

# ManFriday's Seamster

Version 1.0.3 (January 30, 2024)

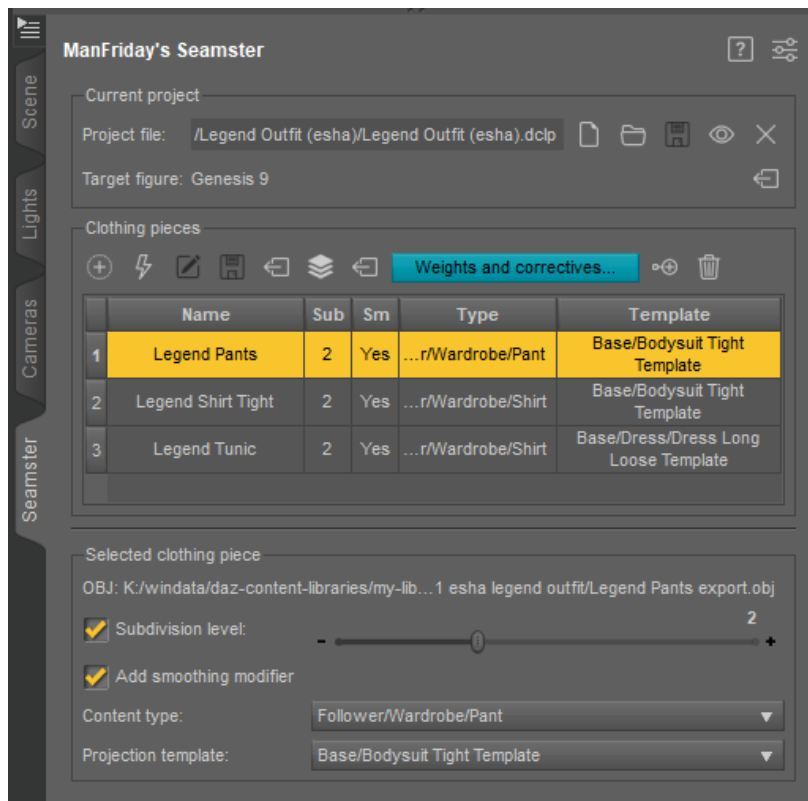


Figure 1: The “Seamster” pane.

## 1 Features

Making 3D clothing for Daz Studio usually involves the following phases:

1. **Modeling** and UV-mapping the clothing with 3D modeling software<sup>1</sup> and exporting it in the OBJ file format.
2. **Importing** the OBJ onto a Genesis figure in Daz Studio and **rigging** it (giving it bones and making them work with weightmaps and corrective morphs).
3. **Texturing**.

Seamster is a plugin for Daz Studio 4.21 and higher by ManFriday<sup>2</sup> that simplifies step 2, the importing and the rigging, which many consider the most tedious part of making clothes for Daz Studio.

<sup>1</sup> Marvelous Designer is a popular choice for this because it maps traditional tailoring with sewing patterns to 3D clothing.

<sup>2</sup> <https://www.daz3d.com/manfriday>. Unlike many of my other plugins, Seamster will not work on Daz Studio versions before 4.21 because it makes use of newer Daz Studio features.

Seamster greatly reduces the many clicks that are normally required, reducing the risk of forgetting something along the way, which can otherwise lead to a lot of lost time and frustration. It imports your clothing OBJ and runs Transfer Utility with just two clicks. It helps you with checking and painting the **weightmaps** for all the bones and with making **corrective morphs** for certain bone rotations. It can also help with creating **rigid follow nodes (RFNs)**.

For a detailed feature list, please see [Detailed feature list](#), page 22. Seamster supports Genesis 8, 8.1 and Genesis 9 figures. Older figures have not been tested.

## 2 Setting up

Seamster has its own “Seamster” pane in Daz Studio (figure 1). After installation (see [Installation](#), page 23 for details), you must add the pane to the Daz Studio user interface once before you can use it.

Do one of the following:

- Right-click on any of the backgrounds next to the existing panes in the user interface and select “Add Pane (Tabs)”. This will add the new tab near where you clicked.
- Go to “Window” → “Panels (tabs)” in the main menu bar (see figure 2).

In either case, you should see “Seamster” as a selectable option; select it, and the new pane should show up. If you do not see the pane in the list, then the plugin has not been installed correctly; see [Installation](#), page 23.

## 3 What is rigging?

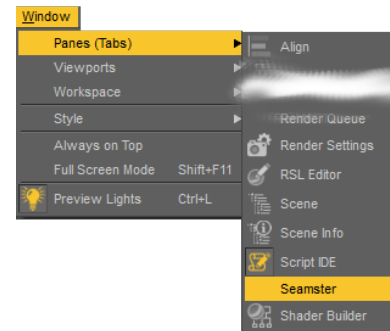
If you have rigged clothes for Daz Studio before and only want to know how Seamster helps you with that, you can skip this section. It is intended for users who haven't and want to know what bones, weightmaps, correctives and followers are. Even though Seamster makes things significantly easier, it will still be helpful to know why clothes move magically when a figure's joints are bent in Daz Studio.

The good news is that even without Seamster, you normally don't have to rig clothing from scratch. There is a tool in Daz Studio called “Transfer Utility”<sup>3</sup> which sets up a basic rig for a piece of clothing by copying it from a Genesis figure, and most of clothes rigging involves fixing the weightmaps of this initial rig and its corrective morphs, and adding new ones.

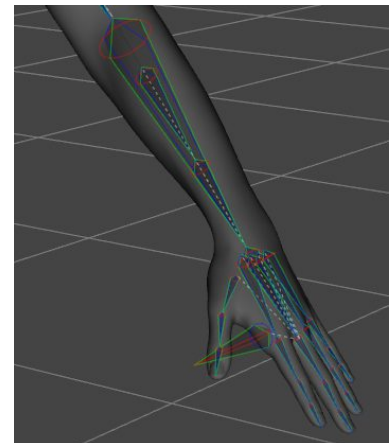
But even these steps can get very tedious and repetitive, so Seamster will help you transfer the rig of a Genesis figure to a clothing piece and fix weightmaps and corrective morphs.

### Bones

Essentially, in all of 3D (not just clothes, and not just Daz Studio), rigging is the process of adding “**bones**” to a 3D mesh. Bones are not visible in renders, but they work behind the scenes to make parts of a mesh move in a useful



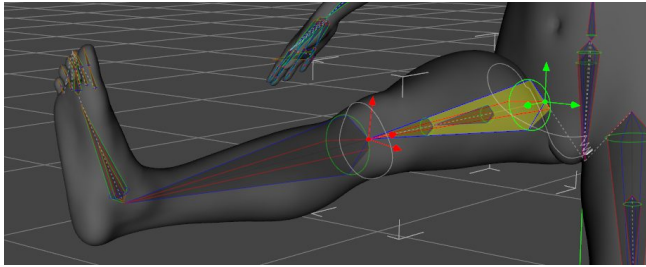
**Figure 2:** Adding the “Seamster” pane.



**Figure 3:** Bones in the arm of a Genesis 9 figure.

<sup>3</sup> “Edit” → “Figure” → “Transfer Utility”.

way. Combined, the bones of a figure are usually called a “**rig**”, or a “**skeleton**”.<sup>4</sup> You can see a figure’s bones in Daz Studio when you switch to the “Joint Editor” tool (figure 3).



<sup>4</sup> Blender uses the term “armature”.

**Figure 4:** Genesis 9 “Right Thigh” bone with “Bend” at  $-90^\circ$ , with the “Joint Editor” tool active.

Bones always define *rotations* (unlike morphs, which define translations). Take a look at figure 4: each bone has a “center point” (displayed in green) and an “end point” (red), which rotates around the center point. When a bone is rotated, the parts of the mesh near it move along. And since the bones of a figure are organized in a hierarchical tree, the “children” of the bone follow as well: when you move the right thigh, the rest of the leg (shin, feet, toes) should follow and not stay in place, exactly like in a human body.

The bones of a figure appear in the “Scene” tab with a bone icon (figure 5). When you select a bone, the “Parameters” pane will show dials for the rotations that are possible for the bone. The rotation dials around the X, Y and Z axes are normally renamed to “Bend”, “Twist” and “Side-Side” or similar.

Any mesh that has such bones (i.e. rigging) is called a “**figure**” in Daz Studio.<sup>5</sup> That includes all Genesis figures, but also animals, clothing and hair.

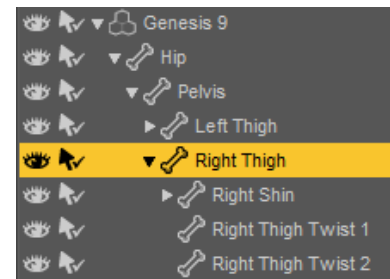
Even an inanimate object like a house can be a “figure” if it has parts that have been made movable with a “bone”, for example a door that can be opened. This illustrates that the term “bone” is purely technical: bones are defined where it should be possible to rotate a part of a mesh. Some of the Genesis bones correspond to real bones in a human body, but many don’t. For example, Genesis has many “bones” in the face to support realistic facial expressions; even the tongue has bones.

## Weights and correctives

To determine which parts of a figure’s mesh should move when a bone is rotated, Daz Studio stores additional data with each bone: weightmaps and corrective morphs.

1. For human figures starting with Genesis 3,<sup>6</sup> each bone has associated with it one **weightmap**, which can be imagined like a painted layer over the mesh.<sup>7</sup> Like the bone itself, this “paint” is normally invisible. It can have an intensity from 0% to 100%, determining how much that part of the mesh should rotate along with the bone, where 100% (red) means that the mesh should follow the bone completely (figure 6).

To see the weightmaps for a bone in Daz Studio, select the bone in the scene tab; switch to the “Node Weight Map Brush” tool; then open the “Tool Settings” tab and click on “General Weights”. You should then see the weights for the bone similar to figure 6. Note that not all bones have

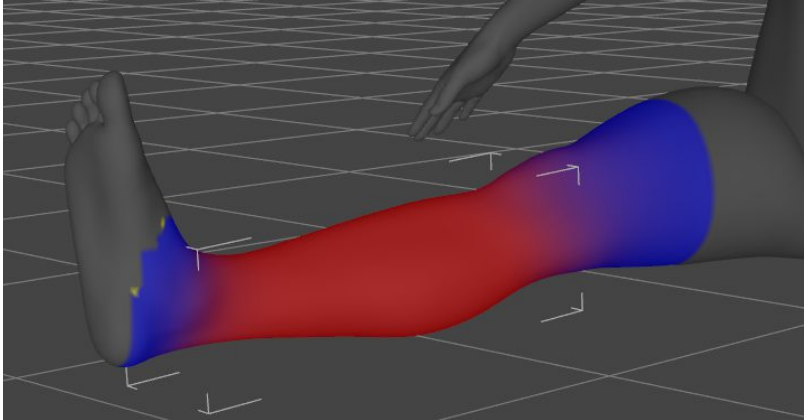


**Figure 5:** Bone nodes in the “Scene” tab.

<sup>5</sup> See **figure** in the glossary.

