

# Using Oculus Quest II VR Headset in Engineer and Technician Education

AGBell

7/29/2024

## Developing VR Content for Unreal Engine

This presentation is show how to “develop” VR content for Unreal Engine (UE). This software can be freely downloaded and used to develop models and environments for VR headsets like the Oculus Quest II.

- Getting Started
- Making a poster
- Getting a model
- Importing a model
- Model building outside Unreal Engine
- Model building inside Unreal Engine

## Developing VR Content for Unreal Engine - Getting Started

There are a number of software packages you will need to install on your computer to build content for VR using Unreal Engine. To install software you will, need admin right on your computer and the computer will need to be capable of supporting VR (high end graphics).

Hardware -

- Quad-core Intel or AMD, 2.5 GHz or faster
- 8 GB RAM
- Windows 10/11
- DirectX 11 or 12 compatible graphics card
- Also, you will need hard drive space to save your files.

I use the Dell Alienware Aurora R16 computer with 32 GB of RAM and a 4TB external hard drive. You will also need a Meta Quest 2 or 3 headset with a PC Link Cable so you can upload content to the headsets. We also purchased some docking stations for charging the controllers and headset.

## Developing VR Content for Unreal Engine - Getting Started

There are a few software packages you will need to build content for VR using Unreal Engine. To install software you will, need admin right on your computer and the computer will need to be capable of support VR (high end graphics).

### Software -

- Epic Games Launcher
- Meta Quest Developer Hub
- Oculus App
- Unreal Engine 5
- Visual Studio - Latest Community Version
- Android Studio
- Oculus Developer Account
- Oculus Mobile App - latest version
- Blender
- Sidequest
- Maybe a CAD program like AutoCAD and/or Solidworks
- Maybe Maya



# Developing VR Content for Unreal Engine - Getting Started

[Epic Games Installer](#) - latest version

[Oculus](#) - latest version

[Meta Quest Developer Hub](#) - latest version

[Android Studio](#) - Android Studio Flamingo | 2022.2.1 Patch 2 May 24, 2023.

[Visual Studio](#) - Latest Community Version

May also need to have the Oculus App on your phone to enable the development mode for each of your headsets.

# Developing VR Content for Unreal Engine

**Tyrannosaurus**  
Temporal range: Late Cretaceous  
(Maastrichtian), 68–66 Ma

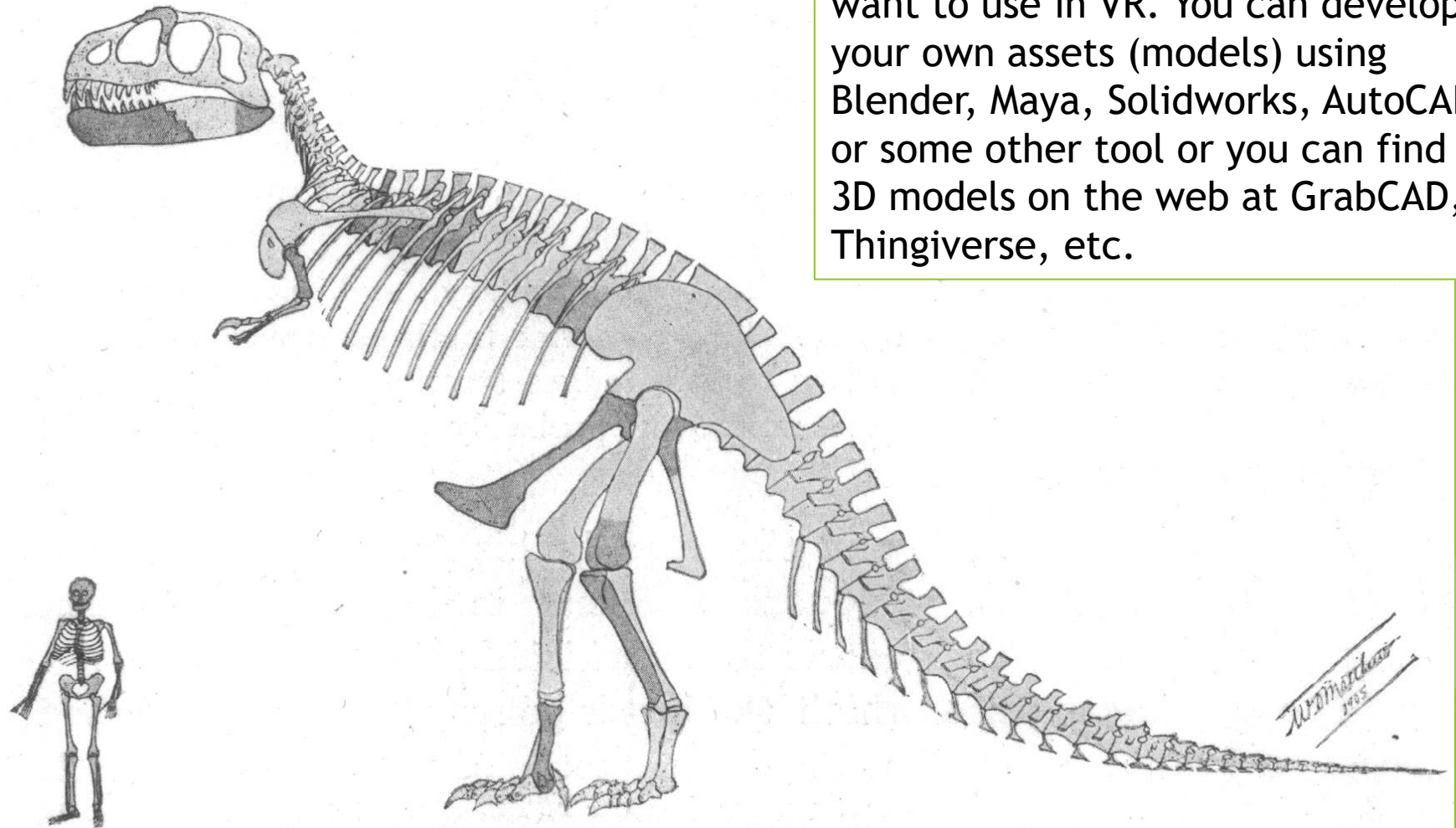
PreC e S D C P T J K PgN



Reconstruction of the *T. rex* type specimen (CM 9380) at the Carnegie Museum of Natural History

**Scientific classification**

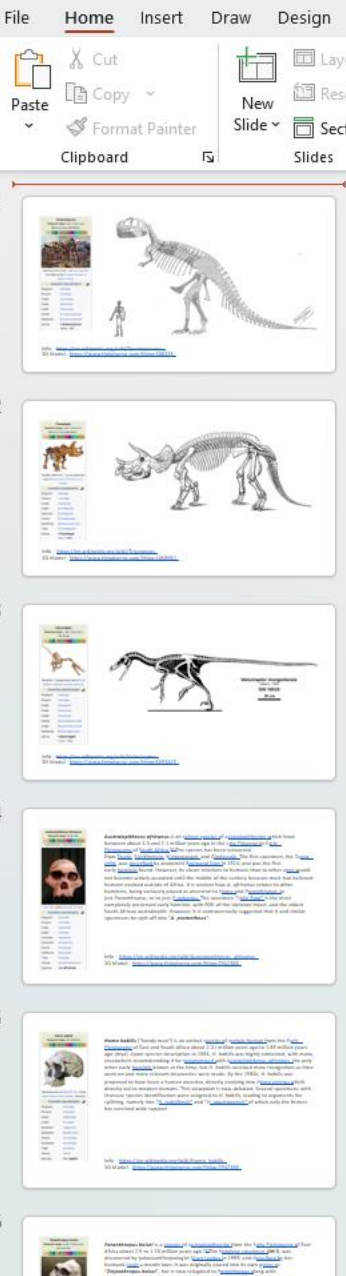
Kingdom:	Animalia
Phylum:	Chordata
Clade:	Dinosauria
Clade:	Saurischia
Clade:	Theropoda
Family:	†Tyrannosauridae
Subfamily:	†Tyrannosaurinae
Genus:	† <b>Tyrannosaurus</b> Osborn, 1905



You must first decide what you want to use in VR. You can develop your own assets (models) using Blender, Maya, Solidworks, AutoCAD or some other tool or you can find 3D models on the web at GrabCAD, Thingiverse, etc.

