Poser Panel Add-on for Blender

Summary

The goal is to produce a morphed figure unimesh OBJ that is in the same vert order as the figure's original source OBJ.

Among other things, this OBJ may be used as a geometry file for the Create Full Body Morph command. This depends on how Blender imports and exports OBJ files. It must maintain the OBJ integrity in such a way that the final exported OBJ is essentially the same (in form) as the original source OBJ. Critically, Blender has an ability to maintain vertex order, poly-groups, and materials in an OBJ from import back through export.

You will need 3 files to create a morphed unimesh:

1. The original source OBJ.
2. A Poser exported OBJ of the ***zero'd*** figure, welded
3. A Poser exported OBJ from the ***morphed*** figure, also welded

Poser

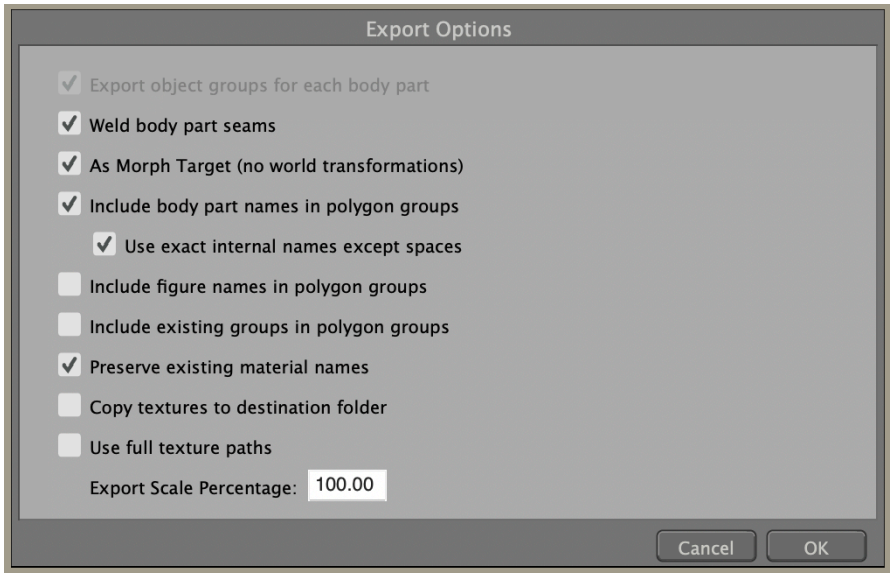
Producing the 3 inputs used by the add-on:

1. The original source OBJ

   For Poser figures, this is in a geometries folder. The easiest way to find it is to look in the figure's CR2 file, for the OBJ is used.

2. A Poser exported OBJ of the ***zero'd*** figure, welded *as Morph Target*.

   The OBJ must be completely zero'd. All IK is disabled, no rotations, no translations, no morphs activated. Only a single frame, and only the one Figure. It should be exported with the following options:

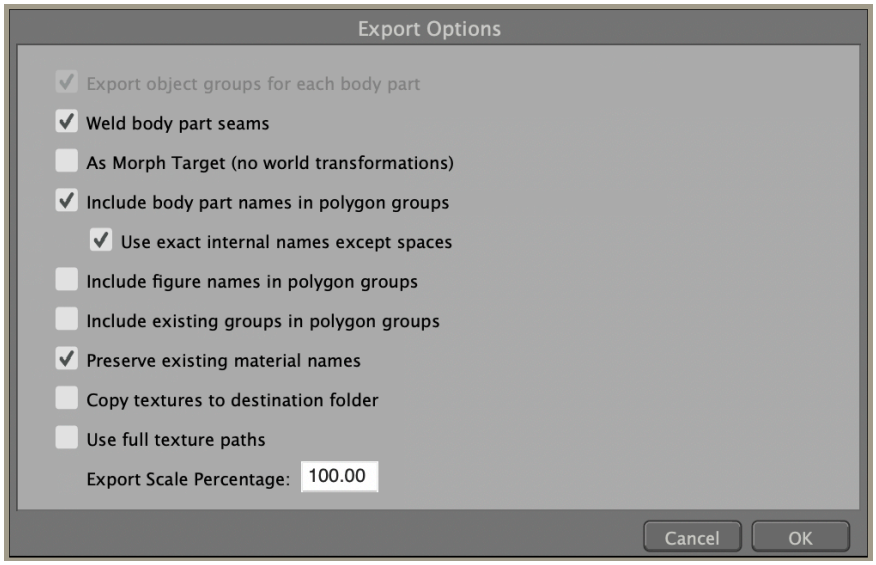


   Note: often figures load in a zero state by default (eg: Dawn). Some figures do not (eg: LaFemme; has IK enabled which causes rotations). You should always check to make sure.

You can easily zero a figure using the following steps (in order):

1. Turn off all IK by unchecking all options under **Figure > Use Inverse Kinetics**
2. Zero the figure by selecting **Figure > Zero**

3. A Poser exported OBJ from the ***morphed*** figure, also welded but *not as Morph Target*.



It doesn't matter what you use to morph the mesh. You can load morphs, dial them, use the morph brush, combine morphs and magnets. All that matters is the shape of the figure's mesh, not how it got that shape.

However, no IK, no scaling, no rotations, and no translations (unless, for some reason you want them to be considered part of your morph). Essentially, you typically do not want the rig to affect the mesh shape. Remember, this is the shape the unimesh will take on.

The same export options as the previous step. The only difference is that you have changed the shape of the mesh.