<sup>6</sup> For older mechanisms, see **rigging** in the glossary.

<sup>7</sup> See **weightmap** in the glossary.



**Figure 6:** Weightmap for the “Right Shin” bone.

weights, and sometimes the weights for one bone are stored under a different bone.

2. **Corrective morphs** are often added to a specific bone rotation to correct the positions of vertices that would otherwise be unrealistic. They are morphs like other morphs in Daz Studio,<sup>8</sup> but they only kick in under certain conditions. Daz Studio uses Enhanced Remote Controls (ERCs) to make this work. These are formulas around property dials which can express complicated if-then relationships, for example, “turn this on if the X-rotation is greater than 55°, but only half as much”.

<sup>8</sup> See **morph** in the glossary.

To emphasize the differences:

1. While there is only *one weightmap* for every bone, which is shared between all rotation axes of the bone, a bone can have *several corrective morphs*, and they can be much more specific, as they are triggered by changes in a specific rotation dial. They can even be so specific that they only turn on for specific amounts of that rotation.
2. While the weightmaps determine the intensity of a *rotation* (with the paint color representing the strength), correctives are *morphs* and therefore cause a *translation* along an axis.

The general rule during rigging is **weights before morphs**: weights should be fixed as perfectly as possible for all bones before working on corrective morphs. Weights determine in broad strokes what should rotate along with a bone, and the correctives fine-tune that.

## Followers

Even though clothing and hair have bones like other rigged figures, they are special in that they can be “fit to” a human figure. Figures that can be fitted to other figures are called “**followers**” in Daz Studio. The most common examples are clothing and hair. For that to work, the follower must have the same bone structure as the figure it is fitted to.<sup>9</sup> “Rigging clothing” in Daz Studio means setting up the bones of a clothing follower so that these mechanisms work when a user loads the clothing piece onto a Genesis figure (or fits the clothing manually).

<sup>9</sup> See **follower** in the glossary.

When you fit clothing to a figure (either simply by loading a clothing piece with a Genesis figure selected, or using a manual “fit to”), the following happens behind the scenes:

1. The **bones** of the follower are linked to the bones of the parent figure, so that when a parent bone is rotated, the follower does the same. This matching happens by bone name: for each bone in the parent, Daz Studio checks if a bone by the same name exists in the follower.
2. **Weight maps** in the follower are not modified during fitting, they stay the same.<sup>10</sup>
3. With **corrective morphs**, the picture is a bit more complicated.
  - If a follower comes with a custom morph that has the same name as a corrective in the parent figure, Daz Studio sets up the follower's corrective to behave like the parent's. This is how it is possible to fix bends in a follower: if the author of a clothing asset defines a corrective morph for a certain bone rotation, it gets turned on automatically if the bone dial in the parent is modified.
  - By contrast, if a corrective exists in the parent but *not* in the follower, even though the bone exists, then Daz Studio transparently projects the corrective from the parent into the follower as soon as the corrective is needed. To emphasize, this does not happen while the follower is fitted to the parent: it happens later, as joints are moved.<sup>11</sup> Morphs of this kind have a “Type” of Modifier/Shape/Generated in their parameter settings, as opposed to just Modifier/Shape.

<sup>10</sup> The weightmaps in the clothing piece were defined once by the artist when the clothing piece was rigged, with or without Seamster, as described in this manual, and saved with the clothing asset.

<sup>11</sup> This projected morph is truly “on demand”, i.e. temporary, and is not saved with the scene or asset data: if you undial the bone property, save the scene and load it again, the corrective will have disappeared.

In other words, a corrective clothing morph can be in one of the following states:

1. It can exist only in the parent figure, but not in the clothing piece. This happens if the rotation for which the corrective has been set up has not yet been used, and no custom corrective has been saved with the clothing asset.
2. It can exist in the parent figure and be dynamically injected into the clothing piece, without modification, as joints are rotated. This corrective is not saved with the clothing piece.
3. It could be a custom corrective in the clothing piece that was created and saved with the figure/prop asset by its author.

With these things in mind, you are now ready to work in Seamster!

## 4 Basic steps in Seamster

ManFriday's Seamster operates on “**clothing projects**”, which are directories on disk that are named according to the Daz Studio clothing product conventions. For example, if you want to start a project called “My outfit” for Genesis 8 Female, the usual project directory would be `People/Genesis 8 Female/Clothing/<Author>/My outfit` in one of your content libraries.

Into such a directory Seamster writes its own project file with a `.dclp` extension.<sup>12</sup> This remembers everything you define for the project as a whole

<sup>12</sup> The file is a plain JSON file and can be inspected with any text editor, even though the Content Library hides it if you look at the directory. The `.dclp` extension is an acronym for “Daz Clothing Project”.

and the clothing pieces in it, as well as your progress.

1. The buttons in the top row of the “Seamster” pane (figure 7) operate on such clothing projects. To **start a new project**, click on the “New” button. A window pops up asking you for the content library, target Genesis figure, your author name and the new project name. When you click on “Create”, the directory and the .dclp file will be created accordingly.

You can save, close and reopen the project with the other buttons in the same row. For your convenience, the “Eye” button will open the project directory in the Content Library pane.

2. You can then **add OBJ files** from your clothing modeler into the project.<sup>13</sup> Press the “(+)” button above the “Clothing pieces” table (figure 8) and browse to the OBJ file you exported from your modeler (e.g. Marvelous Designer). The OBJ will be added to the table, and you can set the basic attributes for it using the controls under the table (see [Clothing piece attributes](#), page 7).



**Figure 7:** Project buttons, from left: New, Open, Save, View in Content Library, Close.

<sup>13</sup> See [OBJ](#) in the glossary for details about this file format.



**Figure 8:** Clothing piece buttons, from left: Add, Auto-rig OBJ, Edit material names, Save support asset, Load support asset, Save wearable, Load wearable, Weights and correctives, Rigid follow nodes, Remove asset from project.

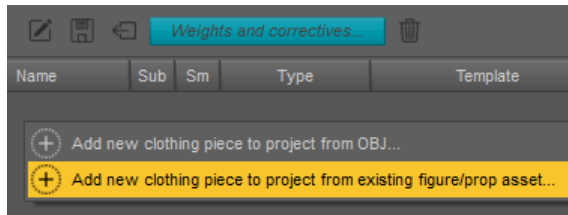
Initially the name of the new asset will be set to the filestem of the OBJ file you imported, but you can **change the asset name** by double-clicking on it in the table. (This will also be suggested for the asset file name on disk later.)

3. You can then **import and convert the OBJ** of an asset by selecting it pressing the “**Auto-rig OBJ**” (lightning bolt) button (figure 9) above the “Clothing pieces” table. (You can also double-click on the asset.) See [OBJ import and conversion to a rigged figure](#), page 8 for the exact steps that Seamster performs here.
4. The first time you convert an OBJ, Seamster automatically opens the “**Edit surface/material names**” window where you can rename the OBJ’s materials (surfaces) if the defaults created by your modeler are less than optimal; see [Renaming surfaces/materials](#), page 8. You can open this window again at a later time with the “Edit” (pencil) button above the assets table.
5. After the converted OBJ has been processed, the “**Save asset**” button (floppy disk) button becomes active, allowing you to save your clothing piece as a Daz Studio figure/prop asset; see [Saving the figure/prop asset](#), page 9.

Note: If you already have a figure/prop asset that you saved earlier and you would like to add it to a Seamster clothing project without having to go through the OBJ import again, you can right-click on an empty space of the clothing pieces table (if a clothing project is already open) and select “Add new clothing piece... from existing figure/prop asset...” (see figure 10).



**Figure 9:** “Auto-rig OBJ” button.



**Figure 10:** Context menu for adding an existing figure/prop asset to a project.

## 5 Clothing piece attributes

To be of any use, a Seamster project should contain at least one clothing piece, but can contain as many as you want. For example, you can create an outfit with four pieces: a shirt, pants, a hat and shoes. Each “clothing piece” will later become a Daz Studio figure/prop asset.<sup>14</sup>

Each clothing piece in a project has the following attributes, which are editable under the “Clothing pieces” table when a clothing piece is selected:

- **Subdivision level:** If set, Seamster will automatically add a subdivision modifier after importing the OBJ and set its subdivision level to what you specify.
- **Smoothing modifier:** If set, Seamster will automatically add a smoothing modifier to the imported OBJ.
- **Content type:** Specify the content type<sup>15</sup> that the figure/prop asset will have. It is highly recommended to set this correctly (e.g. to “Follower/Wardrobe/Shirt” if you are making a shirt piece) for the rest of Daz Studio to treat your asset correctly. What you specify here will be passed to Transfer Utility and also to the “Save figure/prop support asset” dialog.
- **Projection template:** Specify the projection template that Seamster should tell the Transfer Utility to use during import. These templates come with the “Genesis Essentials” packages from Daz and instruct Transfer Utility how to treat weightmaps and other information when converting a prop to a rigged figure. They also create automatic morphs in the target clothing piece.

When you select “None”, the clothing piece will receive a copy of the weights of the Genesis figure. This is rarely the best solution since even for a dress that follows the body shape closely, the weights will need to be lessened for the hips. Depending on the type of clothing piece you are making, there will probably be a better fit.<sup>16</sup>

- Once you have saved out the clothing piece as a **figure/prop asset** (see [Saving the figure/prop asset](#), page 9), Seamster remembers the asset’s path so you can load it again quickly by double-clicking on the item in the table.
- Once you start **weightmapping** the asset (see [Weightmapping](#), page 11), Seamster also remembers which bones you already worked on and saves this information with the project data.

<sup>14</sup> To avoid any confusion, Seamster avoids the term “asset” with its own data until a piece has been saved out as a figure/prop asset. See also [asset](#) in the glossary.

<sup>15</sup> See [content type](#) in the glossary.

<sup>16</sup> Note also that the projection templates for Genesis 9 include a lot more functionality compared to those for Genesis 8 and 8.1, including adjustment and corrective morphs (see [Corrective morphs](#), page 14), so picking a specific template makes a bigger difference than previously. By contrast, the templates for figures before Genesis 9 create only a single “Expand all” morph; see Esha ([Content mastery 6](#)), 6a/17:43; Mada ([YouTube: Sandals](#)), 0:40.



