



3D FOR THE
REAL WORLD.

Cinema 4D R20 glTF Exporter

Introduction

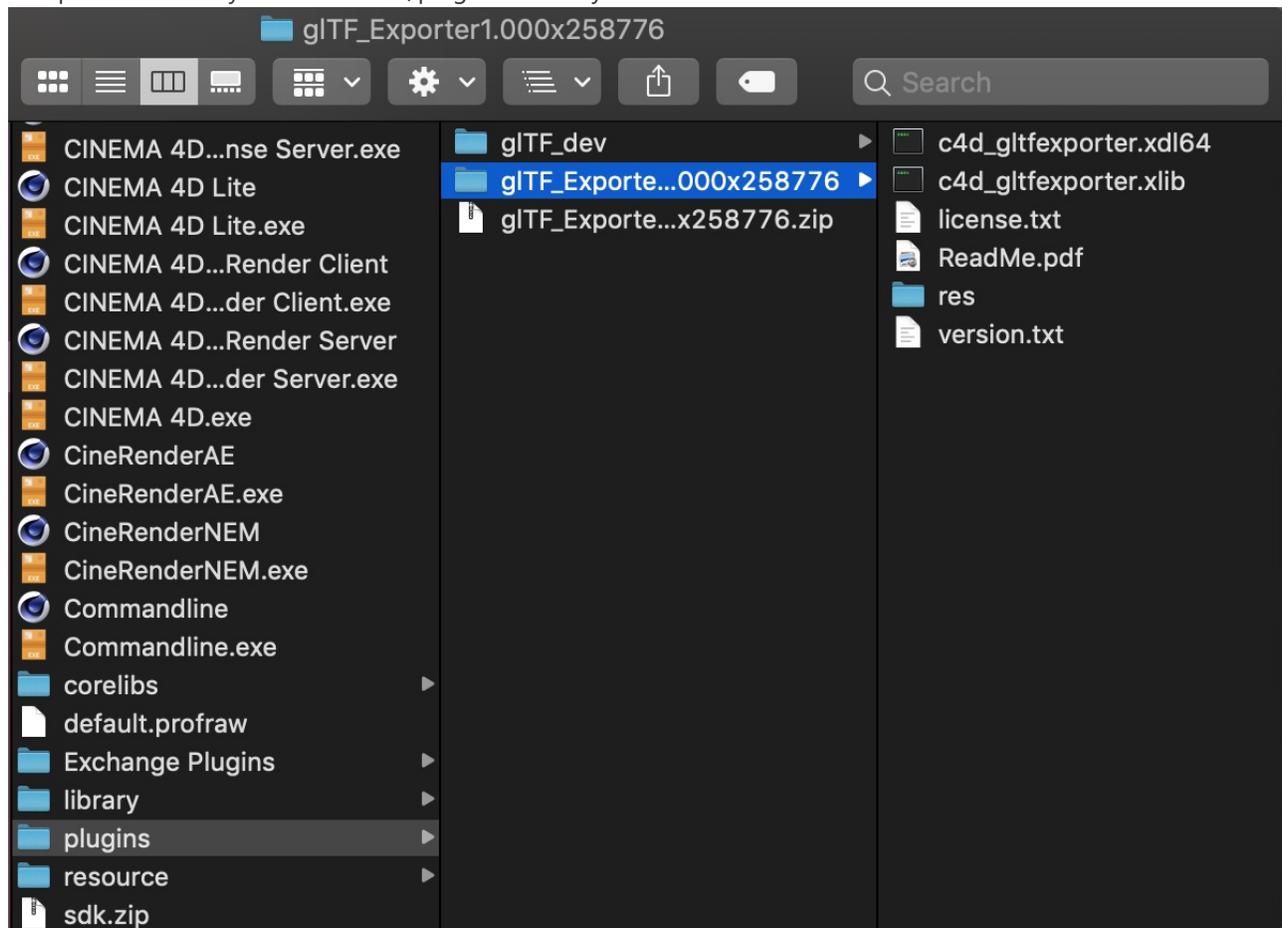
Hi everyone, I've put together this small tutorial to deliver important notes about our glTF Exporter PlugIn. In the following sections I'll explain what is and isn't supported so far followed by what's planned ahead. Hope it will be clear and looking forward to hear your feedback!

