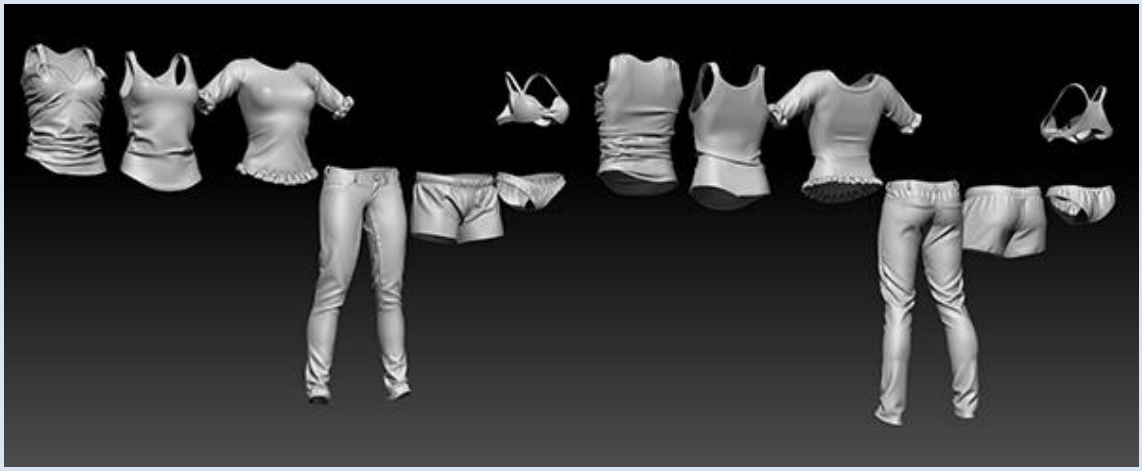
UMA - Unity Multipurpose Avatar

UMA version 2.8

Table of Contents

How is content created?



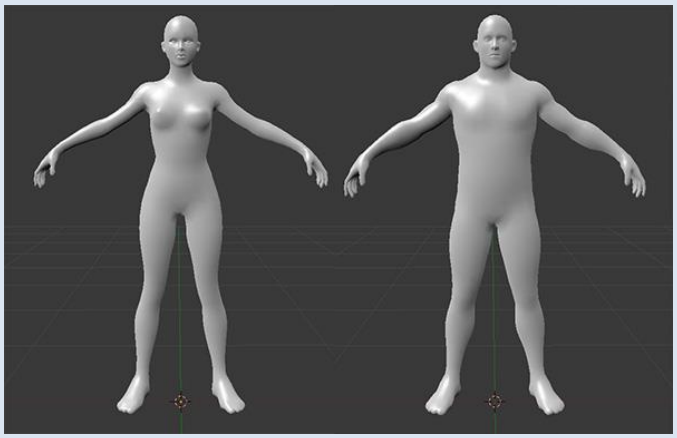
Set of High poly meshes based on UMA final clothes.

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 are able to keep the same data from the original male torso base mesh, 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.

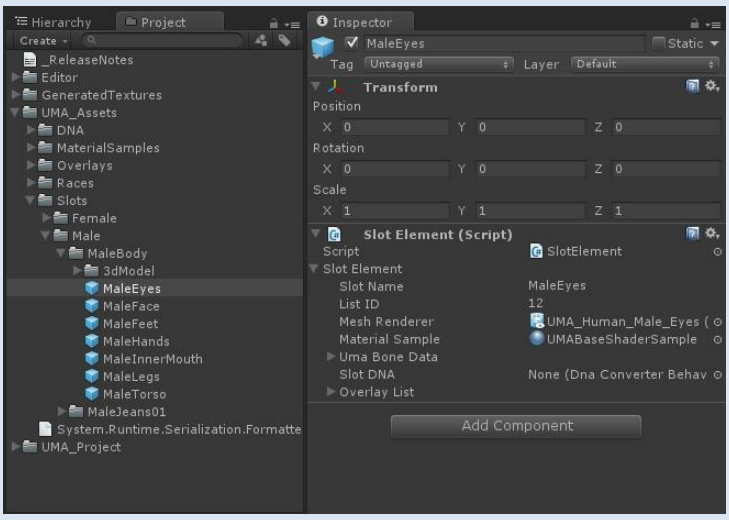
For example the base meshes provided are normally split into several pieces, such as head, torso and legs... and then implemented as slots which can be combined in many different ways.

