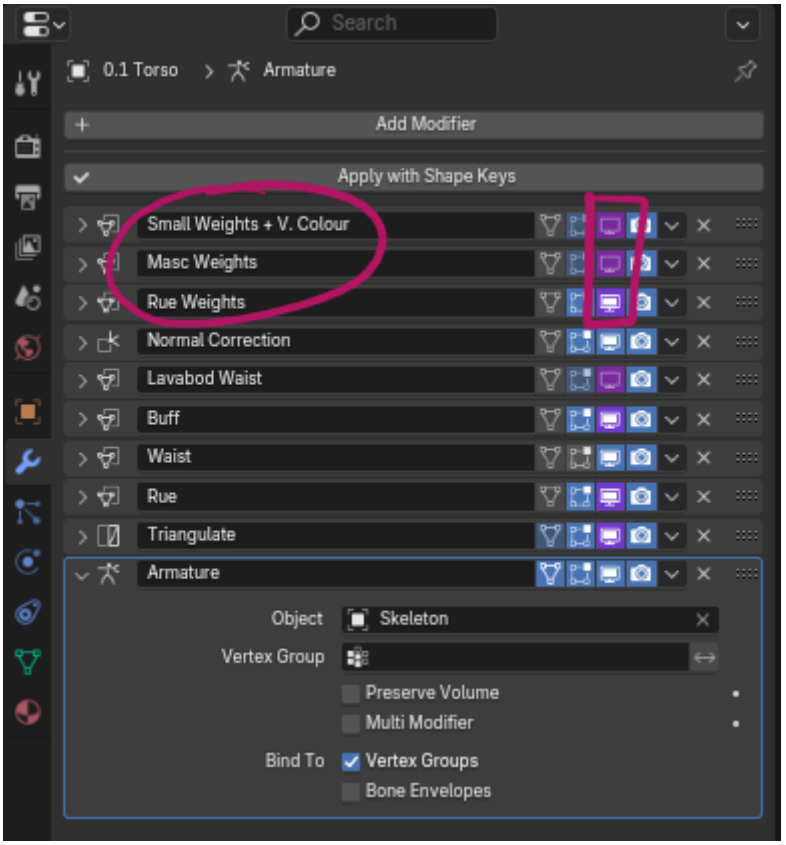
Now it's safe to split the mesh. Go into **Edit mode** and select all of the verts that constitute the Elbow part (or grab everything in between if you still have the seams selected from **Weight Paint mode**). Press **P > Split by Selection**. **Rename** your newly split part with a unique Part number, and repeat for the Wrist.



Because you added your Shape Keys and Data Transfer modifier before splitting them, these new parts will directly inherit all the same shape keys and modifiers. This way you only had to add them once, rather than for each individual part!

Meshes with Differing Weights

Now, while what we have currently works great for changing the *shape* to match different sizes, shape isn't all that changes between sizes. For example, Small has different breast weights to Medium and Large, and Yiggle Rue has additional belly bones that Yiggle *YAB* does not.



If you look at the 0.1 Torso mesh, you'll see something very interesting in the modifiers - Data Transfers for weights that are *also* controlled by drivers!

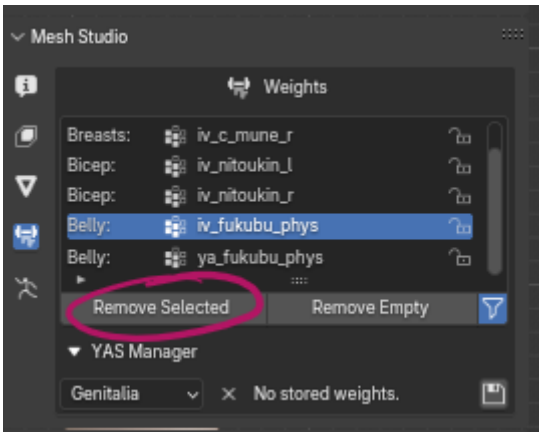
This functions by having a duplicate mesh with correct weights for each size with unique weights in a separate collection and a driver toggling on Data Transfer for each by Topology (adding the weights to the same vertex from an identical mesh). And if you haven't already guessed it, that's exactly what we want to do for our weights.

Set your Chest and Mannequin to Small (or Medium/Large if you refit to Small first), create a **new collection inside your Chest collection**, name it something like **"REFERENCES"** (or use the one you already created for the NORMALS mesh). **Duplicate** your shirt mesh (so my Tank Top piece) with **Shift+D** and rename it to something like **"SMALL RUE WEIGHTS"** (or just small if you aren't including YAS weights - remember the Addon makes it extra easy for us to remove the rue weights for a non-rue size versus adding them after). Drag and drop it into your REFERENCES collection.

Hide your gear for a moment and select SMALL RUE WEIGHTS. Repeat the steps in Weighting to give this piece the correct weights - I'd recommend either limiting it to only the breast area or excluding the seams if your mesh has been split up to ensure you don't create a mismatch at the upper arm seam.

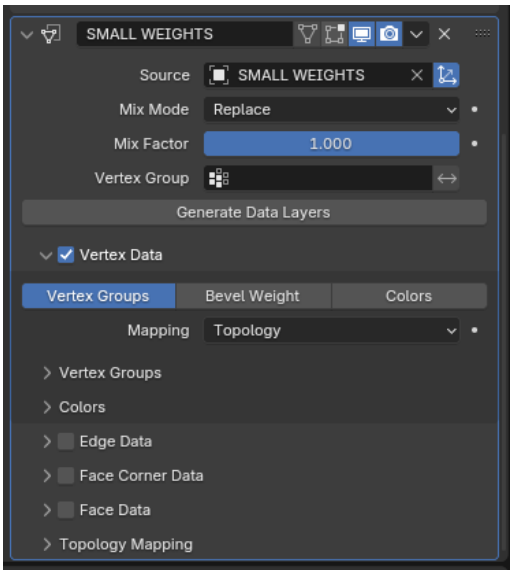
Now for the non-Rue meshes; duplicate your SMALL RUE WEIGHTS and name this new mesh to "**SMALL WEIGHTS**". Duplicate your regular shirt mesh again, and name this one "**RUE WEIGHTS**". You should now have your Shirt mesh in the base Chest collection, then a Rue, a Small, and a Small Rue tucked away in another collection as pictured here.

Select SMALL (non-Rue) and open the **Mesh Studio > Weights Menu**. Find the belly bones *iv_fukubu_phys* and *ya_fukubu_phys* in the list (enable the filter in the bottom right to show only the YAS bones) and hit remove selected. You'll get a message to say the weights were added to the parent bone and it'll vanish from your list.



Repeat this step with your non-Rue Medium/Large shirt as well, which for me is my base Chest collection 1.0 piece.

Once this is done, hide your REFERENCES collection and return to your Chest collection's top. Add a **Data Transfer modifier** and rename it to something like "SMALL WEIGHTS" (you'll end up with several Data Transfer modifiers here, so it's best to keep them identifiable) and select your **SMALL WEIGHTS** mesh as the **Source**. Check **Vertex Data**, **Vertex Groups** and set the **Mapping** to **Topology**.



Duplicate this modifier twice - one called "SMALL RUE WEIGHTS" with SMALL RUE as the Source, the other "RUE WEIGHTS" with RUE WEIGHTS as the Source. Don't apply any of them

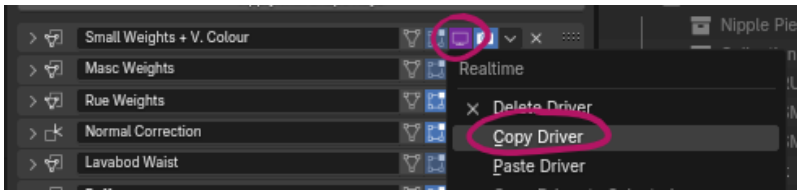
And one last step - on one of the Rue weight modifiers, hit Generate Data Layers to add the fukubu bones back to the base shirt's Vertex Group list - it can't add them to the mesh if they aren't already in the list!

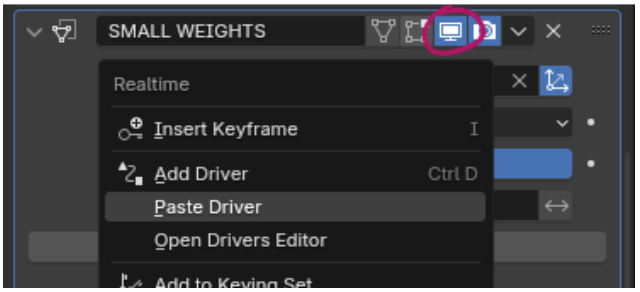
Drivers

To have these modifiers toggle with the size changes we want to employ the same process used to link shape keys - Drivers!

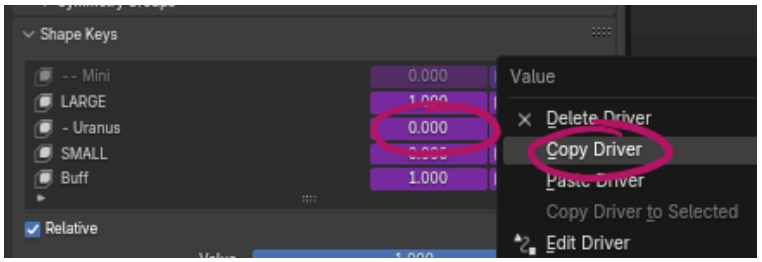
Single Variable Drivers

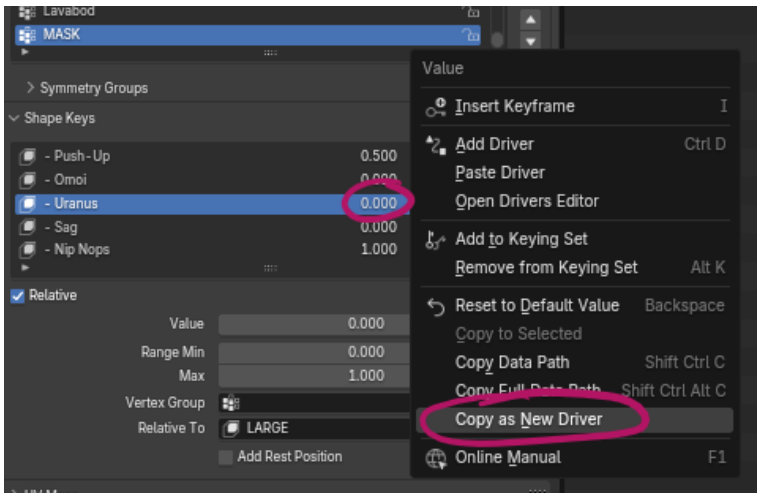
If you're adding Small weights but not including YAS/Rue, we have a little shortcut. Since the 0.1 Torso mesh already has a driver that triggers exactly how we want it, we can simply borrow that one!





Go to the **Modifiers Tab** of your **0.1 Torso**, locate a **Data Transfer** for a Small size, **right click** the purple monitor icon and select **Copy Driver**. Now return to your gear mesh and your newly created SMALL WEIGHTS, **right click** your monitor icon and click **Paste Driver**.

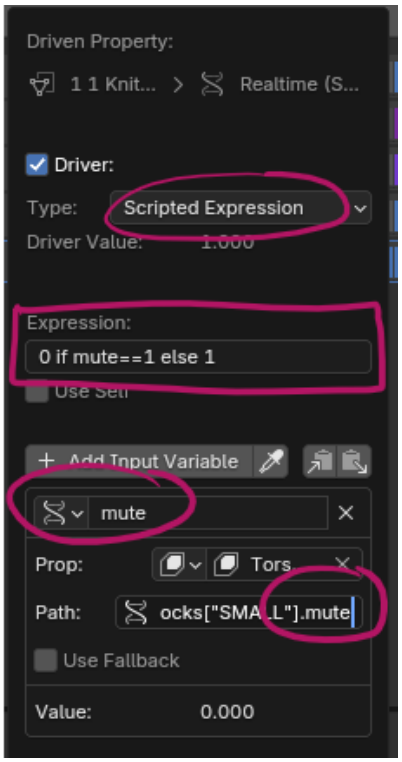




If you're looking at adding a driver for an option that doesn't have an existing 0.1 Torso driver to steal (say if Uranus has different weights or a different Mask to Medium/Large), you can also look at your gear's Shape Key list and copy one from there. Keep in mind that quite a few of these Shape Keys are *not* muted (the checkbox) between sizes, only the *value* changes. Be sure to copy your driver from the value instead.

Additionally, you can find the *non-driver* value you want your modifier to change with (0.1 Torso's Uranus shape key for example), **right click**, then select **Copy as New Driver** near the bottom of the menu.

Toggle your sizes on and off with the **Devkit Overview**, and you should see the monitor of your Modifiers toggle on and off in tandem.



Driver is reversed?

Somewhat confusingly, the logic for mute buttons is reversed; 0 is 'on'/visible and *1* is 'off'/hidden/muted. This means if you end up copying a driver from a mute, it may be reversed. If you end up with a driver toggling backwards from how you intend, the fix is fairly simple!

Right click > Edit Driver and change **Type** to **Scripted Expression**, then enter "**1 if mute==0 else 0**" into the **Expression** field. Translated: '*Modifier* on if the referenced key is *visible*, else *Modifier* off'.

Conversely, if you *want* the driver to be reversed, enter "**1 if mute==1 else 0**". Swap either *just* the middle number, or both the first and last; either method will have the same effect.

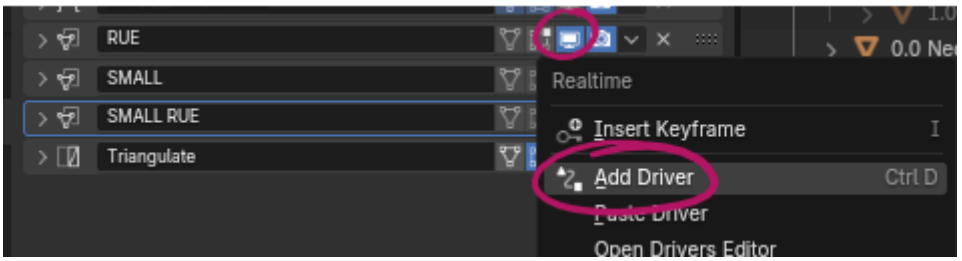
If it gives an error saying **Invalid Python expression**, rename your variable to "**mute**" instead of "key_mute" and check the **Path:** also ends in *.mute*.

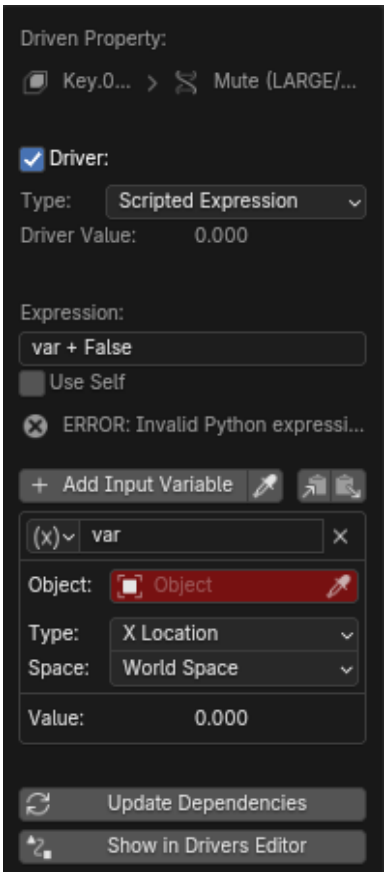
Multiple Variable Drivers

You may encounter scenarios where you need two drivers on one mesh toggling with different variables; for example, my tank top mesh needs drivers to toggle between Medium/Large and Small weights, but also Rue weights.

If you're very careful about where you weight or your top is really close fitting at the underbust, you may have no overlap between the *mune* (breast) weights and *fukubu* (Rue belly) and can limit your Data Transfer Modifiers to only the chest/belly area like the body meshes do, but I find with gear that sits more loosely at the underbust this is very unlikely. To manage this, what I do instead is have my base shirt mesh with Medium/Large weights, a Medium/Large Rue reference mesh, a Small non-Rue reference mesh, and a Small Rue reference mesh.