Version of Cinema 4D

This pluing requires : 20.026 RB251664 or later

Installation

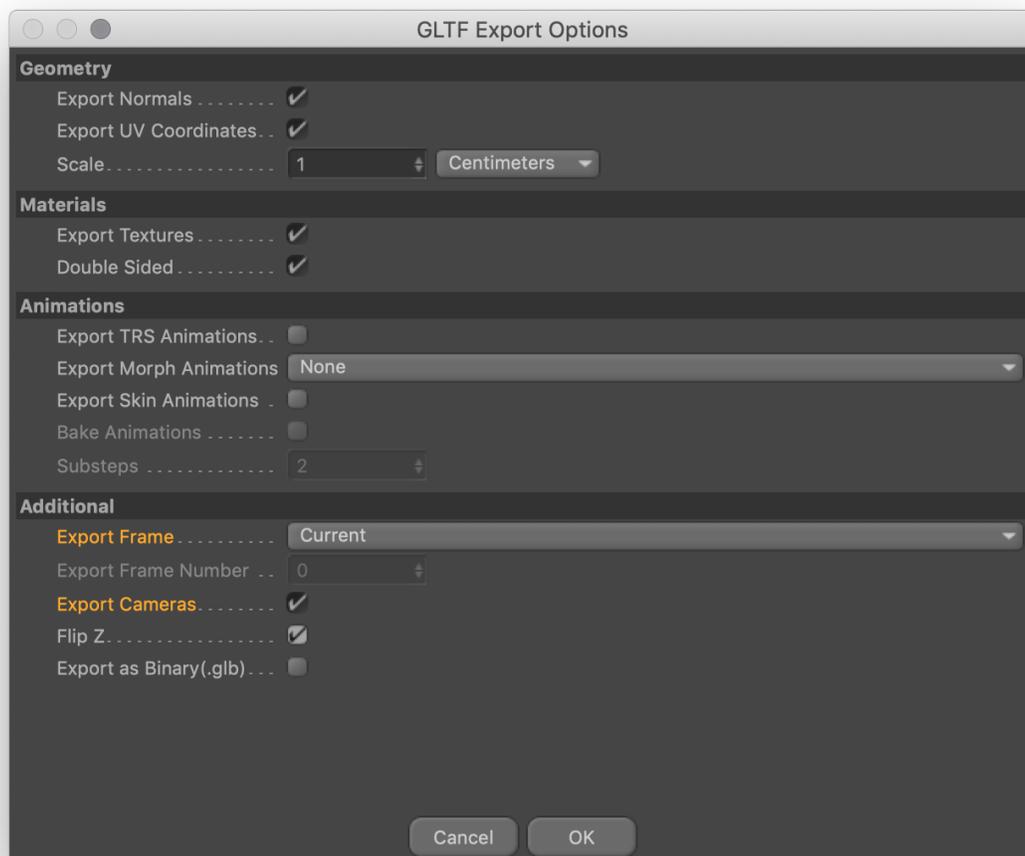
Unzip contents into your Cinema 4D/plugins directory



Exporting and Options

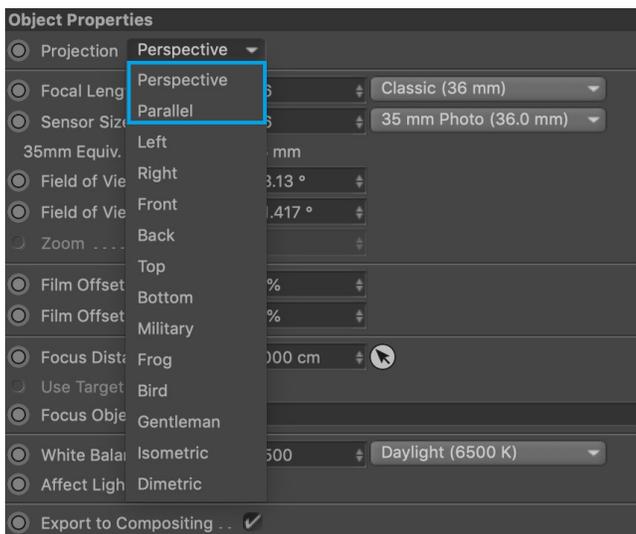
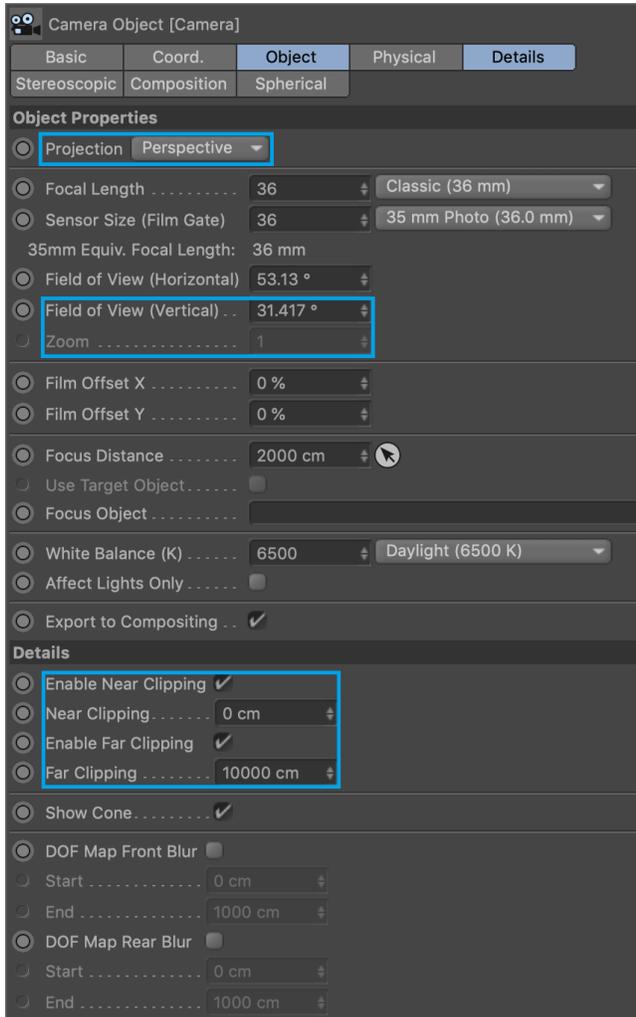
To Export the scene in glTF format go to File>Export>GLTF (*.gltf), select a destination and click Save. An Options window will pop up with the following options:

- Export Normals: Deselect to get Flat Shading.
- Export UV Coordinates
- Scale: Convert the units of the scene.
- Export Textures
- Double Sided: Select to prevent Back-face Culling.
- Export TRS Animations: Export TRS animation tracks.
- Export Morph Animations: Export either Pose Morph animation or PLA animation.
- Export Skin Animations: Export TRS animation tracks of joints.
- Bake Animations: Bake every frame of animation.
- Substeps: Number of substeps to sample between each two frames.
- Export Frame: Export state of document at current frame or at another specified frame.
- Export Frame Number: specify which frame to export.
- Export Cameras.
- Flip Z: Switch from Left Handed to Right Handed System.
- Export as Binary(.glb): Export glTF file in its binary form.



Cameras

glTF 2.0 has a basic support for cameras, perspective projection and parallel(aka orthographic).



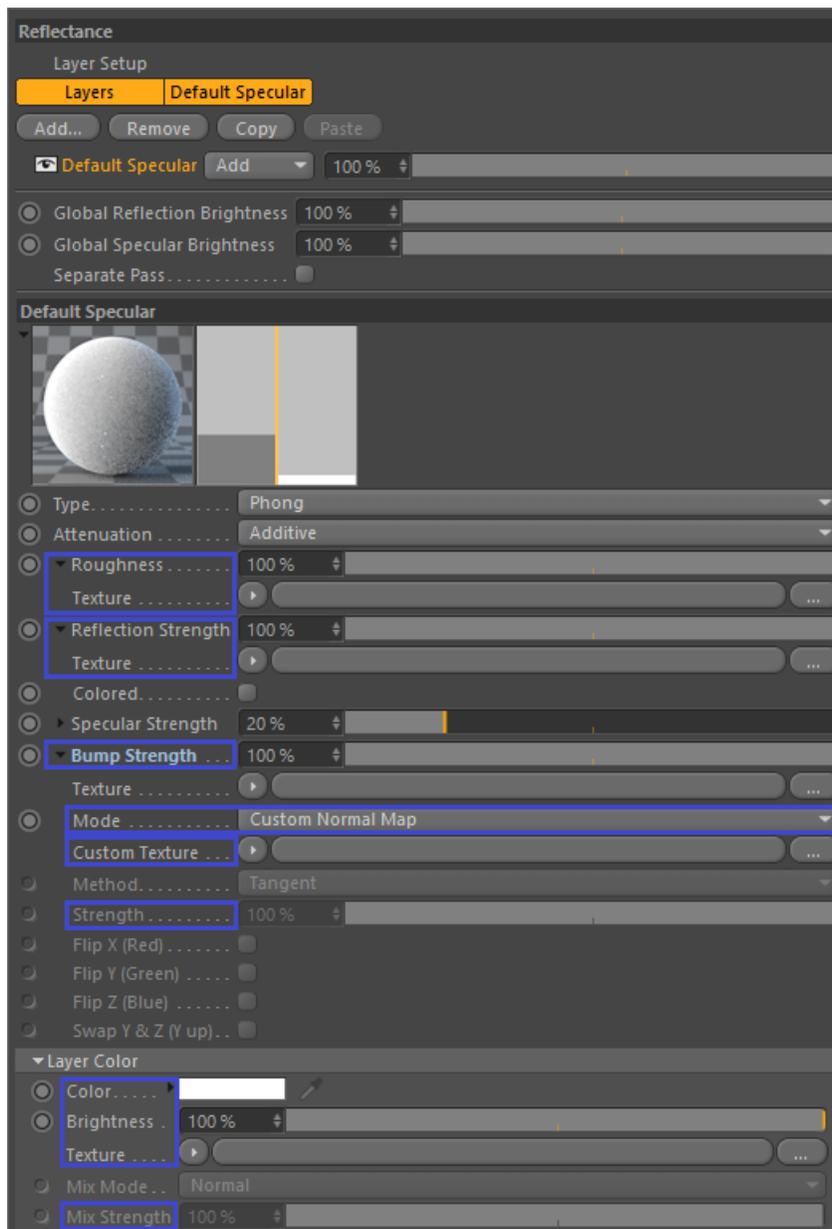
Scale

You can convert the units of the scene at export. I'll provide a few examples to explain how it works:

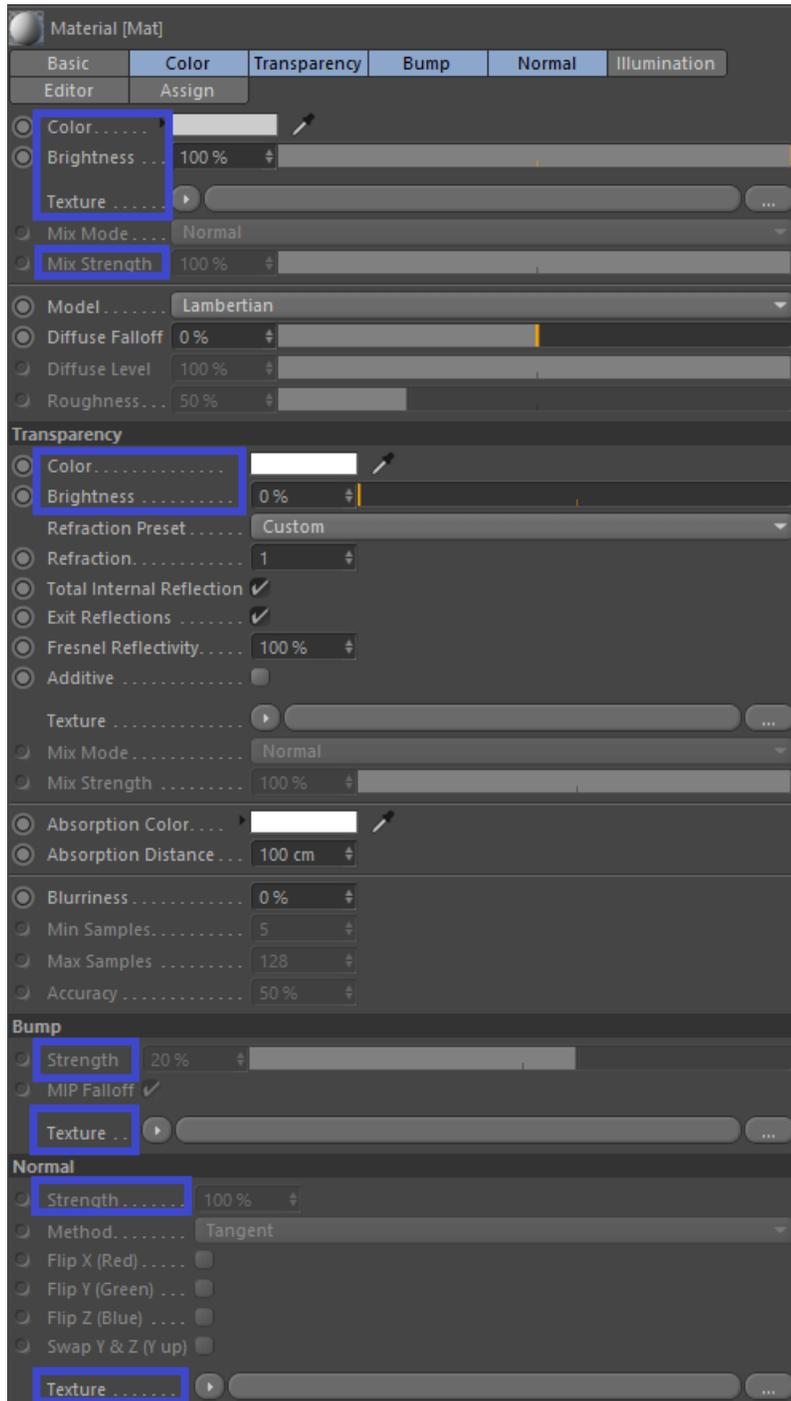
C4D value	Scale option	Exported value
100 cm	1 Centimeter	100
1 cm	1 Meter	0.01
1 m	1 Centimeter	100
1 m	1 Meter	1

Materials

glTF takes the PBR approach to describe Materials, which translates pretty well to Cinema's Reflection Channel, but is limited in some aspects. One of those is the lack for multilayered material support. And so, to get a similar looking result on export, we recommend working with a single Reflection/Specular layer in the Reflection Channel.