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 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.

UMA standard avatar material uses a similar version of Unity´s Bumped Specular Shader, but UMA project provides many other options.

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.



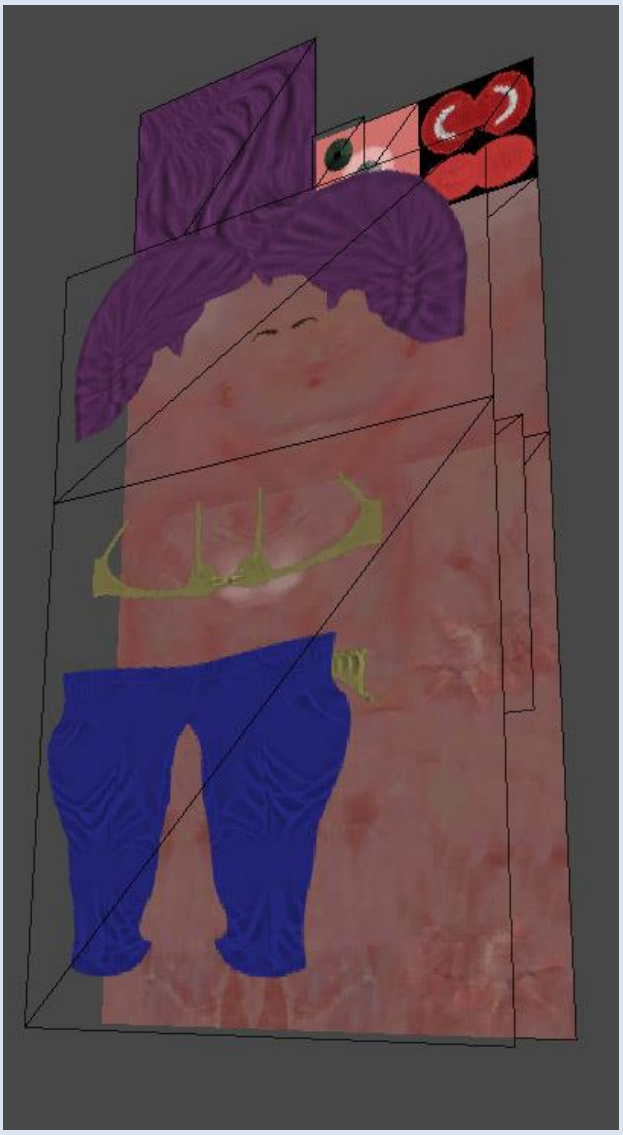
Example of Male slots.

To handle that, we provide a tool for importing meshes that recalculate the normal and tangent data based on a reference mesh. UMA MaterialBuilder will be explained in following pages.

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.

UMA standard shader requires two textures provided by overlays, one texture for Diffuse color (RGB) + overlay mask (A) and one texture for Normal map(GA), Specular (R) and Gloss (B). These non-standard textures let us compress a lot of information in only two textures, reducing final memory usage.



Example of overlay composition

Together, Joen Joensen and I have worked on “UMA material builder”, a tool that receives 3 standard textures for Diffuse(RGB) + mask(A), Normal map(RGB) and specular(RGB) + gloss(A), and compacts the data into the two textures described above. In the process, the specular color is reduced to one channel of data, the resulting average color of the 3 channels provided.

Following Pedro Toledo's post “Brief Considerations About Materials” http://www.manufato.com/?p=902, Joen and I worked together to reach two shader that handle specular color based on diffuse color reference resulting in Dielectric and conductor materials.