- If you use Seamster to add “**rigid follow nodes**” to a clothing piece (see [Rigid follow nodes \(RFNs\)](#), page 19), Seamster will also store information about those with the project data.

## 6 OBJ import and conversion to a rigged figure

When you press the “Auto-rig OBJ” button (lightning bolt, figure 9), Seamster imports the OBJ of the asset into the scene and converts it into a rigged figure. In detail:

1. If the scene does not already contain a **Genesis figure** of the type you defined when you created the project, Seamster will load the “development” version of that figure (e.g. “Genesis 9 Dev Load” if you selected Genesis 9).
2. Seamster will then **import the OBJ** as a prop with the default Daz Studio OBJ import settings (which you can adjust).
3. Seamster will add **subdivision and smoothing modifiers** to the prop, if you enabled those for the clothing piece.<sup>17</sup>
4. It will then turn the prop into a rigged figure by transferring the bones and weightmaps from the Genesis figure to the clothing piece. To do this, Seamster silently runs Daz Studio’s “**Transfer Utility**”, giving it the projection template that you specified for the asset on the pane.<sup>18</sup>
5. Seamster automatically fixes the user-facing **morph labels** created by Transfer Utility from the projection templates. For each generated morph, Transfer Utility normally sets the morph *label* to the morph *name*, resulting in labels like `body_bs_LoosenWaistUpper`. Seamster automatically fixes that, for example to “Loosen Waist Upper”.
6. Seamster will then set up **subdivision and smoothing** again on the new rigged figure that Transfer Utility created, if you enabled those for the clothing piece (Transfer Utility doesn’t).

After this, the rigged clothing piece has become a child node of the Genesis figure and is ready to be saved as a figure/prop asset (see [Saving the figure/prop asset](#), page 9).

## 7 Renaming surfaces/materials

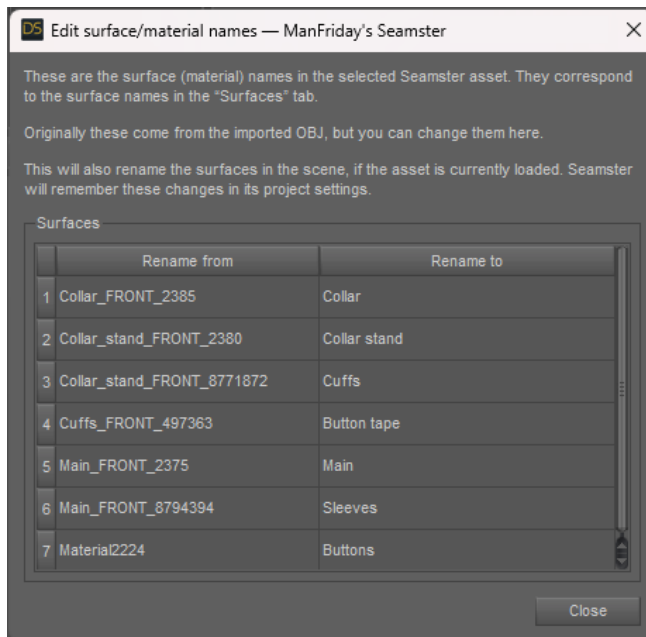
After an OBJ has been converted to a rigged figure for the first time with the “Auto-rig OBJ” (lightning bolt) button (figure 9), Seamster automatically brings up its “**Edit surface/material names**” window. (You can bring this up at any time yourself by pressing the “Edit” button above the clothing pieces table.) The two columns in the table fill with the material names that were found in the OBJ. Marvelous Designer, for example, exports OBJs with rather unwieldy names as shown in figure 11, which you probably want to change.

While you could rename materials yourself in the “Tool Settings” of Geometry Editor, Seamster makes this a bit easier: it automatically switches to the “Surface Selection” tool when the window is open, and you can select

<sup>17</sup> Daz recommends that if a clothing piece is intended to have a subdivided mesh, subdivision should be turned on before running Transfer Utility for better results.

<sup>18</sup> Without Seamster, you would have to run Transfer Utility yourself by going to “Edit” → “Figure” → “Transfer Utility”.





**Figure 11:** Renaming surfaces/materials.

and change the material names directly in the right column (double-click on an item or press F2 to start editing). The change is immediate. The old and new material names are then saved with the Seamster project file and will automatically be applied again if you import the OBJ with Seamster again later.

## 8 Saving the figure/prop asset

After you have imported an OBJ prop and converted it to a rigged figure, you can save out the resulting asset as a Daz Studio “figure/prop support asset”. You could use the standard Daz Studio menu item<sup>19</sup> for that, but Seamster makes this a bit easier as well. Clicking on the “Save asset” (floppy disk) icon brings up

1. a file picker dialog in the project directory with the clothing piece label as the suggested .DUF file, which you can just accept if you’re happy with that;
2. after that, Daz Studio’s own “Figure/prop asset save options” dialog, but with all fields pre-filled with the data from the Seamster project and asset.

Normally you can just use the suggested settings, and you’re done. Seamster then remembers that you have saved out the asset already and will enable the “load support asset” button above the “Clothing pieces” table for it, which quickly loads the support asset onto the figure, as if you had double-clicked on it in the Content Library.

Once you have saved the clothing piece as a figure/prop asset in Seamster, you can also click on the “Save” button above the table to quickly save it again with the same settings (which you should do often as you continue to work on weightmapping and corrective morphs).

<sup>19</sup> “File” → “Save” → “Support asset” → “Figure/prop assets”.

By default, Seamster implements **incremental saving with versioning** for your figure/prop assets. The most current version will always be in the file with the name that you picked when you first saved the asset. Every time you save the asset again, Seamster will rename the previous version to have a date/timestamp suffix, e.g. `Shirt_2024-01-02_17-42-59.duf`.<sup>20</sup> This allows you to restore a previous version of the figure/prop asset in case you made a mistake, or the figure/prop asset broke in some other way: rename the suffixed files and directories to the original files, and you should be back to the previous version.

By default, Seamster will keep the last three versions of your asset; you can change the number of versions to keep, or turn off incremental saving altogether, in the Seamster preferences ([Seamster preferences](#), page 21).

## 9 Details about Daz Studio figure/prop assets

Generally, in Daz Studio, **support assets** have data in the hidden `/data` directory of a content library: geometry, UVs, morphs. This is why when you load a figure or a piece of clothing into a scene, their geometries are not copied into the scene (keeping the scene file small), but are loaded as needed from the hidden `/data` directory by every scene file that uses them.

In the case of a *figure/prop* support asset, the file you picked when you saved the asset will be the small `.duf` that is visible in the Content Library (for clothing, for example, under `/People/Genesis 9/Clothing/<Artist>/<Product>/<Filestem>.duf`). Behind the scenes however, Daz Studio creates bigger `.dsf` files under `/data` in the same content library, which are referenced by the `.duf` (and by every scene that uses the asset).<sup>21</sup>

Among the files under `/data` are:

1. `/data/<Artist>/<Product>/<ItemName>/<Nodename>.dsf`, containing the geometry (see below for details about the naming);
2. a `Morphs` subdirectory with more subdirectories containing all the asset's morphs, including corrective morphs;
3. a `UV Sets` subdirectory with more subdirectories containing the UVs.

The “**artist**” and “**product**” are defined when you create a new Seamster project, and Seamster will create the project's parent directories accordingly. Seamster will copy these into the “Figure/prop asset save options” dialog when you save a figure/prop asset, so a matching path under `/data` will be used for the support files as well, unless you change these during save.

Regarding the *clothing piece names*, things are more complicated:

1. Seamster maintains a **clothing piece label** in the “clothing pieces” table (e.g. “My pants”) and saves it with the project file. Initially Seamster uses the filestem of the imported OBJ. When you load the clothing piece OBJ into the scene using “Auto-rig OBJ”, Seamster will set the **scene node label** in the “Scene” tab to this label (e.g. “My pants” again). Seamster will keep the labels in sync: it will update the clothing piece when you rename the scene node, and vice versa.<sup>22</sup>
2. Additionally, each such node as an **internal node name** that is only visible if you inspect the node's scene identification data.<sup>23</sup> For this,

<sup>20</sup> Seamster will use the same suffix to rename the “item name” directory for your figure/prop asset under `/data`. See the next section for details about these files.

<sup>21</sup> See **asset** in the glossary. Note that Daz Studio treats figures and props the same way in this regard; the only difference is that a “figure” in this sense includes rigging (see **figure** in the glossary).

<sup>22</sup> Note that the label is only displayed to the user in the “Scene” tab; it is not used for naming any asset files.

<sup>23</sup> “Edit” → “Object” → “Scene identification”.

Daz Studio initially uses the label with a polycount suffix (e.g. “My pants\_35966”).<sup>24</sup>

- When you save the figure/prop asset, in the “save as” dialog that pops up first, you can specify an arbitrary **DUF filename** (where Seamster again suggests the clothing piece label). This is what the user will see in the Content Library to load the asset.
- In the “Figure/prop asset save options” dialog that pops up second, the “**item name**” will become part of the DSF path: Daz Studio will create the DSF as /data/<Artist>/<Product>/<ItemName>/<Nodename>.dsf, as indicated above.<sup>25</sup>

So while Seamster suggests the same name for “Seamster clothing piece label”, “Daz Studio node label”, “Daz Studio internal node name” (plus suffix), “user-facing DUF asset name” and “figure/prop asset item name” and tries to keep the first two in sync, the others can theoretically be different.

## 10 Weightmapping

### “Weights” tab in Seamster

After you have saved a clothing piece as a figure/prop asset, you can switch the “Seamster” pane to weightmapping mode (figure 12) by pressing the “**Weights and correctives...**” button.<sup>26</sup> Leave the “Correctives” tab at the bottom right alone for now; this is for creating corrective morphs later.<sup>27</sup>

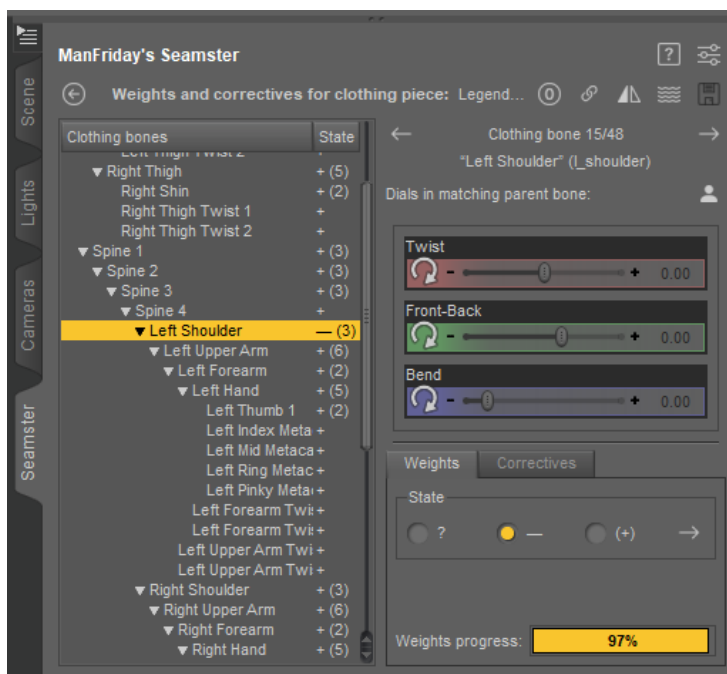
You can **exit weightmapping mode** by clicking on the “arrow left” circle at the top left, and re-enter weightmapping mode again later.

