Using NFS MW 2012 Exporter Blender addon by DGIorio

Car Replacement Tutorial by SwiftZC33S & PolySoupList

Download link for the addons: https://github.com/DGIorio/nfsmw_exporter
See also - How to setup the Blender addon

This tutorial will show you how to:

Replace the car body, wheels and collision model
Move the position of the driver, effects and skeleton bones, or swap the driver model

Along with this tutorial, it is A VERY GOOD IDEA to watch DGI's introduction video and look at the sample Blend files he has made available.
Some existing Blender knowledge is needed to make full use of it.

Read the Advanced Tutorial to learn about:

- How to add other LODs for body and wheels
- How to add animations for damage, spoiler and steering wheel
- How to add vertex ambient occlusion

Table of Contents

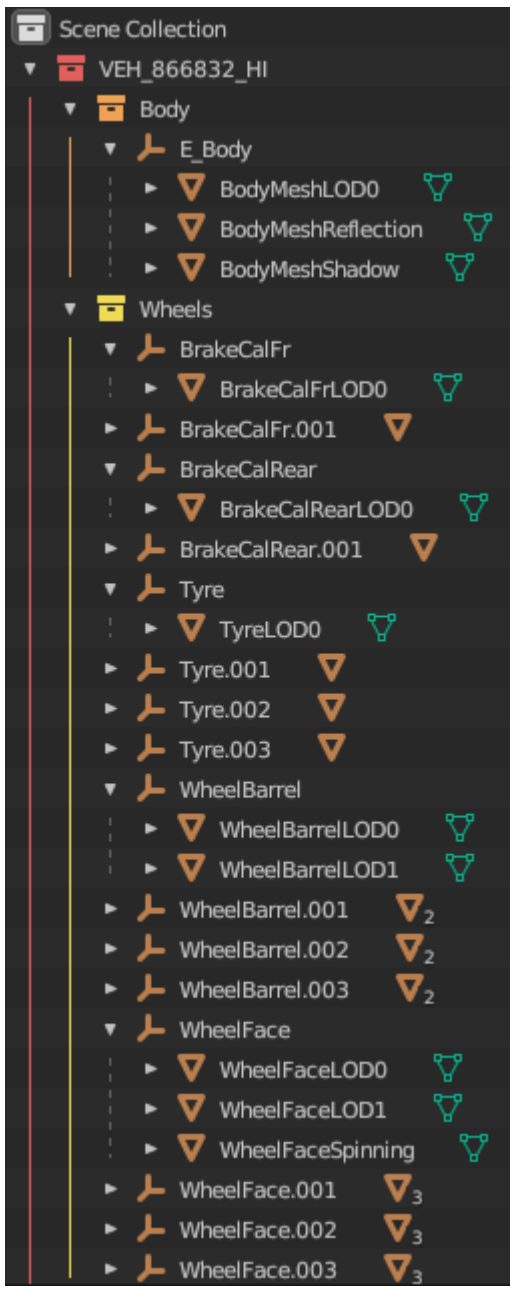
Last updated 11/02/2025

Overview

The Exporter addon requires your Blend file to be set up with a specific hierarchy of **Collections** and **Empty Objects** which will contain the model's **Meshes**.
In order to tell the Exporter what to do with those objects, you will add Custom Properties to them.

Hierarchy Example 1

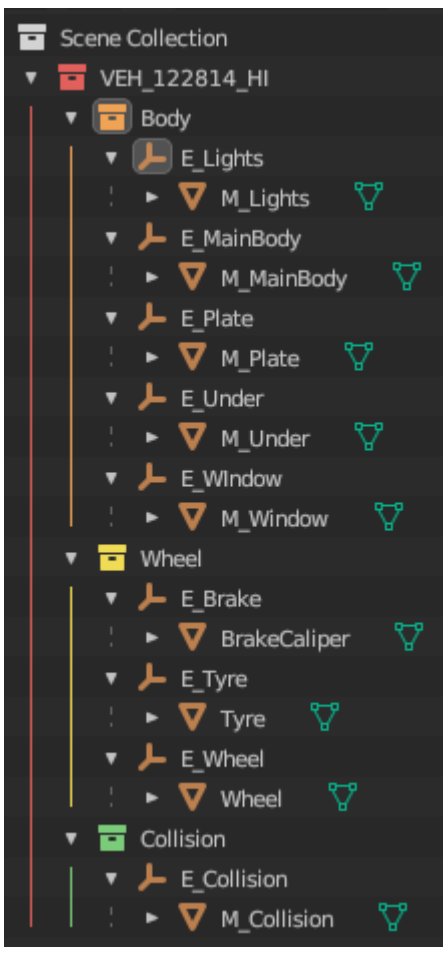
In this example there is one mesh for each LOD. The car body parts will be defined by different Materials (paint, carbon, plastic, glass, etc)



VEH_xxxx_HI	Must be the exact file name of the car being replaced. **Always required.** If it is for a LO file then use VEH_xxxx_LO HI = your car model. LO = AI car model.
Body	A collection to contain everything for the car body. **Always required.** Can be named anything.
E_Body	An Empty Object to contain a body mesh and its LODs (if using) There must be one Empty Object for each mesh, unless the mesh is a LOD. Can be named anything.
BodyMeshLOD0	The highest quality mesh. Can be named anything.
BodyMeshReflection	A mesh for the car's reflection in puddles. It is considered a LOD. Can be named anything.
BodyMeshShadow	A mesh for the car's shadow. It is considered a LOD. Can be named anything.
Wheels	A collection to contain everything for wheels. **Required if replacing wheels.** Can be named anything.
Other Empty Objects under Wheels	There must be one Empty Object for each part of the wheels and brakes. The empties with .001, .002, .003 are duplicates of the original one. One is used at each corner of the car. Can be named anything.
Other Meshes under Wheels	Just like the car body, wheel LODs are placed under one Empty. This is important if you want to make the proper spinning/blur effect on wheels. Can be named anything.

Hierarchy Example 2

In this example each body part is an individual mesh and there are no LODs.



VEH_xxxx_HI	Same as Example 1.
Body	Same as Example 1.
E_Lights (etc)	One Empty Object to contain one body mesh. Can be named anything.
M_Lights (etc)	The highest quality mesh. Can be named anything.
Wheel	Same as Example 1.
E_Brake (etc)	One Empty Object to contain one wheel or brake mesh.
BrakeCaliper (etc)	The highest quality mesh. Can be named anything.
Other Empty Objects under Wheels	Same as Example 1.
Other Meshes under Wheels	Just like the car body, wheel meshes are placed under one Empty each. Can be named anything.
Collision	A collection to contain the collision mesh and its Empty Object.
E_Collision	One Empty Object to contain the collision mesh.
M_Collision	The collision mesh.

Import and Prepare the New Model

Before you start… general info to be aware of:

- **Vertex and Face Limit**: Maximum of 65535 vertices and faces per material (you can use multiple materials on one mesh)
  Collision models should have less than 255 vertices and less than 255 faces per PolySoupMesh.

- **Triangulate Faces**: For Body and Wheel meshes, faces must be triangulated. Only the Collision model can support quads.

- **Texture Resolution**: The minimum resolution is 4x4. The resolution should be a power of two - for example 4, 8, 16, 32, 64, 128, 256, 512, 1024.
  Don't use too many high resolution textures because this can cause crashes. You could use 2048x2048 for a livery, but much less for other textures.

- **Texture Formats**: Supported texture formats are DDS, PNG, TGA, JPG, PSD and BMP. Textures will be auto converted to DDS as that's what MW uses. If you have DDS textures already, they should be DXT1 or DXT5 with mipmaps.