I´ve already managed to integrate some of the most common Unity shaders into UMA and the adjusts 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

Fernandospent a long time recording and producing video tutorials to cover all the basic process of content creation, but the knowledge for actually working on any 3d software is prerequisite. Those original videos are available on his youtube channel here:

(http://www.youtube.com/user/fernandoribeirogames)

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

Secret Anoraks Content Creation

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 an tshirt 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

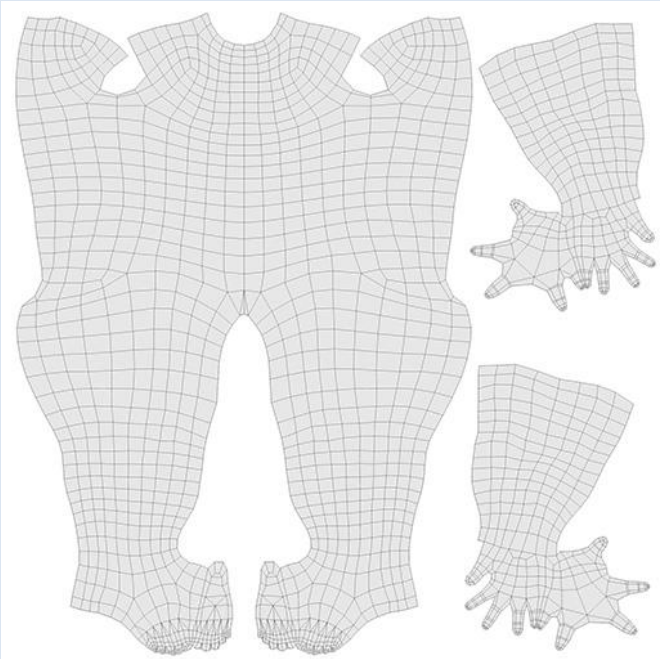
Both UMA male and female base textures have a specific resolution. Below is the list of those base texture sizes in pixels:

- Head : 2048x1024

- Body : 2048x2048

- InnerMouth : 512x512

- Eyes : 256x256



Male body UV layout

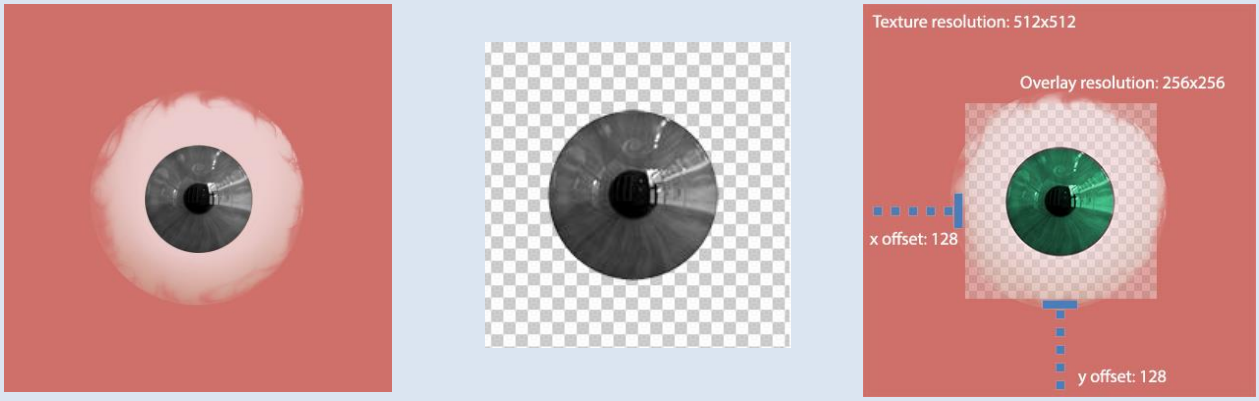
Those values are specific to the provided base meshes as a standard for anyone creating content for them. If you plan following a different standard, it´s possible to use any different resolution.

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 tshirt 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.

