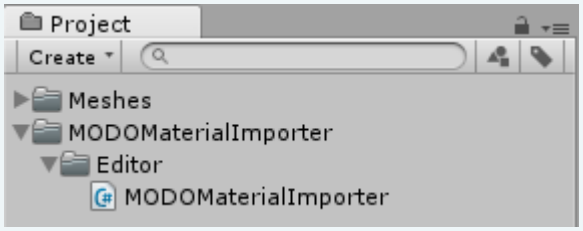
MODO Material Importer for Unity 3D

Summary

The MODO Material Importer is a Unity plugin that sets up standard materials in Unity, ensuring that their settings match those of your materials in MODO. When exporting to FBX from MODO, it is possible to have an XML file generated along with it which contains additional material information that is not supported by FBX. This plugin reads that information and sets up the materials in Unity to match the materials created in MODO using the new Unity material type.

Installing MODO Material Importer

If not installing via the Unity Asset Store, after creating a project in Unity, create a folder called Editor in the Assets folder of your project, and copy the script into it.



Using the MODO Material Importer

After exporting the FBX and XML files from MODO, place them into the Assets folder of your Unity project (or export them directly there). The plugin automatically detects the XML file paired with the FBX file based on their matching file names.

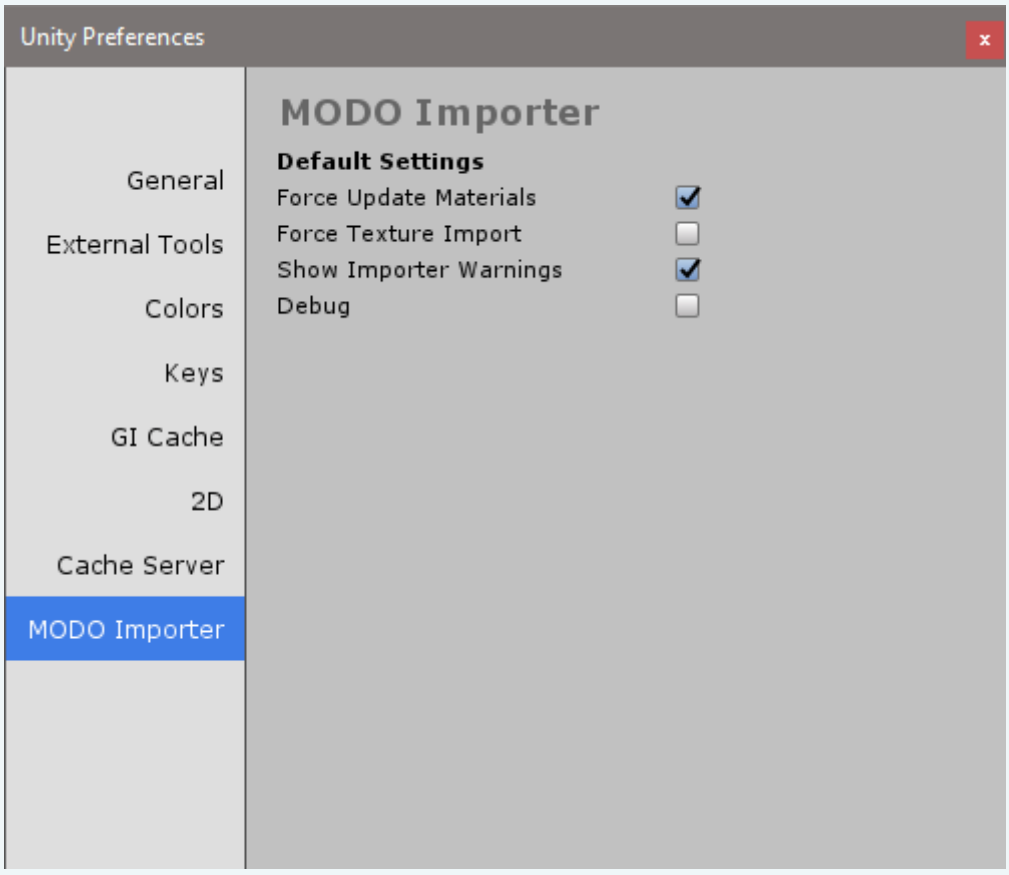
If either the .fbx or the XML file is updated, the plugin automatically applies the material settings.

The regular Unity settings of your mesh for material naming and texture searching are respected. You can check and alter these by selecting your imported .fbx file and viewing it's properties in the Inspector.

Importer Settings

There are global default options for the plugin, some of which can be overridden if desired on each XML file.
These default options can be found in your Unity Preferences;

Edit > Preferences > MODO Importer



- **Force Update Materials**

  When enabled, if either the XML or the FBX file is reimported, the plugin always updates the material's properties to match those specified in the XML file.

  When disabled, the material's properties are only applied to any brand new materials created as a result of the import process.

- **Force Texture Import**

  When enabled, the plugin will always attempt to reimport any textures that are stored externally to your project's Assets directory.

  When disabled, the plugin will first attempt to find a matching texture that already exists in your project and use that.

  Only new or missing textures are imported from an external location.

- **Show Importer Warnings**

  When enabled, if the plugin detected any issues with importing materials, such as incorrect channels or texture conflicts, they will be displayed in a dialog after import.