Info - <https://en.wikipedia.org/wiki/Tyrannosaurus>  
3D Model - <https://www.thingiverse.com/thing:308335>

If you want a poster then an images can be created with the MS Paint, PowerPoint, or other tools. Many Posters are created with PowerPoint and then the slides are saved as images and can be used as materials or textures in Unreal Engine



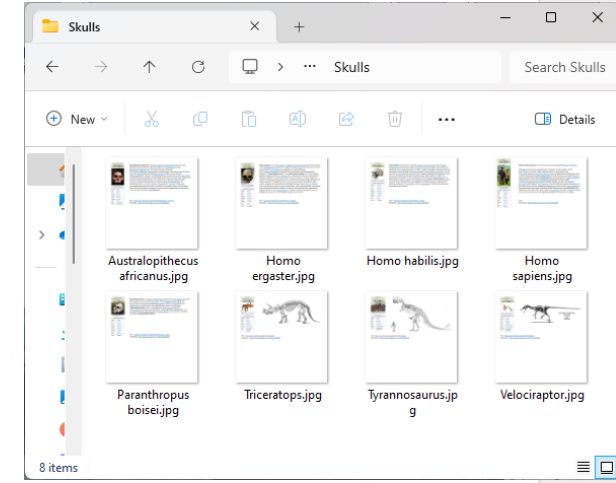
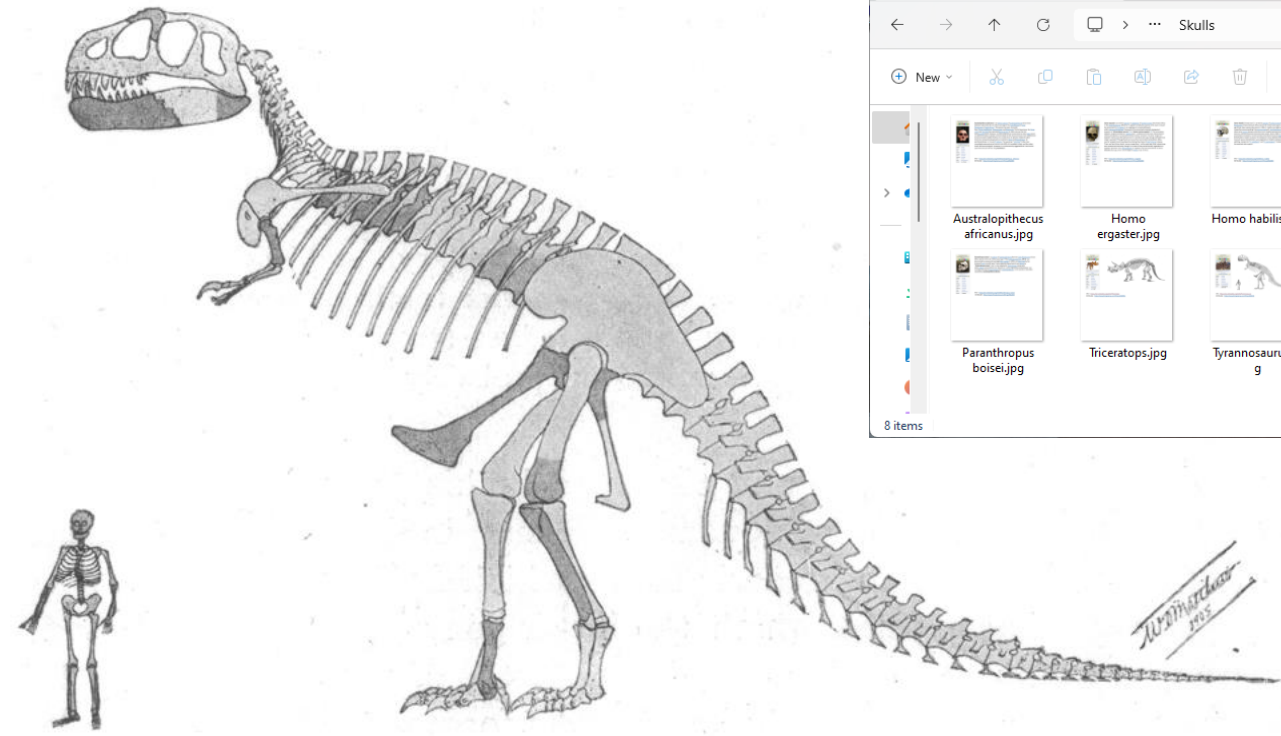
**Tyrannosaurus**  
Temporal range: Late Cretaceous (Maastrichtian), 68–66 Ma

PreC e S D C M J K PgN

Reconstruction of the *T. rex* type specimen (CM 9380) at the Carnegie Museum of Natural History