<sup>24</sup> Note that Daz Studio does *not* adjust the node name when you change a node’s label in the “Scene” tab.

<sup>25</sup> If incremental saving is enabled in the Seamster preferences (which is on by default), Seamster will add the version suffix to the “item name” as well, resulting in one directory under /data for each version of the asset.

<sup>26</sup> For an introduction how these weights work, please see [What is rigging?](#), page 2.

<sup>27</sup> See [Corrective morphs](#), page 14.



**Figure 12:** Seamster in weightmapping mode.

In weightmapping mode, Seamster shows you a copy of the bone hierarchy of the *clothing piece* as a tree on the left. Initially this will be the bones that Transfer Utility copied when you converted your OBJ to a rigged figure.

Seamster also switches the active Daz Studio tool to the weightmapping brush for your convenience. If you are not familiar with this tool, there is an introduction in the next section ([Daz Studio's weightmap brush tool](#)).

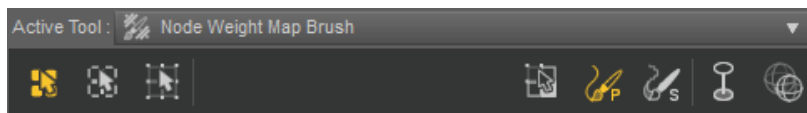
When you click on one of the bones in Seamster's list, Seamster selects that clothing bone in the scene and shows you the rotation dials of the parent *Genesis figure* bone at the top right. You can then change those dials to move the Genesis bones to test how your weightmaps in the clothing react to bone movements.<sup>28</sup>

You can also click through the bones of the clothing piece with the “next” and “previous” buttons at the top right. After you are done weightmapping a bone, you can set its state to “Done”, which will advance the progress bar at the bottom of the pane. Seamster remembers your weightmapping progress for each clothing item and saves it with the clothing project. This allows you to mark bones that need fixing and work through them later.

Typical problem areas which often need fixing include the abdomen (lower and upper) and everything around the shoulders (especially armpits).

Be sure to **save** your clothing asset regularly in order not to lose your work if Daz Studio crashes. Seamster has a one-click “Save” button at the top right of the weightmapping page which saves both the figure/prop asset again (including all the weightmaps) and the current Seamster project (so that Seamster will remember your weightmapping progress).

## Daz Studio's weightmap brush tool



Here are some helpers to get you started with the “Node Weight Map Brush” tool, as Daz Studio calls it, which allows you to edit the weightmaps associated with the currently selected scene node.<sup>29</sup>

When rigging clothing, you do not normally have to *add* weightmaps manually. When you select a bone of a recent Genesis figure, the “General Weights” map is already there and will have been transferred to the clothing when you imported your clothing piece with Seamster.<sup>30</sup>

When the weightmap brush tool is active, there are a number of buttons at the top of the “Tool Settings” (figure 13).<sup>31</sup> Regarding the buttons on the right:

1. When “**geometry selection**” is active, the brush selects geometry like in Daz Studio's Geometry Editor. Depending on which of the three buttons on the left is selected, you can select polygons (faces), edges or vertices.

This is useful when you want to edit weightmaps very precisely: for example, you can select some faces and then, in the “**Weight editing**” submenu of the weightmap brush's options menu, select “Fill selected” or “Smooth selected”. (More about that menu below.)

<sup>28</sup> Traditionally, without Seamster, one would have to keep alternating between selecting a *Genesis figure* bone in the Scene tab to dial in a rotation and selecting the matching *clothing* bone again to work on the clothing bone's weightmap, which can become very tedious for dozens or hundreds of bones. Seamster also automatically switches to the weightmap brush tool and selects the “General Weights” map to save you from having to go to the “Tool Settings” pane and selecting it by hand.

**Figure 13:** Weightmap brush tool options.

<sup>29</sup> For a video tutorial about weightmapping without Seamster, see Mada ([YouTube: Tight top](#)). Seamster eliminates many of the steps shown there, but the tutorial still shows instructively where and how to fix weightmaps using Daz Studio's tool, which hasn't changed. The process to check the bones and change the weightmaps starts at about 4:00.

<sup>30</sup> Daz Studio uses weightmaps in several contexts, but most importantly, a Genesis (3 or newer) figure has one weightmap for every one of its bones that determines which vertices get moved when the bone is rotated. Do not try to add “TriAx” weights, those are used for older figures. Again, see [Weights and correctives](#), page 3.

<sup>31</sup> For details, please refer to the Daz documentation: [http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport\\_tools/weight\\_map\\_brush/editing/weight\\_map\\_direct\\_editing/start](http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport_tools/weight_map_brush/editing/weight_map_direct_editing/start).

2. Most frequently however you will use “**paint brush**” (**P**) mode. Here the mouse pointer becomes a circle and you can paint weights onto the mesh directly (figure 14): where you drag the mouse, the weights will increase to a maximum of 100%.

The inner circle of the paint brush will have the most strength during painting; areas outside of the outer circle will not be affected at all. The area between the inner and the outer circle will have a “falloff”, where the strength decreases with the distance from the center.

If you hold down the **Alt** key while painting, the weights under the mouse will be reduced instead. If you hold down the **Ctrl** key, the brush will temporarily smooth instead (see below).

3. In “**smoothing**” (**S**) mode, the circle smoothes by default, averaging the weights within the circle. (If you hold down Ctrl, it paints like the paint brush instead; if you hold down Alt, it subtracts weights, like the paint brush with the Alt key pressed.)

In other words, smoothing can be done in two ways: with the regular paint brush, by holding down Ctrl while painting; or by switching to the smooth brush, which then smoothes without the Ctrl key held.

Note that any given vertex of a figure may be affected by several bones and thus several weightmaps, but the total combined contribution of all bones to an individual vertex cannot exceed 100%.<sup>32</sup> So when you add or smooth weights, your painting may cause Daz Studio to automatically reduce weights in bones other than the currently selected one.

## Special twist bones in Genesis 9

Genesis 9 changed many of the bones compared to Genesis 3 and 8 for better compatibility with game engines and other 3D software.<sup>33</sup> It also combined the separate “Bend” and “Twist” bones that Genesis 8 had for some joints into one. Instead, it introduced other “Twist” bones without children which hold the weightmaps instead of the “real” bones in the hierarchy.<sup>34</sup> If you select one of the bones that have no weightmap, Seamster will tell you which other bone to use instead.

This applies to the following (and all the matching “Right” bones):

Bone without map	Bones which have maps instead
Left Upper Arm	Left Upper Arm Twist 1 + 2
Left Forearm	Left Forearm Twist 1 + 2
Left Thigh	Left Thigh Twist 1 + 2

Each bone on the left of the table (without a map) is hooked up to move the two bones on the right, which have a weightmap each, but are triggered at different rates (different factors) to make the bone rotation look more natural.

## Tips and tricks

- The weightmap brush has many additional options,<sup>35</sup> which can be ac-

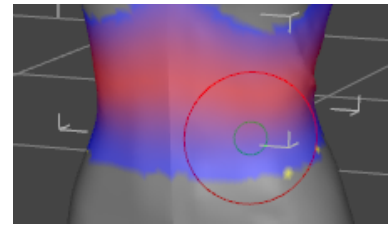


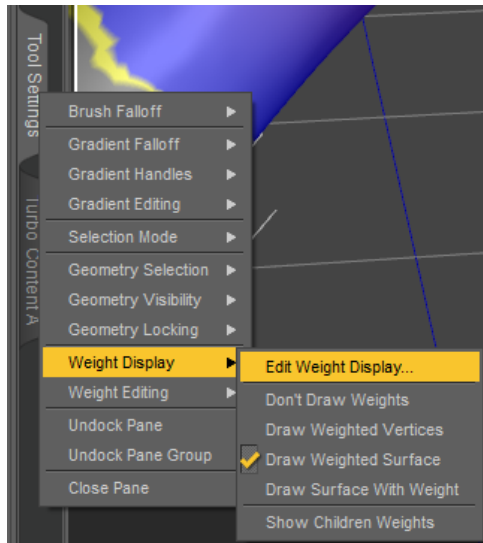
Figure 14: Paint brush mode.

<sup>32</sup> [http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport\\_tools/weight\\_map\\_brush/editing/start](http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport_tools/weight_map_brush/editing/start).

<sup>33</sup> Esha (*Genesis 9*), 08:00.

<sup>34</sup> For example, instead of two bones for “Left Shoulder Bend” and “Left Shoulder Twist” in G8, there is now only one bone in the hierarchy (“Left Upper Arm”); the additional twist bones for the weightmaps have been taken out of the hierarchy and are children of the “forearm” instead. The twist bones are children of the forearm but not parents of the hand; Esha (*Genesis 9*), 06:00.

<sup>35</sup> For details, please refer to Daz’s documentation: [http://wiki.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport\\_tools/weight\\_map\\_brush/tool\\_settings\\_option\\_menu/start](http://wiki.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport_tools/weight_map_brush/tool_settings_option_menu/start).



**Figure 15:** Options of the weightmapping brush.

cessed either by right-clicking into the viewport (figure 15) or by right-clicking on the “Tool Settings” tab.

- Using the mouse wheel with the Ctrl key held down increases or decreases the radius of the brush. Alternatively you can set the radius on the “Tool settings” pane when the weightmap brush is active.
- The “Sensitivity” value on the “Tool settings” pane determines how much strength the brush exerts on the weights under it. Sometimes it helps to lower the default value of 0.3 to 0.2 for more control.
- If you need very detailed information about specific weights, you can do the following (see figure 16):
  1. With the weightmap brush active, choose “Geometry Selection” (Shift+L).
  2. Switch to vertex selection.
  3. In Tool Settings, go to “Selection Analysis” tab and it will display which maps include the selected vertices.

