

UMA - Unity Multipurpose Avatar

UMA version 2.14

Table of Contents

How is content created? 1

 Base Meshes 2

 Slots..... 2

 Overlays..... 4

Asset creation pipeline..... 4

 Texturing and UV mapping..... 5

 3d modeling 6

 Rigging and skinning 6

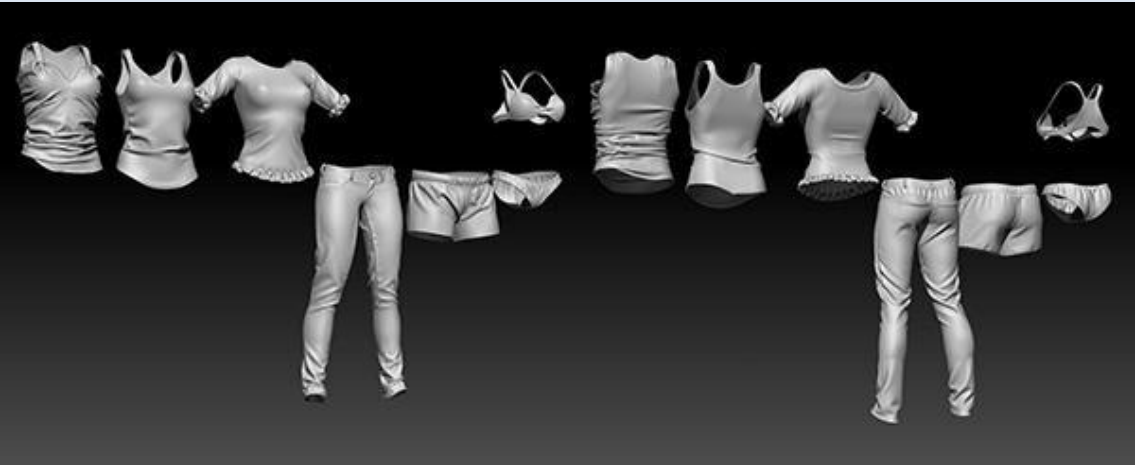
3D software 8

 Blender 2.7 Export**Error! Bookmark not defined.**

 Blender 2.8 Export 8

 3dsmax Export 10

How is content created?



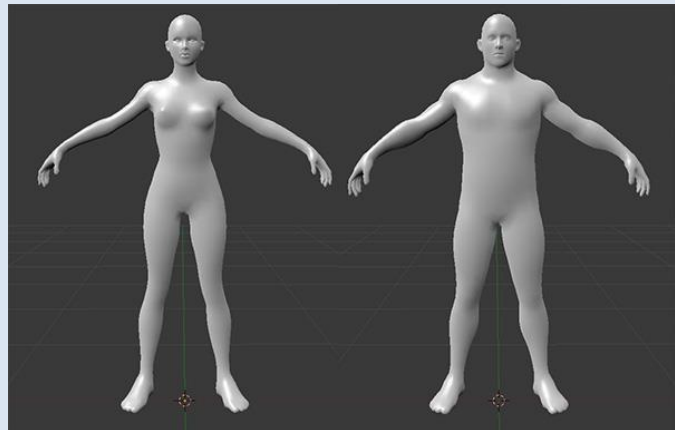
Set of High poly meshes based on UMA final clothes.

Creating content for UMA doesn't require much extra knowledge beyond the usual asset creation pipeline. There are three main type of content: Base meshes, Slots and Overlays.

Base Meshes

UMA provides two starting base meshes, a male and a female. It's possible to create completely different and unique base meshes and still take advantage of UMA. As all content uses base meshes as reference, if you create a minotaur base mesh and use the original male torso base mesh with skeletal deformations, all clothes based on that male torso can be shared with the new Minotaur mesh.

For a completely different race, such as horses, you will need unique content. This could then be shared with other similar races, like unicorns and even dragons. Male and female Base meshes were created considering the average volume those bodies would be able to reach using the adjustments in the UMA rig.



Uma Female and Male base

meshes.

Slots

All UMA content that provides a mesh is a slot. Slots are basically containers holding all necessary data to be combined with the rest of an UMA avatar.