**Scientific classification**

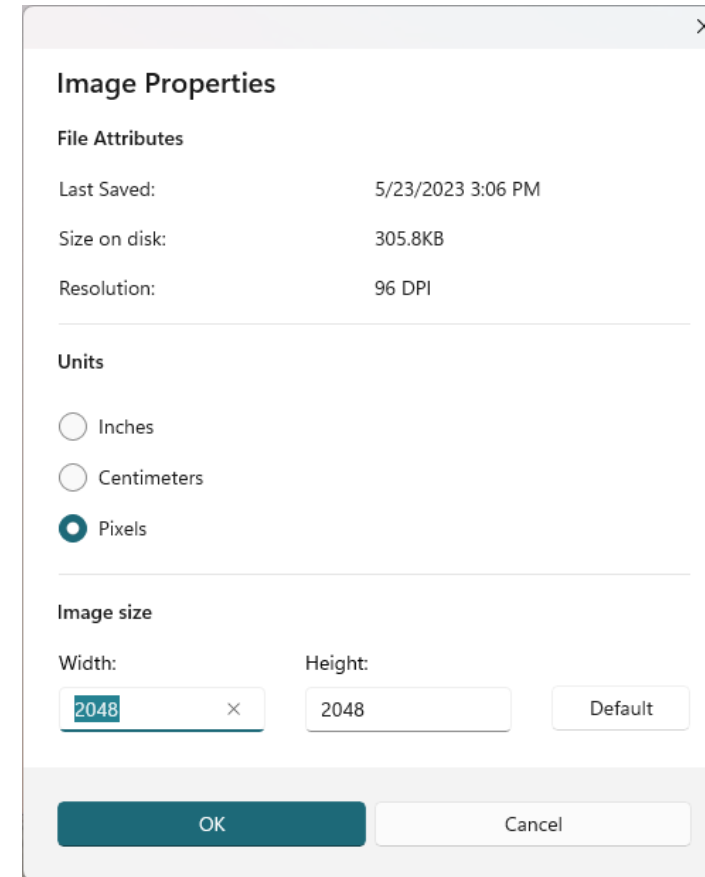
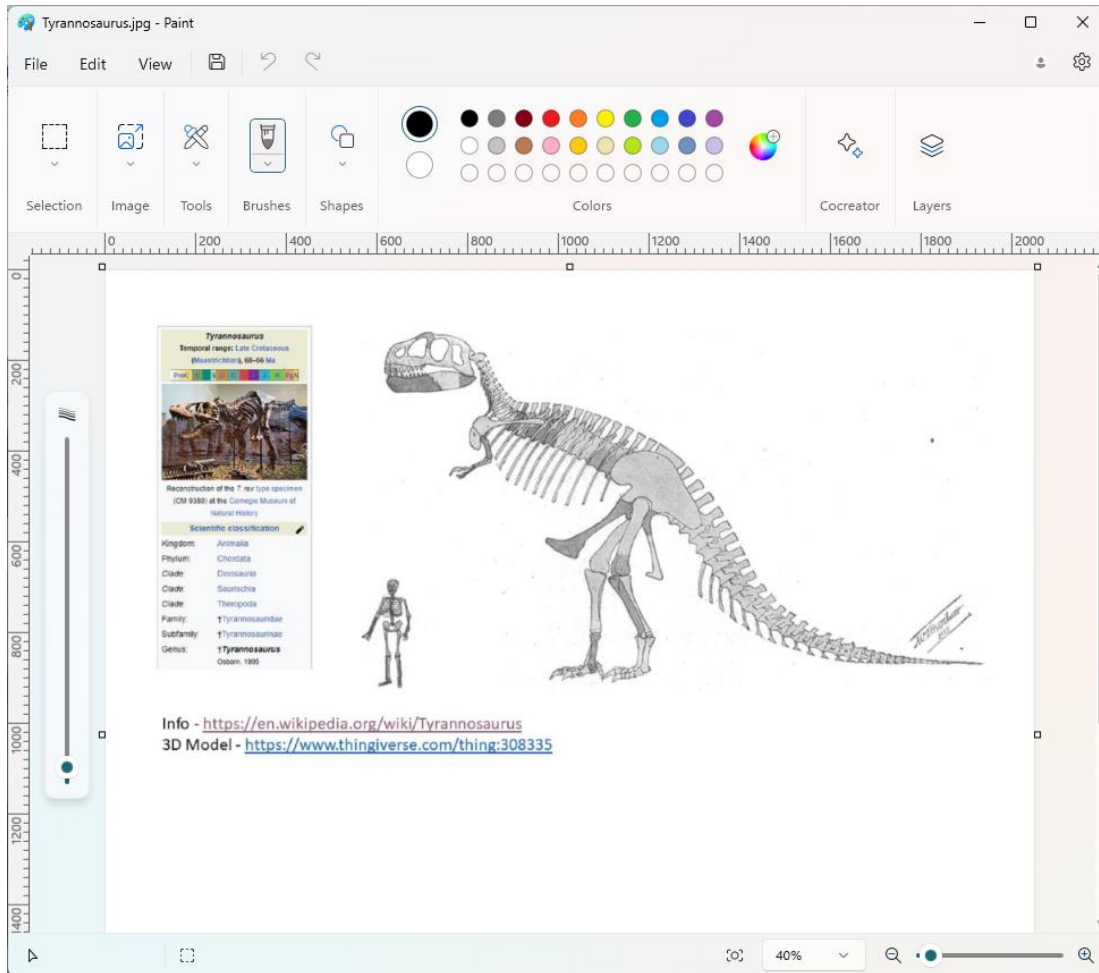
Kingdom: Animalia  
Phylum: Chordata  
Clade: Dinosauria  
Clade: Saurischia  
Clade: Theropoda  
Family: †Tyrannosauridae  
Subfamily: †Tyrannosaurinae  
Genus: †**Tyrannosaurus**  
Osborn, 1905



Info - <https://en.wikipedia.org/wiki/Tyrannosaurus>  
3D Model - <https://www.thingiverse.com/thing:308335>

## Developing VR Content for Unreal Engine - Making a poster

Posters can be presented in Unreal Engine on actors (assets that are displayed in the level) as a material. The image to be applied can be a square image that is related to a power of 2, i.e., 1024 by 1024, 2048 by 2048.





# Developing VR Content for Unreal Engine - Making a poster

Selection Mode | Platforms | Settings

Perspective | Lit | Show



Open the Content Folder and make a directory for "Textures". Import the image you want to use as the poster into your project. Then edit the image.

Import | Save All | All > Content > VRTemplate > Textures | Dock in Layout | Settings

Search Textures

- Australopithec\_aufricanus Texture
- Australopithec\_aufricanus\_Material
- Homo\_ergaster Texture
- Homo\_ergaster\_Material
- Homo\_habilis Texture
- Homo\_habilis\_Material
- Homo\_sapiens Texture
- Homo\_sapiens\_Material
- Paranthropus\_boisei Texture
- Paranthropus\_boisei\_Material
- T\_Grid Texture
- Triceratops Texture
- Triceratops\_Material
- Tyrannosaur... Texture**
- Tyrannosaur...\_Material
- Velociraptor Texture
- Velociraptor\_Material

17 items (1 selected)

Content Drawer | Output Log | Cmd | Enter Console Command | Trace | Derived Data | All Saved | Revision

# Developing VR Content for Unreal Engine - Making a poster



Make three modifications to the image:

- 1.) Compression Settings > UserInterface2D (RGBA)
- 2.) Power of Two Mode > Pad to Power of Two
- 3.) Mip Gen Setting > Sharpen 10

Details x Oodle

Imported: 2048x2048 Method: Streamed  
Displayed: 2048x2048 Format: B8G8R8A8  
Max In-Game: 2048x2048 Combined LOD Bias: 0  
Resource Size: 21845 KB Number of Mips: 12  
Has Alpha Channel: True Encode Speed: Unsupported  
Source Alpha Detected: False