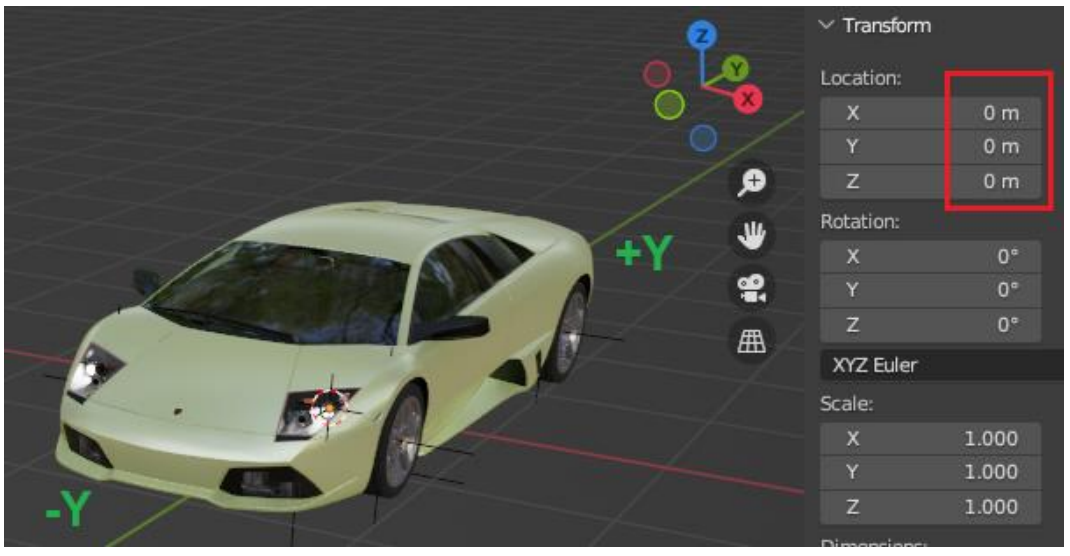
- **Custom Wheels:** If you're adding custom wheels, all original wheel parts will be automatically deleted (whether you have a replacement for all parts or not)

---

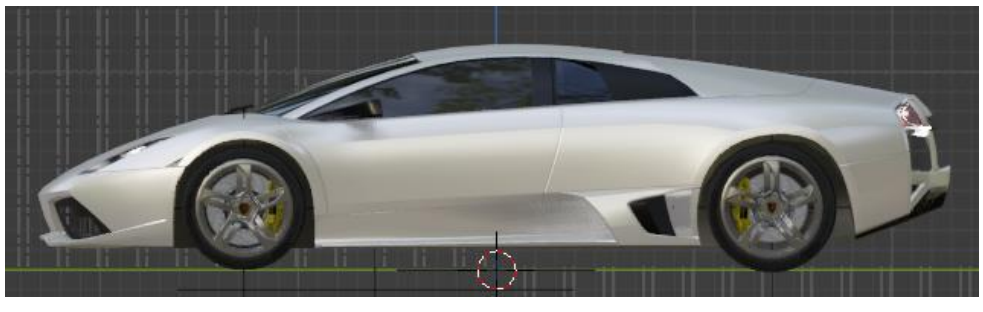
1. Start by importing your new car model into Blender.

2. Separate the car's wheels, brake calipers, brake discs and tyres into individual meshes if it isn't already.
   For the body, this isn't necessary. You can have just a single mesh if you want.

3. Make the front of the car face towards the **-Y** axis and the **Location** should be **0** on the XYZ axes.



4. The bottom of the wheels should be touching the **Y** axis. This represents the ground in the game.



Create the Hierarchy

1. Add Collections

a. On the right under `Scene Collection`, add a new Collection and rename it to **`VEH_number_HI`** where **`number`** is the ID of the car to be replaced. (If there is already a Collection with Blender objects such as Camera and Light, delete them)
b. Add up to 3 more Collections according to what you want to change (Body model, Wheels model, Collision model) They can be named anything you like.

2. Add Empty Objects

a. Add one Empty Object for each mesh. They can be named anything you like.
b. Move the Empty Objects under their relevant Collection. (drag and drop it on top of the Collection)

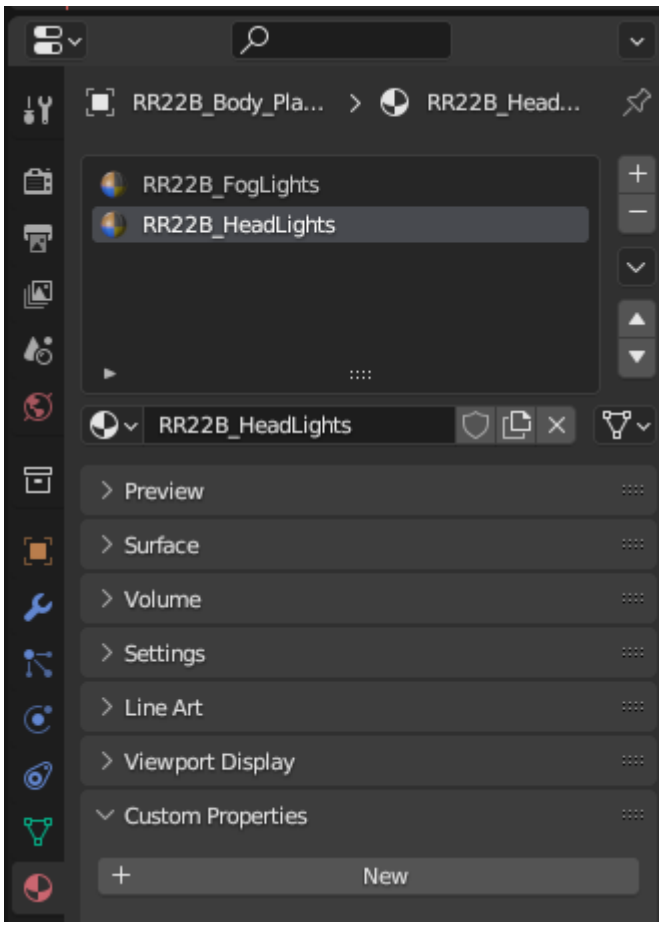
3. Associate Meshes with Empty Objects

a. Put one mesh under each Empty Object. (drag and drop it on top of the Empty, while holding SHIFT)
b. If you have LODs of the same part, they can go under the same Empty. (as seen in Example 1 at the start of this doc)

OPTIONAL
The car's Reflection, Shadow, low LOD of wheel parts, and blur/spinning wheel can be replaced. See the **Advanced Information**.

4. Add Materials to each mesh

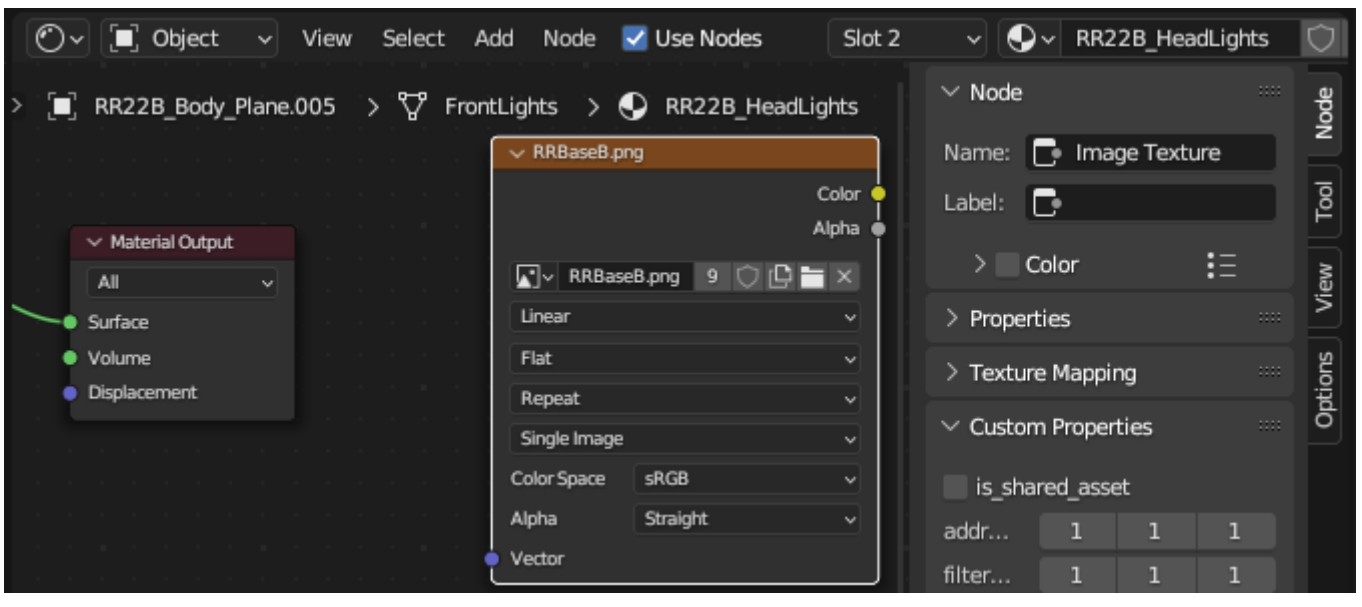
Each mesh needs a minimum of one Material. If you downloaded the model from somewhere else, it might have them already. Otherwise, add at least one.
The Material cannot have **.001 .002** etc in its name - it causes errors. Remove this suffix if present.