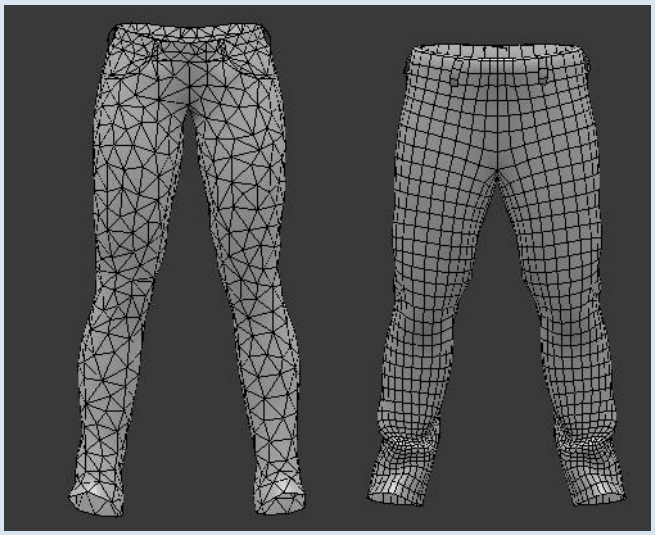


Above image illustrates the use of a cropped overlay in junction with Eye base texture to generate iris color variation

In other hand, UMA 1.0.1.0R don´t support combining textures with different original size: Even cropping an overlay removing the unused masked area, the original overlay size should be the same of the base texture. So if you provide a cropped overlay for eyebrows, it requires a Rect data relative to the 2048x1024 head reference texture. Providing a non cropped overlay with smaller or bigger resolution than the base overlay might generate error messages.

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.



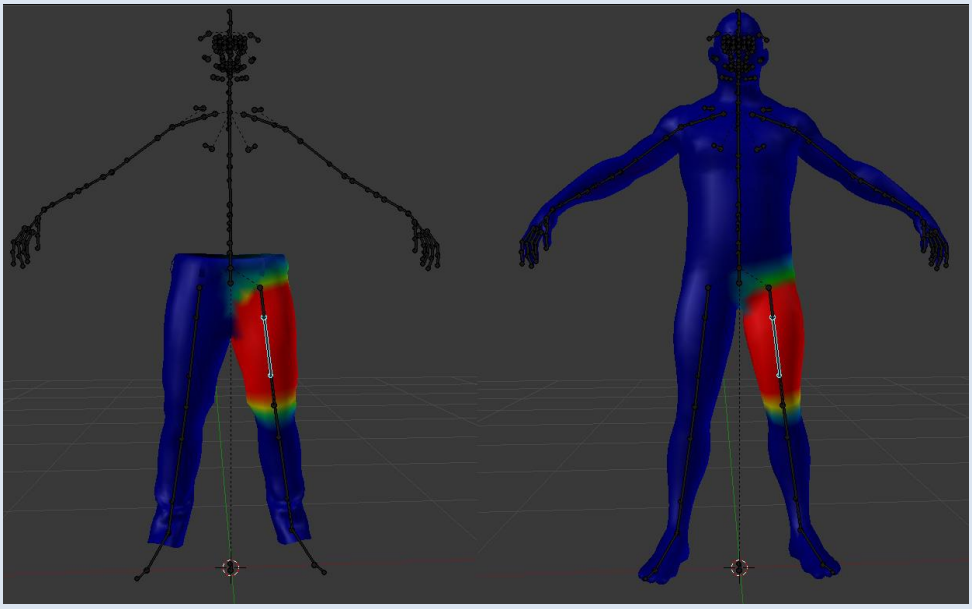
Rigging and skinning

This is where all the UMA magic happens. All shape variations we can achieve on UMA avatars are a consequence of changing bone scales, positions and rotations. A mesh correctly following those changes depends on rigging and skinning entirely and an incorrect skinning process might lead to issues such as a clothing pieces not following the same shape variation as the body.

I´ve provided base meshes 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. I´ve shown this being done in Blender at some video tutorials (http://www.youtube.com/watch?v=ImD6APS0xek), but the same can be done with XSI Gator or other specific solutions.

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.