Search

Level Of Detail

Mip Gen Settings **Sharpen10**

LOD Bias 0

Texture Group World

Advanced

Compression

Compress Without Alpha

Editor Show Final Encode

Editor Defer Compression

Compression Settings **UserInterface2D (RGBA)**

Advanced

Interchange

AssetImportData

Scene Import Asset None None

Node Unique ID Factory\_C:/Users/abell/OneDrive/Desktop

Texture

Power of Two Mode **Pad to Power Of Two**

Padding Color

sRGB

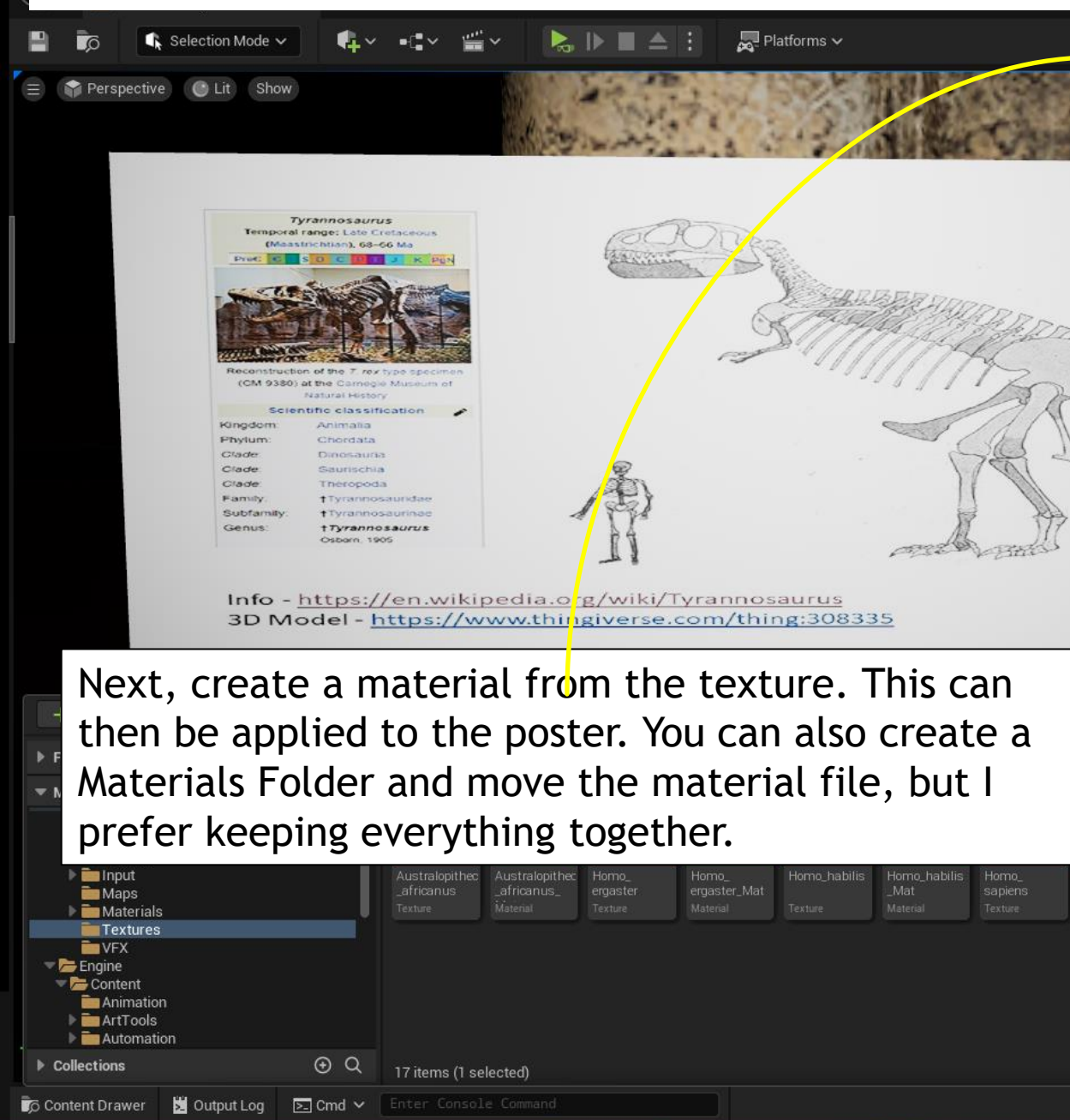
Advanced

Adjustments

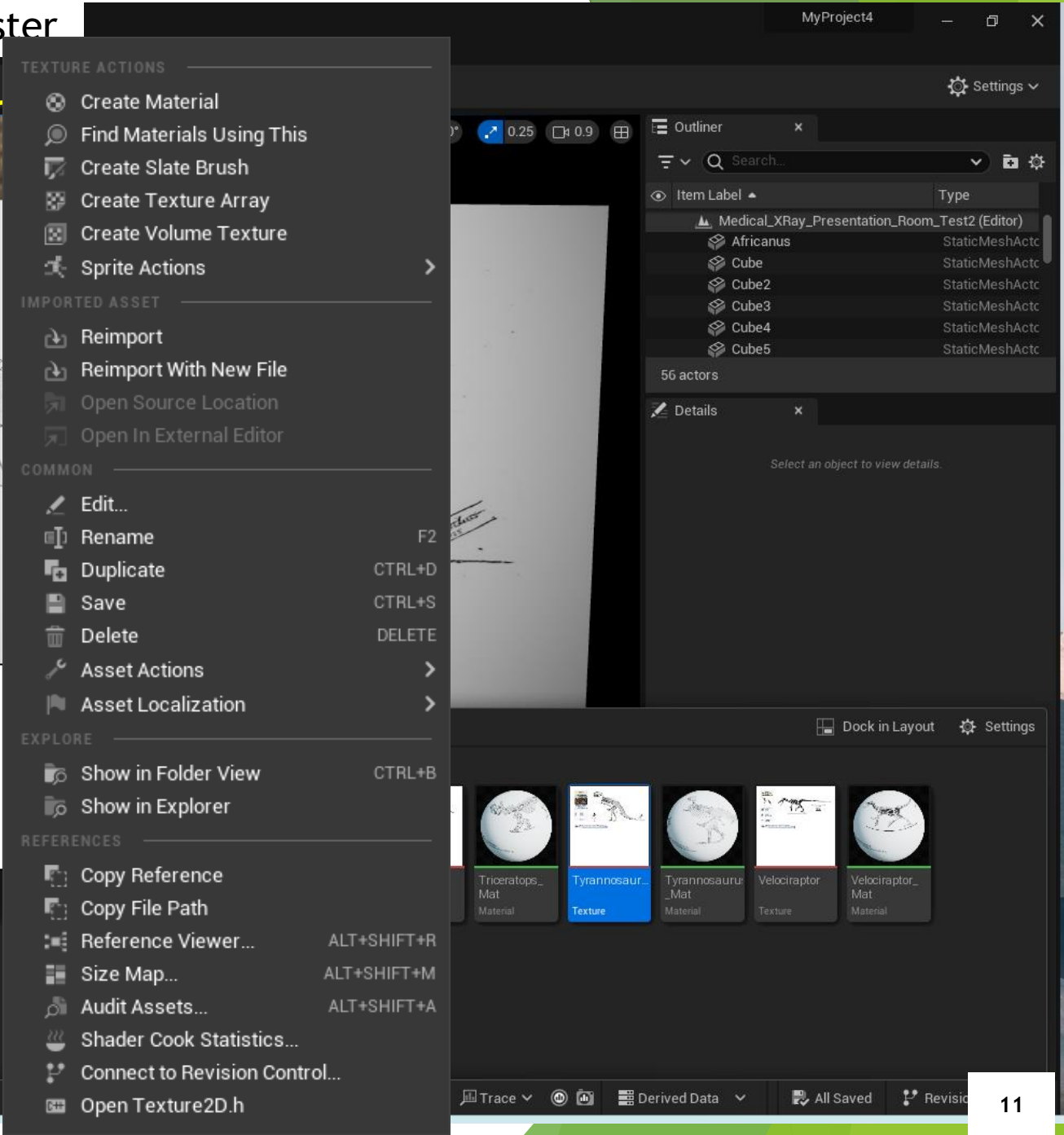
Brightness 1.0

Brightness Curve 1.0

# Developing VR Content for Unreal Engine - Making a poster



Next, create a material from the texture. This can then be applied to the poster. You can also create a Materials Folder and move the material file, but I prefer keeping everything together.





# Developing VR Content for Unreal Engine - Making a poster

The screenshot displays the Unreal Engine 4 interface. On the left, a 3D viewport shows a white sphere with a blue and white dinosaur skin texture applied to its top half. The top menu bar includes options like 'Apply', 'Search', 'Home', 'Hierarchy', 'Live Update', 'Clean Graph', 'Preview State', 'Hide Unrelated', 'Stats', and 'Platform Stats'. The main area is the 'Material Graph' for 'Tyrannosaurus\_Mat', which is currently empty. A 'Texture Sample' node is visible, showing a preview of the dinosaur skin texture. A yellow box highlights the 'Tyrannosaurus\_Mat' material palette on the right, listing various material properties such as 'Base Color', 'Metallic', 'Specular', 'Roughness', 'Anisotropy', 'Emissive Color', 'Opacity', 'Opacity Mask', 'Normal', 'Tangent', 'World Position Offset', 'Subsurface Color', 'Custom Data 0', 'Custom Data 1', 'Ambient Occlusion', 'Refraction (Disabled)', 'Pixel Depth Offset', 'Shading Model', 'Surface Thickness', and 'Front Material'. The bottom left shows the 'Details' panel with various material settings like 'Phys Material', 'Phys Material Mask', 'Material Domain', 'Blend Mode', 'Shading Model', 'Two Sided', 'Use Material Attributes', 'Cast Ray Traced Shad...', and 'Subsurface Profile'. The bottom right shows the 'Substrate' panel with a message 'Substrate is not enabled.' The bottom status bar indicates '1 Unsaved' and 'Revisio'.