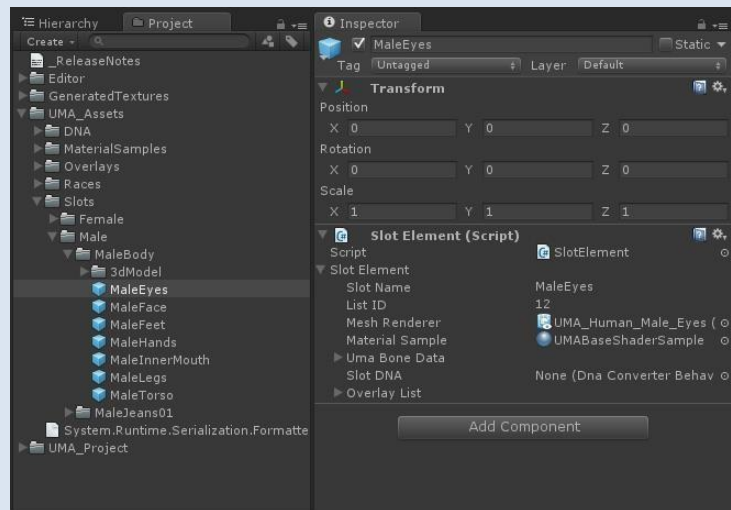
For example, the base meshes provided are normally split into several pieces, such as head, torso and legs... and then implemented as slots which can be combined in many different ways.

An UMA avatar is in fact, the combination of many different slots, some of them carrying body parts, others providing clothing or accessories. Lots of UMA variation can be created simply by combining different slots for each avatar.

Slots also have a material sample, which is usually then combined with all other slots that share the same material. Female eyelashes for example, have a unique transparent material that can be shared with transparent hair. It's necessary to set a material sample for all slots, as those are used to consider how meshes will be combined. In many cases, the same material sample can be used for all slots.

The included UMA materials wrap the most common Standard Shader options. You can, however, create new UMAMaterials for your new slots and set the parameters in the template material and texture channels as needed.

The big difference between body parts and other content is that body parts need to be combined in a way that the seams wouldn't be visible. To handle this, it's important that the vertices along mesh seams share the same position and normal values to avoid lightning artifacts.



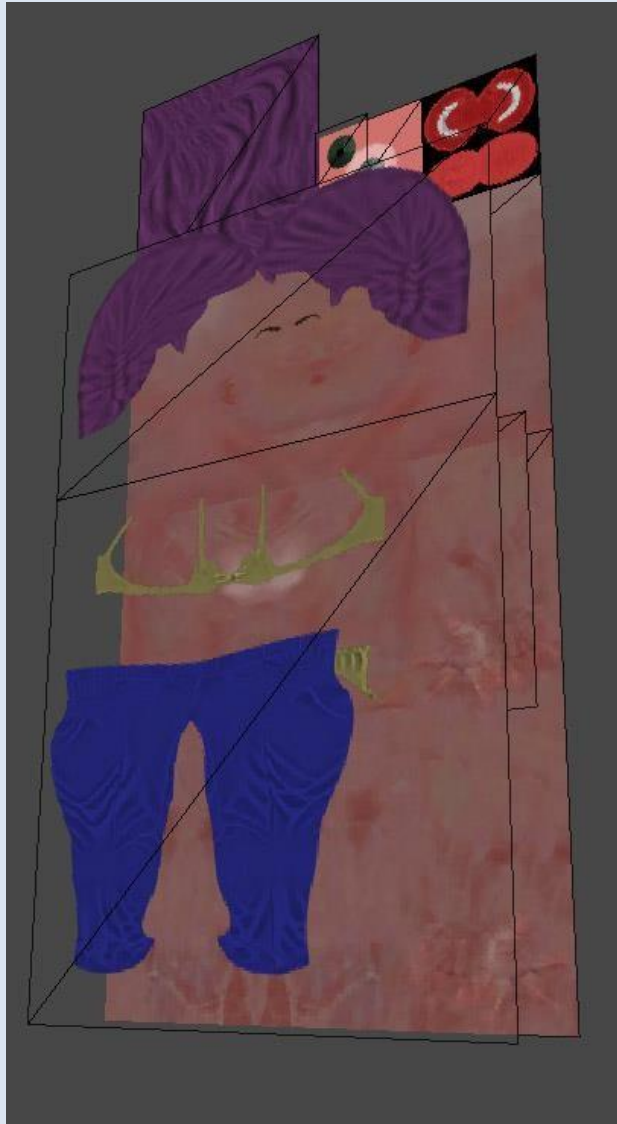
Example of Male slots.