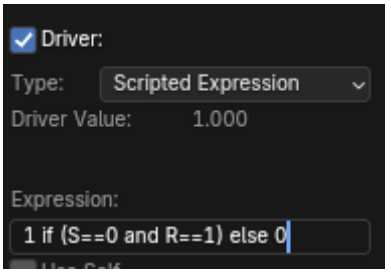
This then means I need to edit my Driver script to toggle with two variables instead of one. I covered this process in Adding Corrective Shape Keys and Drivers if you read through that part, but in case you didn't, here's the rundown looking more specifically at Small/Rue toggles.



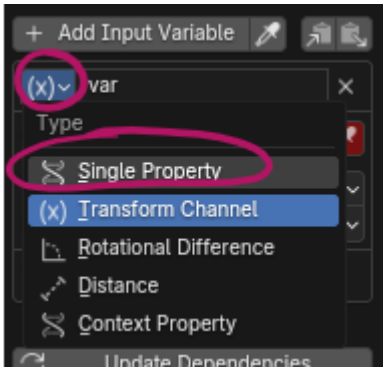


Right click the **monitor button** on one of your Modifiers and select **Add Driver**. I'll do non-Small Rue first. This will bring up a small Driver Editor window, which you can now re-open at any time with **Right Click > Edit Driver**.

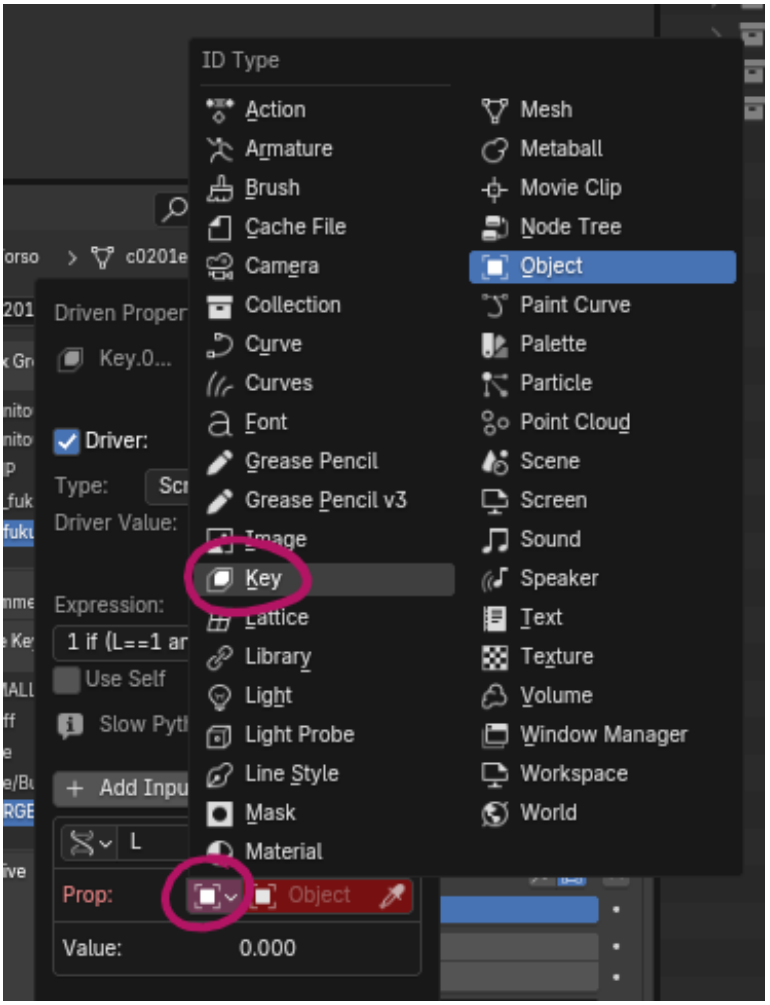
First, ensure **Driver Type** is **Scripted Expression**. In the **Expression field** below, type **"1 if (S==0 and R==1) else 0"**. Essentially *'ON if SMALL is off and Rue is on, else OFF'*. Ignore the error below for the moment: it's because we haven't set "S" and "R" as our variables yet.

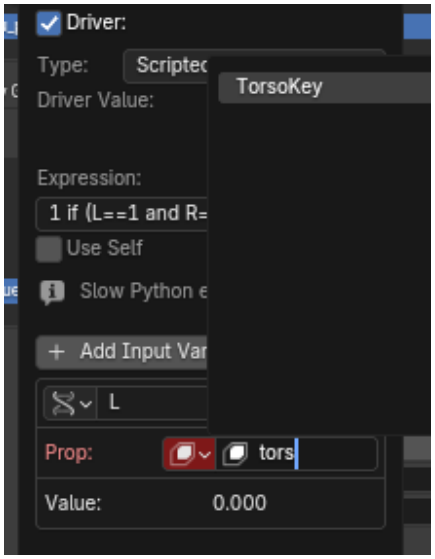


Next, set the existing **Variable** below to use **Single Property** with the dropdown before the name. **Rename it to S** (it doesn't have to be a single letter or anything, I just like to keep it short and sweet).



This will change the fields below to look for a **Prop**. Select the **dropdown** to swap this to a **Key** instead of an Object.



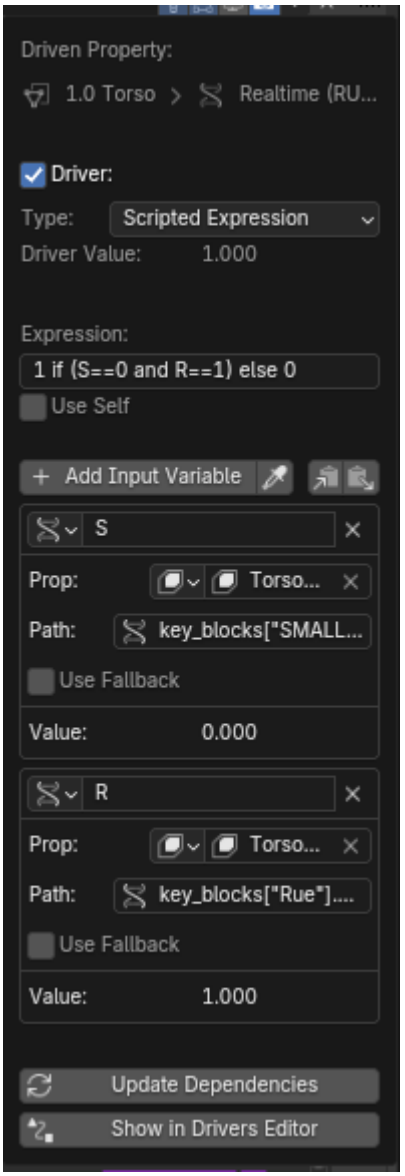


Once this is looking for a Key, search in the field next to it for **TorsoKey**.

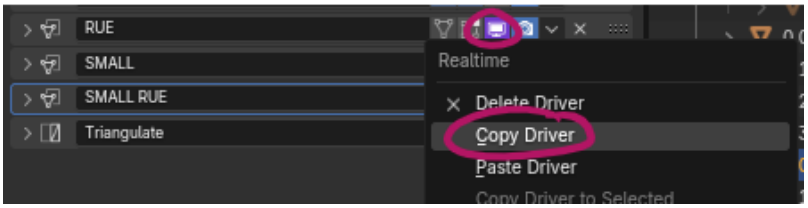
In the new **Path** field that appears, type or paste in "**key_blocks["SMALL"].value**". If you have the Small shape enabled in your overview now, you should be able to see the value below change to 1 to show you the value the path is referencing.

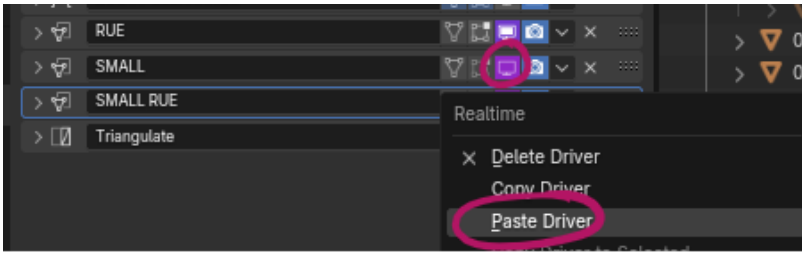
Now we want to hit **Add Input Variable** and repeat for the Rue key - just name it "**R**" and set the path as "**key_blocks["Rue"].value**" instead.

Once you're finished, your Driver Property menu should look like this! >



Now I simply need to **Copy** and **Paste** this fully set up driver to my other Data Transfers...





...and then **right click** > **Edit Driver** to change the Expression slightly for each; "**1 if (S==1 and R==0) else 0**" for Small non-Rue and "**1 if (S==1 and R==1) else 0**" for Small Rue.

Preparing for Export

Shape Keys

Vanilla (shp_) Shape Keys

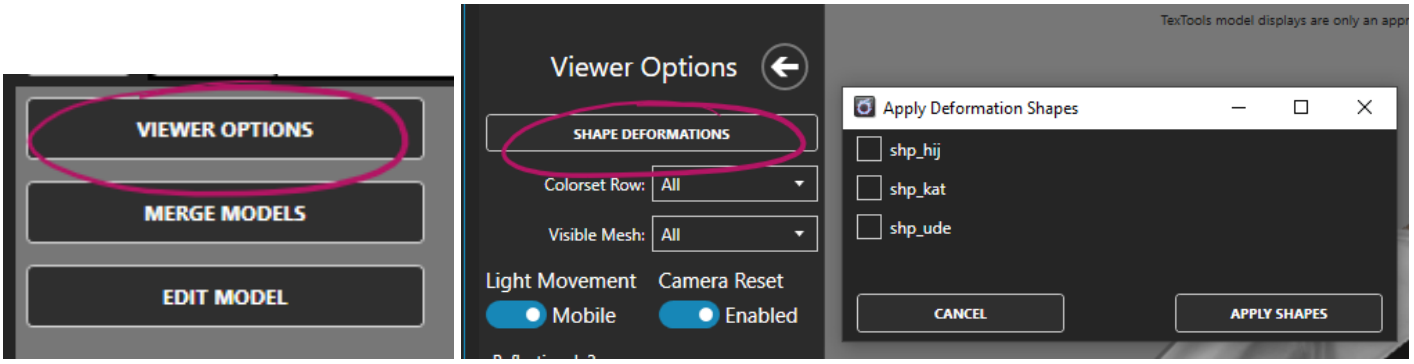
Sometimes called Shape Data, these are shape keys that XIV's engine uses for the purpose of tucking gear in for compatibility with other pieces. These are all named "shp_***", with the latter three letters denoting what area to squish with. If you remember any of the shape keys you deleted after importing your model, this was them!

Here is a full list, but the main ones you'll want to pay attention to are as follows:

- **shp_kat** - Long Gloves.
- **shp_ude** - Mid Gloves.
- **shp_hij** - Short Gloves.
- **shp_kos** - Waistband.
- **shp_mom** - Long Boot.
- **shp_hiz** - Mid Boot.
- **shp_sne** - Short Boot.

The new Addon saves us a bunch of time and hassle with making shape keys. Blender cannot apply modifiers to a mesh with shape keys and even the scripts that allow you to do so struggle with certain modifiers (namely Shrinkwrap), so we used to have to export all our sizes (or apply all our modifiers) and then make the shape data for all of them individually, using Surface Deforms to duplicate the squish from one piece onto another. But, the Addon can now automate this process as part of its export!

To begin with, we should check which pieces actually need Shape Data. I will often search the vanilla gear up in TexTools, click Viewer Options in the top left, then hit the Shape Deformations button to check which already exist on the gear.



You can also check for clipping by enabling the **Gear** collection inside the **Resources** collection in the devkit, which contains the Bozjan Wristwraps/Armguards (**Long Gloves (Buff)**), Molybdenum Armguards (**Long Gloves**), High Allagan Gloves of Healing (**Medium Gloves**), Expeditioner's Thighboots (**Thighboots**), AR-Caean Work Boots (**Medium Boots**)and Rebel Boots (**Medium Boots 2**). Alternatively, if you've exported any WIPs to check in game, you can equip various gloves to see how badly things clip.

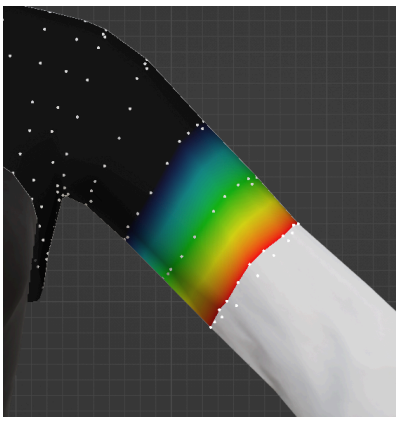
My go-tos for testing these are:

• **Ravel Keeper's/Pink Beryl Halfgloves** and **Bozjan Wristwraps/Armguards** for **Long Gloves**, as the Ravel Keeper's have a small band that shows exactly where the sleeve seam needs to align, and the Bozjan gloves are the lowest sitting long gloves I've found. (Don't forget Buff arms will clip with vanilla long gloves regardless of shape data, so be sure to test with upscaled gloves)
• **High Allagan Gloves of Healing** for **Mid Gloves**, as they also have a band for the mid seam.
• **Radiant's Wristgloves of Striking/Aiming** for **Short Gloves**, as the sleeve will stick out from under the cuff if it needs a shape key.
• **Expeditioner's Thighboots** or **No.2 Type B Boots** for **Long Boots**, as they are form-fitting and simple, and already upscaled to YAB legs. (Much like the long gloves, vanilla boots will clip no matter what you do - always test with upscaled thighboots)
• **Rebel/Urban Boots** for **Mid Boots**, as the top of these are one of the snuggest to the YAB calf I've found.
• **Boarskin Survival Boots** for **Short Boots**, mainly just because I used these a fair amount when I first started modding and found a lot of trousers clipped this way.

I have all of these **favourited in Glamourer** for easy access so they're right up the top of my lists. Unsurprisingly, these baggy sleeves need all three pieces of arm Shape Data.

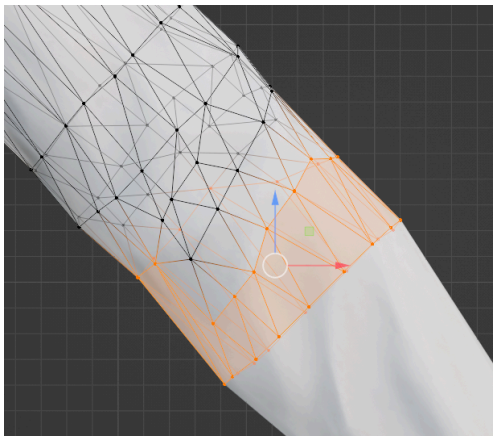
Adding them is as simple as creating a new shape key and giving it the appropriate name!

Add a new **Vertex Group** and select the seam. For even meshes like the upper arm here I will **Grow Selection** with **Ctrl+Num +** and use **Ctrl+X** to **set the weight to 0.5**, then **Ctrl+Num -** to **Shrink Selection** again and **Ctrl+X** to **set the weight to 1.0**. It should look something like this; a softer blend.

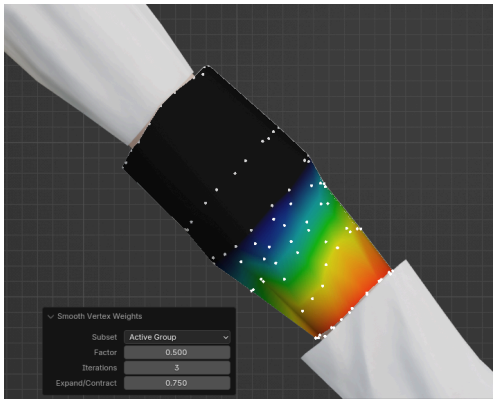


Add a **Shrinkwrap** limited to this group. Instead of targeting the Mannequin however, we want to use the **Chest** meshes as the **Source** - since we don't want the sleeve to exceed that edge anyway. The **Snap Mode** should stay as **On Surface**. If you have skin underneath, you'll want to set the **Offset** to at least **0.001m**, perhaps higher if the sleeve is still fairly low poly.

Make sure your **Chest** is set to the **same size** as your basis - your first shape key - else this shape key may also create some unintended shape edits. Rename the Shrinkwrap as "**shp_kat**" and **Apply as Shape Key**.