Perspective Lit Show

Tyrannosaurus\_Mat > Material Graph

Texture Sample

- UVs [0] RGB
- Tex R
- Apply View MipBias G
- B
- A
- RGBA

Tyrannosaurus\_Mat

- Base Color
- Metallic
- Specular
- Roughness
- Anisotropy
- Emissive Color
- Opacity
- Opacity Mask
- Normal
- Tangent
- World Position Offset
- Subsurface Color
- Custom Data 0
- Custom Data 1
- Ambient Occlusion
- Refraction (Disabled)
- Pixel Depth Offset
- Shading Model
- Surface Thickness
- Front Material

Details Parameters

Physical Material

Phys Material None

Phys Material Mask None

Material

Material Domain Surface

Blend Mode Opaque

Shading Model Default Lit

Two Sided

Use Material Attributes

Cast Ray Traced Shad...

Subsurface Profile None

Substrate

Substrate is not enabled.

Content Drawer Output Log Cmd Enter Console Command

1 Unsaved Revisio

Zoom 1:1

Palette

MATERIAL

This is what the material looks like. Also, shown is the blueprint that is used to apply the material to a shape.



# Developing VR Content for Unreal Engine - Making a poster

The screenshot displays the Unreal Engine 4 interface. On the left, a 3D viewport shows a white sphere with a texture. The central area is the Material Graph for 'Tyrannosaurus\_Mat', featuring a 'TextureCoordinate' node connected to a 'Texture Sample' node. A text box in the center reads: 'To align to the material to the shape you will need to add a TextureCoordinate to the UVs input in the Blueprint for the material. This will improve the image that is displayed on the poster.' On the right, the material palette for 'Tyrannosaurus\_Mat' is visible, listing various material properties like Base Color, Metallic, Specular, etc. The bottom of the screen shows the 'Details' panel with material settings and the 'Output Log'.

To align to the material to the shape you will need to add a TextureCoordinate to the UVs input in the Blueprint for the material. This will improve the image that is displayed on the poster.

# Developing VR Content for Unreal Engine - Making a poster

The screenshot displays the Unreal Engine 4 interface. The top toolbar shows the 'Add' button (a cube with a plus sign) highlighted with a yellow box. A yellow arrow points from this button to the 'Shapes' option in the 'GET CONTENT' menu, which is also highlighted with a yellow box. Within the 'Shapes' sub-menu, the 'Cube' option is highlighted with a yellow box. A white text box with the text 'Create a basic Cube' is positioned over the 'Cube' option. The background shows a 3D scene with a stone wall, a wooden floor, and a large horn-like object. The right side of the interface shows the 'Outliner' and 'Details' panels. The 'Outliner' panel lists several actors, including 'Wall\_0', 'Wall\_1', and 'Wall\_2'. The 'Details' panel shows the properties for the selected 'Wall\_0' actor, including 'StaticMeshComponent' and 'Physics' settings.

Selection Mode

Perspective Lit Show

GET CONTENT

- Import Content...
- Quixel Bridge
- Unreal Marketplace
- Content Browser

PLACE ACTORS

- Basic
- Lights
- Cinematic
- Shapes
- Media Plate
- Visual Effects
- Volumes
- All Classes
- Place Actors Panel

RECENTLY PLACED

- Cube

Create a basic Cube

Medical\_XRay\_Presentation\_Room\_Test2 (Editor)

- T-RexSkull\_Jaw
- T-RexSkull\_Skull
- triceratops\_highres
- Wall\_0
- Wall\_1
- Wall\_2

56 actors (1 selected)

Details

Wall\_0

Wall\_0 (Instance)

StaticMeshComponent (StaticMeshComponent0)

Search

General Actor LOD Misc Physics

Rendering Streaming All

Element 0

MI\_Onyx\_Marble\_Brick

Advanced

Physics

Simulate Physics

Mass (kg) 1815.627319

Linear Damping 0.01

Angular Damping 0.0

Enable Gravity

Constraints

Update Kinematic fro...

Ignore Radial Impulse

Ignore Radial Force

Apply Impulse on Dam...

Content Drawer Output Log Cmd Enter Console Command

Trace Derived Data 2 Unsaved Revisio



# Developing VR Content for Unreal Engine - Making a poster

Reimport Base Mesh | Collision | UV

Perspective | Lit | Show | LOD Auto

LOD: 0  
Current Screen Size: 0.531654  
Triangles: 48  
Vertices: 54  
UV Channels: 2  
Distance Field: 42x42x42 = 0.00Mb always loaded, 0.10Mb streamed  
Approx Size: 100x100x100  
Num Collision Primitives: 1  
Estimated Compressed Disk Size: 0.00 MB (0.00 MB Nanite)

The cube is actually a 100 cm x 100 cm x 100 cm cube. Why?

Details | Socket Manager

Search

Collision

- Primitives
  - Double Sided Geometry
  - Never Needs Cooked Collision D...
  - Simple Collision Physical Material: None
- Collision Presets: BlockAll
- Collision Complexity: Project Default
- Customized Collision:
- Complex Collision Mesh: None

Advanced

Import Settings

- Mesh
- Material
- Transform
- Miscellaneous
- File Path

Ray Tracing

- Support Ray Tracing:

Navigation

- Has Navigation Data:
- Cylinder Collision: 0 Array element
- Box Collision: 0 Array element
- Area Class: None
- Gather Convex Geometry:
- Create on Client:
- Is Dynamic Obstacle:

Thumbnail

Level Of Detail

Content Drawer | Output Log | Cmd | Enter Console Command

5 Unsaved | Revisio

# Developing VR Content for Unreal Engine - Making a poster

Ans - It is the default size.

In the Project Settings > Editor > Appearance we see the Distance/Length is in Centimeter. This can be changed if you want imperial or a different metric value. Some programs may use Meters as a default unit.