To handle that, we provide a tool for importing meshes that recalculate the normal and tangent data based on a reference mesh. This is the "Slot Builder". See the full documentation for information on using the SlotBuilder.

Overlays

Each slot requires at least one overlay set but usually receives a list of them. Overlays carries all the necessary textures to generate the final material(s) and might have extra information on how they are mapped. The first overlay in the list provides the base textures, and all other overlays included are combined with the first one, in sequence, generating the final atlas.

We created materials using 'BetterShaders' to integrate some of the most common needs into UMA and the adjustments required are quite simple. Even more advanced shaders can be integrated and used. Keep in mind it's possible to provide extra textures on each overlay, this way, it's possible to include displacement maps, Sub-surface scattering masks or any necessary data.



Asset creation pipeline

UMA project provides an UMA content creator pack, available on GitHub here:

[Content Pack](#)

It provides:

- base diffuse, specular and normal map textures
- UV layout reference images
- Male and female base meshes in .OBJ, FBX, ZTL, 3dsMax and .Blend
- Sample clothing from Isbit games

- Marvelous Designer avatar files

Fernando spent a long time recording and producing video tutorials to cover all the basic process of content creation, but the knowledge for working on any 3d software is a prerequisite for UMA. Those original videos are available on his YouTube channel here, but much has changed since the initial version of UMA:

<http://www.youtube.com/user/fernandoribeirogames>

SecretAnorak has created a newer set of content video tutorials on his page here:

[Secret Anoraks Content Creation](#)

These videos use the current UMA and are recommended. Secret does this for free for the community. If you would like to support his efforts, he has a patreon here:

[Secret Anoraks Patreon Page](#)

The overall knowledge for generating UMA content is 3d modeling, rigging, skinning and texturing. It's also possible to work on existing content already available. For example, if you have a t-shirt slot, with the right texture work it's possible to provide an chain mail overlay without extra knowledge of modeling, rigging or skinning.

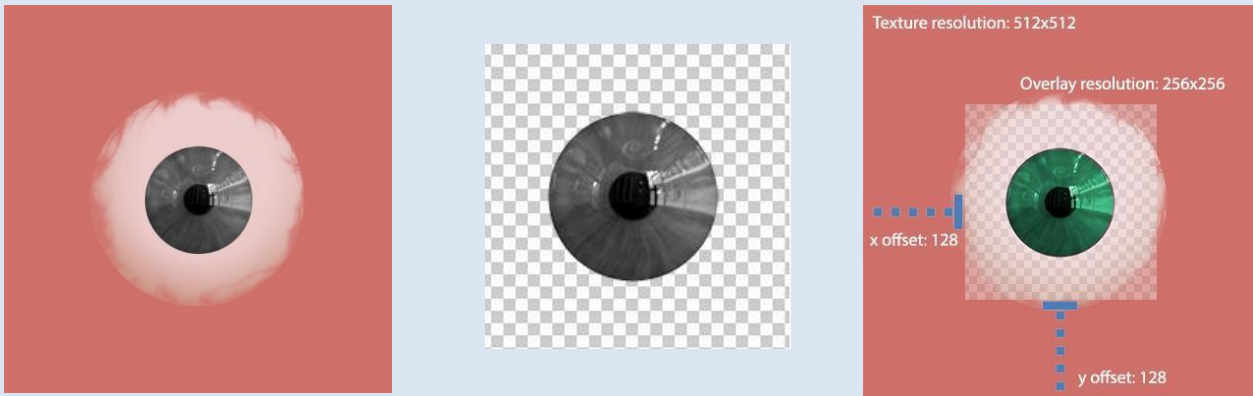
Texturing and UV mapping

When creating new content for UMA, if you're aware it will be covering an area of UMA body texture, it's possible to use that UV area for the new content texture, if it offers enough space. This example can be seen on Female t-shirt and hair, both of them save atlas space, having those textures as overlays for body base texture instead of being base textures themselves.

Also, any overlay texture and it's covered base textures don't need to keep the same size. For example, FemaleUnderwear01 overlay covers only the left half of the base, so it's possible to have that overlay with half the width of body base texture.

The Rect provided together with the overlay elements is responsible for keeping information of the positioning adjust of the cropped overlay, relative to base texture coordinates.

It's possible to provide overlays that will receive color adjusts at atlas creation. In those cases, usually the predominant color is white or gray in those areas, to have a neutral influence over final color.