5. Add Textures

a. Select the Material that requires textures.
b. Drag textures into the Shading window or add them via **Add**>**Texture**>**Image Texture**. You don't have to connect them to the shader, but if you do, it allows you to use the *Identify Texture Type* feature later on.

NOTE: If you don't have textures to add for a Material, don't worry. The Exporter assigns default textures if needed.



A note about Lightmap Textures for lights to work

Red channel = brake lights
Green channel = headlights
Blue channel = reverse lights
Alpha channel = night time running lights

A note about UV mapping

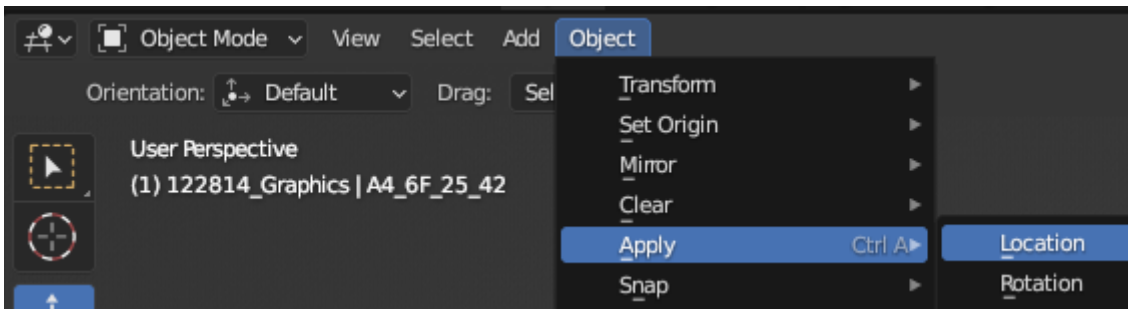
Some shaders in MW use multiple UV map layers. For example BodyPaint has:
UV1Map = diffuse texture
UV2Map = crumple (dents) texture
UV3Map = effects (dirt and scratches) texture and raindrops texture
UV4Map =

You can name your UV map layers as UV1Map, UV2Map, etc or TEXCOORD1, TEXCOORD2, etc
The exporter now gets the UV layer index based on the number. (In older versions before 28 April 2024, it went by what order they were in)
To check what UV layers a shader uses, check an existing car or refer to the `NFSMW_VehicleShaders.json` file in the NFSMW_Library_PC\Shaders folder. The TEXCOORDs are listed under each.

Set the Correct Location and Rotation

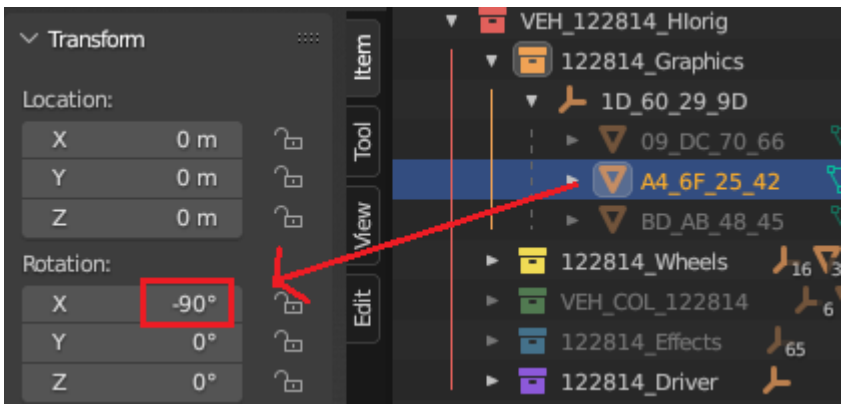
The Location and Rotation of the Meshes must be set correctly. In Object Mode do the following:

a. Check that the **XYZ Location** of all Meshes is 0. If it is not, select them and go to the **Object** menu > **Apply** > **Location**.

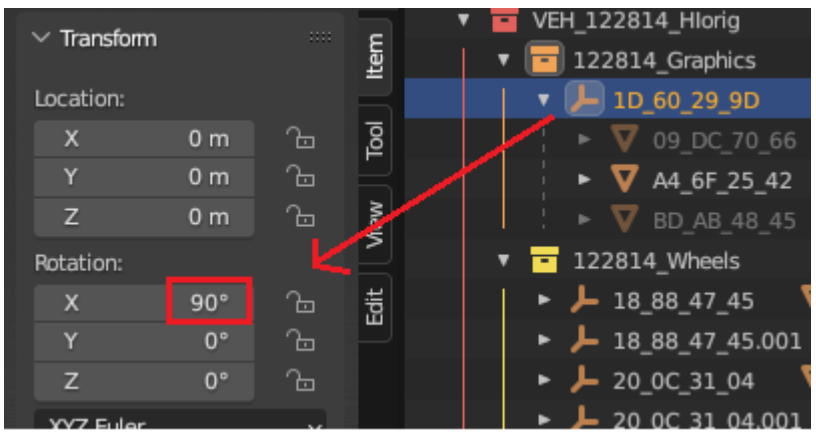


b. In the **Transform** panel, set the **X Rotation** of all Meshes to **−90** (that's MINUS 90) and then go to the **Object** menu > **Apply** > **Rotation**.
TIP: Select all Meshes to be rotated, then hold down the ALT key while you click in the **X Rotation** field. This will edit the rotation for all Meshes at once.

c. If you have changed the Scale of anything, apply the Scale too.



d. Select each Empty Object and use the **Transform** panel to set the **X Rotation** to **90**. But **do not** apply rotation to the empty objects.

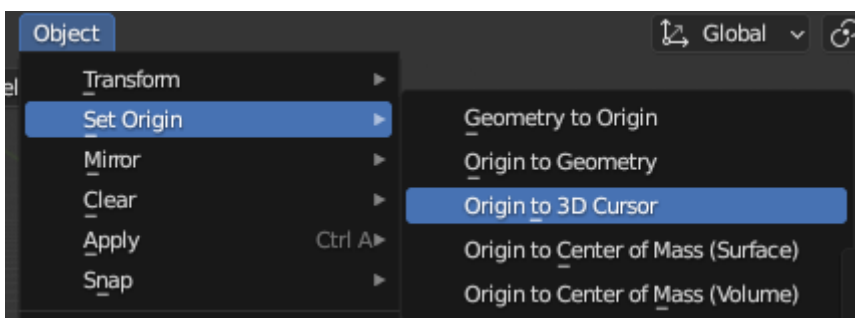


For Wheels - Set the Origin

All Empty Objects and Meshes for wheels need their origin set at the actual location of the wheel like this:

If it isn't like this already, do the following things for each wheel:

a. In Object Mode, select the Tyre Mesh, and go to the **Object Menu** > **Set Origin** > **Origin to centre of mass (surface)**
b. Again go to the **Object Menu** > **Snap** > **Snap cursor to selected**.
c. Select the Caliper, go to the **Object Menu** > **Set Origin** > **Origin to 3D cursor**.
d. Select the Brake Disc, then Rim, and do the same **Set Origin to 3D cursor**.
e. Select the wheel's Empty Object, go to the **Object Menu** > **Snap** > **Selection to cursor**.