For dresses or armor, simply projecting those values might not be enough, and skinning knowledge may be necessary for best results following body variations. What is important to keep in mind if you´re creating your own rig is to keep a pair of parent and children bones anywhere you need that specific area to receive non-uniform changes. This is the case on most of the UMA rig, for example the arms. You´re able to change the arm scale both in uniform and non-uniform ways, but having an non-uniform scale deforms all child bones too, so those changes need to be applied to the child bones working in pair. It´s also important to keep in mind it ´s always these child bones that carry the skinning data, as it will always change directly, and 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

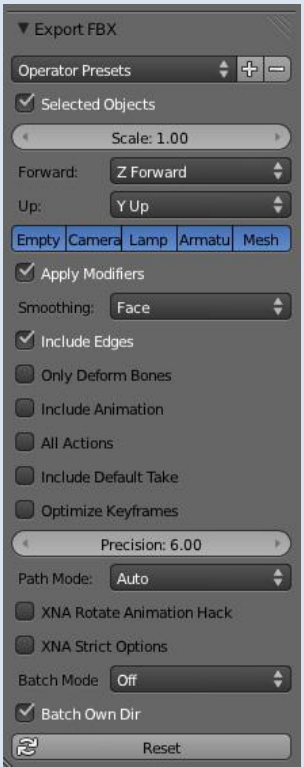
Blender is UMA standard content creation software, because it´s open source and accessible for all developers. Being the standard means I´ll provide most of the 3d tutorials on this specific application but keep in mind most of the available softwares have a set of tools to handle what I´m doing in blender.

Initially, I planned including here the standard import and export setting for each 3d software I manage testing the integration with UMA, but as you´re going to notice, importing setting mostly depends where those files where generated, so it´s hard having an standard. For export setting, I´ll explain the specific setting for exporting content for UMA base mesh.

Blender 2.7 Export

The blend files for the included human meshes are available at the github location above. You can fit your clothing to the skinned mesh and then transfer the weights using the "Data Transfer" modifier. Put the data transfer modifer on your new clothing, the Source Object should be the skinned mesh. Select Vertex Data, Nearest Face Interpolated, Vertex Groups, all layers, and then press "Generate Data Layers" to copy the rigging to the new model. Be sure to apply the modifer. Then, review the wieghts, and adjust using weight painting as needed. Because of the coordinate issue and the old FBX export, the rig should be rotated 180 degrees (do not apply).

To apply transforms – select your mesh, and go to object/apply/rotation and scale, and object/apply/location. Select your rig, and verify your transforms are applied there. If not, go into pose mode, and choose pose/ apply/apply visual transform to pose.



Exporting files (Before Blender 2.8): Exporting fbx files of rigged and skinned content to unity requires having "*forward axis*" as "*Z forward*" and "*Up Axis*" as "*Y Up*".

