



## **IMPORTER MANUAL**



## MANAGING MODS

Imported mods are located in:

`"..\[SteamLibrary]\steamapps\common\PureFarming\  
PureFarming_Data\StreamingAssets\IMPORTER\mod"`

Mods created by others also have to be placed in that folder. After placing new mods into the folder, the game has to be restarted, because mods are being initialized on game start and won't be seen if the game is already running.

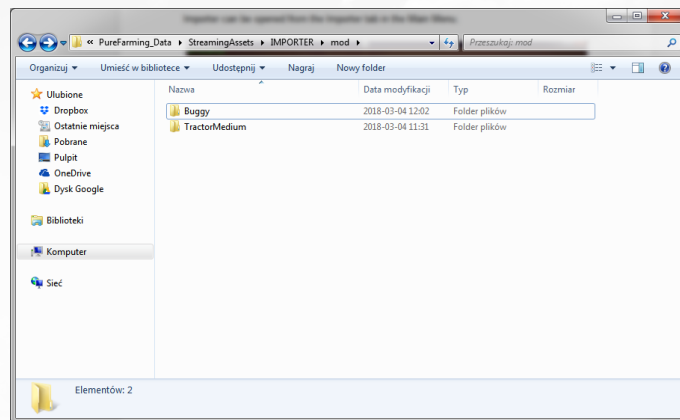
## DELETING MODS

Mods can be deleted in the Importer application or directly from the `"..\mod"` folder location.

## CHANGING MOD'S NAME

To change the mod's name, you have two options:

1. Load the mod in the importer, change its name in the parameters and save it. It will create a new mod with a new name but preserve the previous one.
2. Change the \*.bin and \*.json filenames. Open \*.json file with a text editor and change the name value parameter and save the file. All of the names must match. You also have to make sure the name is unique and that you do not use other mods with the same name.



## MAKING A NEW 3D MOD

Initially, a new mod has to be created in a modeling software like Blender, 3ds Max, Maya, etc. There are many tutorials on the web, teaching how to use each software so it will not be explained here. You can of course use a complete 3D model made by someone else and skip that part.

## PREPARING AND EXPORTING A 3D MOD FOR THE IMPORTER

There are a few key requirements a model has to fulfill before it can be successfully imported. A vehicle mod consists of different parts, because functional parts have to be separated. There

are optional and mandatory parts. Optional means a part can be, but does not have to be, imported either because a vehicle can work without it or because there are default parts that can be used in the importer. Mandatory parts have to be imported.

## HIERARCHY

The model in the modeling software has to match the corresponding object structure:

### Vehicles:

#### Shared parts:

- Body
- (Optional) SteeringWheel
- (Optional) WheelFrontLeft
- (Optional) WheelFrontRight
- (Optional) WheelRearLeft
- (Optional) WheelRearRight
- (Optional) FenderFrontLeft
- (Optional) FenderFrontRight

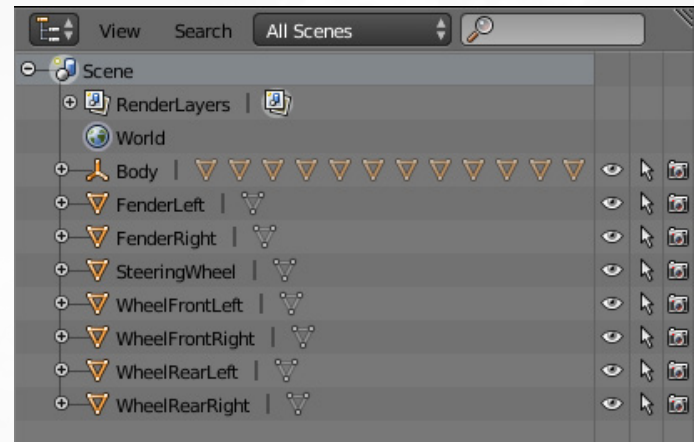
#### Unique parts:

- Dump Truck
- Container
- Caterpillar Tractors
  - (Optional) TracksLeft
  - (Optional) TracksRight
- Combine Harvester
  - GrainConveyor
  - UnloaderPipe
  - GrainTankDoor

### Devices:

- Header
- Body

In Blender Outliner, the object hierarchy could look like this:

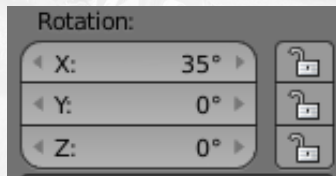
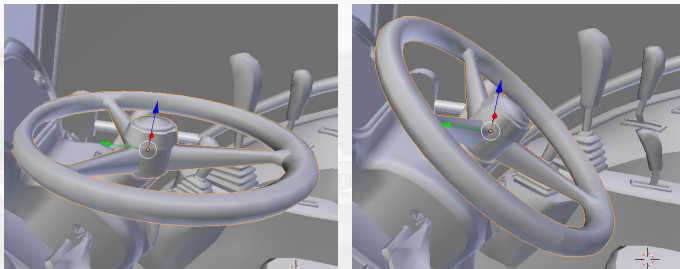
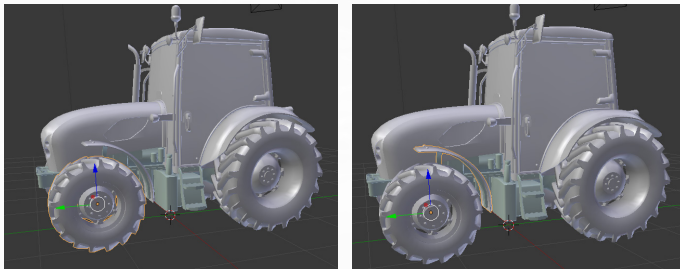


Object names do not have to match the convention, but it helps to automatically assign parts during the import process.



## PIVOTS

Another important thing is pivots. The origin location of an object has to be set to the point at which an object will rotate around. Correct pivots:



**Wheels** – center of the object.

**Fenders** – center of the corresponding wheel.

**SteeringWheel** – the pivot initially should have no rotation.

Rotation should be applied by the translation so the steering wheel can rotate properly in the game.

**UnloaderPipe** – same rules as SteeringWheel.

**Body, Container, Tracks** – do not rotate, so pivots are not that important. The recommended pivot is around point (0,0).

## ANIMATIONS

Some of the mods use animations. The importer supports animations created in modeling software. It is currently the only way to get animations to the mod. Animations are not mandatory. If animations are used, model objects must have the same names as stated in the hierarchy above. Animation names are defined just to help assign them during the import process and do not have to match. Exporting a single animation should cause no trouble. Exporting multiple animations for the same model can be tricky and require animations to be exported as separate takes.

### Dump Truck

ContainerTilt – lifting container in order to unload its contents.

### Combine Harvester

GrainTankOpen – opening door to harvester storage.

HeaderUp – lifting grain conveyor with a header.

### Header

ArmUp – transition between header work mode and standby mode.

Roll – harvesting reel animation, should be looped.

## UV MAPPING

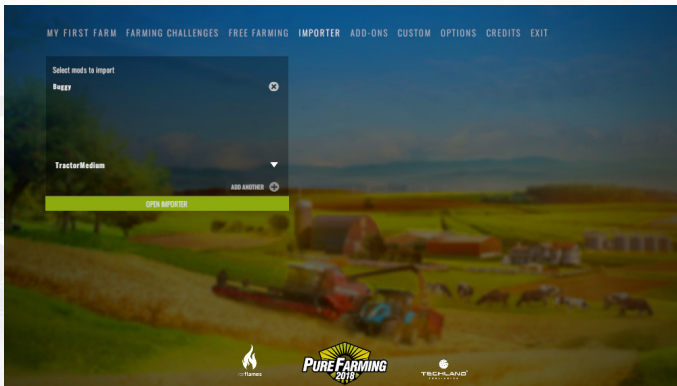
The 3D model should also have texture mapping if textures are going to be used. Assigning textures and material configuration will be done in the Importer.