Set Up the Custom Properties

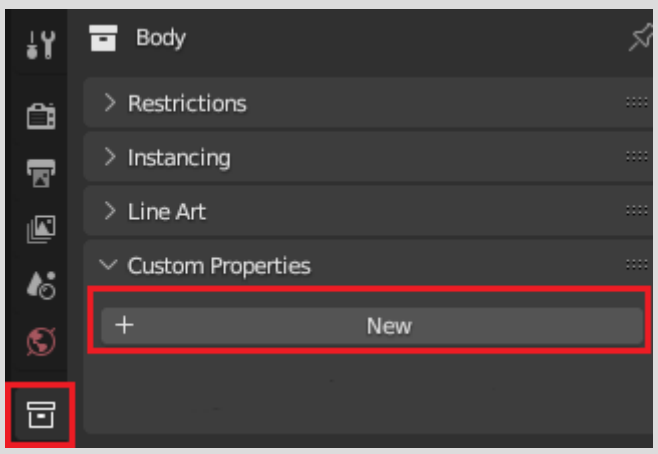
Now that the model is prepared, the hierarchy is created, materials and textures are added, it's time to define what is what so the Exporter can generate your new car.
ProTip: Enable Blender's built-in *Copy Attributes Menu* addon to make copying Custom Properties between objects a lot faster.

You will use the below procedure to add Custom Properties:

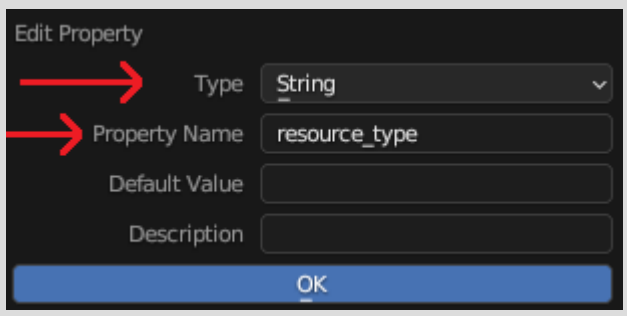
How to add a Custom Property to anything

1. On the right of Blender, click on the relevant Panel.

2. Scroll down and expand **Custom Properties** > click the **New** button.



3. Click the Gear icon next to the field that appears.
4. The **Edit Property** box appears - set the correct **Type** and **Property Name** then click **OK**.



5. Enter the relevant value.

1. Add a Custom Property to all Collections

a. Make sure you are on the Collection Panel
b. Click on a collection.
c. Add one Custom Property with:
d. **Type**: `String` **Property Name**: `resource_type`
e. Enter a value according to that —---------->

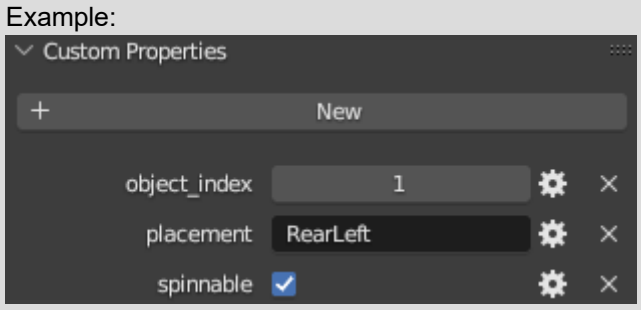
The Collection is for…	Use this resource_type
VEH_xxxx_HI / all Empties and Meshes related to Car body	**`GraphicsSpec`**
All Empties and Meshes related to wheels (Tyre, Rim, BrakeDisc, Caliper)	**`WheelGraphicsSpec`**
All Empties and Meshes for the Collision	**`PolygonSoupList`**

2. Add Custom Properties to all Empty Objects for Wheels

DO THIS FOR WHEELS ONLY
All three of the below properties have to be added on each Empty Object related to the Wheels.
Empty objects with the same name but with .001 .002 .003 etc. at the end act as duplicates of the original one without .001

a. Make sure you're on the Object Panel
b. Add those 3 Custom Properties to each —>

Example:



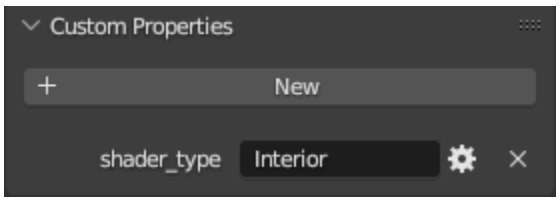
Property Name	Type	Values	Description
`spinnable`	**`Boolean`**	☐ ☑	`Blank` does not spin when the car is moving. Use it on the Brake Calipers `Ticked` spins when the car is moving. Use it on Tyres, Rims and Brake Discs If this property is not set, the Exporter will assume `ticked` by default.

`placement`	**`String`**	`FrontLeft`
		`FrontRight`
		`RearLeft`
		`RearRight`
			Mesh will be placed on the Front left
			Front right
			Rear left
			Rear right

`object_index`	**`Integer`**	`0` `1`	Set 0 on the tyre Set 1 on everything else that isn't a tyre Whatever is 0 has the 'pop' effect, so it should be tyres, unless you want your wheels or brakes to disappear! The exporter assumes the value by default if not present.

3. Materials

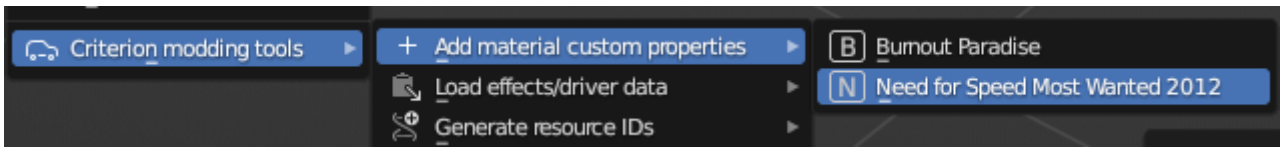
This is where you tell the Exporter what shader to use for each Material.



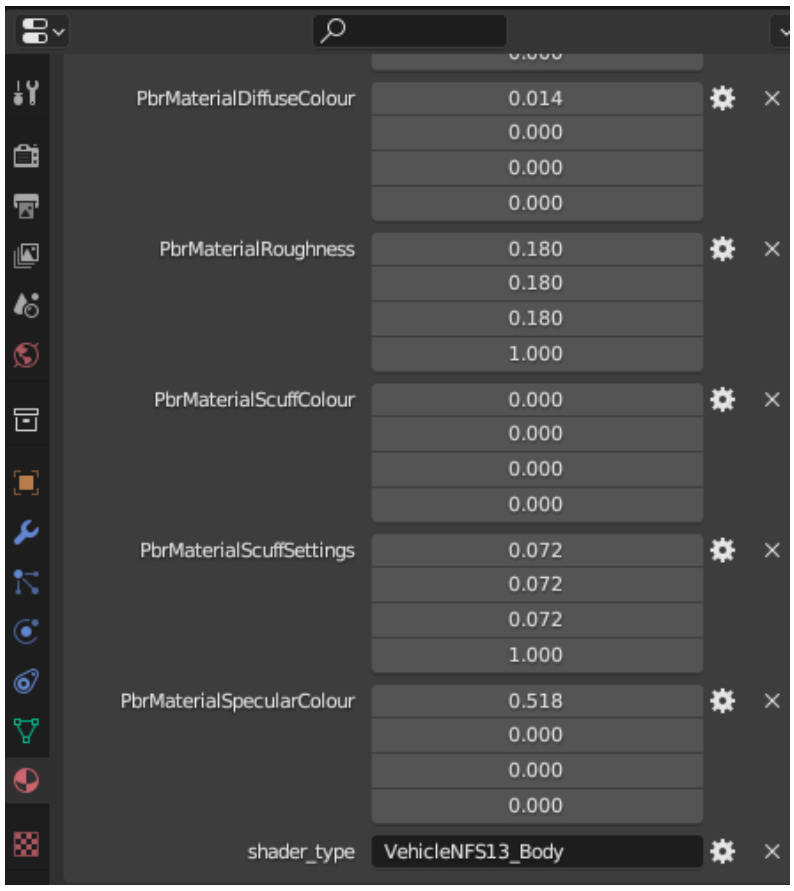
a. Make sure you're on the Material Panel
b. Click on a Material
c. Add one Custom Property with **Type**: `String` **Property Name**: `shader_type`
d. Enter a value according to **Table 1 - Shader Types**
e. In case you missed it the first time, check the name of the Material - remove any **.001 .002** etc from the name as this causes errors when exporting.