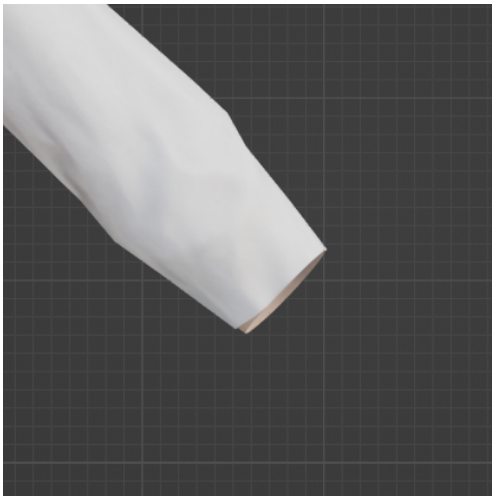


If your mesh's geometry isn't very even like the elbow part here, the Grow Selection will make it lopsided. So, instead, we can add weight to the seam and then make use of **Weights > Smooth**'s **Expand** setting. Select as far as you want it to possibly blend (I ended up selecting everything except the upper arm seam), up your **Expand** to somewhere between 0.750 and 1.000, and then up the **Iterations** enough to make it a smooth blend. Don't forget to check it's set to smooth only the **Active Group**! It can help to add the **Shrinkwrap** pre-emptively to help visualise how it'll squish with those weights.



Rename this one to **shp_ude** and **Apply as Shape Key**.

If you end up with the edge of the seam not quite aligning with the next section of the arm mesh, you can add another **Shrinkwrap** and target the next arm part to line it up and use **Mesh Studio > Mesh Menu > Modifiers** to blend it into your shp_ude Shape Key as we did with editing the different sizes.



If you're including them, check how the shape data applies with Buff arms/Skull Crusher legs, *especially* if you made that shape key with a Shrinkwrap rather than the Mesh Studio's Surface-Deform-esque reshaping. If it clips into the skin at all you probably want to tweak it manually with **Shape Key Edit mode** rather than **Blend from Shape**, as if you apply one with the Buff key enabled, it will attempt to blend the Buff key into it as well - which you don't want for non-Buff sizes.

You can tweak this shape as many times as you like with additional **Shrinkwraps** and use **Vertex > Blend from Shape** or toggling **Shape Key Edit mode**. You can check the fit with the gear in the Devkit inside **Resources > Gear**, too!

As always, repeat with as many of your meshes that require it!

Penumbra (shpx_) Shape Keys

Penumbra has recently added a feature that allows you to toggle custom shape keys on a mesh. YAB has a few of these baked into it behind the scenes, but they work in a few different ways. Additionally, you can make your own! I personally like to add some for booty clearance toggles to stop longer tops floating a mile away from Small Butts or Yanilla legs.

Soft Butt and Alt Hips

The Soft Butt and Alt Hips options that you may have added when you did size conversions are toggled solely in the YAB or Rue modpacks and will apply across all models you equip due to the settings already existing in the body's modpack. You only need to add these shapes while you upscale, ensure they're deforming to your satisfaction and be sure to export later with Shape Keys enabled, everything else is done for you!

These are named *shpx_yam_softbutt* for Soft Butt, *shpx_yab_hip* for Alt Hips on YAB (adds hip dips) and *shpx_rue_hip* for Alt Hips on Rue (removes hip dips), though you don't need to worry about the names - if you want to add them, simply use the **Mesh Studio > Shapes Menu** with **Method: Legs** options.

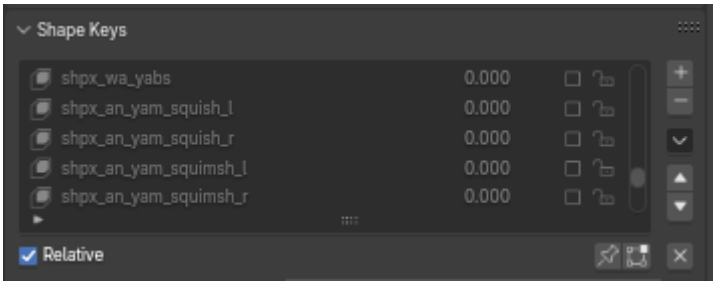
Seams

The next easiest to implement are the seam keys; Shape Keys for the waist, wrists and ankles that will allow a model to differ from the vanilla seam if and *only if* both models either side of the seam have keys with the same name - meaning no need to worry about compatibility issues! These are entirely coded into Penumbra itself, so aside from making the keys on your model, everything else will work out of the box.

These are called *shpx_wa_yab* (YAB regular waist) and *shpx_wa_yabs* (YAB Buff waist), *shpx_wr_yab* (wrists) and *shpx_an_yab* (ankles) These are the keys you may have created earlier if you used the **Mesh Studio**'s **Shapes Menu** via the **Method: Seams** option. You mainly want to add these if your mesh fits tight enough to look obviously blocky at the seam; here's a comparison of the Calfskin Rider's Bottoms without and then with *shpx_wa_yab*! Schmove!

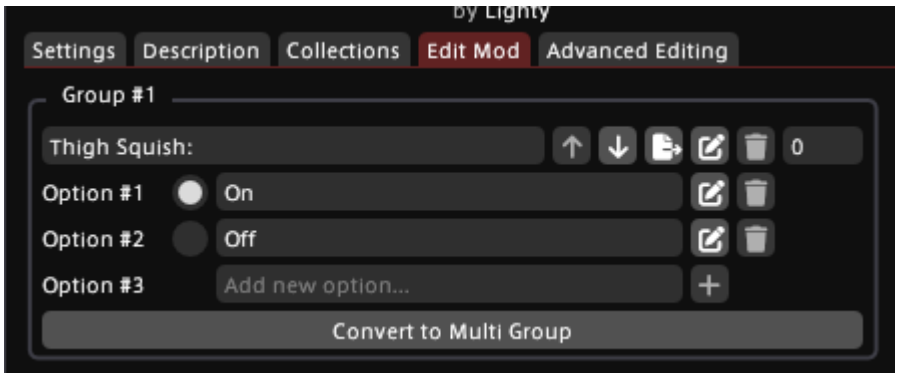
Squish and Squimsh

The other shape keys you have baked into your Devkit meshes are *shpx_an_yam_squish* and *_squimsh* keys for each left and right leg - these are the same shape as the Squish and Squimsh options available in the Devkit Legs overview, simply togglable in game - and separate from one another, too, so you can squish only on one side!



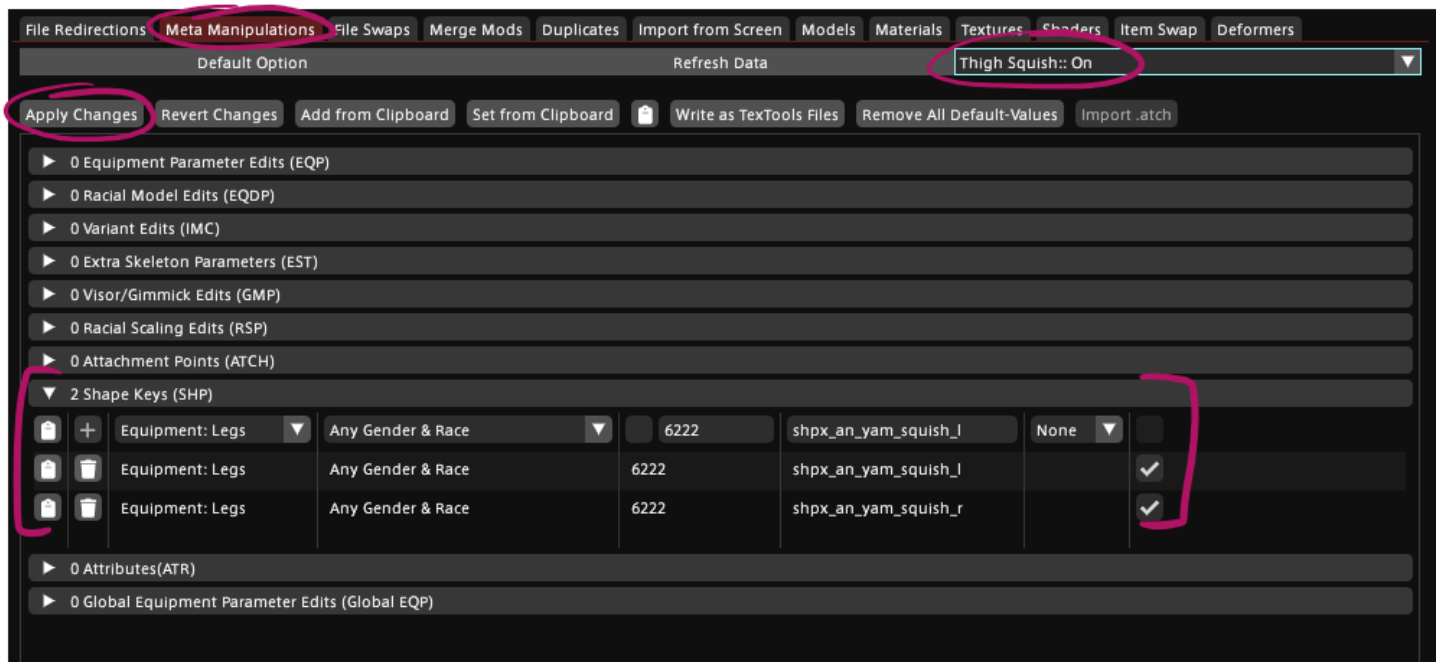
The use case for these would be if you have gear with stockings and you want to give people the choice between thigh squish or no thigh squish, or if you have thigh squish and want the stockings to be togglable.

To implement these, I would use **Mesh Studio > Mesh Menu >** [Method: Selection] to transfer the shape keys to my gear if I want the squish to be togglable (if not, I will have already upscaled it to the squish or squimsh shape).

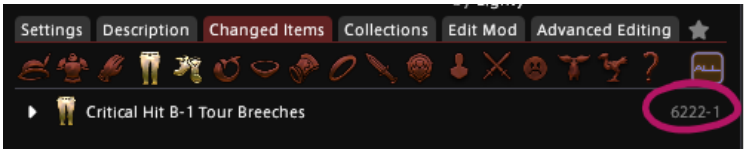


In your ModPack in Penumbra (if you don't have one already you can create one now, hit Export Modpack when you're done and then tell the Addon Modpacker to update that one to maintain the setup you do here), go into the **Edit Mod** tab and **add a new group** with the options you want to include.

Next, open up **Advanced Editing** and **select the option** you want to edit in the top right dropdown. Go into the **Meta Manipulations** tab and open up the **Shape Keys (SHP) Menu** - third from the bottom.



Here you want to select **Equipment: Legs**, **Any Gender & Race** (saves having to add it for every female race individually). **Uncheck the box** - this controls whether it applies to every gear piece in the slot or not - and enter the **Primary ID** for the gear you want the shape key to apply for (you can find this in the Changed Items tab if you've forgotten, ignore the number after the dash, that's the variant ID).



Then, type in or copy & paste the name of the **shape key** you want to activate and click the **+** to add the manipulation. If you're toggling both legs to squish at the same time, you'll want to add both _l and _r. Don't forget to hit Apply Changes up top!

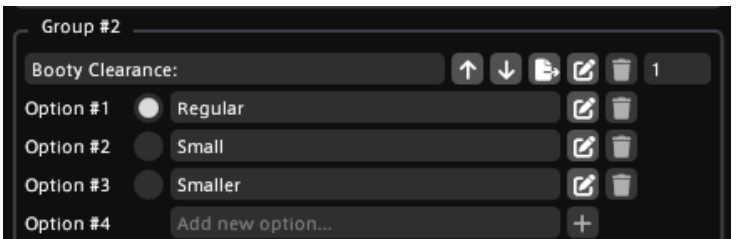
If you wish to add these to toggle with a visible part, you will unfortunately have to set it up as a **Combining Group** rather than an IMC Mask Group. Alternatively if you aren't feeling up to setting up all of that, you can just use an IMC Mask Group and a separate Single Group to toggle the squish and leave clicking that to the end user. :)

This can also be done in the Modpacker inside the Addon if you want to leave this until later.

Custom Shape Keys

And here's the type I use the most! I make my own shape keys to allow users to customise their booty clearance for their tops. All you need to do is **make a shape key** in Blender and edit it as you like, name it something starting with "shpx_" and a max of 30 characters, then add an option to toggle it into your ModPack, the same process as above.

I leave regular clearance for full size YAB/Rue butts as the base size, then add a shape key called

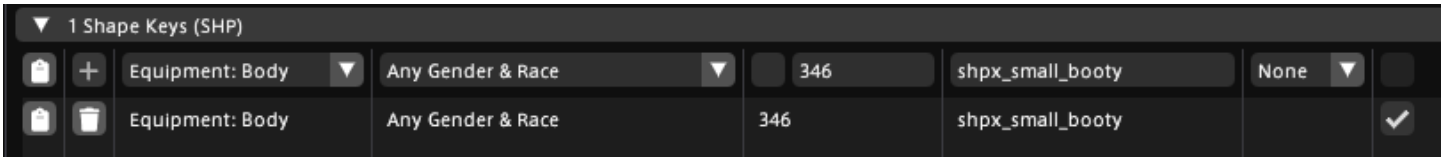


shpx_small_booty for Small/Soft Butts and another called *shpx_yanilla* for, well, Yanilla. In **Penumbra**, I go into the **Edit Mod** tab and make a **Single Group** called “Booty Clearance:” with the options being named “Regular”, “Small” and “Smaller”.

Doesn’t have to be major!

Once I’ve **added a shape key** with the name “*shpx_small_booty*” to all meshes that require it in Blender (if you held off on splitting variant parts away from the main model you won't have to do as many), set the **value to 1** on all of them, **select that as the active key** and **enable Shape Key Edit Mode** on each mesh. Now I can grab all of the meshes and enter **Edit Mode on all of them** at once and alter the booty clearance of my top a little - it can be useful to check how much Small Butt changes on the Mannequin before you move anything as a little reminder. Once you’re happy with it, **set the values** of each to **0**.

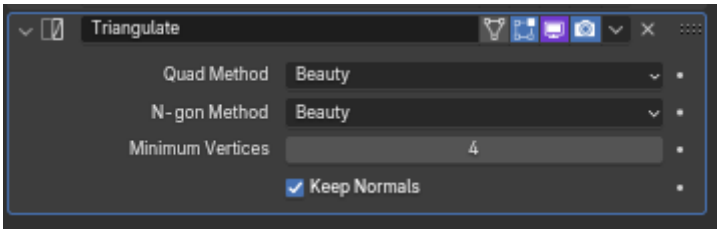
To add *shpx_yanilla*, I simply repeat the process but edit the booty clearance a further than before, including narrowing the hips.



Back in Penumbra, I use **Advanced Editing > Meta Manipulations > Shape Keys (SHP)** (as shown above in Squish and Squimsh) to add my shapes into the options I created. Don’t forget to select your option in the top right dropdown and hit Apply Changes once you’re done - I forget those all the time.

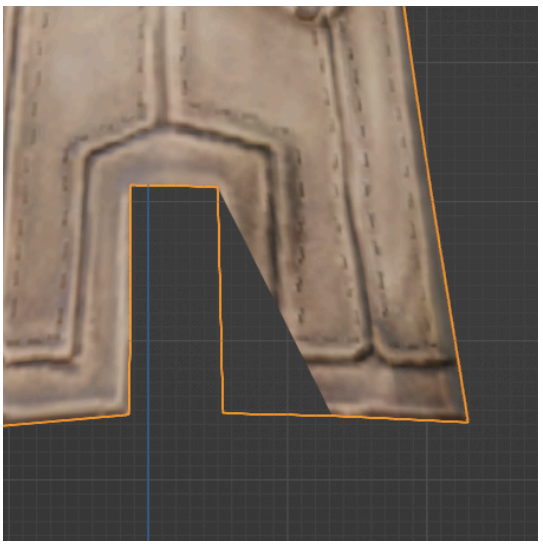
Remember, this can also be done in the Modpacker inside the Addon if you want to leave it until later.

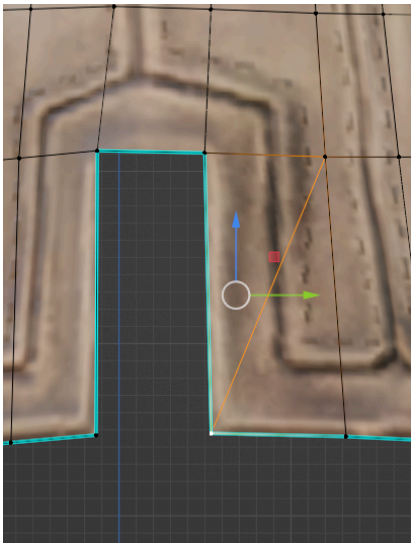
Triangulation



One more tiny step we need to do before we export is, add a **Triangulate modifier** to all of your meshes, if you haven't already. Since the engine requires your meshes to be triangulated to read them, there's an extra check in the **Exporter** to ensure they're triangulated. So triangulate! I'd recommend changing the **Quad Method** to **Beauty** and checking **Keep Normals**.