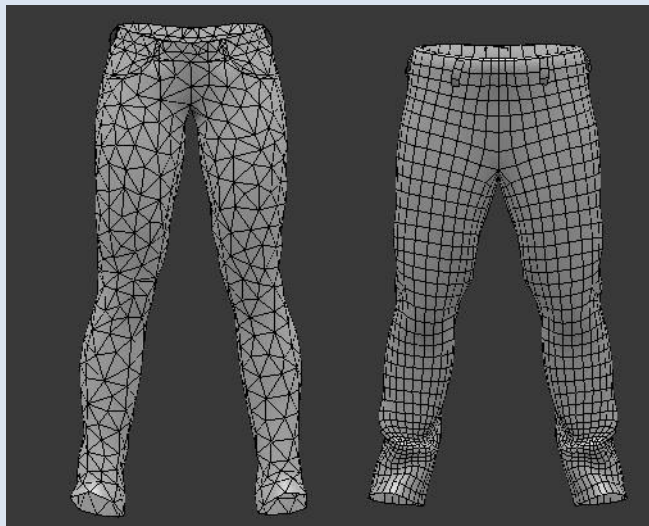


Above image illustrates the use of a cropped overlay in junction with Eye base texture to generate iris color variation

Overlay Rect values should be specified in UV coordinates, so that overlay sizes do not matter.

3d modeling

It's possible to integrate any 3d mesh into an UMA avatar, it's important to follow the same optimization guidelines usually used for traditional characters and clothes, as topology and vertex count. I've included meshes with both uniform and non uniform polygon placement, it's important to keep in mind when each case can be used. The same way, I've worked both on meshes mostly quad based, and meshes entirely based on tris before exporting process. All content can be integrated despite those differences, but stretched polygons might cause poor lightning results, especially visible when not using normal maps. You can even vary the density and topology of additional slots, as long as you provide enough geometry to deform the model as needed.



Female and Male jeans completely different topology

Rigging and skinning

This is where all the UMA magic happens. For the default characters, all shape variations we can achieve on UMA avatars are a consequence of changing bone scales, positions and rotations. A mesh following those changes depends on rigging and skinning entirely

and an incorrect skinning process might lead to issues such as clothing pieces not following the same shape variation as the body. Note that UMA can also use Blendshapes. When using Blendshapes, the shapes have to be propagated from the character to all of the wearable objects also. UMA will keep track of the Blendshapes and make sure they all match.

Base meshes have been provided with rig and skin data because most 3d software has specific tools that allow users to transfer bone weight data between different meshes avoiding most of the time-consuming process involved. Fernando created some of the original videos showing shown this being done in Blender here:

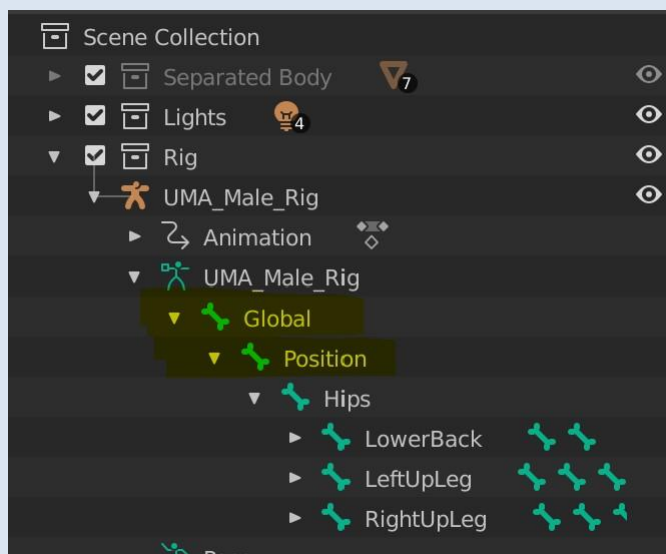
<https://www.youtube.com/@fernandoribeirogames/videos>

SecretAnoraks videos are available here and are more current:

<https://www.youtube.com/@SecretAnorak/videos>

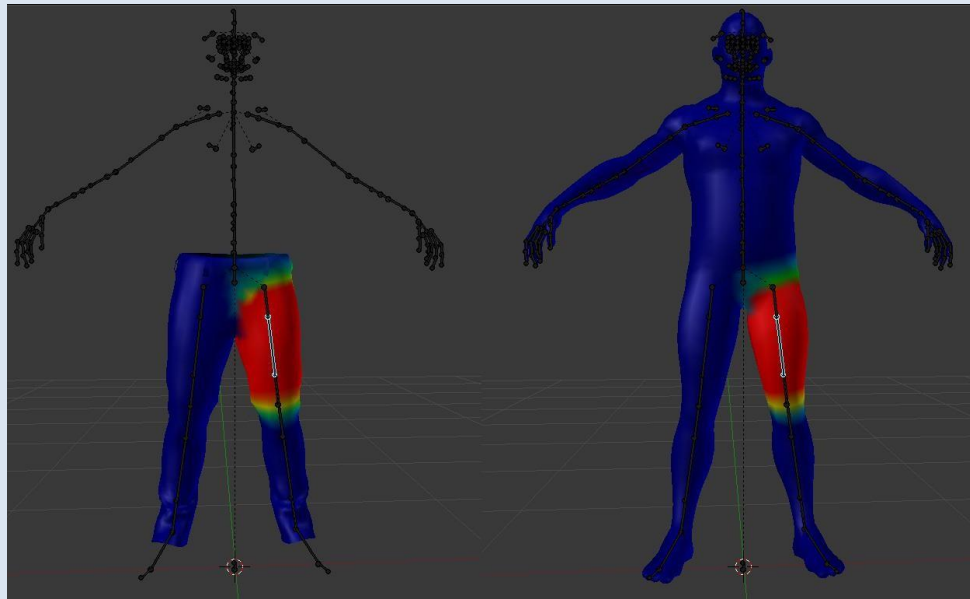
UMA requires that your rig and mesh be normalized – that is, that they don't have any visual transform that is not applied. This will vary depending on the software that you use.

If you are creating a new race, **UMA requires two helper bones at the top of your hierarchy. "Global" and "Position"**, as in the following figure:



For dresses or armor, simply projecting those values might not be enough, and skinning knowledge may be necessary for the best results to follow body variations. It is important to keep in mind that if you are creating your own rig, you should include a pair of parent and child bones wherever you need a specific area to receive non-uniform changes. This applies to most of the UMA rig, such as the arms. You can change the arm scale in both uniform and non-uniform ways, but having a non-uniform scale deforms all child bones too, so those changes need to be applied to the child bones working as a pair. It is also important to remember that it is always

these child bones that carry the skinning data, as they change directly under the influence of the parent bone.



Male jeans skinning and male base mesh skinning. Skinning Data has been projected with Blender Transfer weights tool

3D software

Blender is the UMA standard content creation software, because it's open source and accessible for all developers. Being the standard means I'll provide most of the 3d tutorials on this specific application but keep in mind most of the available software has a set of tools to handle what I'm doing in blender.

Blender 4.X Export

Previous versions of blender required your UMA rig to be rotated 180 degrees. This is no longer required. If your rig is 180 degrees (ie, looking at the "front view" shows the back of the player), then that rotation should be removed. This will usually require you to rotate all the meshes as well (but not in every case).

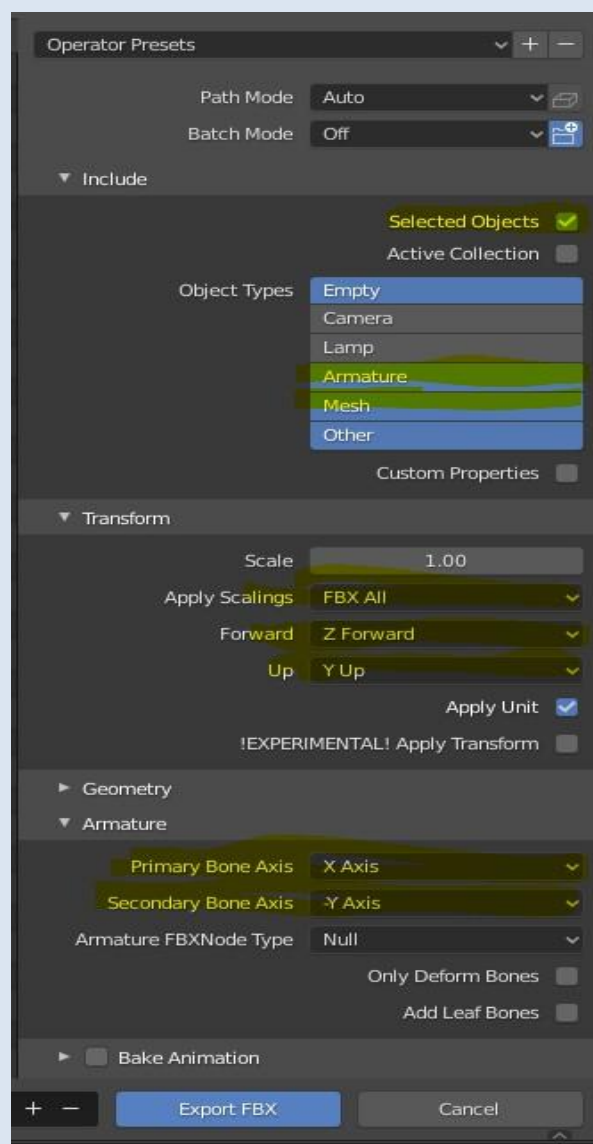
Exporting from **Blender 4.X** requires you to set the "*forward axis*" as "*Z forward*" and "*Up Axis*" as "*Y Up*". In addition, you must change the Armature settings, **Primary Bone Axis should be "X Axis"** and the **Secondary Bone Axis should be "-Y Axis"**, and Apply Scalings should be "FBX All". The asset will **not** need to be scaled in the import settings in Unity.