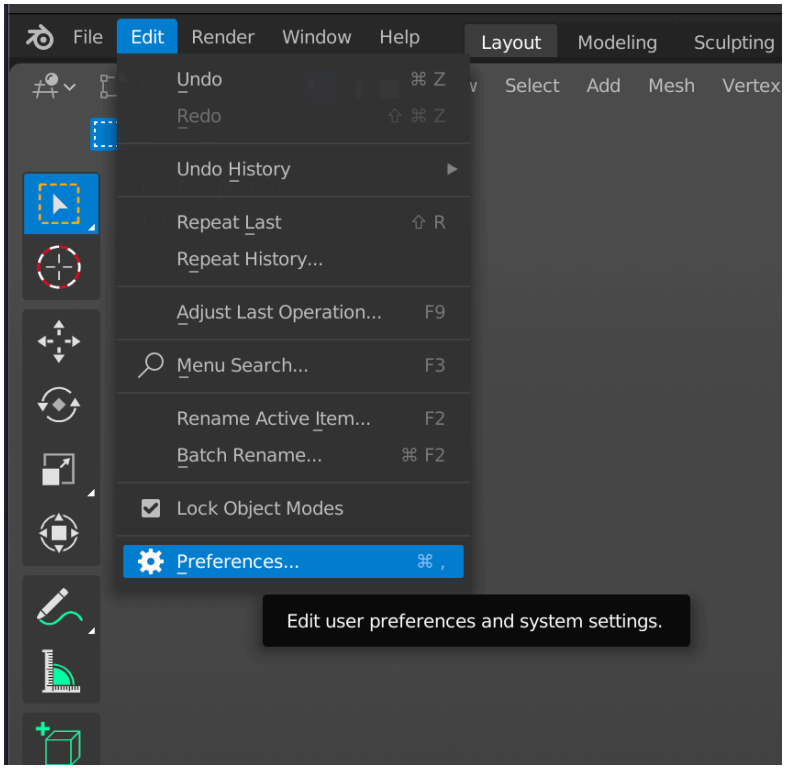
Blender

Install the Add-on

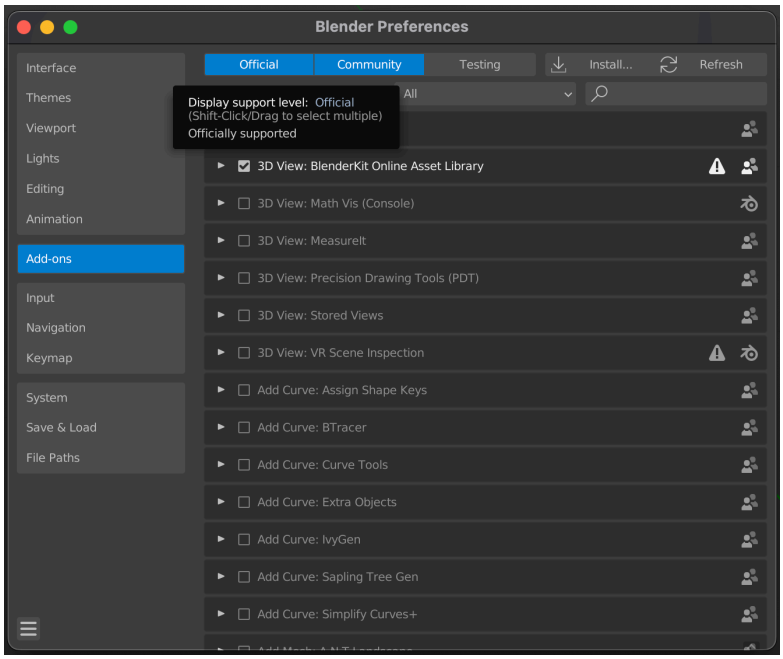
You only have to do this once. If you know how to install an add-on, just install `poser panel.py` file and skip this section.

Make sure you have the file `poser panel.py` somewhere known.

1. In Blender, Open Preferences



2. Select the Add-on Tab



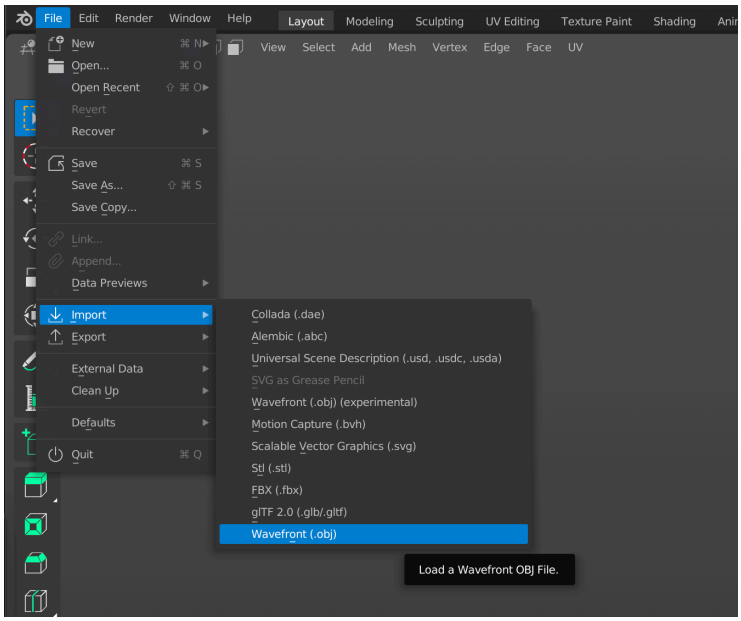
3. Click install, find the `poser_panel.py` file and select Install Add-on.

4. Check the box to activate the Add-on. The panel shows up as **Poser Unimesh** in your panels tabs. Open it and resize it like any other tab.