Sometimes when you add a Triangulate modifier to a quadded mesh that has backfaces with merged seams it can choose to triangulate in a direction that will cause the textures to disappear at a corner due to all surrounding verts of a to-be tri being merged. To fix this, all you need to do is select the vert all by its lonesome and the one opposite it and press J to Join, manually triangulating the face in a manner that will maintain the textures. Be sure to check all areas with backfaces like this, including collars and lapels!





Separating Variant Parts

Now you've set all of your modifiers and shape keys (and tested them, I hope), you're free to detach any parts you want to have as toggles. Belts, bags, pauldrons, buttons - any part that you would like to toggle uniquely will need to be a separate mesh, as they will all need their own attribute.

This is probably the easiest part to come back and tweak later if you decide you want something else to toggle too.

Setting Attributes and Materials

As I mentioned way back in [What's With the Names?], each separate mesh in your scene needs a **unique Group and Part number**. The first number should be the same for any meshes with the same texture - so here, 0 for the body and 1 for the gear. Then every piece that shares textures just needs a unique second number to allow the game to process attributes.

You can name them here in the **Outliner**, or you can edit them with the **Mesh Studio's** first **Menu** which we're going to use to set our attributes.

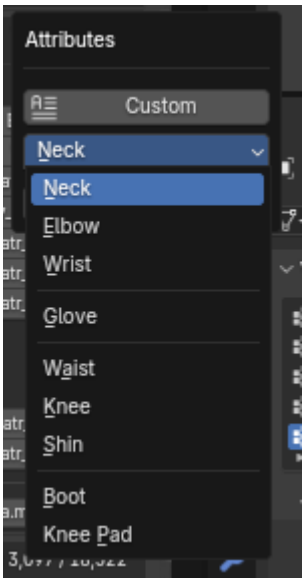
> Note:
>
> **The following applies only to MDL export, which is necessary to use the Modpacker within the Addon**. If you intend to use FBX exporting and use TexTools to pack your mod, you can skip the rest of this section

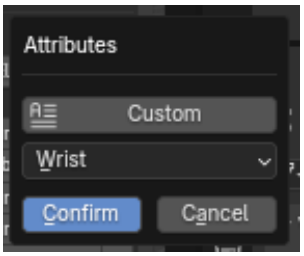
This is the menu where you can set all of your attributes!

Any parts that erroneously have the same numbers will be marked with a little exclamation mark like this:

You can use the up and down arrows before the mesh name to move an object from one mesh to another (onto the other "texture", so to speak) or the left and right arrows either side of the part number to make the numbers unique or reorder the list.

Clicking the **+** button will bring up a small menu to add attributes, including a dropdown that lists all of your default attributes. These are what you want to tag body parts with for them to hide when other gear overlaps with your model in game - for example, anything tagged as a wrist mesh will be hidden with any gloves that cover the forearm. The body meshes should already be set up for you, so you'll likely only need to add them for your gear pieces. The gear I'm looking at right now is the Machinist's Shirt, so I want to **tag** the cuff of the sleeve as **Wrist**.



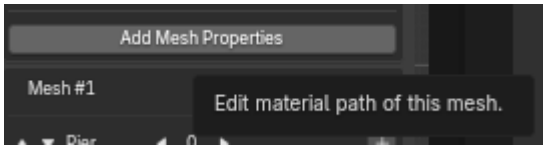


Click **Confirm** once you've selected the attribute you want to add.

If you want to add any other kinds of attributes, you'll want to click on **Custom**, which will bring up a field for you to type your own. The main thing you'll want to use this for is **variant parts** - as you can see from my list, I have some parts tagged as *atr_tv_a*.

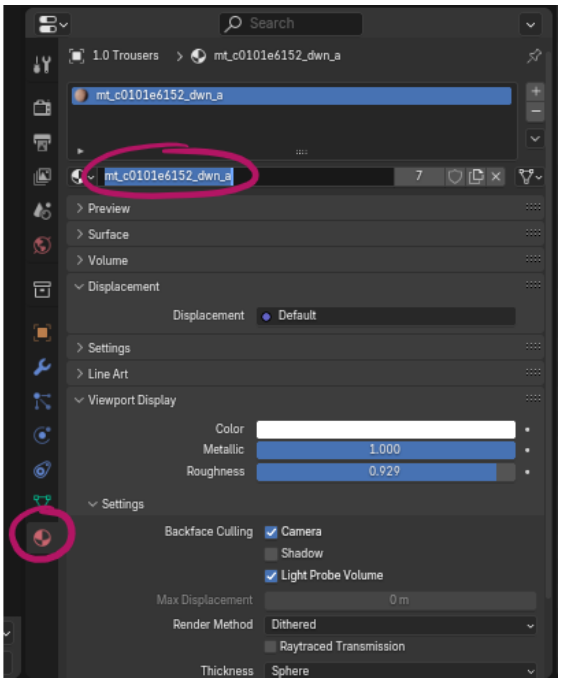
"Atr_" is simply denoting that it's an **attr**ibute (like how the shape keys start with "shp_"), "tv" indicates what the gear slot is (tv for top, dv for bottoms, gv for gloves, sv for shoes, mv for hats), and finally "_a" is which variant part tag it has. You can **add a max of 10 toggles** per gear piece, **A through J**, but the same attribute tag can be put on as many meshes as you want. Some parts you'll see I have two on: I use this for things like belts with bags attached - the bag has both an attribute to hide just the bag, as well as one to hide with the belts, since they would be left floating oddly if only the belt is hidden.

Added the wrong attribute? Not a problem! Simply click on the attribute that's in the wrong place and it'll be removed.



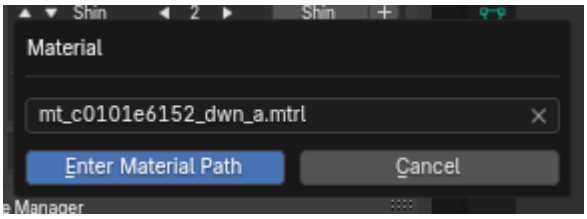
Under each mesh/group list you have a box where you can assign your material. The Bibo material for the body should be the first option in the list.

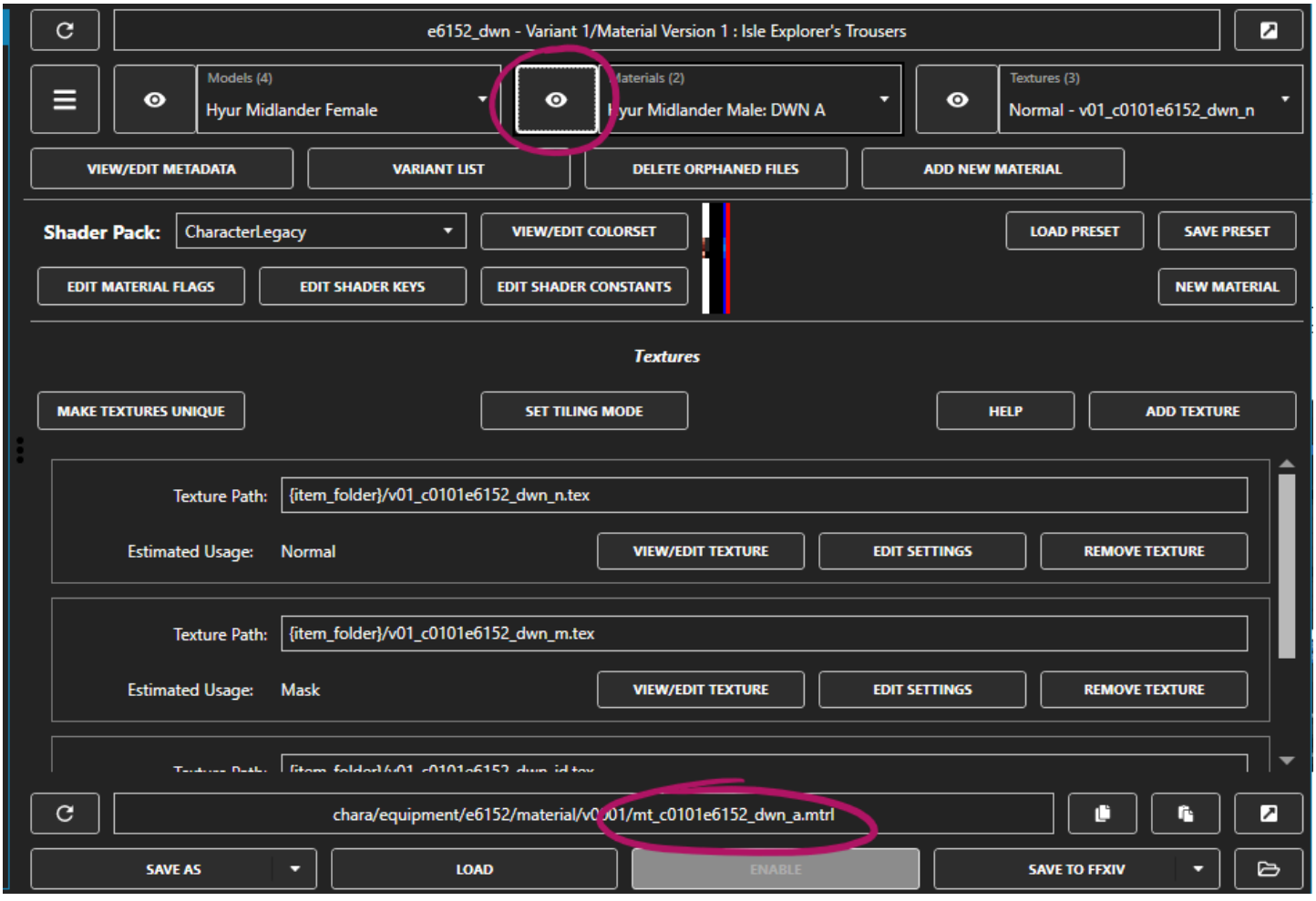
For your gear, however, you'll need to locate your .mtrl path. There are a few ways to do this.



If you imported an FBX and therefore a texture with it, you can open the **Material Properties Menu** (the bottom red icon) and **copy** the text from the field below the material list.

Click **Add Mesh Properties** for your gear's mesh and **paste** it in there and **type ".mtrl" at the end**, then click **Enter Material Path**.





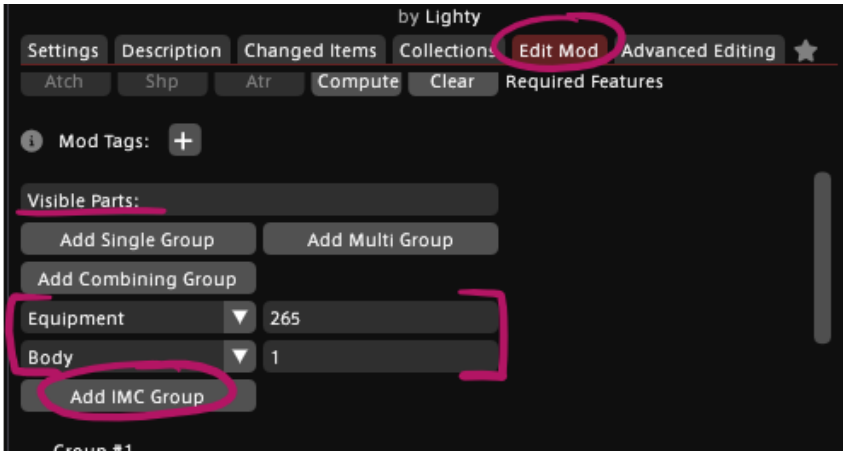
Alternatively, **locate the gear in TexTools** and click on the **Materials tab** - the middle eye icon. At the bottom, **copy** the text from "mt_" onwards. Return the Blender, click **Add Mesh Properties** for your gear's mesh and **paste** it in there. Click **Enter Material Path**.

IMC Mask Groups

While some gear does have vanilla variant parts to make different versions of the same model look slightly different (think the Shisui sets), the main reason you'd want to add them to your mesh is to allow users to toggle parts on and off.

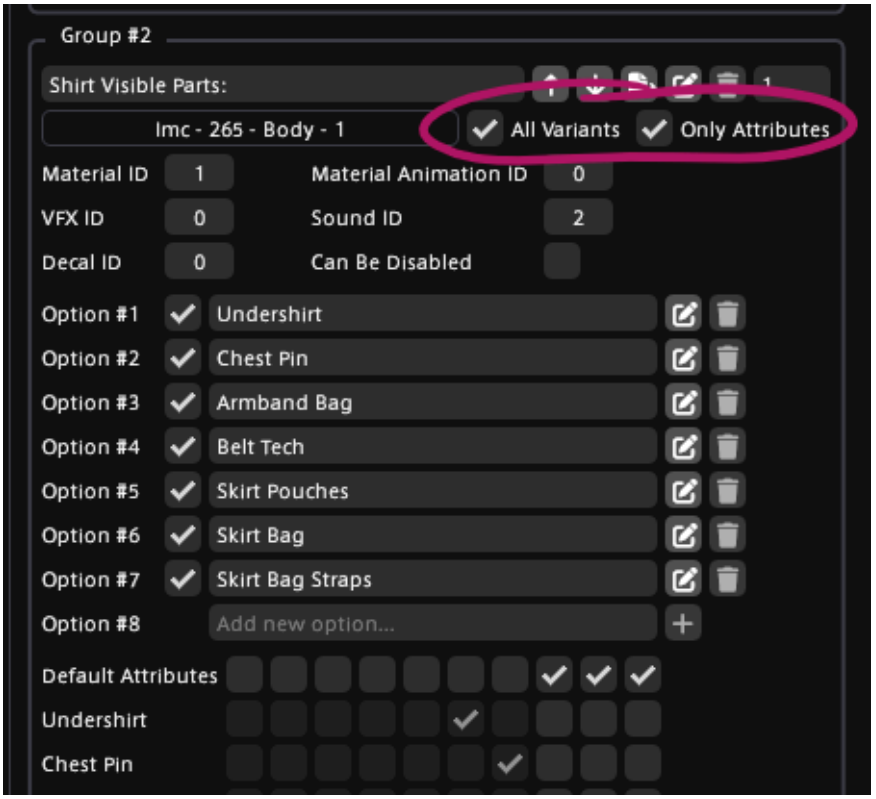
The way we typically handle this in Penumbra is with an **IMC Mask Group** - as you may have read in back in Squish and Squimsh the alternative is making a Combining Group to allow other options to toggle with the parts, such as shpx_ keys, but that's a fairly specific use case so I won't cover how to do that here. (IMC Mask groups can also be created in TexTools if you'd prefer)

IMC Mask groups are special in that, as a mask, they can affect one part of a data block without overwriting the others. We used to have to pack every individual combination of toggles as a separate option, so if you've ever seen a mod with a long list of "both A and B", " yes A but no B", "no A but yes B" "neither A or B" etc, that'll likely be because it predated the introduction of IMC Masks!



To set up one of these, go into **Penumbra**'s **Edit Mod tab** and scroll down a tad to where you can add a group. **Name it**, then enter the **Equipment ID** (can be found in the Changed Items tab if you forgot) next to the Equipment dropdown, **select the slot** the gearpiece occupies and then next to that enter the **variant ID** (or just leave it as 1, we have a button that can make it apply to all variants). Hit **Add IMC Group**.

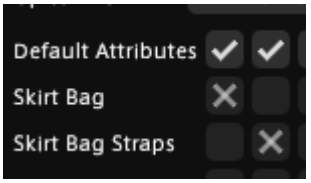
Now when you scroll down you will be greeted by a group that looks like this. Most of these settings you can comfortably ignore, but you'll likely want to check **All Variants** (to apply these toggles to all versions of the gear model) and **Only Attributes** (to stop it also applying the same textures/colours to every variant). If there are variant parts hidden in vanilla you'd like to leave untouched by default you can also check **Can Be Disabled**.