(*) Supported fields are Highlighted in blue



(*) Supported fields are Highlighted in blue

PBR Metallic Roughness

glTF's PBR Material's information is encoded into three main parameters:

- Base Color: set by Layer Color's Color selection or by Color selection in Color Channel.
- Metallic Value: set by main reflection layer's Reflection Strength value.
- Roughness Value: set by main reflection layer's Roughness value.

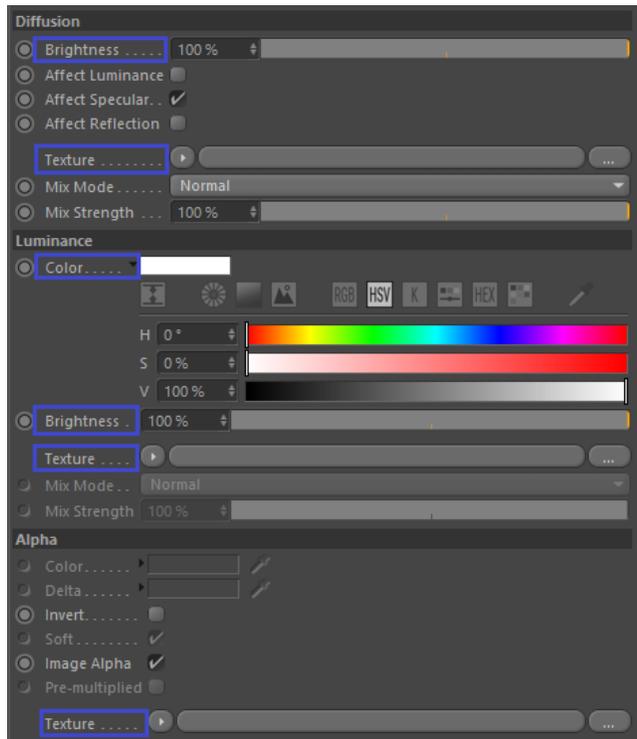
Texture maps to the aforementioned parameters are also supported:

- Base Color Texture: set Bitmap Texture in Layer Color or in Color Channel.
- Metallic Texture: set Grey-scale Bitmap Texture under Reflection Strength.
- Roughness Texture: set Grey-scale Bitmap Texture under Roughness Strength.

Normal Map

Normal Maps can also be set in the main Reflection/Specular layer, under Bump Strength, set Mode to "Custom Normal Map" and set the Bitmap in Custom Texture. Can also be set in the Normal Channel and the Bump Channel.

Other Maps



(*) Supported fields are Highlighted in blue

Occlusion Map

Occlusion Map is set in the Diffusion Channel.

Emissive Factor and Map

Emissive Factor(Color) and Map are set in the Luminance Channel.

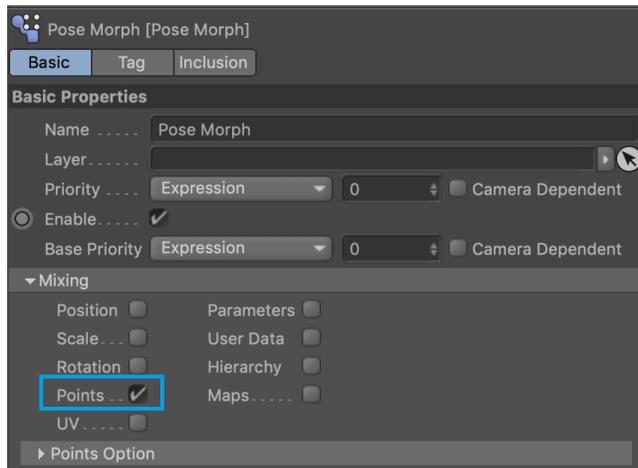
Alpha Map

Alpha Map is set in the Alpha Channel.