OPTIONAL
It's possible to use a shared Material that is not unique to the car (ie. It is shared across cars and exists in the NFSMW library. For example, the original tyre material).
First name the Material with its shared name, then add another Custom Property with **Type**: `Boolean` and **Property Name:** `is_shared_asset` and tick the box.
Take care when doing this, make sure to set the same `shader_type` as the material normally uses.
The Exporter uses default values for each Material where it's needed, but if you want you can also customise those.

After filling in **all** `shader_type`s, in the Object Mode menu click **Add** > **Criterion Modding Tools** > **Add Material Custom Properties** > **Need For Speed Most Wanted 2012**.



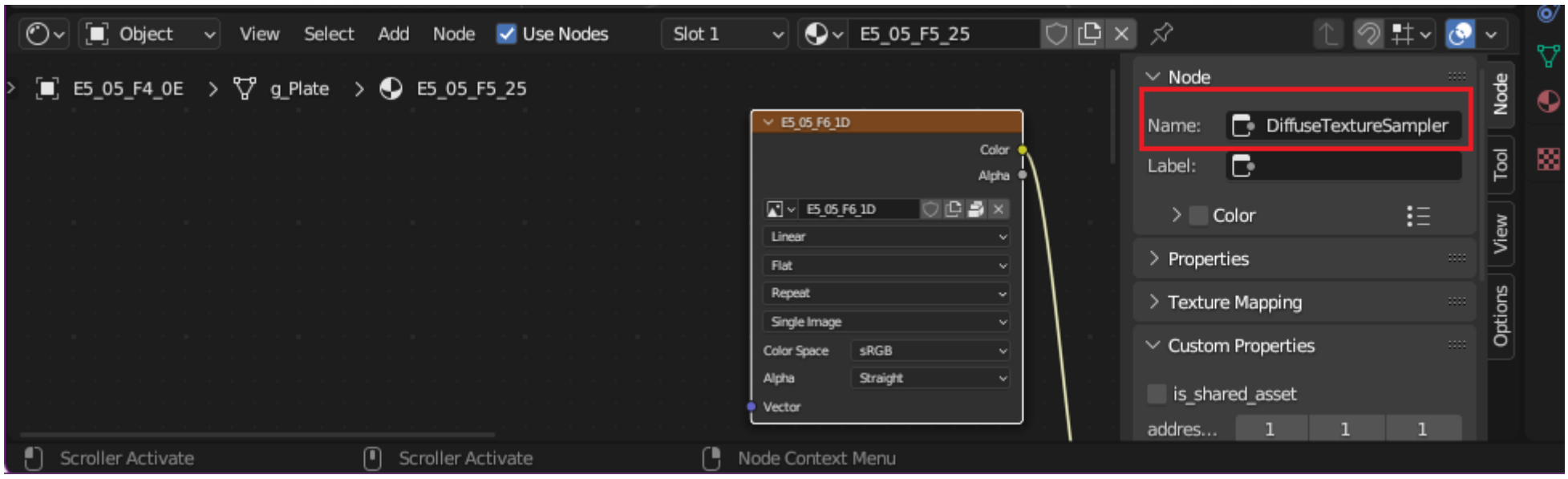
Then look at the Material Panel for each material, some Materials will now have extra properties, you can edit these numbers. (A partial list and explanations are in **Table 3 - Material Custom Property**)



4. Textures

Textures need to be given a specific **Node name** in Blender. The file name DOES NOT matter. It is the **Node name** that's important here.
Name the textures according to **Table 1 - Shader Types** (Textures Supported column)
Also see **Table 2 - Texture Samplers Explained** to check what they do.

Alternatively try the *Identify Texture Types* option. It will attempt to automatically name textures that were connected to the shader. (Currently supports Diffuse, Normal and Specular textures)
To use it, make sure you're in Object Mode and go to **Add** > **Criterion Modding Helpers** > **Identify Texture Types** > **Need for Speed Most Wanted 2012**.
In the Shading window, where you have added the textures earlier, click on each texture and check if it's been named correctly in the **Node panel**.
Remember, you don't have to add all textures. The Exporter assigns default textures where necessary.

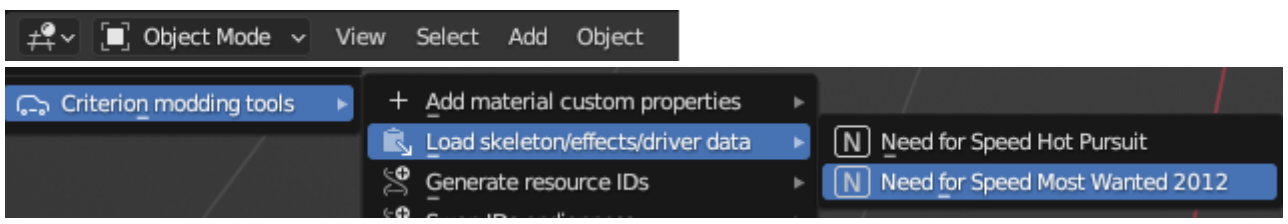


Edit the Driver, Effects and Skeleton Positions

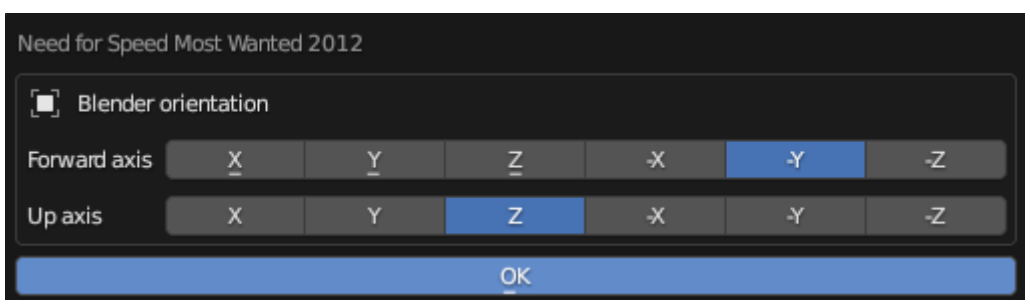
Import the Positions

The original driver, effects, and skeleton positions can be loaded on your model, then you can precisely move them to suit it.

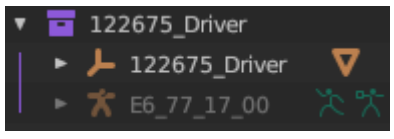
a. Make sure you're in Object Mode, click **Add** > **Criterion Modding Tools** > **Load Skeleton/Effects/Driver Data** > **Need for Speed Most Wanted 2012.**

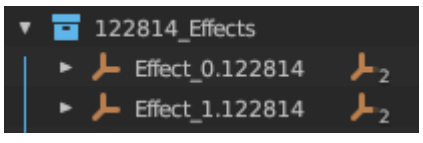


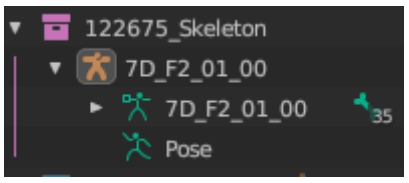
b. A box will appear. If you have the car set up in a different orientation to these defaults, change it to match and click OK.



c. Three new Collections will appear, containing Empty Objects. One for Driver, one for Effects and one for Skeleton.







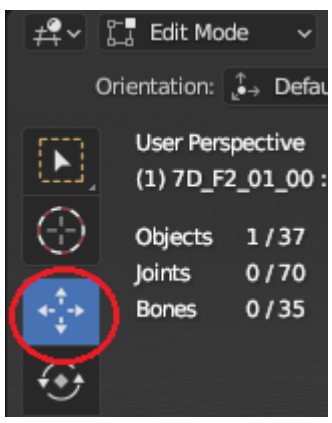
Adjust the Positions

- Select an Empty (driver or effect), and then you can move it. The new positions will be exported.
- The driver's body position is in the Driver Collection, however the hands position is in Effects.
- Each sphere is the position of a bone in the car's skeleton, which is used for various animations.