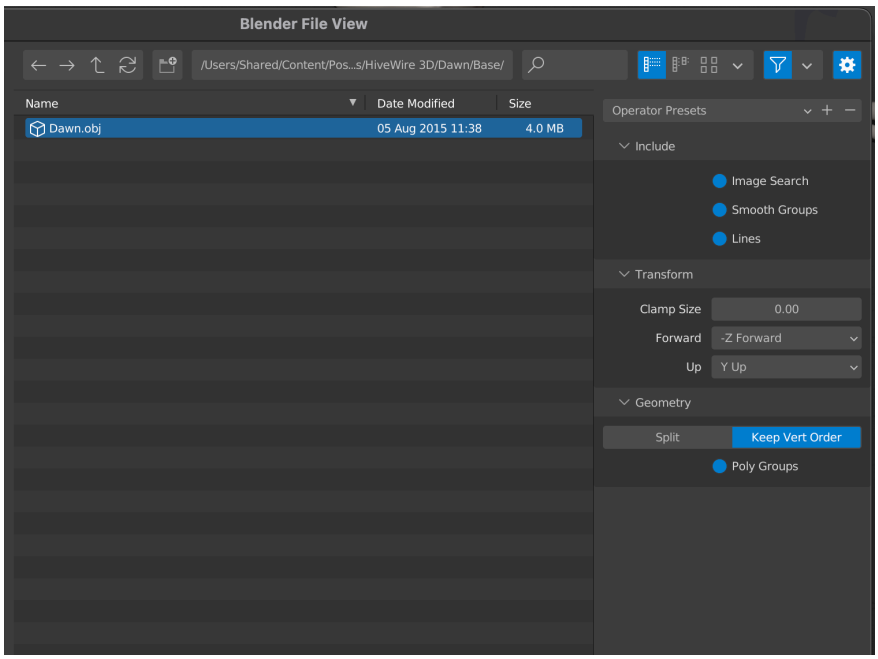
Use the Add-on

Import all 3 OBJ files

Note: as of Blender 3.2 there are two OBJ importers. The newer experimental and the older. It's safer to use the older. I found a bug in the newer, which the developers fixed but there are still edge cases where it doesn't. The new import/export is written in C++ and is blazingly fast for huge OBJ files. Poser OBJ files of single figures are not very big and the old version only takes a split second anyway so the gain is inconsequential in this use case.



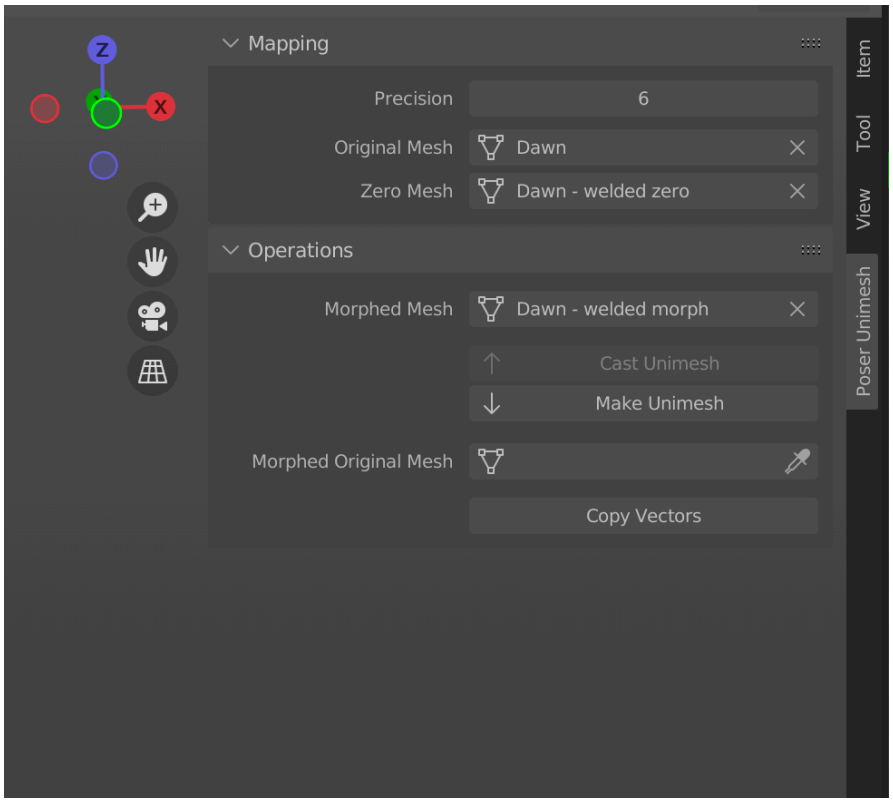
1. Import the original source OBJ



   Select **Keep Vert Order** and **Poly Groups**.