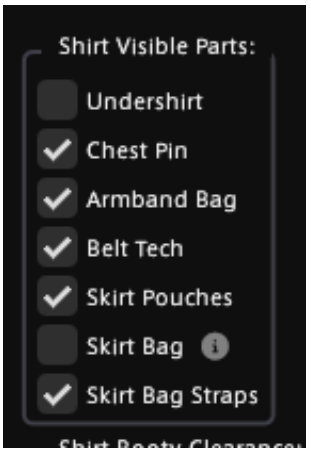
Below you are given your **Options** and **Checkboxes** just like a standard Multi Group. You'll want to make an option for every variant part tag you've added - type in a **name** and click the **plus**. The checkboxes control whether the option is on or off when a user first imports the modpack.

Below that is the most important part - your attribute table. The first row named **Default Attributes** (which will all be checked when you add a new group) is what is visible by default. These 10 checkboxes correlate to the A-J tags you added earlier - if you hover, the letter will pop up. They will also auto fill alphabetically - the first option will affect attribute A, the second B, et cetera. Mine are misordered here because I like to re-order the options (you can click and drag from the list above) to be vaguely head-to-toe order.

Below Default Attributes you'll be given a row for each option you entered above and how it interacts with the variant part tags you have; whether it toggles on (checkmark) or off (cross). When you create a new IMC mask group the default attributes will all be *on* and these options turn them *off*.



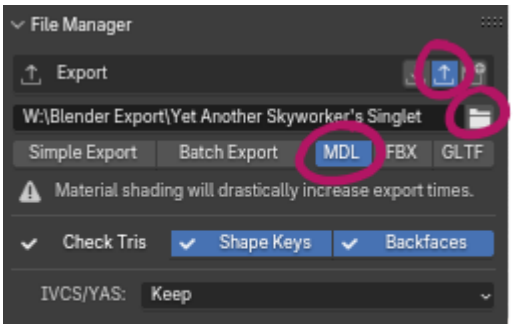
I personally like to invert these just so in the mod options a checkmark indicates the part *is* visible rather than check-to-hide because that just makes more sense to me, but it functions either way around.



You are able to toggle more than one part with each option should you choose to - one place you're likely to encounter this is with nail options on gloves. The devkit mashes are already tagged with the variant part tags, you just need to set up the IMC group like this!

Exporting

MDL Export



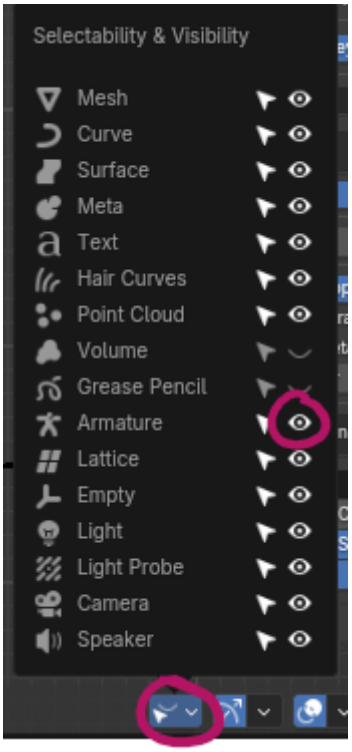
If you've upscaled to multiple sizes, this will probably be your favourite part of the Addon - it's certainly mine!

Open up your **File Manager** tab in the Overview and enter the second menu - **Export**.

Right below that, use the folder button to open a window explorer to **select your export directory**. It will default to the folder you selected way back in Blender Setup, but I personally make subfolders within that for each gearpiece - keep in mind they will export as just the name of the size ("Medium", "Buff - Large", "Rue - Buff - Small"), so this is the best way to keep them organised.

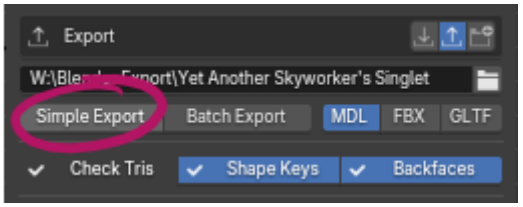
Next, you can select your export format. Select **MDL** if you intend to use the Modpacker (I'll briefly cover FBX export and modpacking in TexTools later just in case you don't).

After that you have the option to **Check for Triangulation** (forced for MDL export, since the game can't read un-triangulated meshes), for the exporter to **maintain your Shape Keys** (the shp_ and shpx_ ones, not the size ones with the drivers), and whether or not it should **generate Backfaces** for any areas you tagged with the BACKFACES vertex group while you worked. I would **leave all of these on** - if you don't have any shape keys or tagged backfaces, it'll simply skip that step.

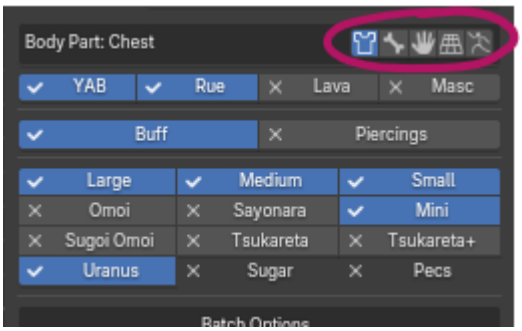


Then you have a dropdown where you can choose if you want to keep YAS weights - **Keep** will leave the weights you put onto your mesh (so if your mesh doesn't have any YAS weights, it will *not* add them for you), **No Genitalia** will... well, keep all YAS bones except the genitals (though if you stored Genitalia weights earlier there won't be any to remove), and **Remove** will get rid of all of your YAS bones and return the weight to the parent bone, much like the Mesh Studio's Weights menu - making it *incredibly easy* to include a non Yiggle variant as well.

If you only have the one size to export, you can stop here and just use **Simple Export** - it will export based on what is currently visible in your scene! Keep in mind that you may need to toggle the Armature to be visible in the viewport via the **Selectability and Visibility Menu** in your viewport header(/footer).



If you're exporting multiple sizes, the following settings are for the Batch Exporter.

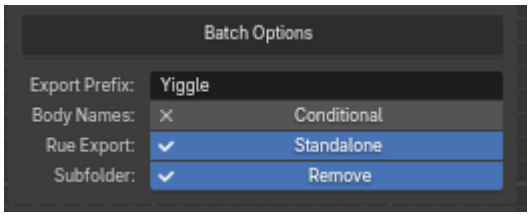


Here is where you can choose the body part you want to export; in order they are **Chest**, **Legs**, **Hands**, **Feet** and a **Body-suit**; both Chest and Legs in the same file.

Each of these will change the buttons below where you can select the bodies and sizes you wish to export, and the size buttons will even change based on which body/bodies you have selected (keep in mind if you have both YAB+ and/or Rue selected as well as Lava, **Lava Omoi/Teardrop/Cupcake are covered by Large/Medium/Small** on the top row - Omoi below Large specifically means YATiddy Omoi!).

The **Piercings** (for Chest), **Pubes** (for Legs) and **Clawsies** (Hands and Feet) buttons control whether those parts are visible or not. Do note that Piercings specifically refers to the nipple piercings, not the belly button one!

The Batch Exporter functions by going through and activating the shape keys for each in turn, so you'll only want to **select the sizes corresponding to the shape keys you created** earlier.



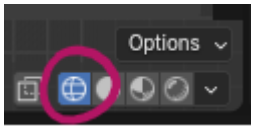
And lastly, your Batch Options!

Export Prefix is text placed before the size - when I export with YAS weights, I use this to add "Yiggle" to the name.

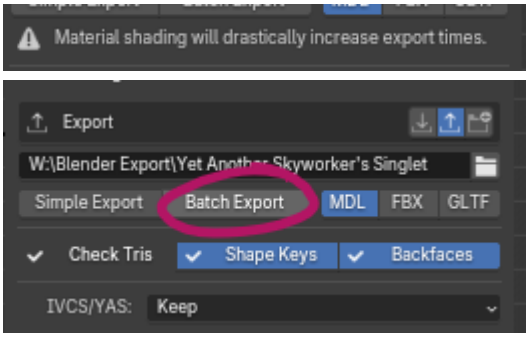
Body Names: Conditional will leave your main body as simply the size name (i.e. "Buff - Medium" for YAB) while others are labelled (say, "Rue - Buff - Medium"), while **Always** will also label your main body ("YAB - Buff - Medium") as well.

Rue Export: Standalone will add Rue as its own body, while **Variant** will limit it to only adding belly to other bodies (LaRue for example).

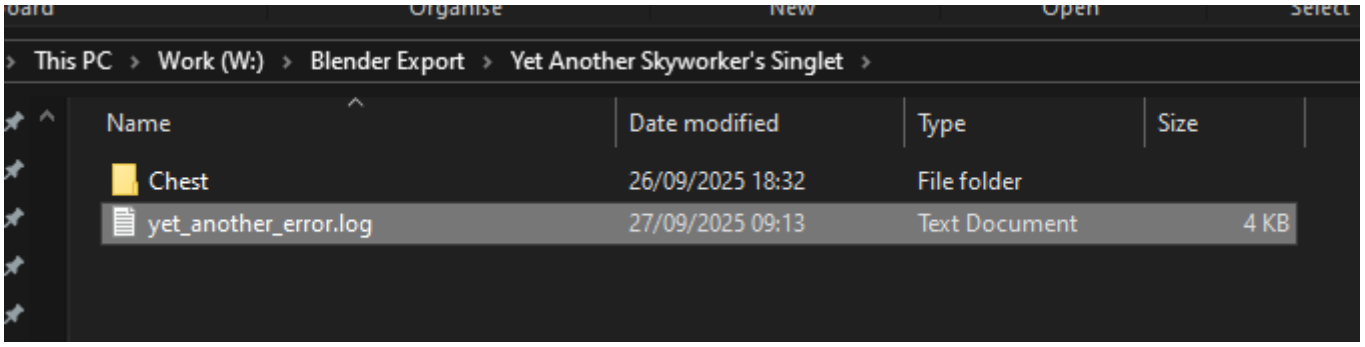
Subfolder, if active (I... think they're labelled backwards) will add a subfolder within your export directory (i.e. "Chest" or "Legs") - handy if you're working on a full set. I just leave it on because I'm forgetful. :D



Once you're happy with your options, change your **viewport shading** to **wireframe** via the header(/footer) to get rid of this little warning. Check all of your sub-key sizes (Uranus, Mini, Sugar) are disabled in the Overview (else the base size will export the same). Then, return to the top of the File Manager and hit **Batch Export**!



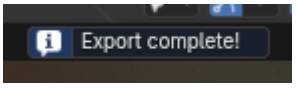
If the process fails, you will get a little warning at the bottom of your blender window. You're most likely to see "**Not Triangulated**" (easily solved by adding a **Triangulate Modifier** and trying again) or a note telling you to view the log in your export folder.

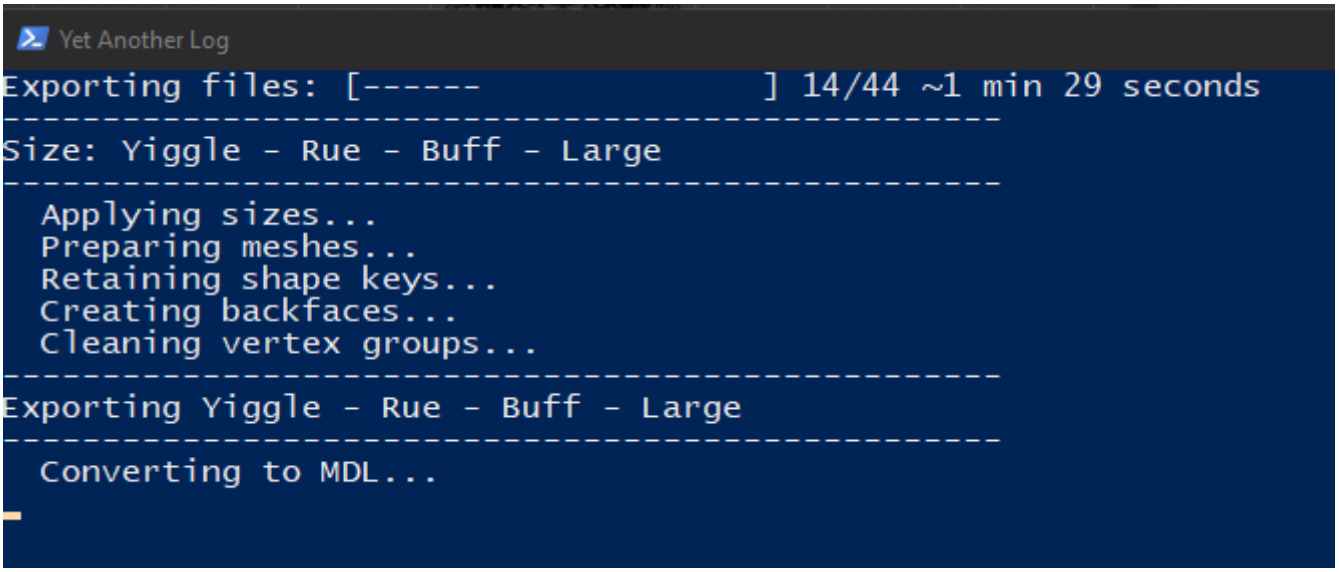



```
    raise XIVMeshIDError(f'{obj.name}: Submesh already exists as "{mesh_dict[group][part].name}".')
bl_ext.raw_githubusercontent_com.yet_another_addon_ffxiv.xiv.io.model.com.exceptions.XIVMeshIDError: 1.0 Tattoo: Submesh already exists as "1.0 Shirt".
LAST PROCESSED ITEM:
================================================
LOD0
```

Open up the error log and scroll down to the bottom to find out what the issue is. I've only encountered the error log so far for two meshes having the same part numbers, which you can fix by **renaming them** or using the buttons as described in Setting Attributes and Materials, but anything you can't decipher might be worth reporting to the discord!

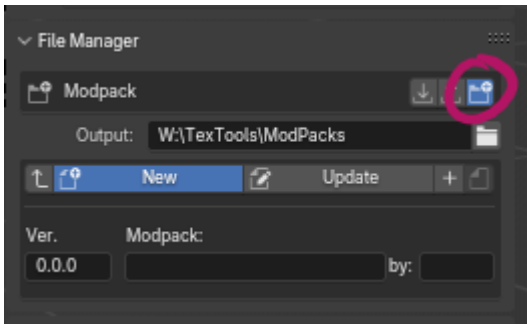
If it works, you will either see "Export complete!" at the bottom of the window, or you may see a Yet Another Log window appear as it processes.





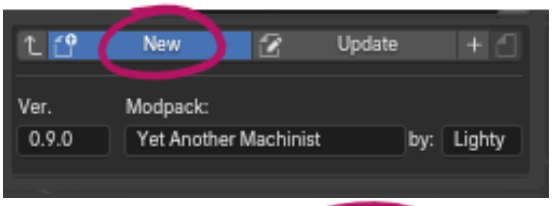
Modpacker

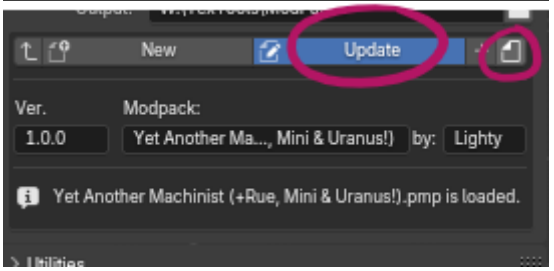
There's one more tab in the File Manager that we haven't covered (the last one you can read about here), and it's possibly the best one yet - the **Modpacker**, denoted by a little folder! Using this we can skip TexTools entirely (well... bar what we've already done with it).



This is what you'll see when you first tab into the Modpacker. Much like the Exporter tab, the very first option you have is a place to **choose your output directory** which will default to the folder you chose in the settings way back in the setup.

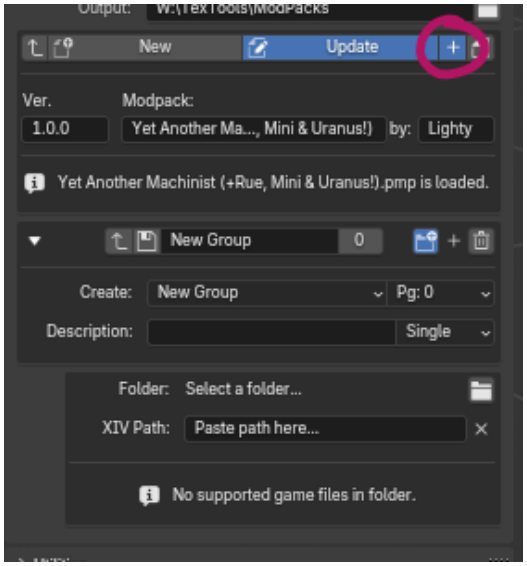
Next you can choose whether you're going to make a new modpack or update one you've already made. For **New**, simply fill out the three fields below; a **Version Number** (I like to use 0.9.0 for pre-release testing), a **Name** and an **Author**. For **Updating** a modpack, click the file button to **select your modpack**, which will give you a message to say the modpack is loaded - now you can change the name, if you'd like.