## Platforms

- [Android](#)
- [Android Material Quality - OpenGL Mobile](#)
- [Android Material Quality - Vulkan](#)
- [Android SDK](#)
- [Android SM5 Material Quality - Vulkan](#)

Display Units on Applicable Properties	<input checked="" type="checkbox"/>
Display Units on Component Transforms	<input type="checkbox"/>
Advanced	
Distance/Length	1 Array element
Index [ 0 ]	Centimeters
Mass	
Index [ 0 ]	Micrometers
	Millimeters
	Centimeters
	Meters
	Kilometers
	Inches
	Feet
	Yards
	Miles
Angles	
Speed/Velocity	Meters Per Second
Temperature	Celsius
Force	Newtons
Torque	Newton Meters
Reference Viewer	
Show Searchable Names	No Preference
Default Max Search Breadth	20

MyProject4

Settings

GAME SPECIFIC SETTINGS

- World Settings
- Project Settings...**
- Plugins

SELECTION

- Allow Translucent Selection
- Allow Group Selection CTRL+SHIFT+G
- Strict Box Selection
- Box Select Occluded Objects
- Show Transform Widget
- Show Subcomponents

SCALABILITY

- Engine Scalability Settings
- Material Quality Level
- Preview Platform

REAL TIME AUDIO

Volume

SNAPPING

- Enable Actor Snapping CTRL+SHIFT+K
- Distance
- Enable Socket Snapping
- Enable Vertex Snapping
- Enable Planar Snapping

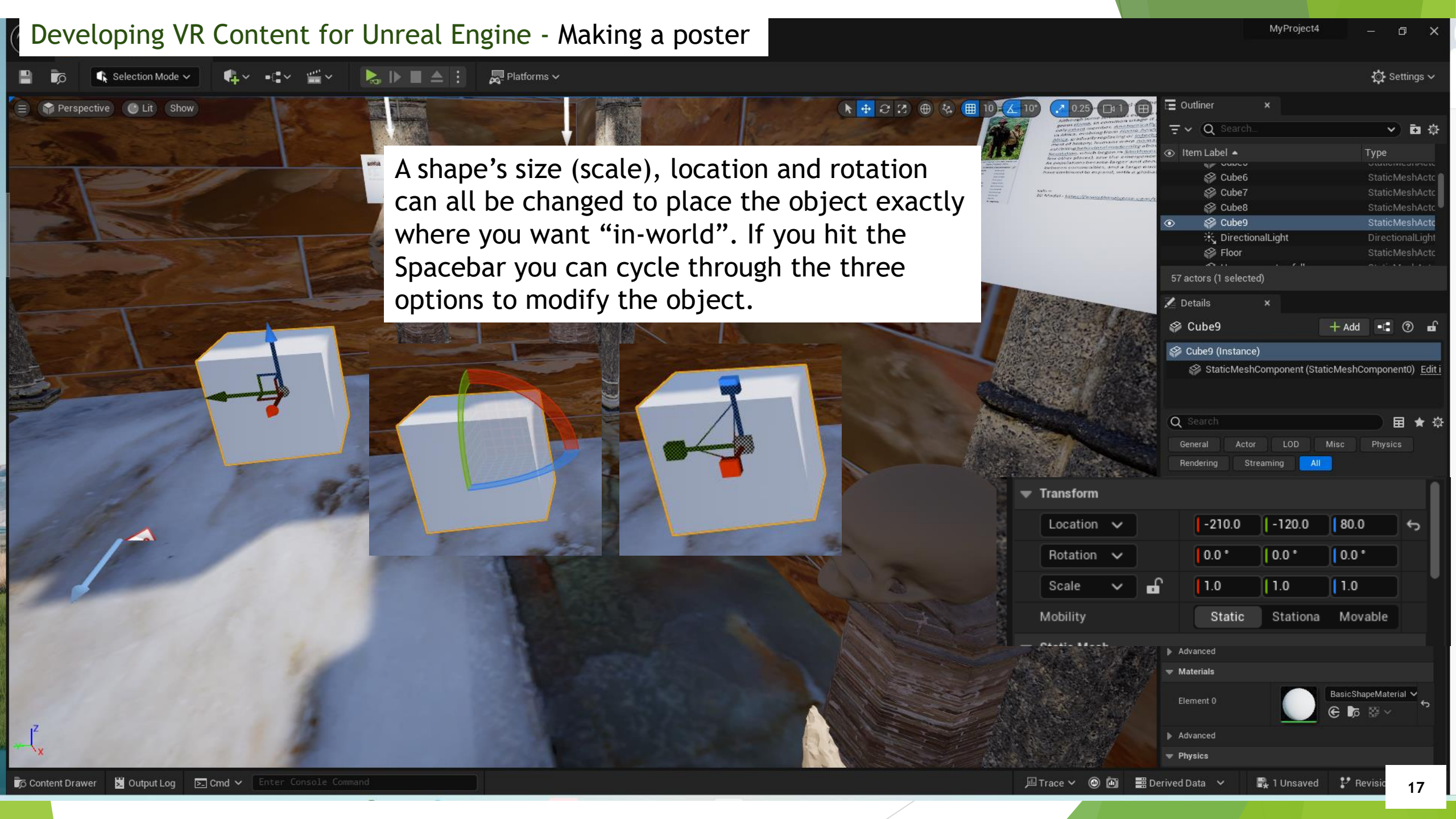
VIEWPORT

- Hide Viewport UI
- Previewing



# Developing VR Content for Unreal Engine - Making a poster

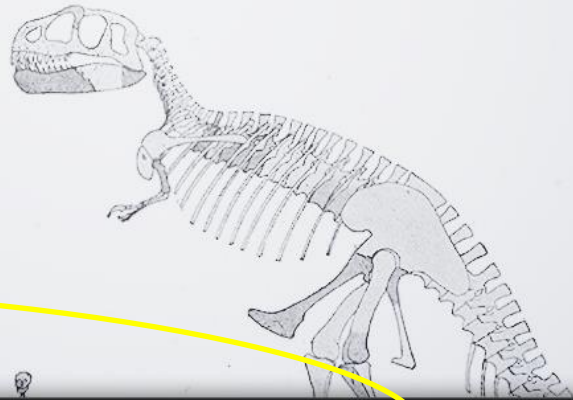
A shape's size (scale), location and rotation can all be changed to place the object exactly where you want "in-world". If you hit the Spacebar you can cycle through the three options to modify the object.



# Developing VR Content for Unreal Engine - Making a poster

The poster displayed here is actually 1 meter wide by 0.56264 meters tall with a 0.01 meter thickness (16:9 aspect ratio). UE is not a modeling tool so you must get use to scaling or building models outside UE.

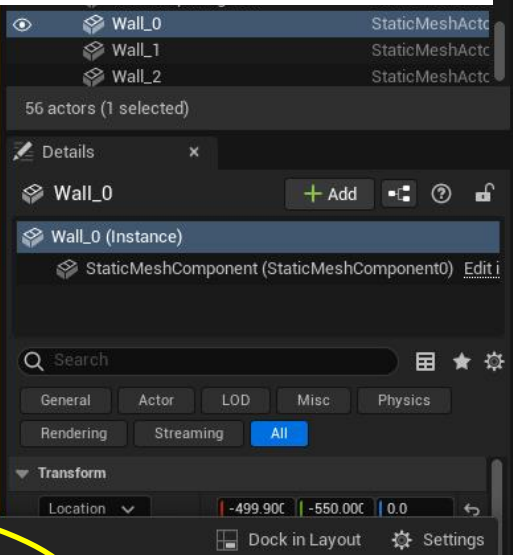
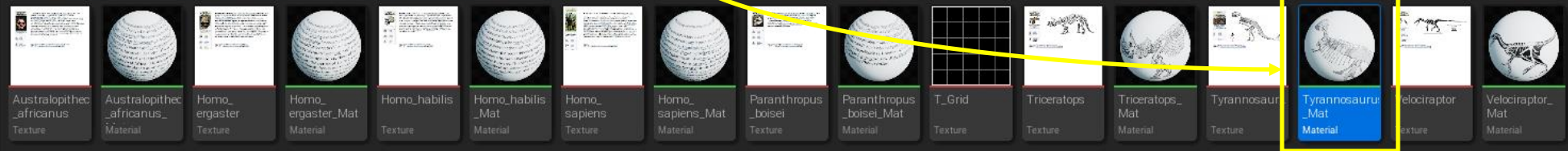
The desired material can be added to the poster that is sized and placed where we want it.