Item	Value
▼ Chest Lower	
▼ General	
23325	0,00369268
23524	0,00131228
23526	0,00296025
▼ Abdomen Upper	
▼ General	
23325	0,295598
23524	0,121141
23526	0,249226
▼ Pelvis	
▼ General	
23325	0,0527504
23524	0,0814832
23526	0,057557
▼ Abdomen Lower	
▼ General	
23325	0,647959
23524	0,796063
23526	0,690257

**Figure 16:** Analyzing weight values.

## 11 Corrective morphs

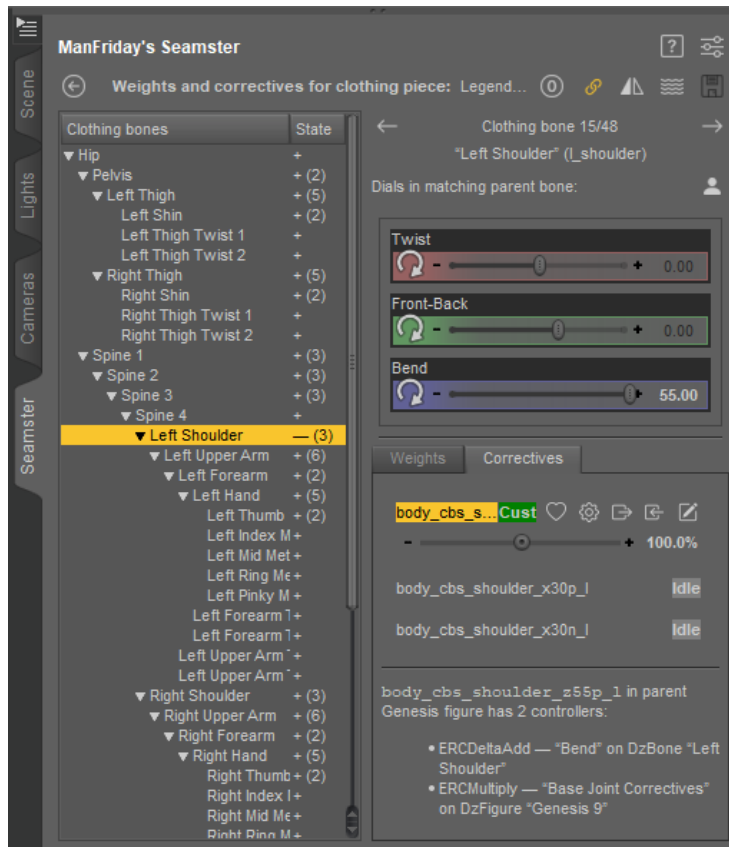
After you have pressed the “Weights and correctives” button on the main pane for a clothing piece, which switches Seamster into “Weights and correctives” mode (see [Weightmapping](#), page 11), you can select the “Correctives” tab at the bottom right, which will show you the corrective morphs that exist in the clothing node for the selected bone, and allow you to modify them.

For an introduction how correctives work, please see [What is rigging?](#), page 2. As noted there, a clothing piece can have many corrective morphs for a single bone, and there is a formula behind each corrective that determines under what conditions it gets activated.

When you select a a clothing bone with the “Corrective” tab active, Seamster shows you the correctives that reference the selected bone in their formulas and color-codes their states:

- If the corrective exists only in the parent figure but has not yet been injected into the follower (the clothing piece), it is marked “**Idle**” and gray.
- If the corrective has been dynamically injected into the follower, for example because you moved the bone dial it is linked to, it is marked “**Gen**” (for “generated”) and yellow.<sup>36</sup>
- Custom correctives that already live in the clothing piece are made as “**Custom**” and green.

<sup>36</sup> As noted in [What is rigging?](#), page 2, generated morphs are temporary and will disappear again if the clothing piece is deleted and reloaded. Dynamically generated morphs will never be saved with the clothing piece.



**Figure 17:** Example “custom” corrective morph.

You can edit corrective morphs in the “generated” and “custom” states because those have active morph deltas in the clothing piece. If a morph is still “idle”, find a dial that activates it and turn it up until the morph turns to “generated”.

For example, with Genesis 9, the `body_cbs_shoulder_z55p_1` corrective is linked to the “ZRotate (Bend)” property of the “Left Shoulder” bone, but with a factor: the corrective is dialed to 100% if the bend value is 55, and it ignores negative values (see figure 17).

When you start working on a piece of clothing after importing it for the first time, the correctives in it depend on the target Genesis figure and the projection templates you specified for Transfer Utility:

- With a clothing piece for **Genesis 8**, the standard projection templates contain no correctives at all. So, correctives only start showing up in the clothing piece once Daz Studio injects them into the clothing piece as you

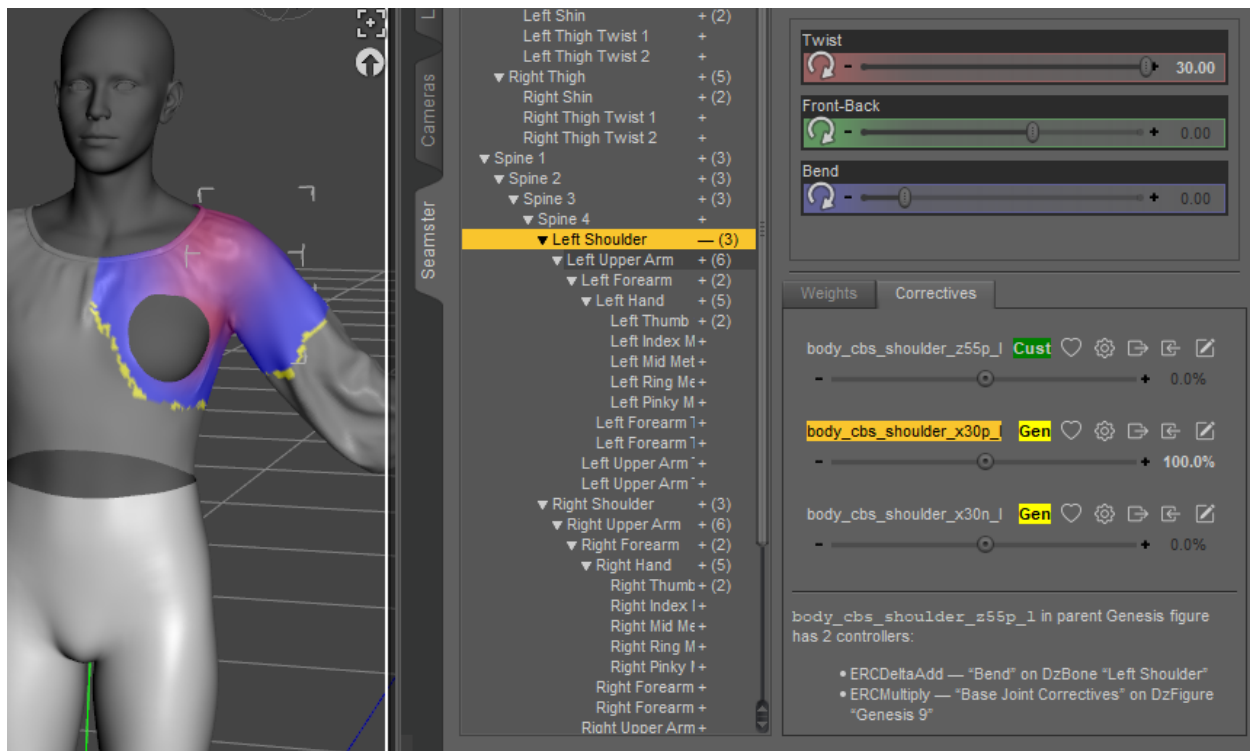


move the individual dials for the bones of the Genesis figure, and they will be gone again if you reset the bone dials and load the clothing asset a second time.

- By contrast, with **Genesis 9**, the projection templates already come with a number of corrective morphs to make rigging easier. As a result, your clothing piece will already contain some correctives in “Custom” state even if you haven’t moved any bones yet.

A typical example is shown in figure 18. Dialing up the “Twist” rotation in “Left Shoulder” has created some poke-through that cannot be fixed by painting weightmaps alone. This rotation has turned on the `body_cbs_shoulder_x30p_1` corrective to 100%, and a corrective got injected into the clothing piece. However, it is not good enough and needs to be fixed.

It is of course possible to fix a corrective morph without Seamster, but getting the exact process with all the required settings right is tedious and error-prone. Below are the steps how Seamster makes this easier.



**Figure 18:** Fixing one corrective morph.

## Fixing a corrective with Blender

To edit a corrective morph in **Blender**, you can export an OBJ and import it again to create a morph. Without Seamster, you would have to do many steps manually and eventually use Morph Loader Pro.<sup>37</sup> The default settings in Seamster for fixing a corrective are designed to work with Blender and reduce the process to the following steps:

1. Make sure the corrective morph you want to edit is dialed up to 100% and click on the “**Export OBJ**” button in its row in Seamster. This will export the clothing shape as an OBJ, in its current pose, which includes

<sup>37</sup> See **Morph Loader Pro** in the glossary.

the morph, so you can then import it into Blender. By default this will also export the posed parent Genesis figure as a second OBJ so you can see the interaction between the two in Blender, but you will only want to re-import the *clothing* OBJ later.

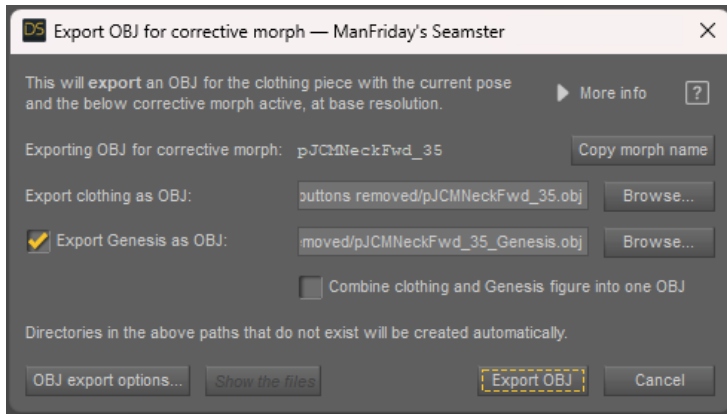


Figure 19: Export window.

The button brings up the window shown in figure 19. This might seem complicated, but normally you can accept all options and press the “Export OBJ” button.

After the export, the window stays open and the “Show files” button becomes active, which opens the exported files in Windows Explorer for your convenience.