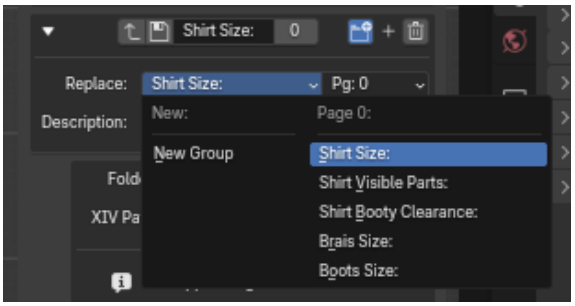


Whether creating new or updating, the process from here on is much the same. Additionally, once you create a modpack here, it will automatically convert to updating!

Size Selection

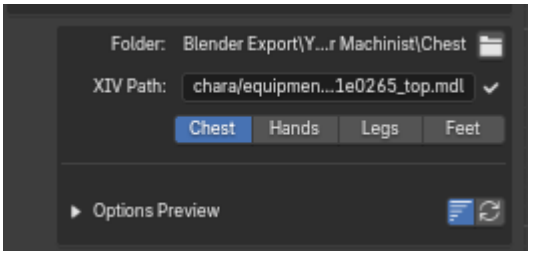


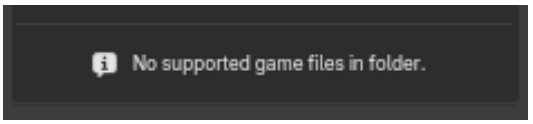
Once you have your modpack setup/loaded, click the **+** to **add a group**. This will cause an additional menu to show up below. The first thing you want to do is use the **Create:** field to select whether you are making a **New Group** or **select the group** you are Replacing. You can **Rename** it with the field above.



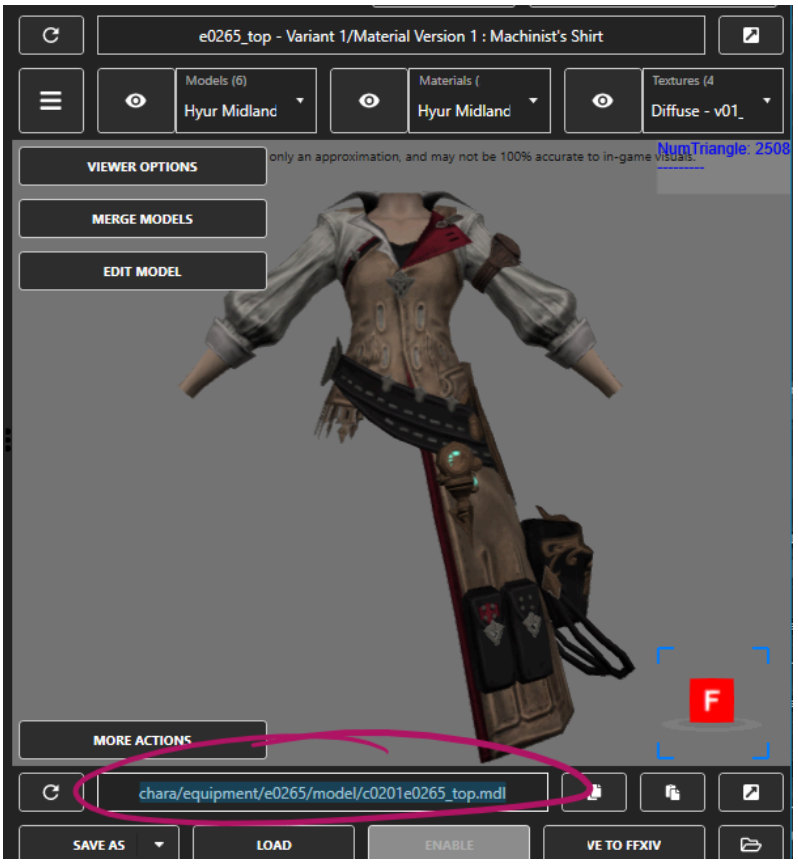
You can also **add a description** for your group if you want (the name alone can be descriptive enough). You have the ability to alter the **priority** with the number next to the renaming field, which controls which group's choices take priority if you were to have two groups that affect the same gearpiece, and also customise which **page** it appears on for people importing with TexTools - I leave both of these as 0 for most modpacks. And finally, next to the description is where you can change the type of group; **Single**, **Multi**, **Combi** or **Phyb**. Since we're packing our sizes first, we want **Single** - you can only have the one model active at a time!

The next field you have is for you to select the **Folder** where your MDLs have exported. Be sure to pick the one with the .mdl files, or you'll get a "No supported game files in folder" warning!





Once it detects a valid folder, an Options Preview will appear below. You can open this rollout to check it's detected all of the sizes you exported. The two buttons are for **Yet Another Sort** (enabling default YAB size order, else it will be sorted alphabetically) and for **Refreshing** the list if you export some more - note that this only refreshes the Options Preview list, it's not necessary for packing the mod.

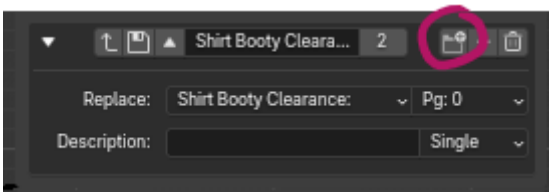


Then below that, you'll need to enter the XIV path for the model you're intending to replace. It's structured in this format: "**chara/equipment/e[SET CODE]/model/c0201e[SET CODE]_[BODY SLOT].mdl**". It uses the same codes I mentioned in What's With the Names?, if you remember that.

If you remember the codes or want to go looking for them, you can enter this manually. Alternatively you can open the item up in TexTools and **copy** the text below the model viewer - it's already in the correct format for you!

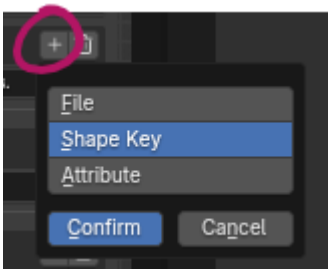
Once this is complete, return to the top of this group's options and click the up arrow to pack the group!

Shpx Key Selection



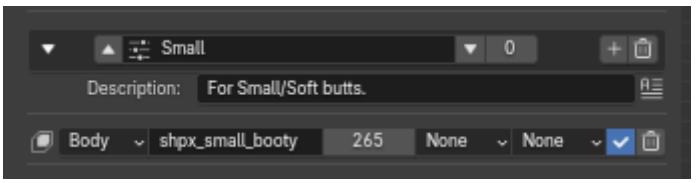
If you've included any custom shpx keys (such as my booty clearance options) or you want to toggle the leg squish, you can set up the group for that in the modpacker too.

Create another group as before with the **+** button next to the Update button. Then, toggle the **folder** button **off** to convert this to a customisable group. Remember to select whether this is a **Single** or **Multi** option group - if your keys are not intended to be toggled at the same time you want Single (such as my booty clearance keys), but if you want to be able to toggle both at the same time (such as L/R squish being separate toggles but you want to be able to trigger both) you should pick Multi.



This then allows you to click the + button here, which can be used to add a new option, which will pop up like this. **Name** your Option where it says "Option #1" and give it a description, if you want.. To turn this into a shpx toggle, click the **+** on the right and select the second option; **Shape Key**. Click **Confirm**!

This will add a new row below your description with a few fields in it, so let's cover them from left to right.



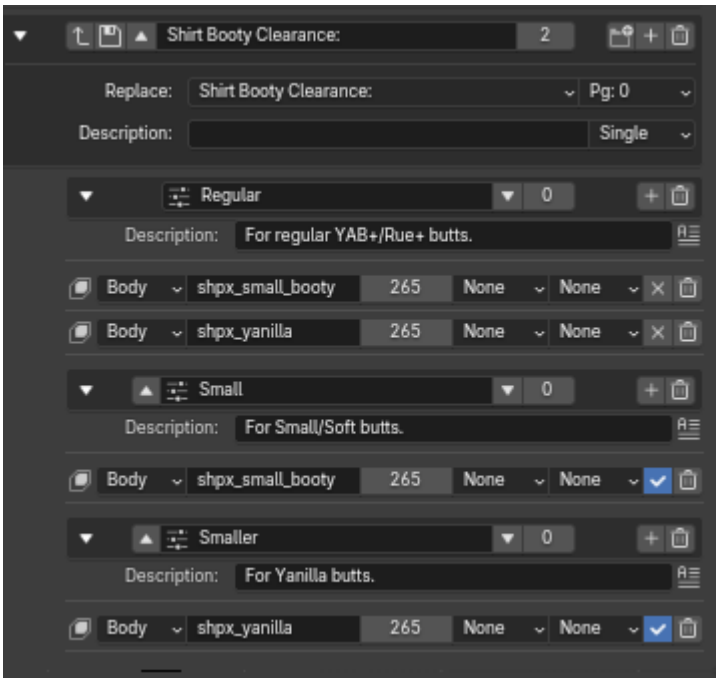
First, a dropdown where you can select which **body slot** is affected by the key.

Next, the field to enter the **name** of your shpx_ key. You can copy-paste this from your Attributes > Shape Keys list if you'd like.

In the middle is where you need to select the **Model ID** or **Set Code** for your gearpiece. This field doesn't like leading zeroes, but rest assured any 3-number code - 265 in this case will be read by the game as "0265". Leaving this as -1 will make this shape key trigger on all gear pieces with the same shape key - which I could use for my booty clearance keys, but you very much do not want to use this for leg squish keys - it'll squish any YAB/Rue legs!

The next dropdown is to select whether this shape key should activate if the gearpiece across a seam shares a key with the same name, and **which seam** this applies to. This is how the smooth waist (*shpx_wa_yab* and *shpx_wa_yabs*) function - though those are already set up to trigger automatically! You would only need to use this if you're making your own custom seam shape keys.

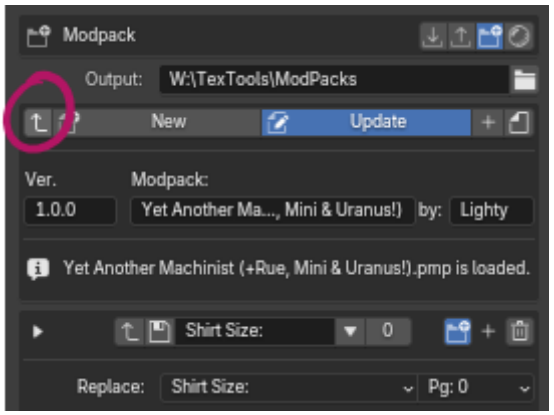
And the final dropdown is where you select **which race** this applies to, or "None" if it applies to all. I leave this applying to all since I don't want to enter every race separately, and unless someone working on male bodies happens to use the same shape key name this wont cause any issues.



And the final option of all is a checkbox where you control whether the shape key turns **on or off**. You'll likely only need to turn one on, but I personally add removing the shape keys to my default option - just in case!

So when all is said and done, your options should look something like this!

Once you have all of your groups set up, you can **pack them all in one go** by returning to the top of the File Manager window and hitting the **arrow** next to New!



Doing it with TexTools Instead

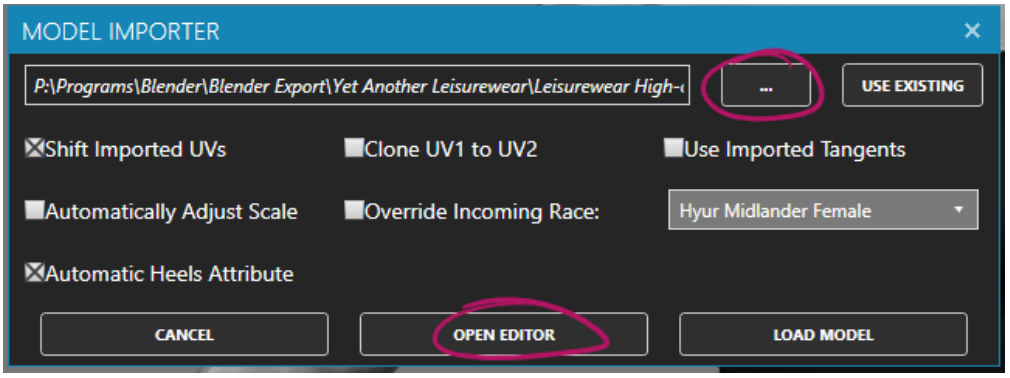
Exporting

If you'd rather not use the Modpacker, the other option is TexTools! For this, we'll need to go all the way back to our exporting step. The process is exactly the same, except for one thing: you want to export as FBX instead of MDL.

Instead of repeating all of that, I'm just going to link back to MDL export - but remember, select **FBX** instead!

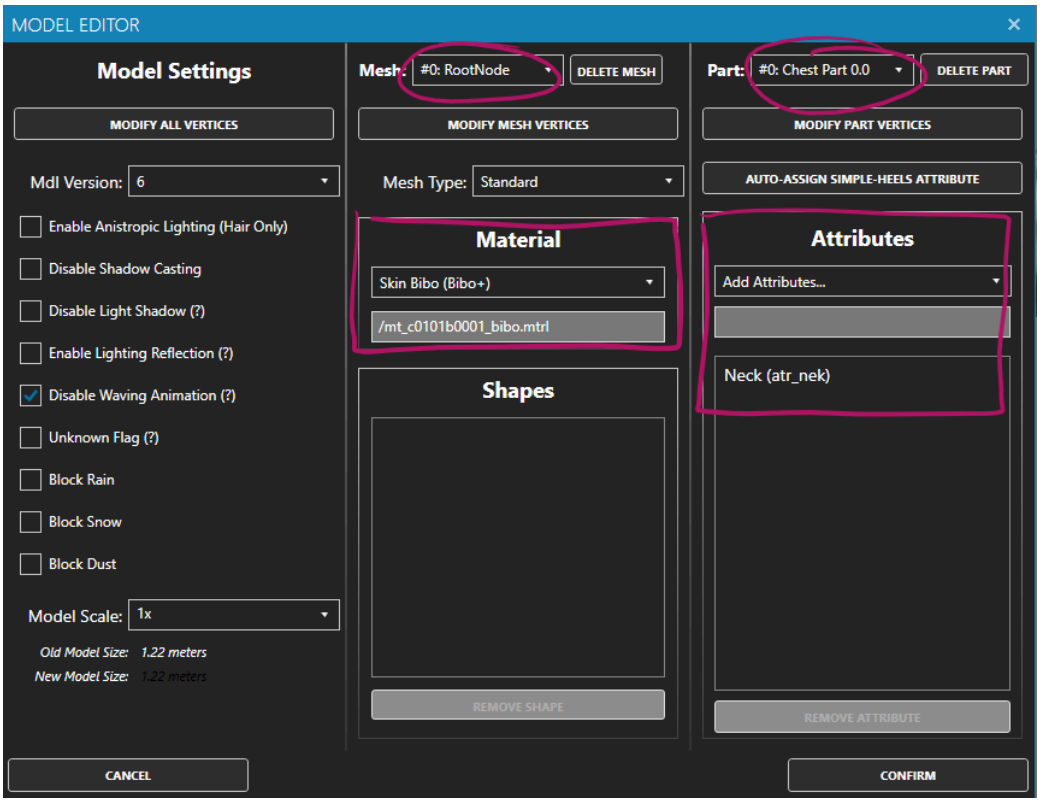
First Import

Importing into TexTools is pretty simple! Ensure you have a **Transaction** set up before you proceed. Navigate to the gear piece in question, ensure you're on the **Models tab** and are looking at the **Midlander Female** model, then hit **Load** at the bottom. **Click the three little dots** and **find your model**, then hit **Open Editor** - all of the other settings should be **default**.



The only other setting you may need to pay attention to is **Shift Imported UVs**; if you hover the prompt will explain that this should be toggled on for anything exported from TT 3.0 (basically anything since DT) but off for anything from 2.0 (pre-DT). If you happen to be importing an upscale on a model you exported aaaages ago or if the textures appear patchy and strange, try toggling this off.