Scientific classification  
Kingdom: Animalia  
Phylum: Chordata  
Clade: Dinosauria  
Clade: Saurischia

Content > VRTemplate > Textures

Search Textures





# Developing VR Content for Unreal Engine - Making a poster

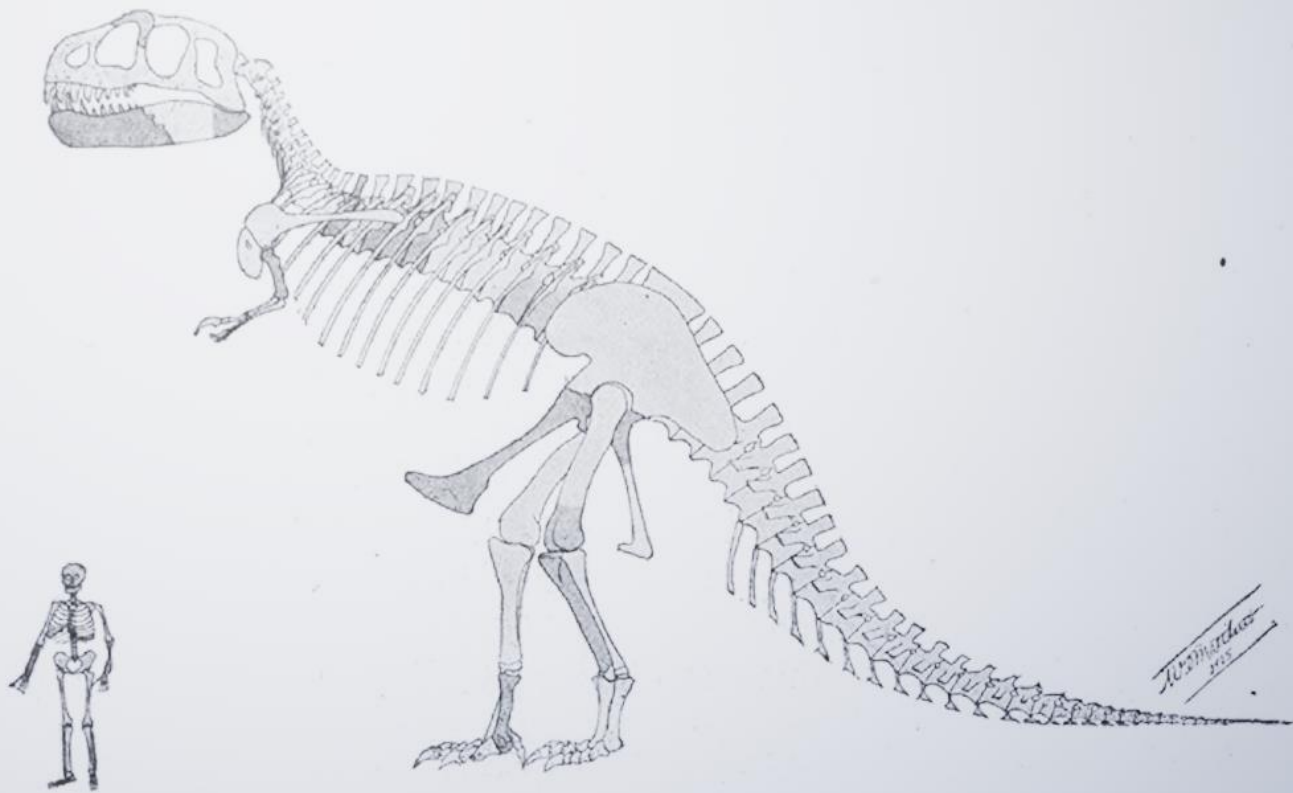
Finished poster.

**Tyrannosaurus**  
Temporal range: Late Cretaceous (Maastrichtian), 68–66 Ma

Reconstruction of the *T. rex* type specimen (CM 9380) at the Carnegie Museum of Natural History

**Scientific classification**

Kingdom:	Animalia
Phylum:	Chordata
Clade:	Dinosauria
Clade:	Saurischia
Clade:	Theropoda
Family:	†Tyrannosauridae
Subfamily:	†Tyrannosaurinae
Genus:	† <i>Tyrannosaurus</i> Osborn, 1905



Info - <https://en.wikipedia.org/wiki/Tyrannosaurus>  
3D Model - <https://www.thingiverse.com/thing:308335>

MyProject4

Settings

Outliner

Item Label	Type
Medical_XRay_Presentation_Room (Editor)	
T-RexSkull_Jaw	StaticMeshAct
T-RexSkull_Skull	StaticMeshAct
triceratops_highres	StaticMeshAct
Wall_0	StaticMeshAct
Wall_1	StaticMeshAct
Wall_2	StaticMeshAct

56 actors (1 selected)

Details

Wall\_0

Wall\_0 (Instance)

StaticMeshComponent (StaticMeshComponent0)

Search

General Actor LOD Misc Physics

Rendering Streaming All

Transform

Location: -499.900, -550.000, 0.0

Rotation: 0.0°, 0.0°, 90.00000°

Scale: 11.0, 0.5, 4.0

Mobility: Static Stationary Movable

Static Mesh

SM\_Cube

Advanced

Materials

Element 0: MI\_Onyx\_Marble\_Brick

Advanced

Physics

# Developing VR Content for Unreal Engine - Making a poster

Posters can be placed in Unreal Engine and provide value information to the student. In world the student can move around to better view the posters. Poster size and location are important so that students can see the posters.



MyProject4

Settings

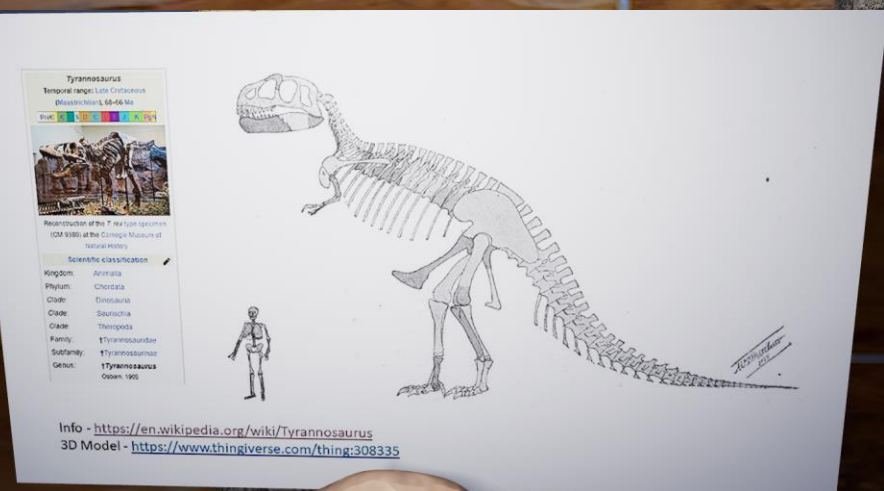
Outliner

Item Label	Type
Medical_XRay_Presentation_Room (Editor)	
Cube7	StaticMeshActc
Cube8	StaticMeshActc
DirectionalLight	DirectionalLight
Floor	StaticMeshActc
Homo_ergaster_full	StaticMeshActc
Homo_Habilis_whole	StaticMeshActc

56 actors

Details

Select an object to view details.