It's very important when exporting content being sure that the transforms are all applied – that position, rotation and scale are normalized, this means you need to bake all those into vertices position. Select all meshes / rigs you want to export. To do this, go

into object mode, and select all meshes and the rig you want to export, and choose Object/Transforms/Apply All.

If you are creating a new race, or adding bones to an existing rig (like for a ponytail) You should also make sure that the bones are normalized. To do this, go into pose mode, select all the bones, and use 'Apply pose as rest pose'.

You usually don't need to include animations when exporting clothes and accessories, but it's important that the rig is exported along with the mesh for your content – otherwise, Unity will not create a SkinnedMeshRenderer on import, and this is required.



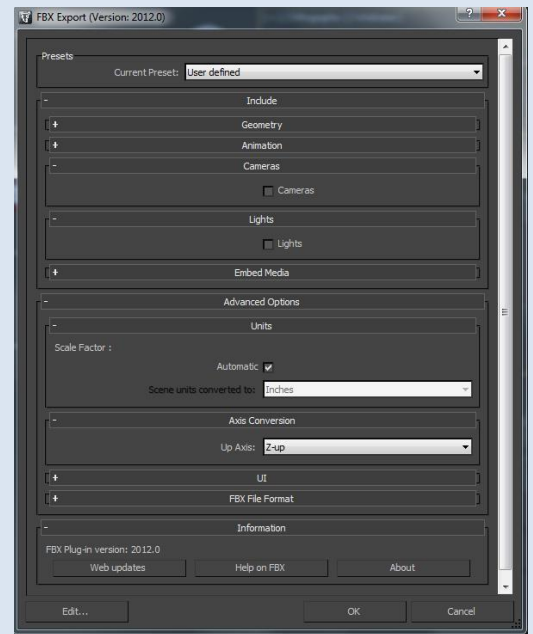
3dsmax Export

Importing files: 3dsMax files are available in the GitHub content link posted above. We recommend using Blender whenever possible.

3ds max provides modifier “*skin wrap*” as a solution for copying skinning data between UMA base mesh and your own content.

Exporting files: The same warning I’ve provided for blender users apply here too: It’s very important when exporting content being sure both it’s position, rotation and scale are normalized. It’s possible to adjust mesh pivot using the “*Affect Pivot Only*” button. There’s a specific adjusts required when creating your own content, the Z rotation value of the mesh should be set to 180 degrees, this is usually done entering *Affect Pivot Only* mode and rotating the pivot itself.

When exporting the fbx files, you need to have both your content mesh and rig selected. It’s usually a good idea not including the cameras and lights on the fbx files. For exporting, units can be kept on “*automatic*” and “*Up Axis*” need to be set to “*Z up*”.



Maya Export

Exporting files: The same warning I’ve provided for blender users apply here too: It’s very important when exporting content being sure both its position, rotation and scale are normalized.

In Maya then make sure that you have your bones should be oriented in Maya World axis default Y-Up and not as Blender which use Z-Up, if so, all you need then is to rotate Root -90 on X and then save the maya file (do not use FBX).