Morph Animations

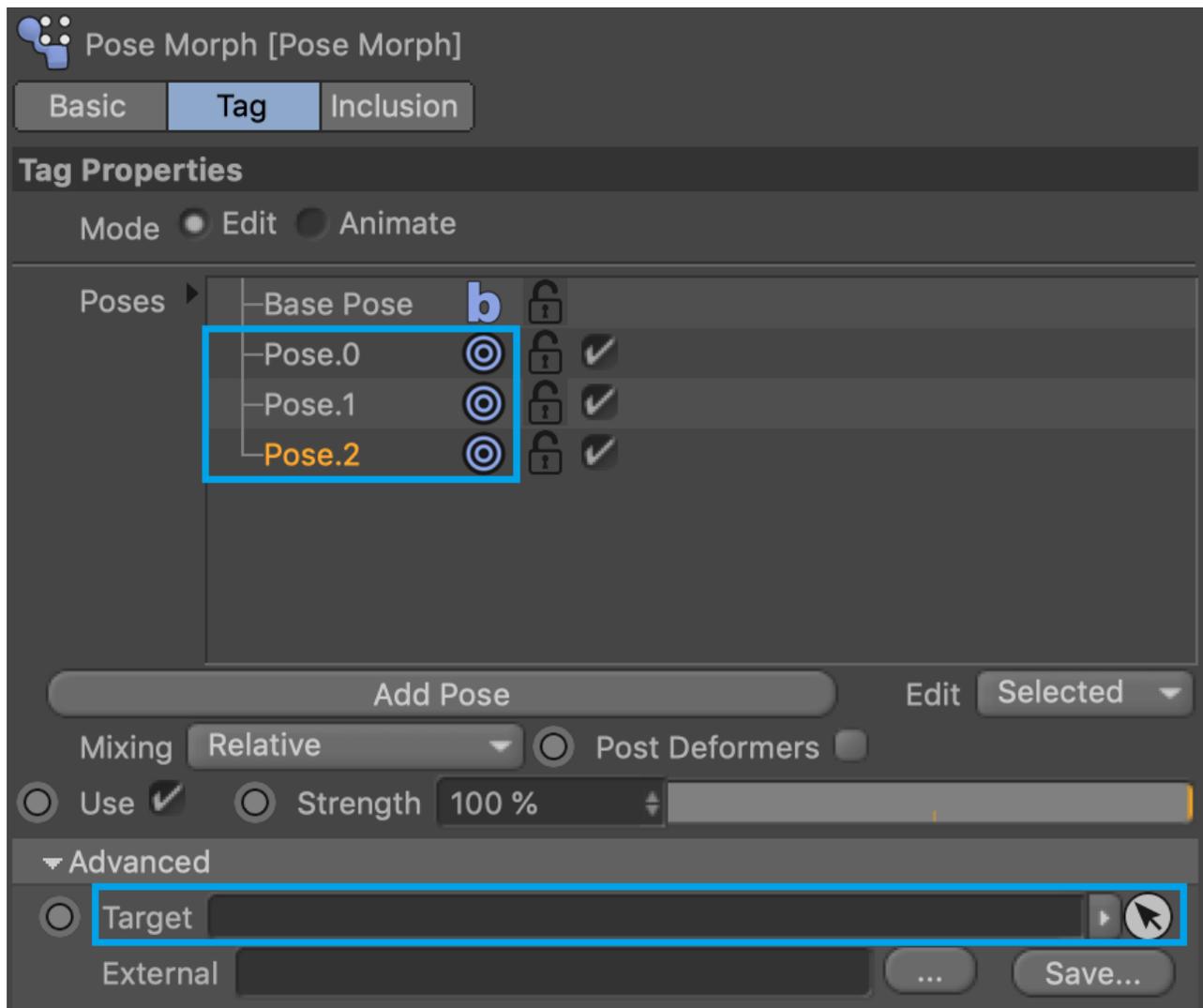
Pose Morph

The current glTF standard supports morph targets which are added, with different weights, to the original mesh. The weights can also be animated.



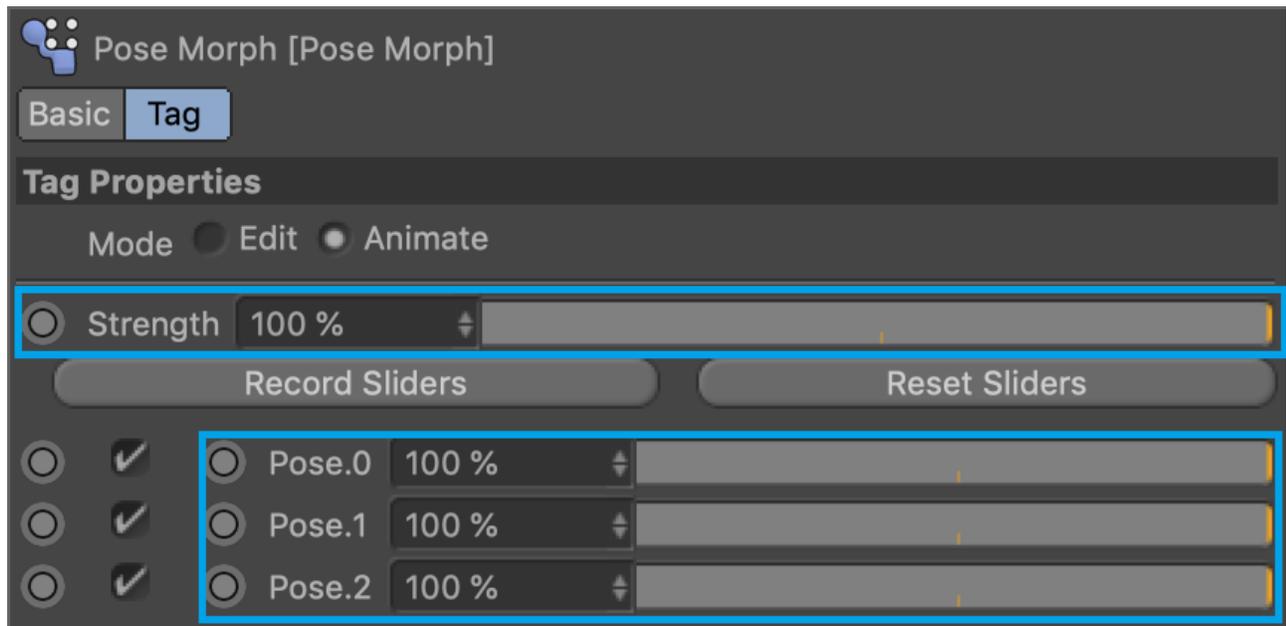
(*) Supported fields are Highlighted in blue