2. Repeat this step for the Poser exported welded zero OBJ

3. Repeat this step for the Poser exported welded morph OBJ
4. In the Add-on Panel, select those Objects in the appropriate places



In the panel there are two sections: **Mapping** and **Operations**.

Mapping holds the two meshes used to calculate the map. In this case, we're calculating a map between the source OBJ and Poser's welded (but differently vert ordered) OBJ of the same mesh. The map will allow us to resort verts into the original order :)

The fill-ins are Blender **Object Selection** boxes. You can type the object name, click on it for a drop down of all objects, or click the eyedropper then directly select an object by clicking on it in the 3d view. I tend to use the drop down. Also, here is where naming things clearly helps :)

The Object is actually tracked internally. Which means once you've selected the Object in the Add-on, it doesn't matter if you change it's name elsewhere. The name you see in the Add-on always updates.

I always check that this setup produces the correct output. If you've accidentally gotten any of the assorted import or export OBJ options wrong, things will likely not go as you expect.

5. Operations is where you trigger the Add-on to do something. Actions only become available when the Add-on has sufficient information. The Add-on was created to mainly **Make Unimesh**.

The 4th Object is Morphed Original Mesh. This is the target mesh. If it's blank, the Add-on copies the SRC mesh and uses that. If it's already a copy of the source, it will re-shape it based on the Morphed Mesh.

You have 4 objects:

1. The original source OBJ

2. A different OBJ (eg: different vert order) of the exact same shaped mesh

3. A differently shaped version of (2)

4. A version of (1) shaped like (3)

6. Export this the Morphed Original Mesh as an OBJ. You minimally want vert order and poly-groups checked. You can use this OBJ as a geometry file for a Poser FBM. I usually test this before editing in Blender or using the OBJ elsewhere.

Tech Details

At its heart, the add-on maps between two forms of the same mesh. “Same mesh” means each vert in one mesh (keyed by XYZ) has at least one vert in the second mesh. In a simple wireframe 3d: mesh 1 looks exactly like mesh 2. Same vertices and same faces, all in the same places.

Why?

Because Poser can’t export an OBJ of a morphed mesh that it can use itself as an input OBJ for a FBM. You heard that right :D

This is due to a few things:

1. In OBJ format, a vertex ID (unique identifier for a specific point) is determined by its order in the file. OBJ is a text format, with vertices listed in order, one per line. For instance, vert 10 is the 10th vertex in the list. It could have had a serial number to make it clearer. But with 1 vert per line, the line number is guaranteed to be unique. And it’s essentially an ID already. Efficient.

2. For an OBJ to be used as a FBM, you have to know which input vert index refers to which target (Poser) vert.

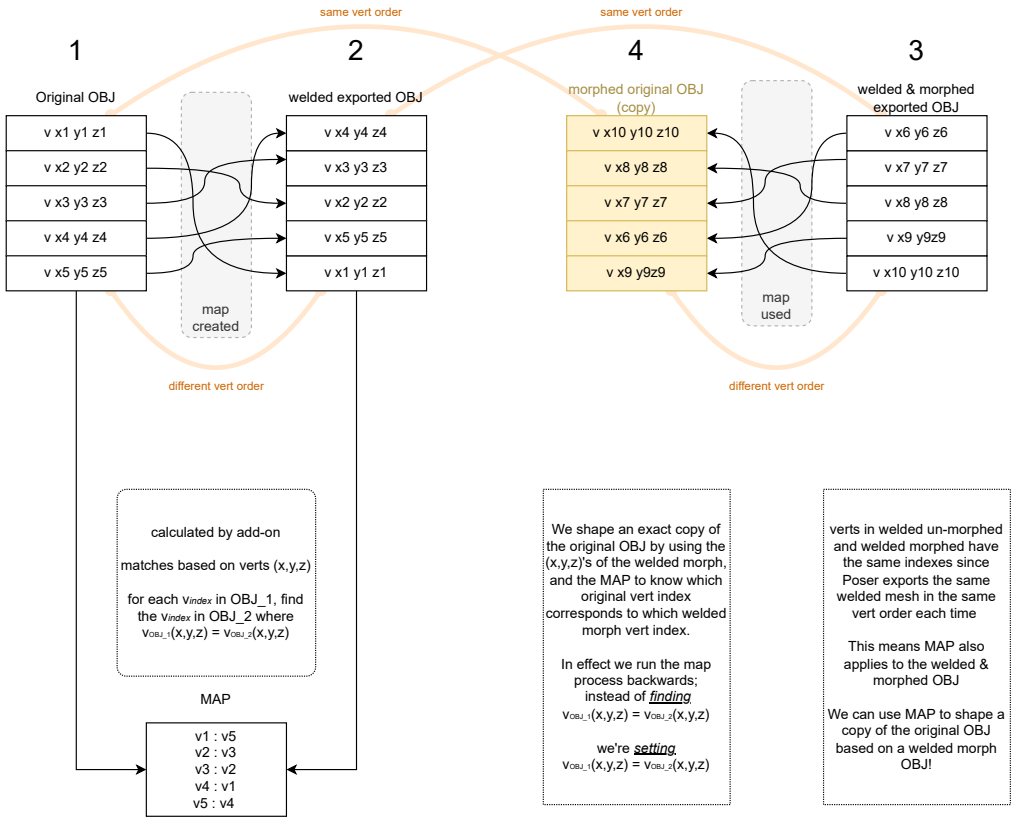
    1. With vert IDs being the line number, for two OBJs to refer to the same mesh, they need the same number of verts (“wrong number of verts” error).