Example of hands positions (highlighted orange)

Spheres represent bones

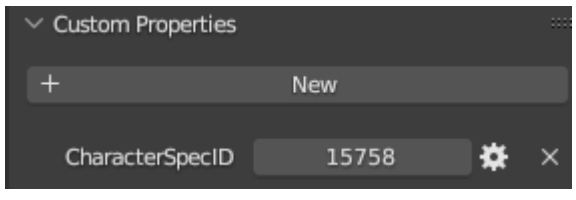
- To move a bone, click on any sphere (they will all be selected), then go into **Edit mode.**
- The view will change a bit - each bone is split into two spheres joined by a line.
- Click on one of the spheres to be moved, press **CTRL+ L** to select both, then use the **Move** tool so they move together.



How to Swap the Driver Model

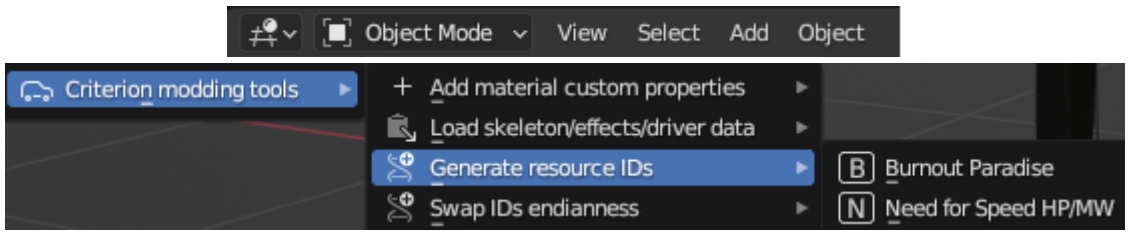
The Empty Object under the Driver Collection has a Custom Property called `CharacterSpecID`. It has the original ID of the driver model. You can change it.

- `15757` or `8D_3D_00_00` - Cop
- `15758` or `8E_3D_00_00` - Driver with normal seatbelt
- `1538141` or `5D_78_17_00` - Driver with harness

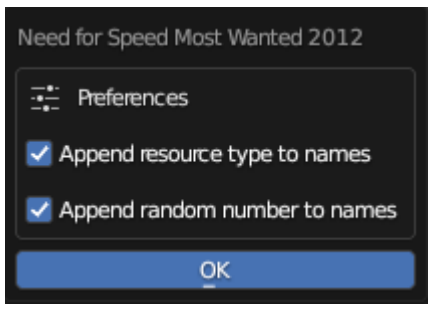


Generate the Resource IDs

The last step before exporting is to generate resource IDs so everything you've added can work in game.
Make sure you're in Object Mode, click **Add** > **Criterion Modding Tools** > **Generate Resource IDs** > **Need for Speed HP/MW.**



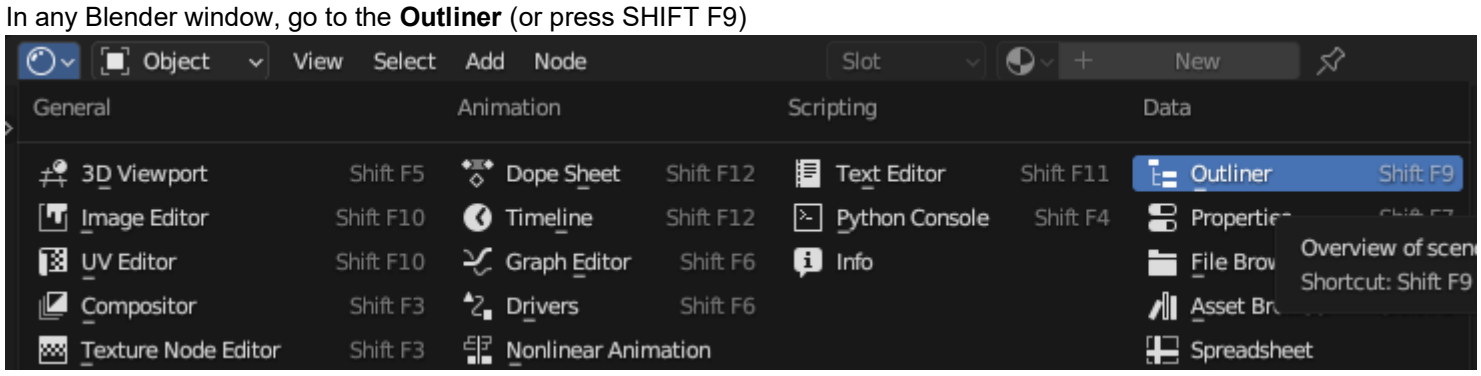
A box will appear. Leave both options ticked and click OK.



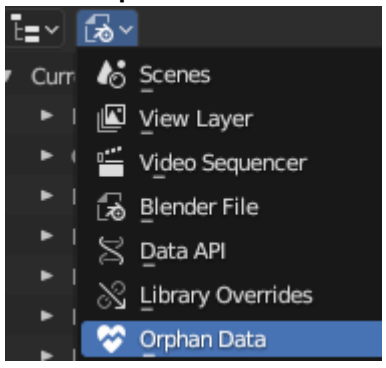
Clean Up the Blend File (optional)

You can remove unused objects that may have been created while you worked on the car. (this is optional) Sometimes, duplicated but unused objects could cause problems with exporting.

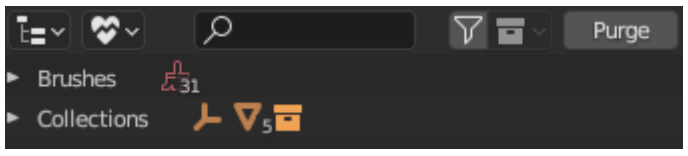
In any Blender window, go to the **Outliner** (or press SHIFT F9)

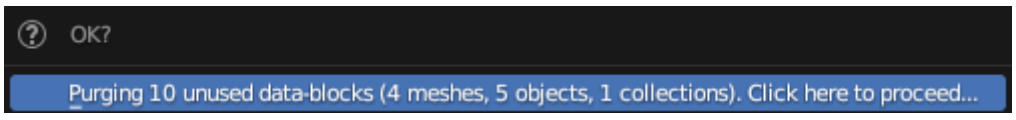


Select **Orphan Data**.



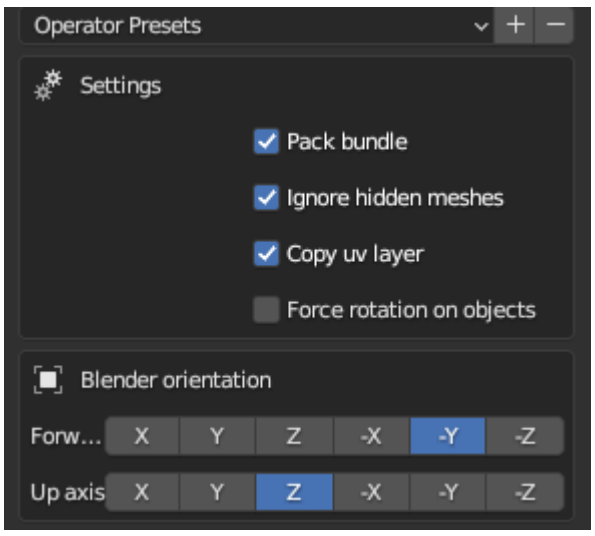
Click the **Purge** button, and click on the popup to confirm.





Export the Car

a. Before you start, ensure everything to be exported is visible
b. If you previously imported the original car to use some of its data, ensure everything related to it is hidden
c. To export go to **File** > **Export** > **Need for Speed Most Wanted (2012) (.dat)**
d. Give it a name and select the options you want on the right panel, then click the **Export to folder** button.