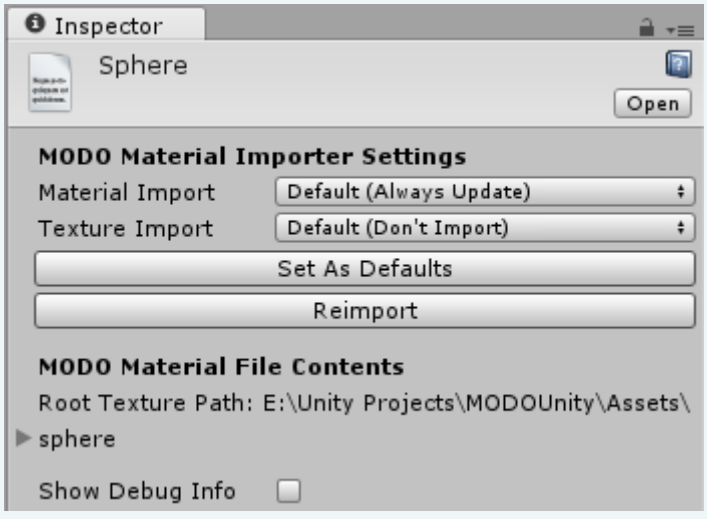
  Even when disabled, more detailed descriptions of the warnings will always be printed to the Console output, but the user will not be presented with a dialog listing them.

- **Debug**

  When enabled, the plugin will output more detailed information to the Console during material importing. This may be useful to hunt down the cause of any unexpected issues you experience.

Script Inspector

Selecting a valid MODO Material XML asset displays a custom Inspector containing options for the import settings as well as a listing of the XML contents.



Importer Settings

The Importer Settings allow you to override the global defaults specified in the editor preferences for the selected XML file.

- **Material Import**

  This overrides the global option for "Force Update Materials" for this specific XML file.

- **Texture Import**

  This overrides the global option for "Force Texture Import" for this specific XML file.

- **Set as Defaults**

  Apply the current XML import options as the global defaults.

- **Reimport**

  This reimports the selected XML file.

File Contents

This gives a summary of the information stored in the XML file. Each material defined in the XML has a drop-down, listing all of the information about the material.

These details are read-only and are available only for your information.

Material Rules

The following rules apply for material parameters:

- **Transparency**

  Transparency is currently not supported for Unity in MODO, nor is it exported to the XML file.

  However, you can use the regular Transparent Amount texture effect with Invert enabled. To match Unity, set your Transparent Amount texture to be the Albedo texture, but swizzled to output just the Alpha channel.

  To do this, check Swizzling and set the Channel to Alpha. Then, in Unity, manually set your material's blend mode to Fade or Transparent. For more details on channel swizzling, see Channel Swizzling.

NOTE: The standard shaders in Unity use the Alpha of the Albedo texture for their opacity mask.

- **Normal Maps**

  Textures set to Normal or Detail Normal will have their texture types set to Normal when imported from an external location.

- **UV Channel & Texture Wrap**

  In Unity, all of the main texture wrap values are taken from the settings specified for the Albedo texture.

  Similarly, all of the detail texture wrap values - and UV channel - are taken from the settings specified for the Detail Albedo texture.

  This means that, in MODO, if you have an Albedo with 2.0 Wrap U and 2.0 Wrap V, but a Normal map with 1.0, when exported to Unity, the Normal map will take the 2.0 Wrap U and Wrap V settings of the Albedo texture.

  The same is true for detail textures; the values from the Detail Albedo are used for all detail textures.

- **UV Channel Ordering**

  The UV maps for each mesh are exported to FBX in alphabetical order (they order they are listed in the Vertex Map Lists in MODO).

  Unity assumes that the first UV map in the list is the main UV set and all of the main textures will use that for their UV map.

  The detail textures can optionally use the second UV map in the list. On export, if a detail texture does not use the same UV map as the Albedo texture (the main UV set) it is assumed to be using this second UV map and the importer will assign it as such.

NOTE: It is therefore important that you name your UV maps in MODO in such a way that the detail or secondary UV map is second in the UV Map lists in MODO.
For example, using "Texture" as your main UV map and "Texture Detail" as your secondary UV map.

Meeting Unity's Material Specification

Unity has hardcoded settings for certain texture map channels it takes its values from.

For example;

Unity expects to take the Smoothness value from the Alpha channel of the texture used to define Metalness (which is taken from the Red channel of the same texture).

If you had your Smoothness set up to use a different texture in MODO, it would not be possible to apply that to your Unity material. The alpha channel of the Metalness texture would be used automatically.

NOTE: As of Unity 5.4, it is also possible to use the Alpha channel of the Albedo texture to supply the Smoothness value. If you have your textures set up this way in MODO, the plugin will detect this automatically and enable the option in the material.

Texture Channels

The following table contains groups of values and the corresponding texture of MODO, from which the Unity importer takes them.

Each group is hardcoded in Unity to come from the same texture.

Texture Name	Channel Use
Albedo	**RGB:** Albedo Color **Alpha:** Transparency or Smoothness*
Metalness	**Red:** Metalness **Alpha:** Smoothness*
Occlusion	**Green:** Occlusion
Height	**Green:** Height
Detail Mask	**Alpha:** Detail Mask

* In Unity 5.4 or later, Smoothness can optionally be controlled by the Albedo texture's Alpha channel instead.

TIP: MODO's channel swizzling feature allows you to set your textures up in MODO the way that Unity

requires them to be. For more information, see Channel Swizzling.