## EXPORTING

A model prepared this way can be exported to FBX.

## USING THE IMPORTER

The Importer can be opened from the “IMPORTER” tab in the Main Menu.



All imported mods will be visible in the dropdown menu. Every mod has to be added to the list, before it will be available

in the game. Selected mods are being loaded to RAM during map loading, so they can be easily accessed by the game.

## IMPORTING STEPS

Select model type by clicking the corresponding icon:



Left bottom toolbar:

1. Quit the Importer and go to the Main Menu.
2. Go to the model selection screen, which is the first step to import a mod.
3. Load the already imported mod to the Importer for additional modifications.
4. Delete an imported mod from the hard drive. This action cannot be reversed.

## LOAD 3D MODEL

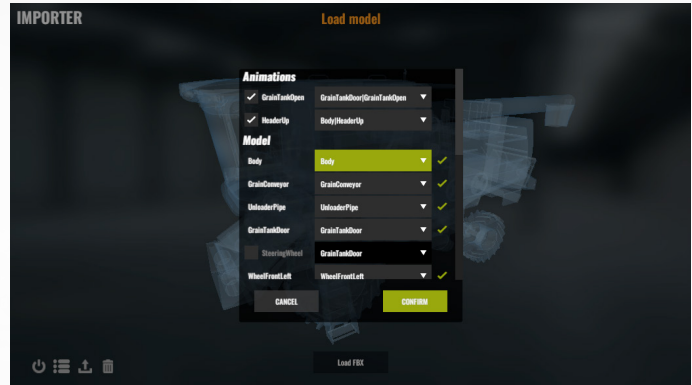
Select an fbx file from a hard drive.



## VALIDATION

If the file has a correct format and is loaded successfully, you will see the validation window. Here you can select which model parts will be imported. The left column represents a mod structure and the right column represents 3D model hierarchy. Each model part is unique and can be selected only once, so for example you cannot import a steering wheel both as a steering wheel and as a front wheel. If the mod can import animations, there will also be animation items to select. If the part has a toggle button, it means it is optional and does not necessarily have to be imported. Also, if the model

objects were properly named in the modeling software, they will be automatically assigned here.

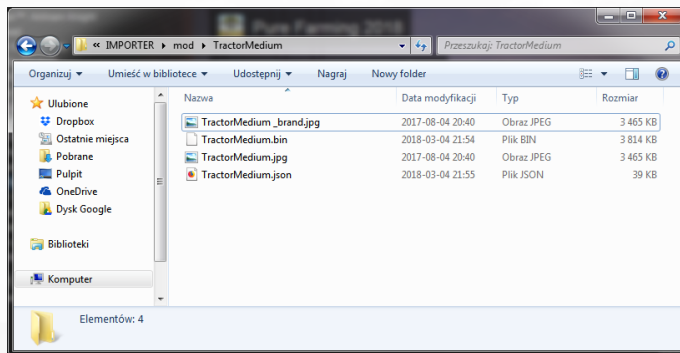


## CONFIGURE AND SAVE

If everything went fine, you will see your model in the next window. From here you can modify parameters, configure the vehicle's functional and visual parts, apply textures and save changes. At this stage, it is already recommended to click the "Import" button, which will create a folder with the mod's name and save the mod to that folder. The saved mod consists of two major files:

- binary (\*.bin) - contains mesh data. the fbx file is no longer needed.
- json (\*.json) - contains modification data. Parameters, model positions, rotations, material data, etc. are stored

in this file. Direct edition of this file is possible but is not recommended.