The options are:

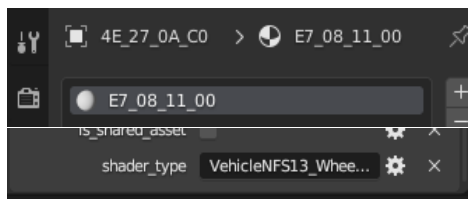
- **Pack bundle** - Pack all exported files into the BNDL file
- **Ignore hidden meshes** - If you have hidden any meshes they won't get exported
- **Copy UV layer** - Copy the first UV map layer to the other UV map layers where needed. Some shaders use more than one UV (for dirt, lightmap etc) so this allows them to work properly.
- **Force rotation on objects** - Check this if your car got exported in the wrong orientation (for example if you forget to set the 90° rotation on each mesh as mentioned earlier)
- **Blender orientation** - Forward Axis and Up Axis - Set the orientation of your model if it is different

To check the progress or any error messages click **Window** > **Toggle System Console**.

Custom Property Reference

Table 1 - Shader Types

Car body		
shader_type	**Use it for…**	**Textures Supported**
Badge	A badge texture Shader ID 8A_EF_09_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler SpecularTextureSampler
BodyColor	Body parts that are not paintable. Set a colour with the Base Colour setting in Blender's Principled BSDF shader and that colour will work in game. Shader ID 92_EF_09_00	EffectsTextureSampler CrumpleTextureSampler
BodyLivery	Body parts that can be painted, and also support a livery texture. Shader ID 72_EF_09_00	DiffuseTextureSampler EffectsTextureSampler CrumpleTextureSampler
BodyPaint	Body parts that can be painted. Shader ID 76_EF_09_00	EffectsTextureSampler CrumpleTextureSampler
BodypaintLight	Body parts that can be painted and has a lightmap (on vanilla cars it's used for the area behind the license plate - the plate's light glow texture is visible on it) Shader ID 74_EF_09_00	CrumpleTextureSampler EffectsTextureSampler LightmapLightsTextureSampler
BodyPaintNormal	Body parts that can be painted and has a normal map. Does not support damage. Shader ID 6E_09_00_00	NormalTextureSampler EffectsTextureSampler
CarbonFiber CarbonFiber2	Body parts made of Carbon Fiber. Each gives a different carbon pattern texture by default. Shader ID 78_EF_09_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler SpecularTextureSampler
Chassis	Part of the car's chassis. It comes with the default chassis texture if not supplied. Vanilla cars use this for the grille without transparency. If you use this for a grille, set the Diffuse texture.	Same as CarbonFiber
Chrome	Chrome parts. Gives default settings to make it look like chrome.	Same as BodyColor
CopLight	Cop lights. Yep. Shader ID: 7A_EF_09_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler SpecularTextureSampler EmissiveTextureSampler
DullPlastic	Body parts that are plastic. Gives default settings to make it look like dull plastic.	Same as BodyColor
Engine	Engine bay	Same as CarbonFiber
Glass	Windows Shader ID A9_EF_09_00	EffectsTextureSampler CrackedGlassTextureSampler CrackedGlassNormalTextureSampler
GlassColourise GlassColour GlassColor	Coloured glass coverings for tail lights, usually. Use whatever one you want. Same shit, different names. Shader ID AB_EF_09_00	NONE
Grill	Grille that requires transparency.	Same as Badge
Interior	Interior Shader ID 9B_EF_09_00	DiffuseTextureSampler NormalTextureSampler SpecularTextureSampler LightmapLightsTextureSampler
LicensePlate	License plate background. It comes with the default plate background if not supplied. Shader ID 7E_EF_09_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler LightmapLightsTextureSampler
LicensePlate_Number	Numbers and letters on the license plate. It comes with the default letters texture if not supplied. Shader ID 9C_D4_10_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler LightmapLightsTextureSampler
LightCluster	Option 1: Headlights, brakelights, reverse lights. How lightmap textures work Shader ID 7C_EF_09_00	NormalTextureSampler SpecularTextureSampler LightmapLightsTextureSampler
LightRefracted	Option 2: Headlights, brakelights, reverse lights. This option can make the lights look better if you have all the correct textures. Shader ID A1_EF_09_00	EmissiveTextureSampler ExternalNormalTextureSampler InternalNormalTextureSampler DisplacementSampler ColourSampler
LightGlass	On vanilla cars it's the cracked damaged glass layer. Shader ID A7_EF_09_00	DiffuseTextureSampler EffectsTextureSampler, CrackedGlassTextureSampler CrackedGlassNormalTextureSampler LightmapTextureSampler
Mirror	Mirror glass. Gives default settings to make it reflective like a mirror.	Same as Glass

Wheels		
shader_type	**Use it for…**	**Textures Supported**
BrakeDisc	Brake disc Shader ID B5_EF_09_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler SpecularTextureSampler
Caliper	Brake caliper Shader ID B5_EF_09_00	Same as BrakeDisc
CaliperBadge	Badge on the brake caliper Shader ID FC_BF_19_00	Same as BrakeDisc
Rim	Rims/wheels Shader ID B5_EF_09_00	Same as BrakeDisc
RimSpin	Spinning rim (if replacing this LOD, set RimSpin on both the lower LOD and the spinning LOD for a good result) Shader ID B9_EF_09_00	DiffuseTextureSampler NormalTextureSampler EffectsTextureSampler SpecularTextureSampler BlurDiffuseTextureSampler BlurNormalTextureSampler BlurEffectsTextureSampler BlurSpecularTextureSampler AmbientOcclusionTextureSampler
RimBadge	Badge on the rims Shader ID FC_BF_19_00	Same as BrakeDisc
RimBadgeFade	Badge on the spinning rim (if replacing this LOD, set RimBadgeFade on both the lower LOD and the spinning LOD) Shader ID BB_EF_09_00	Same as BrakeDisc
Tyre Tire	Tyres Note: This won't look exactly the same as the vanilla tyres because it's using a different shader. There is no texture change between track/offroad tyres. Shader ID 9B_EF_09_00	DiffuseTextureSampler NormalTextureSampler SpecularTextureSampler LightmapLightsTextureSampler
	To get vanilla tyres (same textures and works the same) Name the material `E7_08_11_00` (This material already should exist in each car' BNDL) Set the `shader_type` to `VehicleNFS13_Wheel_Tyre_Textured_Normalmap_Blurred` Add a Custom Property as Type: `Boolean` Name: `is_shared_asset` and tick the box.	

The table above lists the most commonly used shader types - DGI & PolySoupList have kindly given them simplified names for us to use. But there are many more…
For a complete list, refer to the `NFSMW_VehicleShaders.json` and `NFSMW_Shaders.json` file in the NFSMW_Library_PC\Shaders folder.

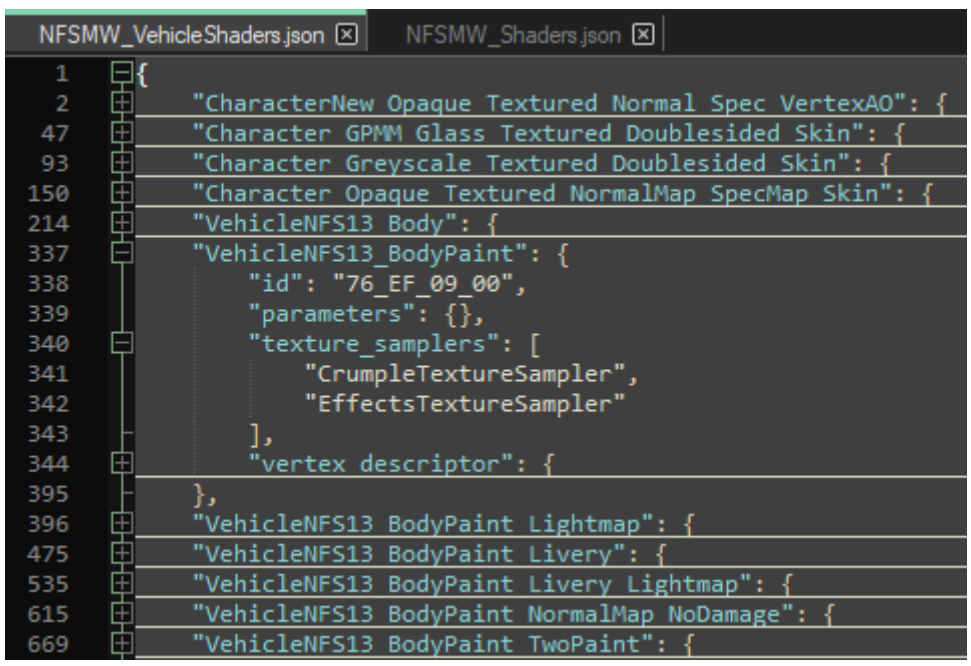
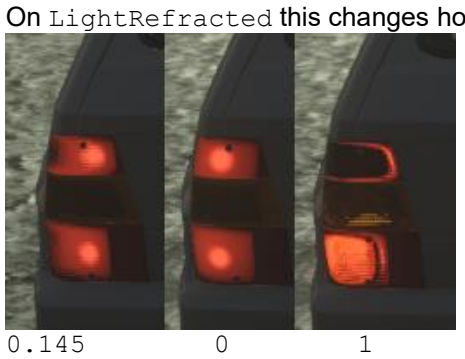