From here you need to go through your **Mesh Groups** (the first number) and **Parts** (the second number) and assign the correct materials and attributes. You can safely ignore the Model Settings panel, though I would recommend checking the **New Model Size** in the bottom left to check it's not wildly different.

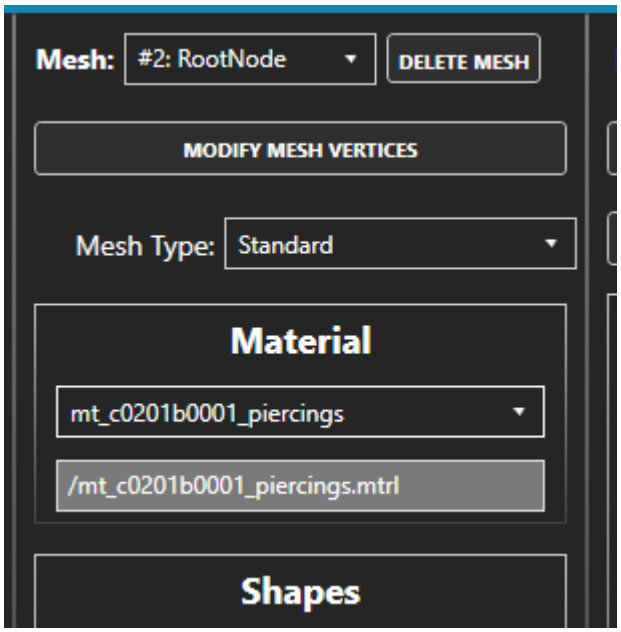


Since I left my body parts as 0.x, I'll go through skin parts first. In the **Material** dropdown select **Skin Bibo (Bibo+)**. Then in **Attributes**, use the **Add Attributes...** dropdown in tandem with the **Part:** selection dropdown in the top right to apply the following;

- **Neck (atr_nek)** on **Neck Part 0.0**
- **Elbow (atr_ude)** on **Elbow Part 0.2**
- **Wrist (atr_hij)** on **Wrist Part 0.3**

Torso Part 0.1 needs no attributes. Remove any other attributes left on the piece so it doesn't hide when it shouldn't.

Now the same for gear pieces! Remove incorrect attributes and add the correct ones based on which part you're working with - this is another place where naming the parts with what they are is very handy. Any parts you want the option to hide need variant attributes - this "**atr_tv_a**" option. You can add up to ten of these - A through J - but only the first three or those that are in use on the model already are visible by default. If you want to add one that isn't in the list, just click **Custom** and type "**atr_*v_**" and append it with the next letter (note that the 'tv' part changes based on the body part - tv for tops, dv for bottoms, sv for shoes. Just match it to the other atr_ options in the list and you'll be golden).



Last but not least, if you have piercings or nails on your gear piece, they need to point to their individual textures. From the **Material** dropdown, select **Custom** and add "**/mt_c0201b0001_**" to the field below and finish it with "**yafinger.mtrl**" for fingernails, "**yatoe.mtrl**" for toenails and "**piercings.mtrl**" for... well. piercings. No need to hide the piercings with attributes, as if a user has no piercings installed it will point to a blank texture and will appear completely transparent. Magic!

Once you're happy, click **Confirm**.

Once textures and attributes are all set, click **Import** and wait to see your errors. The main ones you want to pay attention to are the following:

- **"You may wish to scale the model up/down"** - This error is fixed by the **Import**'s **Cleanup** operations with the Devkit Skeleton selected as the Armature.
- **"Part x.x does not have a valid skin element"** and/or **"Part x.x Vertex y has no valid bone weights"** - The part in question either has no Armature modifier or is pointed to an incorrect skeleton, or you didn't export your Skeleton. Ensure all parts have an **Armature modifier** on them pointed to the devkit Skeleton and that the **skeleton is visible** in the Outliner before you export.
- **"Vertex xyz has abnormal bone weights. The weights have been reset."** - Usually crops up if a vertex group has been removed and the weights haven't been normalised after. It will make that vert misbehave and stick out *really badly*, so it does need to be fixed. Go back into **Weight Paint mode** on the part with the issue, select all (**2** for vertex selection then **A** for all), click on the **Weights menu** and hit **Normalise All**. Set it to **Deform Pose Bones**. Re-export and re-import.
- **"Part x.x: xyz vertices had major corrections made to their weight data"** - You're probably going to see this a lot. And the number will sometimes be very high. XIV's engine can only handle a single vert being attached to 8 bones, so any with 9 bones or weights that do not equal 1.0 (this is what **normalising** does, by the way) will throw up this error. I know a lot of people say to ignore it if it's less than 100 verts, but go back to the drawing board if it's more, but I find sometimes TT's normalisation does just fine. If you get this error, check your weights carefully in game, but it's entirely possible everything will work just fine. To fix it for real though, enter **Weight Painting mode**, press **A** to **Select All**, then hit **Weights > Limit Total**, set the total to **8** followed by **Weights > Normalize All**, then export and import again.

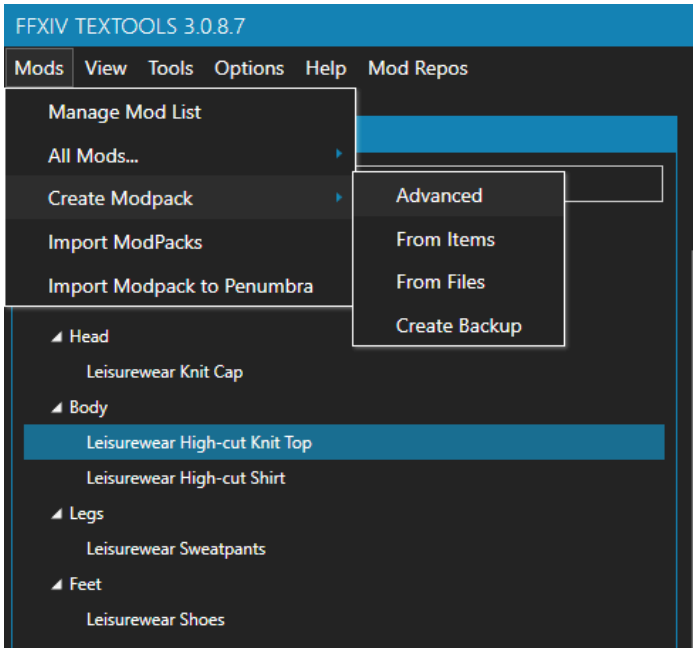
Once you have an import that isn't throwing any *concerning* errors, the bottom right of the TexTools window should have changed to **Save to Penumbra** with a little red exclamation point to tell you what you have isn't yet applied to your Penumrba modpack. Click that, and all your model settings should be saved!

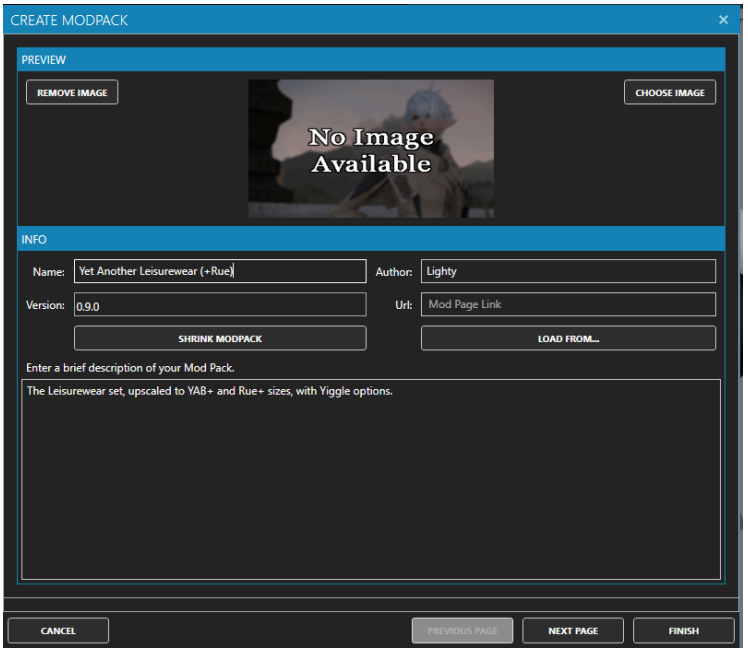
Making the Modpack

Ensure you have imported your model and set the attributes before this (just via a Transaction is fine!) - having it already set up will be a huge time save here if you're adding multiple sizes.

If you have multiple sizes, you'll most likely want to use an **Advanced Modpack** (**Mods > Create Modpack > Advanced Modpack**).

On the first page, **Name** your modpack, enter your **Author's name** and give it a **Version number** (I like to use 0.9.0 while testing) and add a **description** (note, if you upload a mod to Heliosphere, the helio description will overwrite this) and an image if you so desire.

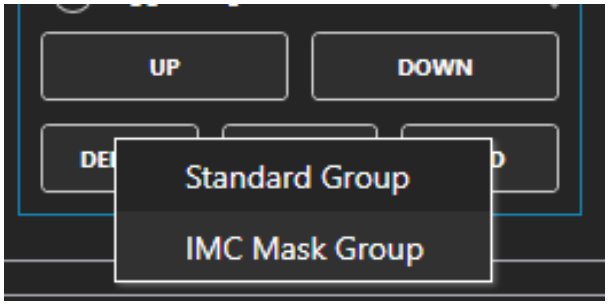




On the next page, hit the **Add** button at the bottom to add a new **Standard Group**. Select it in the listing and hit the **Edit button**. Name it at the top, then **add all of the size options** you'll be adding in the order you'd like them to appear (they can be re-ordered and renamed later). To add a model, select the size you imported a moment ago, use the item list search to find the gear piece, select the **Models tab**, ensure you have **c0201e____** selected (the midlander female model), **uncheck Include Child Files** (this will add metadata, textures, materials, etc, which if you're upscaling a vanilla piece you do not need and if you're upscaling a modded piece you shouldn't include), then hit **Add** to include your current imported model.

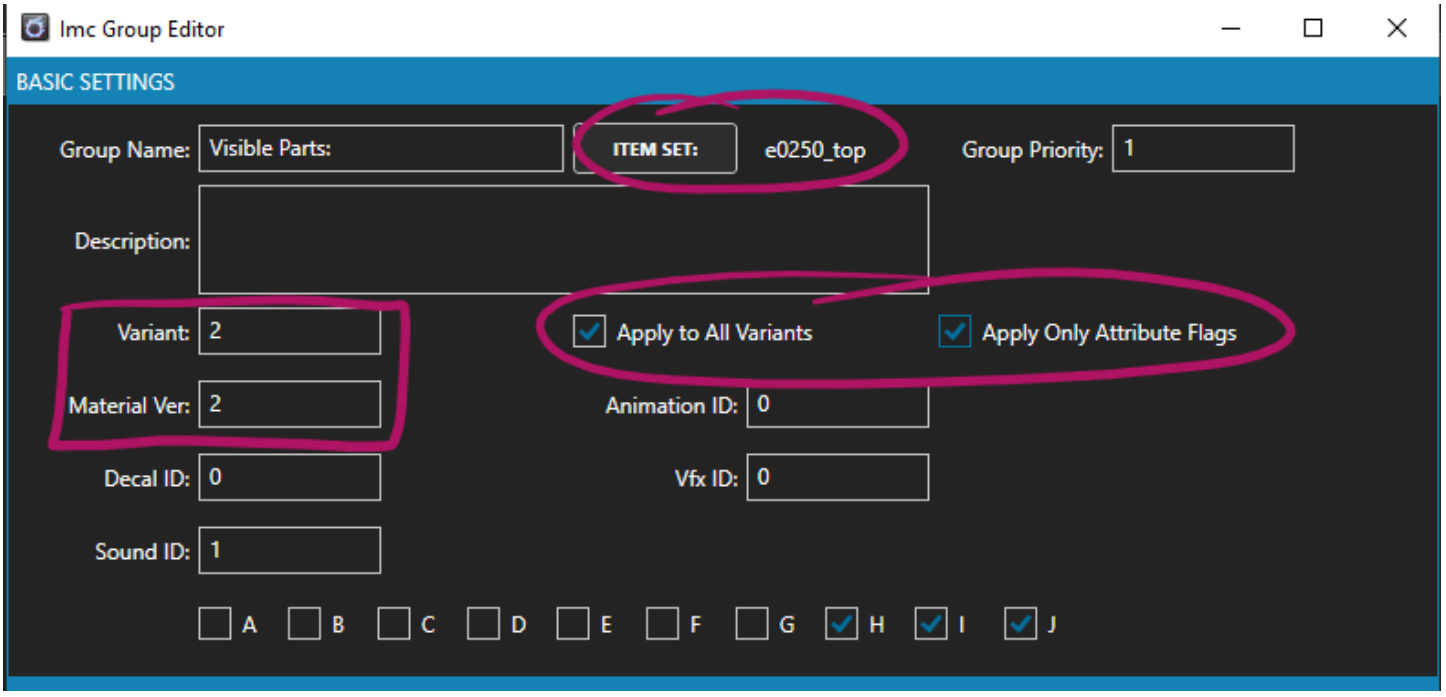
Advanced Modpacks have a shortcut here for adding additional sizes - instead of saving the modpack, importing something new, opening the modpack again and using the Add button, we can select another size from our options list (Medium Buff, for example), **select c0201...** again, then hit **Load**. This will open up the import window you saw before. This is why we wanted to have one already imported and saved in TexTools or a Transaction - any you add via the Load button now will have the same attributes and textures as the *current* mesh in TT. No need to add them for each model manually!

Repeat this step for all of your sizes that you have models for, then click **Done** in the top right.



IMC Groups

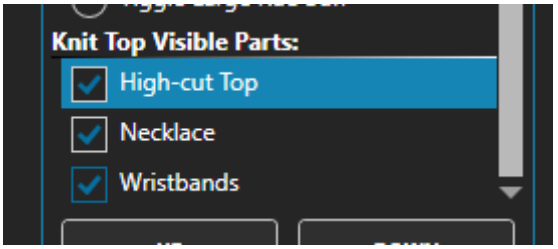
If you had any parts you wanted to add toggles for, **Add** an **IMC Mask Group**, select your new group and click **Edit. Rename** at the top and click the **Item Set button** to select your gear. Fill in the metadata at the top; the main thing to remember is the **Variant** and **Material Ver**; not having these filled in will cause the IMC group to throw an error and not function. You can find all of this information in the **Additional Options** (the burger menu next to the eye for the Models tab) > **View/Edit Metadata** tab > **Variant Settings** at the bottom (which, alas, you can't check while you're trying to make a modpack, but you *can* open a second instance of TT to check the values).



This applies to only the one Variant, so the **Apply to All Variants** checkbox will force apply it to all shared models. By default this will also apply the selected material to all variants as well, but the Only Attribute Flags option will fix this. If something has no shared models, you don't need to check either! (but do note if a new variant is added at some point, these IMC edits will not affect it)

The A through J checkboxes are for which parts are visible by default with toggles turned off - I prefer to have the parts toggle *on* when an option is ticked in the mod options since that makes most sense to me, but you can also have all of these parts checked in the basic settings and use the toggles to turn things off.

Once you've done this, begin adding options and name them, then select the corresponding "**atr_*v_***" attribute you added to the mesh. The checkbox here *toggles* the attribute in question, so if I had A ticked in my basic settings, the option to toggle A would *hide* the part instead. Click **Done** once you're finished.



Since I want all of my parts visible by default, I will check all of the boxes back in the **Option List** of the **Modpack Creator**. Once you've added all the groups as you want to include in your modpack, hit finish and save it. It will try to name the file the same thing as you titled it in the modpack, which is perfectly fine by me.

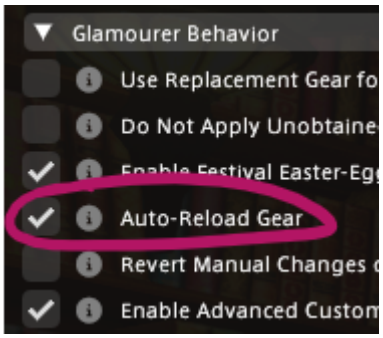
Now you can import it into **Penumbra** to test!