The mod can have textures as well, which should be stored in the same folder. You can also use custom vehicle images for the game shop in the tablet. To do that, place the image in the same folder as the \*.json file and use the same name as the \*.json file which is the actual mod name. You can also add a brand icon by adding “\_brand” to the filename. These images will be loaded when loading the mod to RAM. If you named image files correctly, you will see your images in the game.

## PARAMETRIZATION

Here you can change parameters that affect vehicle physics behaviour in the game and things like fuel tank or storage size. The name is also the mod's ID, so it has to be unique. You cannot have two mods with the same name in the game.



## TRANSFORMATION

Here you can set the position, rotation and scale of the key vehicle elements. At first you have to make sure the vehicle is pointed in the right direction. To check that, click the vehicle icon on the right toolbar to enable ghost reference. If the vehicle is pointed in the opposite direction, select "MODEL" from the "Model Parts" and rotate it in the Y axis or any other axis if necessary. You can also move the model until it has a position similar to the ghost reference. It helps to properly position the model in the game. Now you can start configuring Manual Parts. You can select/deselect an item by clicking on its name. Some of the parts are optional and can be turned on or off. After selecting Manual or Model part, the Transform window will appear in the top right corner. You can select the Local/Global pivot, position, rotation or scale gizmo and directly type transform values.

## SCENE NAVIGATION

**Right Mouse Click + WSADQE** – navigation

**F** – focus camera on selected part

**W** – position gizmo

**E** – rotation gizmo

**R** – scale gizmo

## MANUAL PARTS

General:

**FPPCamera** – position of the first person camera inside the vehicle.

**TPPCamera** – anchor at which the camera will rotate in third person view.

**CenterOfMass** – physical center of vehicle mass. It has high impact on the physics and should be placed close to the ground level for better stability.

**VehicleEnterPoint** – position of the trigger which will detect the farmer to enter a vehicle. Only the position is configurable. The size of the trigger is generated automatically.

**Driver** – model of the driver when the vehicle is being used by the player or AI.

**Collider** – collision bounds. You can add multiple colliders for better precision. You should not apply colliders to the wheels, as they have colliders generated automatically.

**Light** – vehicle lights. You can set range, intensity and shape of the light. Shape is used for optimizations so you don't have to create two lights for the left and right headlight. Each additional light will be enabled in the game by pressing the light input as many times as the vehicle light configuration requires.



Tractors:

**ExhaustSmoke** – smoke coming from the exhaust pipe.

Combine Harvester:

**StrawDumpParticles** – harvesting particles of dumping straw.

**StrawCutParticles** – harvesting particles of cutting straw.

**StrawDustParticles** – dust particles of spreading straw.

**GrainTankBottomLayer** – bottom texture layer of the grain tank storage.

**GrainTankTopLayer** – top texture layer of the grain tank storage. Texture will move between top and bottom layer, depending on the storage content.

**PipeCollider** – collision bounds of the unloader pipe.

**UnloaderParticles** – particles which will come out of the pipe when unloading combine storage.

**PipeOpenPosition** – position/rotation of the pipe when it is fully opened.

**PipeInnerCollider** – bounds which will detect collisions when closing the pipe.

**PipeOuterCollider** – bounds which will detect collisions when opening the pipe.

**HeaderDetachTrigger** – bounds which will detect collisions when detaching the header from the grain conveyor.

**HeaderAttachPoint** – anchor to which header will be attached.

Header:

**HarvestingDustParticles** – dust particles coming out from the header when harvesting.

**AttachingPoint** – anchor with which the header will be attached to a combine.

**CuttingArea** – length of the header harvesting area.

Dump Truck:

**ResourcesBottomLayer** – bottom texture layer of the containers resources.

**ResourcesTopLayer** – top texture layer of the containers resources. The texture will move between the top and bottom layer, depending on the storage content.

Tractor Truck:

**TractorTruckHook** – anchor to which the trailer will be attached and at the same time, the bounds of the trigger which will detect the trailer.