Table 2 - Texture Samplers Explained

Texture Sampler Name	What is it for
DiffuseTextureSampler	The general texture of something, like a livery or a carbon fibre pattern
NormalTextureSampler	Normal map - Gives a more detailed 3d effect to otherwise flat surfaces. The alpha channel of these is for roughness or specularity (not sure exactly which) The Exporter will use a default texture if not supplied.
EffectsTextureSampler	The texture responsible for dirt and scratches on cars. Probably don't need to change this. The Exporter will use a default texture of 1C_8D_0D_00.
SpecularTextureSampler	Texture that determines the shininess or reflectivity of what it's used on. Areas containing more white make it more shiny and areas closer to black are the opposite. The Exporter will use a default texture if not supplied.
CrumpleTextureSampler	The texture responsible for the dent effect on the car body when it's damaged. Probably don't need to change this. The Exporter will use a default texture of 49_02_06_00.
LightmapLightsTextureSampler	Shows when the headlight/brake/reverse lights are on. Also used for the license plate's light up pattern
CrackedGlassTextureSampler	The cracked glass pattern when the windows are damaged - only used by `Glass`, `LightGlass` and `Mirror` shaders. Probably don't need to change this. The Exporter will use a default texture of 7F_07_11_00
CrackedGlassNormalTextureSampler	The Normal map that goes with the cracked glass texture - only used by `Glass`, `LightGlass` and `Mirror` shaders. Probably don't need to change this. The Exporter will use a default texture of 80_07_11_00
EmissiveTextureSampler	Texture that looks like it's emitting its own light - only used by the `LightRefracted` and `CopLight` shaders
ExternalNormalTextureSampler	The Normal map used by the outer portion of the light mesh - only used by the `LightRefracted` shader
InternalNormalTextureSampler	The Normal map used by the inner portion of the light mesh - only used by the `LightRefracted` shader
DisplacementSampler	Makes a light displacement effect creating the illusion of more depth - only used by the `LightRefracted` shader (it's a green and yellow texture on vanilla cars)
ColourSampler	Like DiffuseTextureSampler but only used by the `LightRefracted` shader

Table 3 - Material Custom Properties

Property Name	What is it for
LightmappedLightsAlphaChannelColour	On `LightCluster` this is the idle/running/position lights colour. The values are Red Green Blue Alpha
LightmappedLightsBlueChannelColour	On `LightCluster` this is the reverse lights colour. The values are Red Green Blue Alpha
LightmappedLightsGreenChannelColour	On `LightCluster` this is the headlights colour. On `LicensePlate` and `LicensePlate_Numbers` this is the plate glow colour. The values are Red Green Blue Alpha
LightmappedLightsRedChannelColour	On `LightCluster` this is the brake lights colour. The values are Red Green Blue Alpha
mSelfIlluminationMultiplier	On `LightCluster` this is the brightness multiplier. It's used on other things that have a lightmap too
BrakeColour	On `LightRefracted` this is the brake lights colour. The values are Red Green Blue Alpha
HeadlightColour	On `LightRefracted` this is the headlights colour. The values are Red Green Blue Alpha
ReversingColour	On `LightRefracted` this is the reverse lights colour. The values are Red Green Blue Alpha
TaillightColour	On `LightRefracted` this is the idle lights colour. The values are Red Green Blue Alpha
mExternalGlassColour	On `LightRefracted` this is the colour of the outer glass. The values are Red Green Blue Alpha
EmissiveLuminance	On `LightRefracted` this is brightness of the lights when they're on
ParallaxConstants	On `LightRefracted` this changes how the textures inside the light look, a displacement effect. Google "Parallax mapping"  `0.145` `0` `1`
DebugOverride_GlassVolumeColour	On `Glass` this colours the glass (you can have tinted windows) The values are Red Green Blue Alpha
FresnelFactor	On `Glass` it does this. Only the first value is used. 
OpacityMin	On `Glass` this is minimum opacity. I guess don't set it too close to 1. Only the first value is used.

	
SurfaceSoftness	On `Glass` it does this. Only the first value is used. 
PbrMaterialDiffuseColour	The base colour. The values are Red Green Blue Alpha. In this example I've set `0, 1, 0, 1` while keeping the stock Specular Colour. 
PbrMaterialSpecularColour	Colour of reflected light. The values are Red Green Blue Alpha. In this example I've set `0.5, 0, 0, 1` while keeping the stock grey Diffuse Colour. 
PbrMaterialRoughness	Makes it appear more/less glossy/matte. 0 for full glossy, 1 for full matte. Or anything in between. Only the first value is used. 
PbrMaterialDirtColour	Colour of Dirt. The values are Red Green Blue Alpha
PbrMaterialDustColour	Colour of Dust. The values are Red Green Blue Alpha
PbrMaterialScuffColour	Colour of scratches. The values are Red Green Blue Alpha
PbrMaterialScuffSettings	Changes the appearance of the scratch texture.
PbrMaterialClearcoatFresnel	Changes the amount of reflected light at different viewing angles, for the "clearcoat" of the material.
PbrMaterialClearcoatSpecular	Like `MaterialSpecularColour` but for the "clearcoat" of the material.
PbrMaterialFresnel	On `LightRefracted`. Probably similar to ClearcoatFresnel but this didn't appear to make any difference when I messed with values. Another one to ignore.
	This is just the properties for the more common materials. For a complete list, refer to the `NFSMW_VehicleShaders.json` and `NFSMW_Shaders.json` file in the NFSMW_Library_PC\Shaders folder. However it does not contain explanations for them.

Troubleshooting

Frequently encountered errors we have seen are listed below.

Textures not showing in game	Did you use the correct TextureSampler in the Node name? Textures that are in DDS format must have mipmaps.

Material errors
`in main`
`  is_material_shared_asset = mat["is_shared_asset"]`
`TypeError: 'NoneType' object is not subscriptable`
	The material's name cannot have .001 .002 etc in it. Don't have these numbers in the name of anything at all, except Empty Objects for the wheels.

Getting a struct.error
`  g.write(struct.pack("<%s" % data_type, x, y, z, w))`
`struct.error: short format requires (-32768) <= number <= 32767`
 	
`location: <unknown location>:-1`
	If you have scaled any of the objects, apply the scale.

IndexError (ao_layer_index)
`Traceback (most recent call last):`
`  File "F:\ BlenderPortableMW3611LTS\3.6\scripts\addons\e`
`  status = main(context, self.filepath, self.pack_bundle_file, self.ignore_`
`  File "F:\ \BlenderPortableMW3611LTS\3.6\scripts\addons\e`
`  meshes_info. indices buffer, vertices_buffer, object_center, object_radiu`
`  File "F:\ \BlenderPortableMW3611LTS\3.6\scripts\addons\e`
`  color = list(mesh_vertex_colors[ao_layer_index].data[loop_ind].color[:])`
`IndexError: bpy_prop_collection[index]: index 0 out of range, size 0`
	In the Color Attributes of each mesh, the VColor1 layer must be **Face Corner** and **Byte Color**