In Cinema 4D, Morph targets are called *Poses*, and the weights are *Strengths*.



(*) Supported fields are Highlighted in blue

Setting a Target overrides the internal poses defined in the tag.



(*) Supported fields are Highlighted in blue

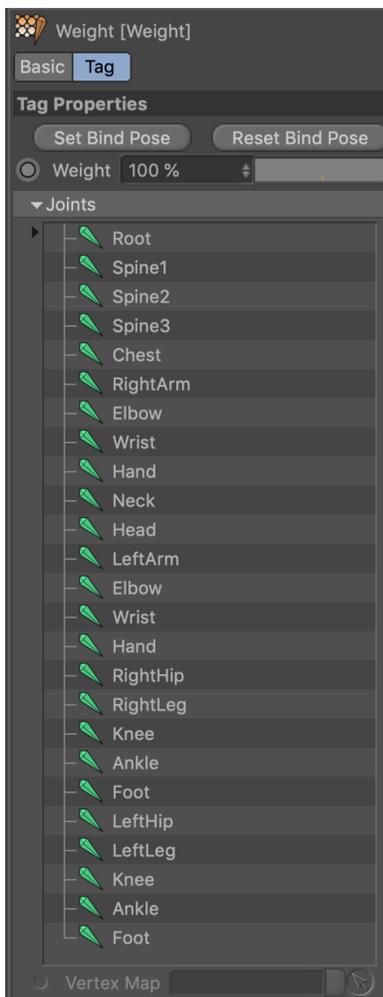
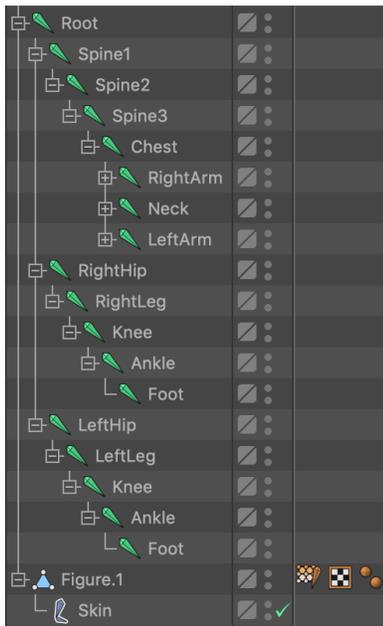
The strengths can be animated.

PLA

PLA isn't technically supported in the current glTF standard, but I was able to achieve this by considering the object at different keyframes to be different morph targets.

Skin Animations

Skin animation is described as the deformation of a mesh according to TRS tracks of joints that are bound to it and assign weights to its vertices.