Pickup:

**StorageContainer** – bounds of the storage contents. The bigger the bounds the more bags and boxes will fit in.

## MODEL PARTS

Mesh parts of the model. Some of the parts are optional and can be turned on or off. Some of the parts have default items which can be selected from the dropdown menu. Currently there are three default parts:

## Wheels

FrontHook

BackHook

Default wheel can be applied to any vehicle if the wheel was not imported from an fbx. FrontHook and BackHook are used in tractors to connect devices. Currently using default hooks is the only way to connect devices to modded tractors.

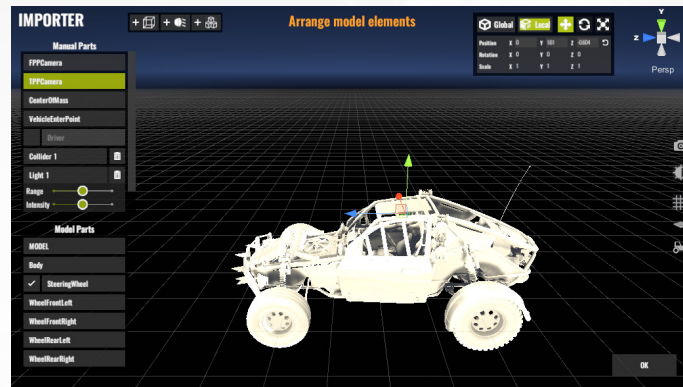
Top left buttons

1. Add collider.
2. Add light.
3. Add storage container.

Right toolbar

1. Reset camera position.
2. Switch background color to light or dark.
3. Toggle scene mesh on/off.
4. Toggle scene floor on/off.
5. Show/hide reference model ghost.

Top right buttons allows you to change camera perspective.



## TEXTURING

### MATERIALS

Here you can add new materials and delete unused ones. If you select a material, its configuration options will appear on the right. Textures before loading should be placed in the “..\IMPORTER\mod\” folder, preferably in the currently edited mod folder.

Shader – select shader used by the material.

Name – material name.

Standard Shader:

**Rendering Mode** – Opaque for solid objects, Transparent for transparent objects.

**Color** – base color of the material. If a texture is added, the color will change the texture tint.

**Albedo** – base texture of the material.

**Metallic** – level of the reflectivity and light response of the surface.

**Metallic Map** – metallic level controlled by the texture.

**Smoothness** – level of the reflectivity and light response of the surface.

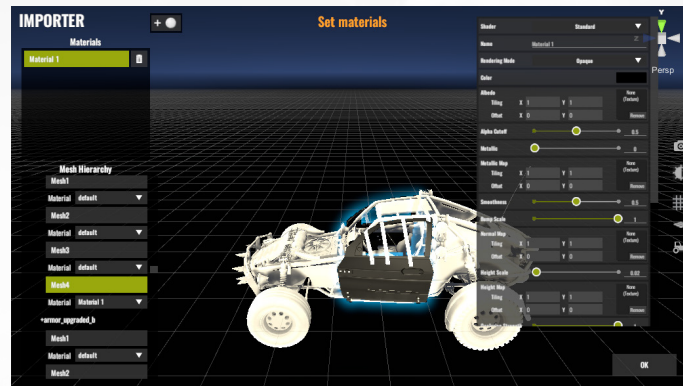
**Bump Scale** – intensity of the normal map.

**Normal Map** – normal map texture.

For detailed explanation of the Standard Shader check the official documentation <https://docs.unity3d.com/Manual/StandardShaderMaterialParameters.html>

## MESH HIERARCHY

Here you can assign created material to the mesh.

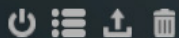


## IMPORT

Clicking this button will save the mod onto the hard drive. If this is the first time saving the mod, a new folder with the mod's name will be created in the "..\IMPORTER\mod\" directory and mod files will be placed there. If a mod with the same name already exists, then it will be replaced with new files.

# IMPORTER

Configure and save



PARAMETERS

TRANSFORM

TEXTURES

IMPORT