Testing

For the most part, bugs that desperately need fixing will be visible to you right away. If you're lucky enough to not immediately see anything, there's a few things I specifically poke around and search for:

- **Check Sizes** - Especially check the sizes of the chest for any sizes you made a sub key for - so Large if you made a Uranus size as well, Medium if you made Mini, et cetera. If you accidentally left one of these sub keys *on* when you exported, Large will have Uranus size.
- **Check for Seams/Holes** - Check along the edges of your separate parts to ensure the model is staying in one piece and doesn't have an obvious normals seam. Common culprits are the arm/leg seams (especially when the limb is bent) and the undersides of things like shawls and skirts - it's easy to miss those when piecing a mesh together.
- **Check Body Seams** - if your gearpiece covers a seam between body parts (e.g. the waist - think how the Skyworker's Singlet needs to meet the trousers at the waist seam), equip the naked bodypiece of the body you're testing on the other side of the seam to ensure there isn't a gap. Be sure to also check the smooth waist seams here if you added any!
- **Check Weights** - Use Glamourer to check the model at min+max bust slider on several different races (Au Ra is good for min, Roe is good for max) to ensure there's no clipping or gaping where weights don't align. Check tricky areas like underarms (the Miqo'te idle with the arms above the head is good for this) and hips (idle poses with more weight on one leg are good, but nothing beats making your character do the splits for between the legs). Check skirt weights of a top by putting a vanilla skirt underneath and making sure it doesn't clip though while moving.
- **Check C+ Compatibility** - Repeat the above step, but add C+ scalers if you want it to be compatible. I tend to only stress about ensuring compatibility with YAB+'s prepackaged scalers, but I do have an extreme scaler to make it more obvious where the problem areas are. I find it's usually just in the top of the cleavage and the front sides of the skirt are the common culprits.
- **Check YAS weights** - Move around with Yiggle turned on to check your booty and belly (for rue) are yigglin', or go into /gpose and ensure you can move the additional bones. Check the Rue belly doesn't have a seam along the waist when worn with Yiggle bottoms, and check the belly weights aren't on non-rue sizes.
- **Check Booty Clearance** - I like to /mogdance in gpose with very different coloured bottoms equipped, and certain /sits like roe's first sit /cpose are very good as well.
- **Check Attributes** - Equip different lengths of gloves and boots to ensure parts are hiding as required, no parts have accidentally ended up hiding when they shouldn't, and that shape data is squishing enough and not clipping.
- **Check Meta Options** - If you added any options for hiding parts, ensure they show/hide as intended.
- **Check Variants Work** - Check all versions/shared models of the gear to ensure none display strangely. Especially important if you included Meta options, as they often have unique metadata and there's a possibility one variant already has a part used hidden/shown.
- If your character is anything like mine, don't forget to check *under* the hair, without C+ scalers, without/under whatever else you may have changed/hidden. :)

Tip!



If you don't already have the "Auto-Reload Gear" option enabled in Glamourer's Behaviour settings, I highly recommend it! It will redraw your gear when you change any mod settings automatically, including being able to swap between sizes/enabled or disabled without your character vanishing for a second. Really helps to tell the differences between shapes and sizes!

If you made good use of the Addon's Animation and Pose functionality as well as the save to Penumbra feature, you can weed out a *lot* of these bugs before you pack your mod. But I still tend to find things wrong, even with these tools! Remember that just about anything can be fixed, and there's no limit on how many times you can go back and tweak, so don't stress over having a lot to fix. I find it helps to make a little list of things I've noticed that need fixing while I test, since I do often forget one or two once I'm back in Blender.

When it comes to fixing clipping for things like shawls, I will always toggle the mod on and off to check how much it clips in vanilla, too. You're never going to be able to completely remove all clipping, especially on larger busts, so don't drive yourself crazy trying to make it completely clip-free. I try to get mine to only clip about as much as vanilla with a little extra leeway the larger the chest gets, but if you aren't too picky feel free to be more lax.

For the rest of this section, I'm just going to cover some bugs and issues you may encounter aside from those listed above and how you can tackle them. I'll add more as I encounter them!

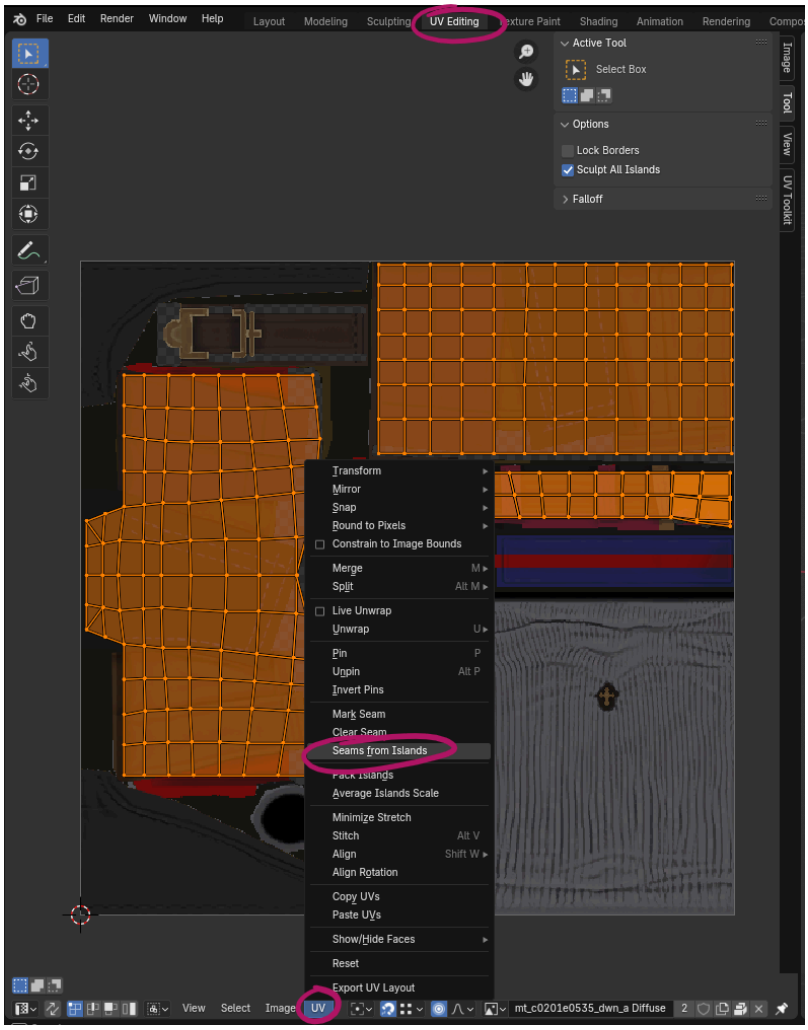
There’s a Weird Squiggle!

There’s a chance you may encounter a strange squiggle near texture seams. I encounter it a lot down the middle of skirts, but it can look like a strange discoloured squiggle like this, or stretched textures right along a seam. [VISUAL PENDING]

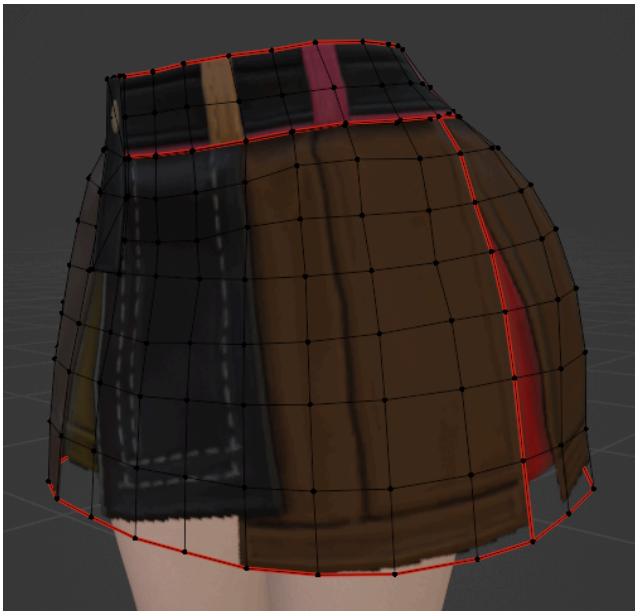
This is because the mesh isn’t quite splitting how it should across the UV (texture) seam. The solution for this is a lot more simple than it might sound - we just need to give it a much better idea where the seams are!

Select the mesh with the issue and enter the UV editing tab. In the left pane, the UV editor, press **A** to **Select All**, then go into the **UV Menu** from your header/footer > **Seams from Islands**.

This will give your model in the right pane some nice red lines to indicate it’s got a UV seam there - the exporter should be able to handle it much better now.



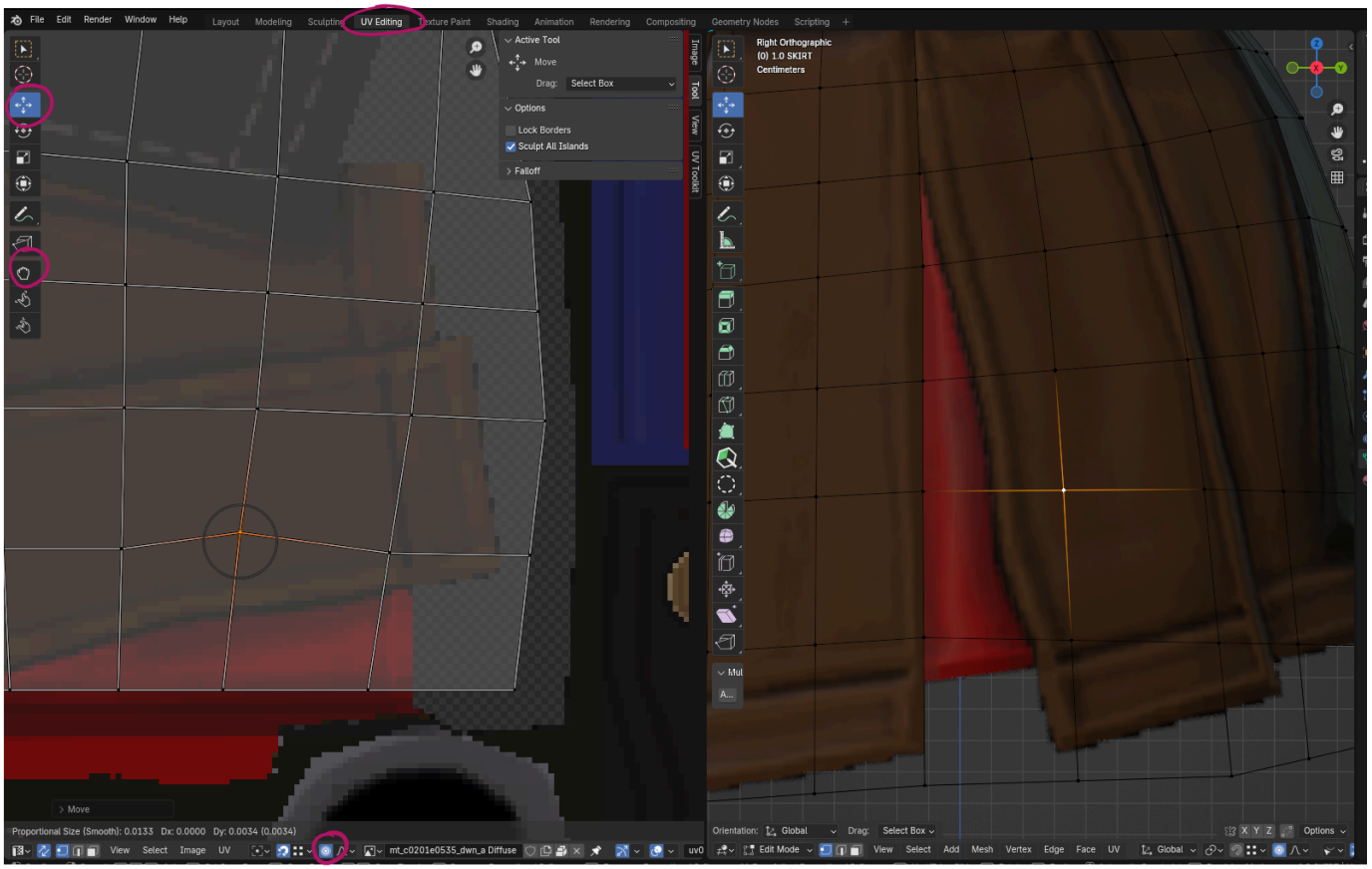
Re-export, re-pack, and re-test! :)



My Textures Are Wonky!

The difference in the shape of the model can sometimes mess with how the textures sit, like the white cut-outs right below the breasts here. You *could* adjust the model to get them to fit right, but sometimes doing that will prevent you from having the topology where you need it.

So instead, what you can do is go into the UV Editing mode tab at the top of your Blender window and select the verts where the issue is, then drag them around the texture map with either **Move** or **Grab** to get things to lie a little neater. Highly recommend having Proportional Editing on for this! Always aim to leave the seams fixed in place.



Once you've made a note of what needs to be fixed, the rest of the process is simply going back into Blender and tweaking the model or shape keys, exporting again, overwriting the modpack with the new models and testing again until you're happy with it. I can sometimes spend days on this step and go through upwards of 20 versions of the testing modpack because I'm very particular. As I mentioned before, sometimes it's easier and tidier to just start over if things are particularly messy - once you're as familiar with this process as I am, a full upscale of a relatively simple gearpiece only takes a few hours, but the troubleshooting step can take three times as long. Sometimes a fresh start is the most time efficient (and least frustrating!) option.

Once I'm happy with the condition of the mod and I can't see anymore bugs to fix, this is where I would get other people to test; send a draft to a commissioner, or pass the modpack off to friends to abuse for a few hours - I'll have been staring at this model for a day or two now and will likely miss a few things, and I as a single person can't test it for compatibility with absolutely everything!

Once they also can't find anything, after I've fixed what they've found, it's time to finish up, pose a nice preview pic and post it! Or you can just keep it for yourself, if you want to. :)

Afterword

Congratulations on your finished (maybe vanilla) upscale! I'm proud of you! And thank you for making it this far, I know it was a lot to read!

This guide is really a labour of love. I don't even want to think about how much time I've spent just writing and rewriting and editing (and fighting the images and tables causing everything to have a seizure - you're a butt, docs). But I do it because I want to share the joy of upscaling (or just the joy of seeing a piece in the size you like if you don't find the process fun, lol), and I also love to teach! I find teaching is a great way to deepen my understanding of a subject.

If you'd like to say thanks, you can drop me a message on Discord (or you can ping me in the YAB server!) or BlueSky (both are @fightylighty), or if you really want to you can drop me a tip on Ko-Fi. These are also the best places to get in touch if you need some help or spot a mistake I should go back to fix! I promise I don't bite! Alternatively, if you'd prefer to stay anonymous, I made a form where you can submit feedback: https://forms.gle/S47jgtHXnA4eQevb9

I really hope you enjoyed the process at least half as much as I do 😘

Glossary

There are a lot of 3D modelling terms that might be unfamiliar to you mentioned within this guide, so here's some extra information about them.

Edge - straight lines on the edges of *Faces*.

Element - connected *Faces*, *Vertices* and *Edges* within an *Object*. Ctrl+L is used to select linked, which will select just the current *element*(s).

Face - a single flat plane on a 3D model.

Mesh - a 3D model.

N-gon - A face with more than four edges.

Normals - this can refer to either the *Normal Map* or *Vertex Normals*. The *Normal Map* is a texture that typically uses RGB channels to show XYZ three dimensional data that the engine uses to pretend pixels are facing different directions to make a model look more detailed than it actually is - the purple looking texture. *Vertex Normals* are model data that, very similarly, controls what direction the vertices are facing. Unifying vertex normals across neighbouring faces can smooth sharp angles on edges.

Object - a collection of *Elements*. Individual meshes or models listed in your outliner tab. Each of your unique "Part x.x" meshes is a separate *object*.

Polygon/Poly - see Face. Typically used to refer to specifically 4-sided faces.

Quad - a four-sided *Face*.

Tris/Triangles - see *Face*. Used to refer to specifically 3-sided faces. Most game engines require models to be split into tris - XIV's included.

Triangulation - the process of turning *faces* (both *quads* and *n-gons*) into *tris*.

UV/Unwrap - 3D shapes need to be flattened onto a 2D plane to apply textures, this act of flattening is called "unwrapping", much like unwrapping a gift. "UV" (originally "UVW" - a string of letters chosen to differentiate the 2D unwrap from the "XYZ" 3D coordinates) is used to refer to these unwraps.

Vertex (plural **Vertices**) or **Vert(s)** - a corner between *faces* on a 3D model, individual points.