Editor Mode

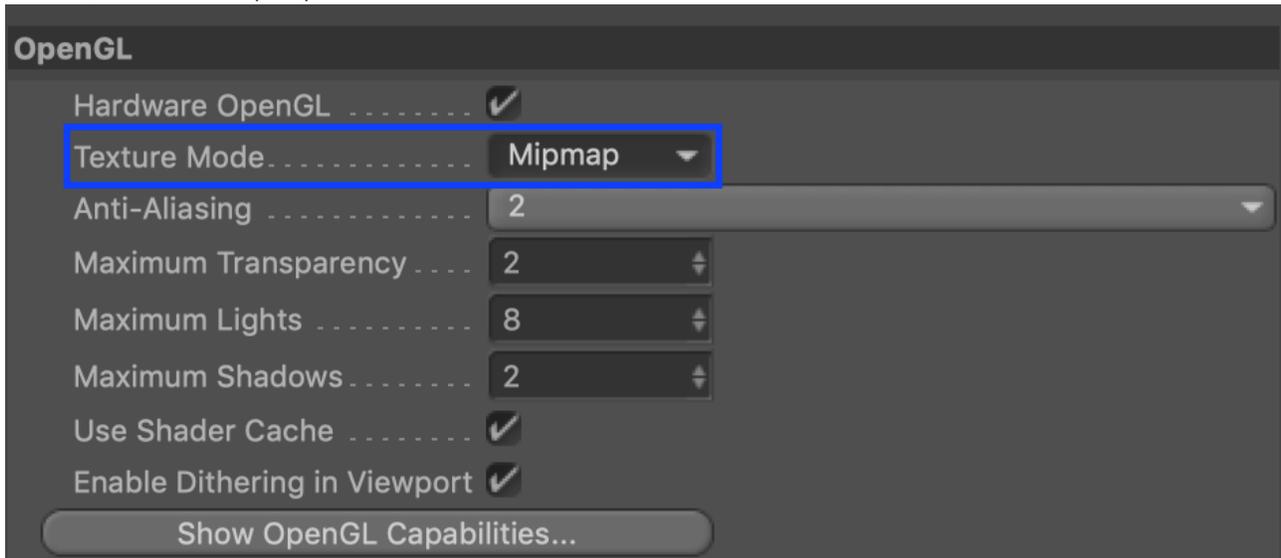
Visibility of the object can be turned off by setting the Editor Mode to Red. Technically there's no support for a visibility boolean in the current standard, this is achieved by setting the material's alpha to 0.



(*) Supported fields are Highlighted in blue

Texture Mode

The Texture Mode of the gltf file is defined by the Texture Mode option in Preferences>OpenGL, select between Nearest, Linear and Mipmaps.



(*) Supported fields are Highlighted in blue

Important notes:

1. Reflection Strength translates very closely to glTF's Metallic Value, but there's a difference... In Cinema 4D, Reflection Strength describes how much light is reflected off the surface, whereas Metallic Value describes how much the reflective behavior of the surface resembles that of a metal. 0% Reflection Strength means that the surface doesn't reflect any light, meaning it's black. But that is not true for 0% Metallic Value.
2. As of now, the automatic baking of Texture Shaders is not yet implemented. If you do use Shaders, please manually Bake them using the Bake Texture tag and re-set them as Bitmaps.
3. In glTF, the Metallic and Roughness Maps are actually single Bitmap where the values are incorporated into the Green and Blue Channels. So when setting a Reflection Strength and Roughness maps, please make sure they are the same dimensions. If they are not, then the Reflection Strength map will be scaled to match the Roughness map.
4. Same goes for the Color Texture and Alpha Map, as they will be merged into a single RGBA Bitmap.
5. Mix Strength: glTF mixes between the Main Color and the Texture by component-wise multiplication of the RGB values. This required an adjustment method of the Color value or the Texture pixels at Export to achieve the same mix as in Cinema 4D. It's still WIP and artifacts may appear in the result.
6. When setting the Normal map in the Reflection Channel... The strength, or "scale", of the Normal Map is defined by the product of Bump Strength and Strength.
7. Proper Error Descriptions are yet to be included, if you run into an error, it will most likely say "Out of Memory..." but it's very rarely so.
8. Polygon Selection is not yet supported, please manually Split the object and apply the different materials accordingly.
9. If Color Channel is enabled, its properties override the Layer Color properties set in the Reflection Channel(i.e. Color, Brightness, Texture and Mix Strength)
10. If Normal maps are set in the Reflection Channel, Normal Channel and Bump Channel. Then the Normal map and its Strength are retrieved from the Reflection Channel first, Normal Channel second, and Bump Channel last.
11. Cubic Spline animation curves: in Cinema 4D, two coefficients define the tangent of every key: "value" and "time". In glTF, a tangent is defined only by "value". If your exported animation doesn't look the same, consider baking the animation.
12. Linear/Step curves: An animation property(Translation/Rotation/Scale) will be exported as Linear/Step Interpolation only if all its keys interpolations(in the x,y,z curves) are set to Linear/Step, otherwise it will be exported as Cubic Spline.
13. Pose Morph OR PLA: It's not possible to export PLA and Pose Morph animations together.
14. Pose Morph global Strength: The global Strength value and its curve in the Pose Morph tag cannot be exported, it is multiplied by the other strength values.
15. Skin Animation Weights: in glTF 2.0, a vertex may be assigned up to 4 weights by 4 joints only! If more than 4 joints influence a vertex, the 4 that assign the largest weights will be exported as the influencers of that vertex.
16. Bake Animation: If your exported animation doesn't look right, try enabling the Bake Animation option.
17. [Animation Range: Exported animation range depends on the Timeline limits defined in the Cinema 4D document.](#)

Enjoy!

Happy Exporting!

Basel