By default, Seamster will create a “Correctives” subdirectory in your clothing project, with another subdirectory for the clothing piece, and write the OBJ under the morph name there.<sup>38</sup>

The OBJ export options are designed for work well with Blender, including a correct scale, but you can change them with the “OBJ export options” button. Seamster will remember these OBJ export options between sessions and keep using them.<sup>39</sup>

2. Import both OBJs that you just exported into Blender and fix the problem in the geometry; see [Required import and export options in Blender](#) below.
3. When you are done, export the clothing OBJ from Blender again (but not the Genesis figure). If you overwrite the OBJ that you just imported, Seamster will be able to use that new OBJ in one click because it remembers the OBJ name between the export and import dialog windows.
4. The fresh OBJ now contains the shape that the clothing *should* have. If you import this new OBJ again with the **Import OBJ** button of the corrective in Seamster (figure 20), Seamster will replace the existing corrective morph with a new one that contains the changes you have made.<sup>40</sup>

## Required import and export options in Blender

For Daz Studio’s Morph Loader Pro to be able to re-import the OBJ that was exported in Blender and create a morph, it is essential that certain options

<sup>38</sup> You can change this in Seamster’s “Preferences” window (see [Seamster preferences](#), page 21); if you prefer to keep the corrective OBJs outside of your Daz Studio content library, you can specify a different path there.

<sup>39</sup> Seamster keeps a separate set of these settings for exporting and importing correctives, with defaults that work well for Blender. These are independent of the regular OBJ export and import settings, and also of the settings that Seamster uses for importing clothing pieces; see [OBJ import and conversion to a rigged figure](#), page 8.

<sup>40</sup> In order to do that, Seamster silently runs Daz Studio’s Morph Loader Pro tool with the correct settings; see [Morph Loader Pro](#) in the glossary.

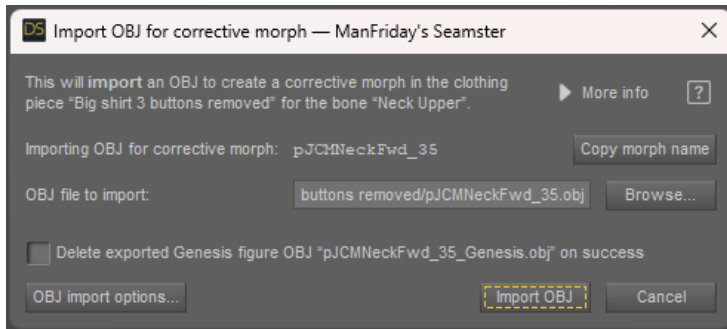


Figure 20: Import window.

be set correctly in Blender's OBJ importer and exporter. Otherwise Morph Loader Pro will fail with "Geometry did not match".

- With the legacy OBJ importer/exporter in **Blender versions before 4.0**,<sup>41</sup> it is critical that "**Keep Vert Order**" and "**Poly Groups**" be enabled on both import and export. While Blender remembers these settings in the same session (if you import / export several OBJs), unfortunately it does not remember them between restarts; you have to re-enable them every time after you have opened Blender. Blender allows for defining saving the current import and export settings as "operator presets" in these dialog windows, which saves saves you a few clicks when doing it frequently.
- When **exporting** from Blender, even with Blender 4.0, you must turn on "**Limit to Selection Only**" if you have multiple items in the scene (e.g. the Genesis figure and the clothing piece), or they will all be exported in the same OBJ (causing Morph Loader Pro to fail). Also make sure the clothing piece is selected instead of the Genesis figure.

## Fixing a corrective with ZBrush

Many artists prefer to work with ZBrush. The following steps will work:

1. Use GoZ to bring both the Genesis figure and the clothing piece into ZBrush.
2. Make the required changes.
3. Export the clothing piece as OBJ from ZBrush to a directory on disk.<sup>42</sup> As the filename, use the morph name; you can right-click on the corrective in Seamster and select "Copy morph name... to clipboard".
4. Click on the "Import" button, then "Browse" and navigate to the OBJ you just exported. You probably will have to adjust the OBJ import settings once to work with ZBrush as the defaults are designed to work with Blender.

## Creating custom correctives from scratch

Sometimes even the many correctives that Daz Studio provides are not enough, and a new corrective needs to be created from scratch, with a custom formula that differs from the predefined ones. Seamster does not yet

<sup>41</sup> For many years Blender had a slow and cumbersome OBJ importer and exporter written in Python. Blender 3.2 received a new importer/exporter written in C++ that is significantly faster and gets vertex order right without special options (see <https://aras-p.info/blog/2022/05/12/speeding-up-blender-obj-import/>; [https://wiki.blender.org/wiki/Reference/Release\\_Notes/3.2/Pipeline\\_Assets\\_IO](https://wiki.blender.org/wiki/Reference/Release_Notes/3.2/Pipeline_Assets_IO)). The old importer/exporter was then renamed "legacy" and removed entirely with Blender 4.0 ([https://wiki.blender.org/wiki/Reference/Release\\_Notes/4.0/Import\\_Export](https://wiki.blender.org/wiki/Reference/Release_Notes/4.0/Import_Export)).

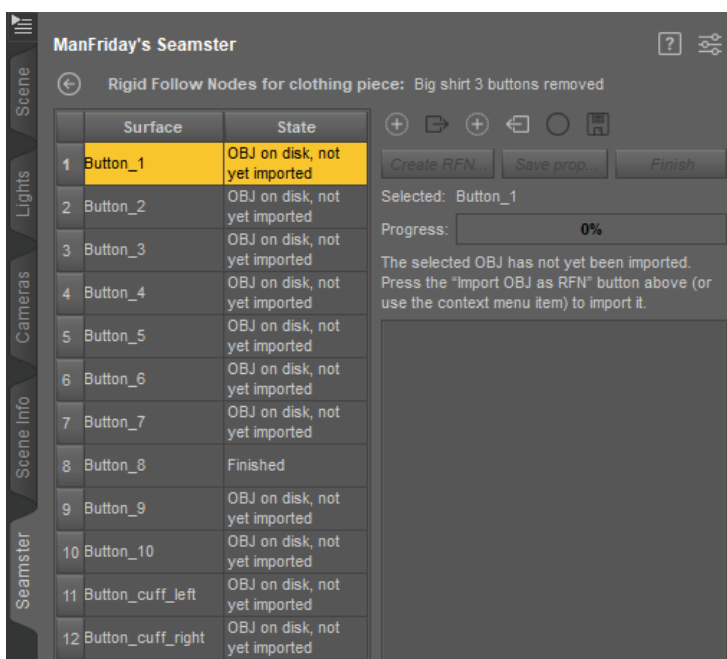
<sup>42</sup> Many artists do not use GoZ to bring geometry back from ZBrush into Daz Studio as it has been proven to be somewhat unstable, and prefer a manual transfer via OBJ.

provide a user interface for that scenario, so for the moment you will have to go through the traditional process using Morph Loader Pro and Enhanced Remote Controls (ERCs).<sup>43</sup>

## 12 Rigid follow nodes (RFNs)

In Daz Studio, rigid follow nodes are part of a mesh that are taken out of the ordinary rigging so that they stay rigid when other parts of the mesh are bent / rotated. Examples are buttons: the cloth around the button should follow the body and stretch with it, but the button itself should only move, but not shrink or grow.<sup>44</sup>

When you press the “Rigid follow nodes” button for a clothing piece, Seamster switches into yet another mode (figure 21).



<sup>43</sup> There are many tutorials for this, for example Esha (*Content mastery 6*), 6a/1:30:00.

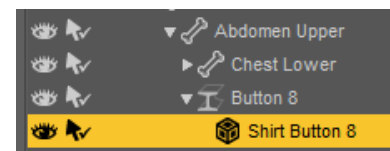
<sup>44</sup> For a great introduction to RFNs and how to make them without Seamster, please see Mada (*YouTube: Rigid follow nodes: Belt buckle*). However please note the Seamster saves you many of the steps explained there.

**Figure 21:** Seamster in “Rigid follow nodes” mode.

Since RFNs can only be loaded by a wearable preset, before entering this mode, you must save a wearable preset for your clothing piece first: right-click on the clothing piece and select “**Save wearable preset as...**”. This way Seamster can automatically save all of the parts of your clothing piece while you are working on your RFNs. (Also, the wearable preset must currently be loaded into the scene.)

In Daz Studio, each RFN consists of three things (see figure 22):

- the actual “rigid follow node”, which is marked with a bar in the scene;
- under it (parented to it), a prop asset which contains the geometry for the node (for example, a button). This geometry must come from a separate prop asset, which is usually first created as an OBJ. In other words, there must be one OBJ for each RFN. We will call this the “**RFN mesh object**” from now on.
- Additionally, Daz Studio saves with each RFN which polygons of the owning mesh the RFN should stick to, and some rigging information. This information is saved in the wearable, not in the RFN.



**Figure 22:** Rigid follow node and prop parented to bone in the scene.

With Seamster, there are two ways to create RFNs:

- If you have an existing clothing piece where the geometry for the RFNs is part of the clothing mesh, Seamster allows you to extract these pieces as separate OBJs before defining them as RFNs.

For this to work, each future RFN must be a separate surface in the mesh: for example, there must be “Button 1”, “Button 2” etc. surfaces in the mesh for the extraction to work. If needed, use Daz Studio’s Geometry Editor to create new surfaces and reassign polygons.

To create RFNs this way, right-click on an empty space of the RFN table and select “**Add RFN from surface...**” (see figure 23). A window pops up showing all the surfaces of your clothing piece, and you can select one or several to be exported. Seamster will then write the OBJs to a subdirectory called “RFN OBJs” under your project directory (which will be created if it does not exist). Each OBJ path is remembered with the project settings so you can import it again with one click.

- If you already have separate OBJs for your RFNs because you planned this process from the beginning, you can import those directly without having to extract them first. Select “**Add RFN from OBJ...**” and pick an existing OBJ from disk, whose path will also be remembered with the project settings.

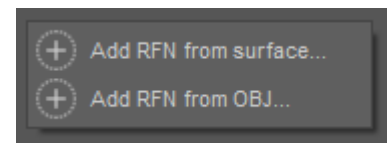
Either way, the OBJ will display “OBJ on disk, not yet imported” in its “State” column. From here, follow the following steps:

1. Select “**Import OBJ as RFN**” for the selected table row (either by pressing the button on the top right, or selecting the context menu item), and Seamster will import the OBJ into the scene.

After import, Seamster will change the label of the new scene node to the “surface name” shown in the table. Since for every RFN there will be two scene nodes (the mesh object and the actual RFN), which cannot have the same label (e.g. “Button 1”), many artists prefix the mesh node with a product acronym to differentiate them (e.g. “ABCD Button 1”). You can click onto the surface name in the table, and it becomes editable.