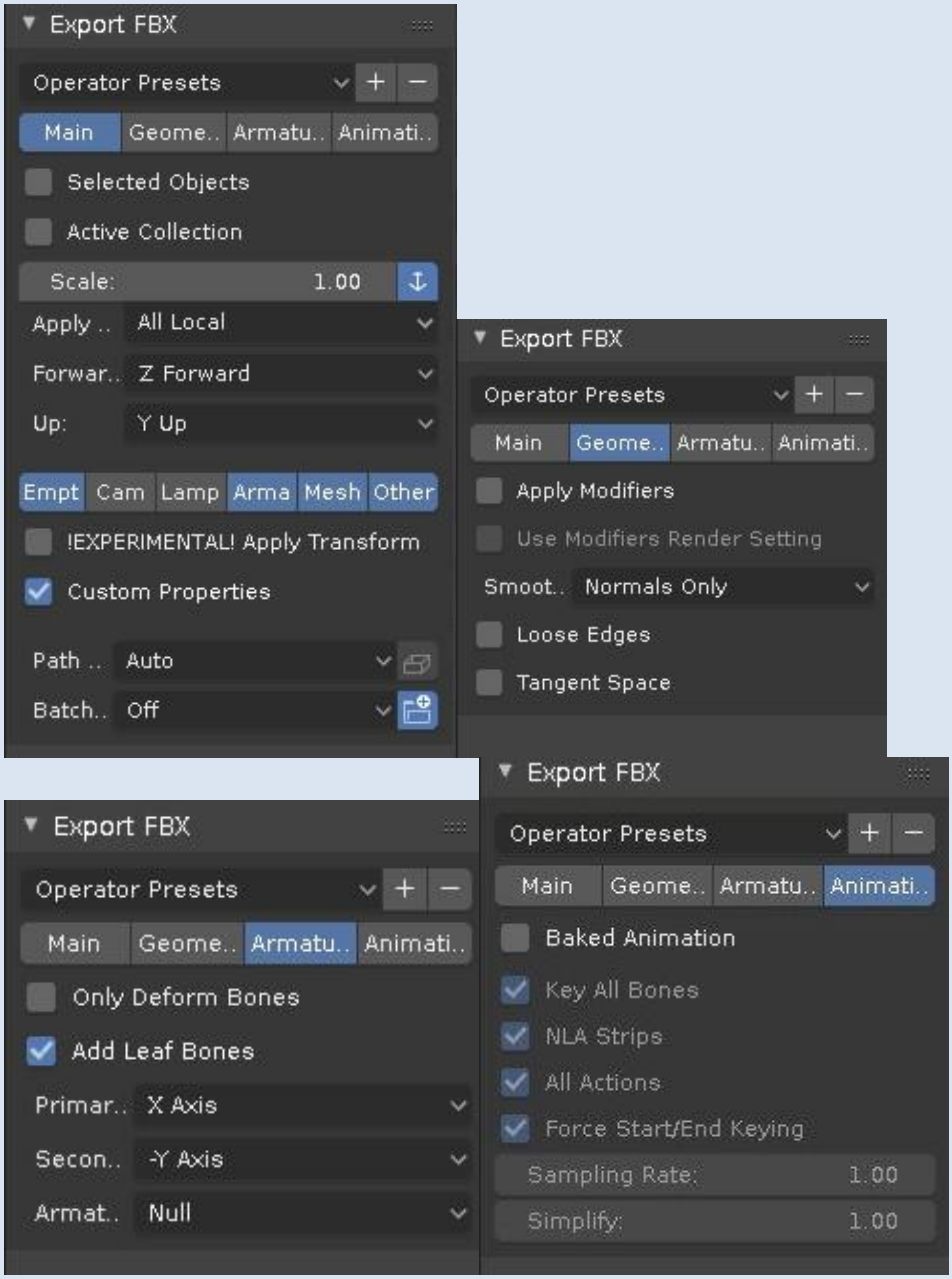
Blender 2.8 Export

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 2.8** requires you to set the "*forward axis*" as "*Z forward*" and "*Up Axis*" as "*Y Up*". In addition, you have to change the Armature settings, **Primary Bone Axis should be "X Axis"** and the **Secondary Bone Axis should be "-Y Axis"**. The asset will usually need to be scaled 100x in the import settings in Unity.

It´s very important when exporting content being sure both it´s position, rotation and scale are normalized, this means you need to bake all those into vertices position. This can be made with "*Ctrl+A*" or "*Object/Apply*".

You usually don´t need including animations when exporting clothes and acessories, but it´s really important that the rig is exported along with the mesh for your content. Including the rig might raise the disk space required by that file, so it´s usually a good idea to keep a bundle of meshes on the same fbx file whenever possible.


Export FBX
Operator Presets
Main
Geome..
Armatu..
Animati..
Selected Objects
Active Collection
Scale:
1.00
Apply ..
All Local
Forwar..
Z Forward
Up:
Y Up
Empt
Cam
Lamp
Arma
Mesh
Other
!EXPERIMENTAL! Apply Transform
Custom Properties
Path ..
Auto
Batch..
Off
Export FBX
Operator Presets
Main
Geome..
Armatu..
Animati..
Apply Modifiers
Use Modifiers Render Setting
Smoot..
Normals Only
Loose Edges
Tangent Space
Export FBX
Operator Presets
Main
Geome..
Armatu..
Animati..
Only Deform Bones
Add Leaf Bones
Primar..
X Axis
Secon..
-Y Axis
Armat..
Null
Export FBX
Operator Presets
Main
Geome..
Armatu..
Animati..
Baked Animation
Key All Bones
NLA Strips
All Actions
Force Start/End Keying
Sampling Rate:
1.00
Simplify:
1.00

3dsmax Export

Importing files: 3dsMax files are available in the github content link posted above. These have been imported and manually cleaned up by Will B from Will B gameart.

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*".