    2. And in the same order (‘morph makes mesh explode” error)

3. BUT… When Poser made the figure mesh it split the mesh into segments, one per actor. When it does this, obviously, some vertices (those whose edges are common to multiple actors) will have to be duplicated in 2 (or more) actors.

    Rant: this would be trivial for Poser to keep. And it would be an easy modification for Poser. This Add-on’s main mapping logic is not a dozen lines of python. The other 185 lines is GUI and error handling. Even better, Poser could easily save this in the figure file when the file is first made. It would never have to calculate it again. The figure file already contains all the information needed. It’s. Just. Not. Done. :(

This add-on constructs a map between 2 objects that have the same mesh. It can use this map to copy vert positions in either direction.



Advanced Use

Blender Shape Keys

Creating a unimesh morph as a Blender Shape Key. A shape key is similar to a morph in Poser. That is, it's just a different shape for the mesh (ie: the vert XYZs may have different values. Same verts, though). Shape Keys are stored in Blender Objects. It's nothing more than list of alternate values for the mesh verts.

> Aside: In Blender you can treat Shape Keys similarly to morphs in that you can combine them, set them at different values, etc. Blender allows you do create an attenuation vertex weight map between 2 keys. Which can be useful in morph creation. The Blender GUI is not purpose built for Figure Posing so some features (like per actor activation of a morph) are not directly available via Blender's GUI without some "grind work".
>
> It's small things like this that has me move meshes back and forth between poser and Blender. Some things Poser does more easily, some things Blender does.

To add a a morph as a shape key, check "as Shape Key"

- If there's no target mesh, then the source mesh is copied, a basis key established, then a new shape key containing the morph mesh shape is added.
- If there's a target mesh, but no shape key selected, then a basis is added and new shape key created.

• If there's a target mesh with shape key selected, then that shape key is changed to mesh shape.

Note: if you target an object with shape keys but don't select "as shape key" it will alter the base mesh. But you can't see it, as it's hidden beneath the basis shape key. You have to delete all shape keys to see the base mesh.

Reverse transfer (casting unimesh)

Instructions not specified at this time But it takes the morphed unimesh and writes it back to the morph mesh. Not normally useful unless the morph mesh comes from a split mesh and you want to sculpt a unimesh in Blender but (for some reason) write back to a morphed split mesh. Note: a morphed split mesh is also imported by Poser correctly as a FBM. It's just not sculpt-able in Blender due to the seams.

Copy Verts

This copies the positions of selected verts one one mesh to another. Useful when used between meshes of the same geometries. You can also do something similar by weight mapping between shape keys.