After this, the table row will display “Imported OBJ, needs RFN”.

2. **Select polygons** in the clothing piece mesh (not the imported OBJ!) to which the RFN should be attached. To make this easier, Seamster switches the active tool in Daz Studio to the Geometry Editor and selects the clothing piece in the scene. You can simply click on the polys. A good rule of thumb for this is to try and select the polygons “under” the RFN prop so that their total surface is not much bigger than that of the RFN mesh object over it.<sup>45</sup>
3. Press the “**Create RFN...**” button at the top right. This will bring up a window where you can enter the name of the future RFN, for example “Button 1”.
4. After that, the “**Save prop...**” button becomes active which allows you to quickly save the imported OBJ as a prop asset. This brings up the standard



**Figure 23:** Context menu to add a new RFN.

<sup>45</sup> See Mada (*YouTube: Rigid follow nodes: Belt buckle*) 3:26. But do not use the context menu as shown in the video, use Seamster’s button instead.

Daz Studio “Save figure prop asset” dialog with meaningful values filled out, which you can normally simply accept.

5. Seamster switches the active tool to the Joint Editor now to allow you to do two things:
  - a) **Fix the rotation axes** of the imported OBJ so that it rotates correctly. The center of the green arrows should be in the center of the RFN mesh object, the red error should be “outside” it to define the rotation axis of the RFN (figure 24). When you look straight at the RFN, the two points should align (figure 25). Then test the rotation dials of the RFN mesh node to see whether the RFN rotates correctly.
  - b) Right-click into the scene and select “Align” → “Align Node”.
  - c) In the Seamster pane, Seamster once again displays the tree of bones of the clothing piece. **Select the bone** whose influence area is “under” the RFN (e.g. “Chest Upper” for a typical shirt button). Change the dials of that bone to make sure the RFN moves correctly with the clothing mesh.
  - d) Then press “Finish”.

The RFN now displays “Finished” in its “State” column. Press the disk icon at the top right to save all data (the RFN prop asset, the wearable preset and the Seamster project settings with the updated RFN state).

### 13 Seamster preferences

Clicking on the “dials” icon in the top right of the Seamster pane (figure 26) will bring up Seamster’s “Preferences” window (figure 27). The settings explain themselves when you hover the mouse over them.

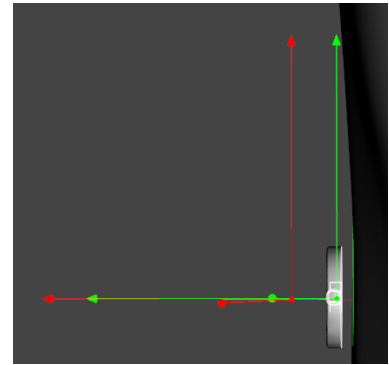


Figure 24: RFN axes, side view.

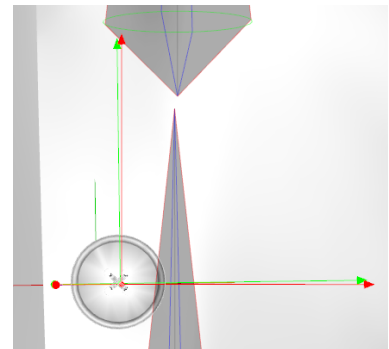


Figure 25: RFN axes, front view.



Figure 26: Preferences button on the main pane.

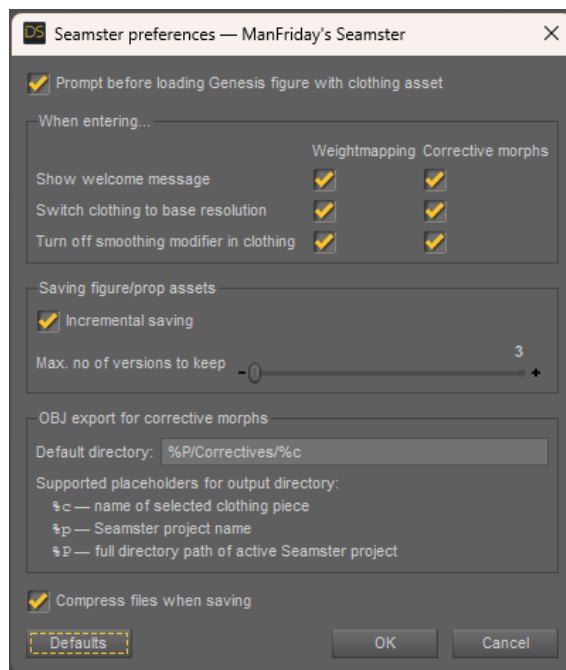


Figure 27: Seamster’s preferences window.

## 14 Detailed feature list

- Project management: creates clothing project directory and manages project settings for every clothing piece in it.<sup>46</sup>
- **Imports** an OBJ from a clothing modeler such as Marvelous Designer with just a few clicks, automatically sets up subdivision and smoothing modifiers according to user specifications and runs Transfer Utility with the required settings.
- Auto-renames the adjustment morphs that Transfer Utility creates from the projection templates and sets their limits to 0% and 100%, respectively.
- Consolidates all the typical per-clothing settings such as subdivision, smoothing, content type and projection template into one set of settings for each clothing piece, which are applied automatically where needed.
- Offers to rename surface names after OBJ import and remembers the names entered, in case the same OBJ is imported again.
- Saves out a **figure/prop asset** with just three clicks, and saves it again later again and again with a single click on the “save” button.
- Incremental saving of figure/prop assets with backups of older versions.
- Simplifies editing the **weightmaps** for rigging:
  - Everything is in one place: bones in the clothing item and the matching rotation dials of the parent figure’s bone are shown together, eliminating the need to switch back and forth dozens of times.
  - Has buttons for “next” and “previous” bone and remembers progress.
  - Switches to weightmapping tool automatically and selects “General Weights”.
- Simplifies editing **corrective morphs**:
  - Shows correctives for the currently selected bone and marks them as “idle”, “generated” or “custom” for a complete overview.
  - Export and import buttons for exporting and importing OBJs with the correct morph name.
  - Can copy corrective name to clipboard for easier naming of OBJs in external software.
  - Automatic import with correct settings by running Morph Loader Pro silently with the correct settings.
  - Working OBJ default settings for Blender (configurable).
- Turns off “Base Joint Correctives”, subdivision and smoothing automatically when entering weightmapping and correctives as required, restoring them on exit (configurable). Buttons exist to turn them on and off.
- Simplifies creating **rigid follow nodes (RFNs)**:
  - Can extract surfaces from a clothing piece as OBJ.
  - Can import OBJs for creating rigid follow nodes.
  - Simplifies setting up the rotation axes and correct parenting.
- Has **“save” buttons** for each context to save wearables, figure/prop assets and RFNs with one click.
- Comprehensive **PDF manual** with instructions about the Daz Studio clothes rigging process with Seamster.

<sup>46</sup> Basic steps in Seamster, page 5.



## 15 Installation

ManFriday's Seamster is best installed through the Daz Install Manager (DIM), which will also take care of updates automatically when they become available.

To verify that the plugin has been loaded correctly, in Daz Studio, you can check "Help" → "About Installed Plugins". There should be an entry called "ManFriday's Seamster" in the list, and it should have a green plug.

Because ManFriday's Seamster is a Daz Studio plugin, not a script, it must be installed to your Daz Studio's "plugins" directory (where Daz Studio itself is located) instead of one of your content libraries, where scripts would reside. If you must install the plugin manually (assuming a default installation of Daz Studio):

- On Windows, the file is `MF_seamster.dll`, and the directory is `C:\Program Files\DAZ 3D\DAZStudio4\plugins\` (or wherever you installed your variant of Daz Studio). You probably need to be a Windows administrator to copy the file there.
- On Mac, the file has a `.dylib` extension instead and needs to be in `/Applications/DAZ 3D/DAZStudio4 64-bit/plugins`.

After copying, restart Daz Studio.

If you have multiple editions of Daz Studio installed (e.g. the public beta in addition to the general release), the DLL needs to be in each program copy's `plugins` subdirectory. DIM will take care of that automatically.

In the worst case, if everything breaks, you can delete the DLL again (when Daz Studio is not running) and all should be OK.

## 16 Acknowledgments

Huge thank yous go out to the experts who answered my many questions, helped me with suggestions and who kept testing Seamster for months to get it into shape:

- **Esha:** <https://www.daz3d.com/esha>
- **Josh Darling:** <https://www.daz3d.com/josh-darling>
- **Lyoness:** <https://www.daz3d.com/lyoness>
- **Lyrria Madril:** <https://www.daz3d.com/lyrria-madril>
- **Mada:** <https://www.daz3d.com/mada>

ManFriday's Seamster uses some code from the following script coding samples at Daz (with many thanks to **Rob Whisenant** for maintaining those):

- "Silent OBJ Import"<sup>47</sup>
- "Silent OBJ Export"<sup>48</sup>
- "Silent OBJ Morph Loader Import"<sup>49</sup>

The plugin's beautiful icons are from the free and open-source Bootstrap Icons collection.<sup>50</sup>

This PDF user manual was written in Markdown, converted to LaTeX with Pandoc<sup>51</sup> and then typeset as PDF with LaTeX using the `tufte-handout` style.<sup>52</sup>

<sup>47</sup> [http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/scripting/api\\_reference/samples/file\\_io/import\\_obj\\_silent/start](http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/scripting/api_reference/samples/file_io/import_obj_silent/start).

<sup>48</sup> [http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/scripting/api\\_reference/samples/file\\_io/export\\_obj\\_silent/start](http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/scripting/api_reference/samples/file_io/export_obj_silent/start).

<sup>49</sup> [http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/scripting/api\\_reference/samples/file\\_io/import\\_obj\\_morph\\_loader\\_silent/start](http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/scripting/api_reference/samples/file_io/import_obj_morph_loader_silent/start).

<sup>50</sup> <https://icons.getbootstrap.com/>.

<sup>51</sup> <https://pandoc.org/>.

<sup>52</sup> <https://github.com/Tufte-LaTeX/tufte-latex>.

## 17 Glossary

**asset:** in Daz Studio, content from a content library that can be loaded into a scene; for example a Genesis character, a piece of clothing, a prop, or a material for any of the above. More technically, a “file segment that may be referenced from within either the same file, or from outside the current file scope”<sup>53</sup>. Assets normally have user-facing **DUF** and hidden **DSF** files. Older assets will often still have Poser file extensions such as `.cr2` and `.pz2`.

<sup>53</sup> [http://docs.daz3d.com/doku.php/public/dson\\_spec/format\\_description/asset\\_types/start](http://docs.daz3d.com/doku.php/public/dson_spec/format_description/asset_types/start).

**bone:** a definition of a rotation for a part of a mesh. Often bones are named after bones in the human body (e.g. “left shin”), but that is not a requirement. In Daz Studio, every bone has: 1) a parent, unless the bone is the root bone of a figure; 2) a center point and an end point, which rotates around the center point; 3) a rotation order (e.g. “XYZ”) defining the main rotation of the bone and around which axes the rotations should occur; 4) a face group, defining which vertices belong to the bone; if no weightmap exists, the face group defines the part of the mesh that rotates with the bone; 5) optionally, but with modern figures practically always, one or several **weightmaps** which determine for each vertex of the mesh how much it should be rotated with the bone: if 0%, then it is not moved at all; if 100%, it rotates with the bone completely. See **rigging** for the different weightmapping modes supported by Daz Studio.

**category:** in Daz Studio **metadata**, a grouping of similar assets that is useful for searching across products, for example “Props/Food” or “Wardrobe/-Socks”. A category can contain an arbitrary number of assets, and an asset can have more than one category.<sup>54</sup> Daz defines hundreds of categories, which are visible on the left side in the “Smart Content” pane. Categories are simple text strings, but form a tree hierarchy: semantically, a character that is in the “Figures/People/Male” category is also in the parent “Figures/People” category.

<sup>54</sup> In database terminology, that is an *m:n* relation. However, Daz Studio enforces that each asset have at least one category assigned to it: if it doesn't, it automatically receives a sub-category under the special “Lost and Found” category.

**compatibility base:** optional **metadata** assigned to a content file to allow Smart Content to display other content files that are compatible with what is currently selected in the scene if the “Filter by context” checkbox is enabled. Content files that are meant to be loaded only on certain scene nodes can declare that they are “compatible with” a compatibility base.<sup>55</sup> Like **categories**, compatibility bases are simple text strings in the database, but hierarchical: when an asset announces that it is compatible with a certain base, it will be considered compatible with all the *children* of that base as well.

<sup>55</sup> For example, a piece of clothing for Genesis 8 Female will set its “compatible-with” to the “/Genesis 8/Female” base. Other assets can introduce their own compatibility bases; for example, a piece of clothing that comes with its own materials can declare its own compatibility base “/Super Outfit/Super Pants”, and the matching materials can set their “compatible-with” information to that for Smart Content to work correctly.

**content type:** in Daz Studio **metadata**, a means to classify assets. Unlike **categories**, content types predate Smart Content (introduced with Daz Studio 4.9), and every asset can have at most one content type (not several). Even though content types like “Preset/Materials/MDL” may look hierarchical, they are a simple list of strings in the database without parenting.<sup>56</sup> Daz Studio relies on content types during asset loading for wardrobe and other **followers** to work correctly.

<sup>56</sup> See [http://docs.daz3d.com/doku.php/public/dson\\_spec/format\\_description/metadata/content\\_types/start](http://docs.daz3d.com/doku.php/public/dson_spec/format_description/metadata/content_types/start) for a (slightly outdated) list of content types.

**DSF:** “DSON Support File”, a Daz Studio content file in **DSON** format with a `.dsf` extension, which is normally in the hidden `/data` directory. Support files normally get loaded behind the scenes by other files and contain data like geometry that should be shared between other files instead of being copied into each of them. Contrast **DUF**.

**DSON:** “Daz Scene Object Notation”, the file format Daz Studio has been using for most of its files since version 4.5.<sup>57</sup> It is a variant of **JSON** and thus human-readable, but DSON files are often compressed using `gzip`<sup>58</sup> and must be decompressed before one can look at them in a text editor.

<sup>57</sup> [http://docs.daz3d.com/doku.php/public/dson\\_spec/format\\_description/file\\_types/start](http://docs.daz3d.com/doku.php/public/dson_spec/format_description/file_types/start).

<sup>58</sup> <https://en.wikipedia.org/wiki/Gzip>.

**DUF:** “DSON User File”, a Daz Studio content file in **DSON** format with a `.duf` extension, particularly for scene files and presets saved by the user. Contrast **DSF**.

**figure:** in Daz Studio, the term has two meanings: it can refer to (1) the family of Daz human characters, including all Genesis figures; (2) any mesh with **rigging**, i.e. having joints and bones, as opposed to a prop, which has no bones. That includes all the human figures, but in this sense, a shirt or a shoe is also a figure, or even a building with a door that can be opened and closed with a dial.

**follower:** in Daz Studio, a **figure** (in the “rigged mesh” sense) whose **bones** and morphs are synchronized with the those of another figure (most importantly, clothing and hair). For example, when you load a piece of clothing onto a Genesis figure (or manually select “fit to”), Daz Studio sets up all the bones and morphs of the clothing to automatically move along with the bones of the parent, which is how Daz Studio moves clothing with the parent figure.

**JSON:** “JavaScript Object Notation”, an open standard file format that originates from the JavaScript programming language, but has become ubiquitous for data exchange in the the entire computing industry.<sup>59</sup> It is plain-text and human-readable like XML, but more compact. Daz Studio uses it for most of its own files; see **DSON**.

<sup>59</sup> <https://en.wikipedia.org/wiki/JSON>.

**metadata:** in Daz Studio, the data about content files that has been written into the Content Database, consisting of **assets**, **categories**, **compatibility bases**, **content types** and **products**.

**morph:** a shape modification (deformation) for an existing mesh (called “blend shape” in Maya, or “shape keys” in Blender<sup>60</sup>). A morph is created by comparing two meshes that have the same number of vertices (and edges and faces between them) and will then contain only the differences (the “deltas”) for each vertex. A morph can only *move* the vertices of a mesh (displace them), but not rotate them around an axis. It also cannot add or remove vertices, or change the connections between them (the edges or faces). Much of Daz Studio’s functionality depends on morphs: every character shape is a morph, modifying the shape of a standard Genesis figure. Technically, a morph is an “indexed set of vertex offsets”, usually expressed as a map of 3D vectors for every vertex of a mesh that should be moved. A morph property then defines a single factor for how much the deltas of a morph should be applied to a mesh.

<sup>60</sup> [https://docs.blender.org/manual/en/latest/animation/shape\\_keys/introduction.html](https://docs.blender.org/manual/en/latest/animation/shape_keys/introduction.html).

**Morph Loader Pro:** a tool built into Daz Studio<sup>61</sup> that operates on the currently selected scene node doing the following: 1. it loads an **OBJ** from disk; 2. it compares the position of each vertex in the OBJ to each vertex in the selected scene node; 3. it will create a **morph** dial in the scene node from the difference (the “deltas” between the vertices). For this to work the two meshes must be identical (have the same number of vertices and connections) except for the vertex positions, or the process will fail with “Geometry did not match”.

**OBJ:** an old and open file format commonly used for storing 3D data.<sup>62</sup> Almost every 3D program (Daz Studio, Blender, ZBrush, Maya, Modo, Marvelous Designer, Hexagon, ...) can import and export data as OBJ with varying degrees of compatibility. While OBJ files can contain vertices, edges, faces, vertex groups, UV maps and (with additional `.mtl` files) material information, the OBJ file format cannot express rigging, lights, cameras or even parenting information<sup>63</sup> and is therefore not suitable for saving Daz Studio scenes or even assets. However, it is still useful for exchanging geometry between different 3D programs, and it is often used to import geometry into Daz Studio and for creating **morphs**.

**product:** on the one hand, what is sold in the Daz store; but on a technical level in Daz Studio **metadata**, a list of **assets** in Daz Studio's Content Database. Daz Studio enforces that every asset in the database belong to *at least one* product. Any asset that doesn't have a product (including files saved by the user) is automatically added to the LOCAL\_USER product, which can grow very large over time.

**rigging:** the process of adding **bones** to a mesh and defining which part of the mesh should rotate with each bone. In Daz Studio, the modern mechanism introduced with Genesis 3 is called “dual quaternion”, where each bone has only one weightmap for all its rotation axes. This weightmap is called “General Weights”. By contrast, Genesis 1 and 2 used a mechanism called “TriAx weighted” where each rotation of a bone had a separate weightmap. An even older system called “Parametric” or “Joint Parameter” was used by Poser. All of these mechanisms are still supported by Daz Studio, and you might find them in the program if you dig deep enough.

**weightmap:** an intensity value between 0% and 100% that is associated with every vertex of the mesh.<sup>64</sup> In Daz Studio, when the weightmap brush tool is active and a weightmap is selected in the “Tool Settings”, those percentages are represented with colors. Grey means 0%. The colors then move through yellow and blue to red for 100%. Daz Studio uses weightmaps for rigging, but also for modifying dForce simulation parameters for certain parts of a mesh only.

<sup>61</sup> Under “Edit” → “Object” → “Morph Loader Pro”.

<sup>62</sup> See [https://en.wikipedia.org/wiki/Wavefront\\_obj\\_file](https://en.wikipedia.org/wiki/Wavefront_obj_file) for the long and winding history.

<sup>63</sup> [https://docs.blender.org/manual/en/latest/addons/import\\_export/scene\\_obj.html](https://docs.blender.org/manual/en/latest/addons/import_export/scene_obj.html).

<sup>64</sup> [http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport\\_tools/weight\\_map\\_brush/editing/start](http://docs.daz3d.com/doku.php/public/software/dazstudio/4/referenceguide/interface/viewport_tools/weight_map_brush/editing/start).

## 18 Change log

- **Version 1.0.3** (2024-01-20): Fix weird dialog sizing issues on the Mac only.
- **Version 1.0.2** (2024-01-16): first version submitted to the Daz Store.

## References

Esha. *DAZ Studio content creation mastery, part 6: Rigging and morphing clothing items*. 2016. URL: <https://www.daz3d.com/daz-studio-content-creation-mastery-part-6-rigging-and-morphing-clothing-items>.

– *Essential introduction to Genesis 9 for content creators*. 2022. URL: <https://www.daz3d.com/essential-introduction-to-genesis-9-for-content-creators>.

Mada. *YouTube tutorials: 01 Flat sandals*. URL: [https://www.youtube.com/watch?v=5\\_D6OZ-d1ko](https://www.youtube.com/watch?v=5_D6OZ-d1ko).

– *YouTube tutorials: 02 Tight top*. URL: <https://www.youtube.com/watch?v=Z22WaVS68NU>.

– *YouTube: Rigid follow nodes: Belt buckle*. 2018. URL: <https://www.youtube.com/watch?v=wp4x6DmDfc8>.