# Developing VR Content for Unreal Engine - Getting a model

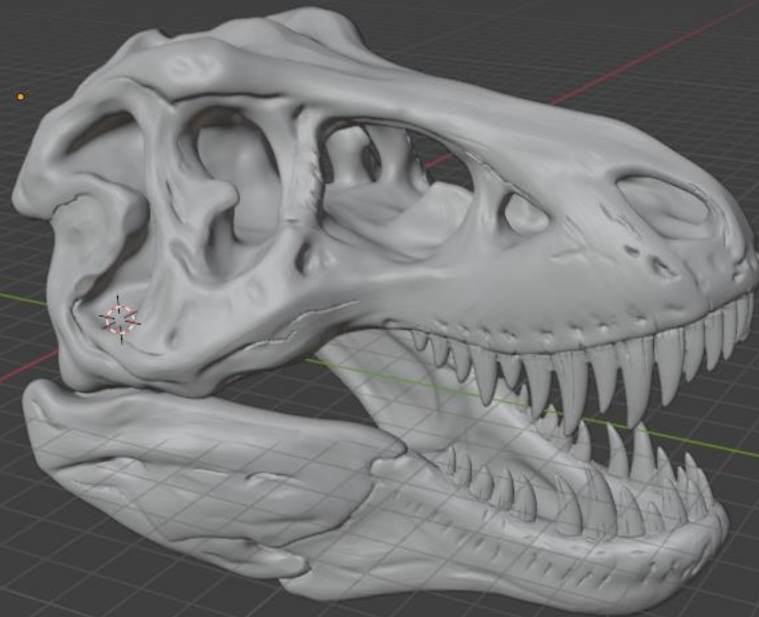
If you find something on the web you may need to convert it from an STL to a FBX or OBJ file. There are online tools you can use but I prefer to use Blender since you can also evaluate the complexity of the mesh.

The screenshot shows a web browser window displaying a page on Thingiverse. The page title is "The T-Rex Skull" by MakerBot, dated May 06, 2014, and marked as "Remixed". The main content is a 3D model of a T-Rex skull on a wooden stand. To the right of the main model is a vertical list of four smaller 3D models, all in blue, representing different parts or variations of the skull. A blue button labeled "Download all files" is positioned above this list. Below the main model, there are five interactive buttons: "Post a make", "Post a remix", "Watch", "Report thing", and "Tip designer". The browser's address bar shows the URL "thingiverse.com/thing:308335". The browser's top bar includes various navigation and utility icons, and the page header features the "UltiMaker Thingiverse" logo and a search bar.

# Developing VR Content for Unreal Engine - Getting a model

In Blender this is a very large model (286,062 tris). The physical size is 132 meters and needs to be scaled.

This is two models that were combined into a single model (skull and lower jaw)



Scripting + Scene ViewLayer

Active workspace showing in the window.

User Perspective  
(1) Collection | T-RexSkull\_Jaw

Objects 4  
Vertices 143,013  
Edges 429,093  
Faces 286,062  
Triangles 286,062

Transform

Location:

X	3.0969 m
Y	-20.082 m
Z	51.84 m

Rotation:

X	-0.000002°
Y	-27.624°
Z	-90°

XYZ Euler

Scale:

X	1.000
Y	1.000
Z	1.000

Dimensions:

X	132 m
Y	97.4 m
Z	64.2 m

Scene Collection

- Collection
  - Camera
  - Light
  - T-RexSkull\_Jaw
  - T-RexSkull\_Skull

T-RexSkull\_Jaw

T-RexSkull\_Jaw

Transform

Location X	3.0969 m
Y	-20.082 m
Z	51.84 m

Rotation X

X	-0.000002°
Y	-27.624°
Z	-90°

Mode XYZ Euler

Scale X	1.000
Y	1.000
Z	1.000

> Delta Transform

> Relations

> Collections

> Instancing

> Motion Paths

> Visibility

> Viewport Display

> Line Art

> Custom Properties

Playback Keying View Marker

1 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210 220 230 240 250

Select Rotate View Object

# Developing VR Content for Unreal Engine - Getting a model

The screenshot shows the Unreal Engine 4 interface. In the center, a 3D model of a T-Rex skull is displayed on a grid. The 'Export' menu is open, listing various file formats. The 'FBX (.fbx)' option is highlighted. A tooltip 'Write a FBX file.' is visible next to it. On the right, the 'Properties' panel shows the transform data for the selected object, 'T-RexSkull\_Jaw'. The 'Scene Collection' panel on the far right shows a hierarchy of objects including 'Camera', 'Light', 'T-RexSkull\_Jaw', and 'T-RexSkull\_Skull'. The bottom of the screen features a 'Playback' panel with a timeline from 1 to 250.

Export menu options:

- Collada (.dae)
- Alembic (.abc)
- Universal Scene Description (.usd\*)
- Grease Pencil as SVG
- Grease Pencil as PDF
- Wavefront (.obj)
- Stanford PLY (.ply)
- Motion Capture (.bvh)
- STL (.stl)
- FBX (.fbx)**
- glTF 2.0 (.glb/.gltf)
- X3D Extensible 3D (.x3d)

Properties panel (T-RexSkull\_Jaw):

Property	Value
Location X	3.0969 m
Y	-20.082 m
Z	51.84 m
Rotation X	-0.000002°
Y	-27.624°
Z	-90°
Mode	XYZ Euler
Scale X	1.000
Y	1.000
Z	1.000

Model must be converted to be imported into Unreal Engine. We will use the FBX format.



# Developing VR Content for Unreal Engine - Getting a model

The screenshot displays the Unreal Engine 4 interface in Object Mode. A grey T-Rex skull model is centered in the viewport. The 'Apply' menu is open, with 'All Transforms' selected. A tooltip above it reads 'Apply the object's transformation to its data.' The Transform panel on the right shows the following values:

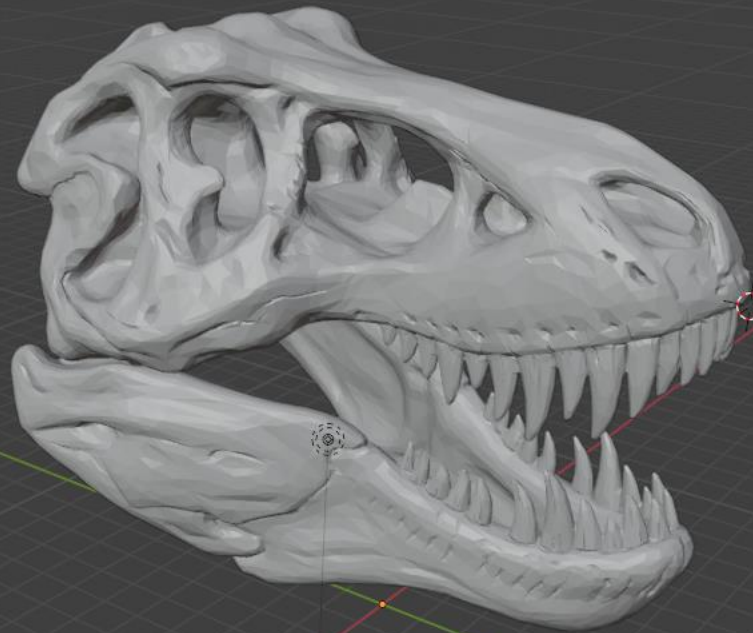
Property	X	Y	Z
Location (m)	9.0782	0.7888	-4.9961
Rotation (°)	0°	-111.05°	-90°
Scale	0.100	0.100	0.100
Dimensions (m)	11.6	9.74	15.9

The Outliner on the right shows the 'Decimate' modifier applied to the 'T-RexSkull\_Skull' object. The 'Decimate' panel shows a Ratio of 1.0000, Symmetry checked for X, Y, and Z, and a Face Count of 285334.

Prior to export it is a good practice to apply all transforms.

# Developing VR Content for Unreal Engine - Getting a model

This is the decimated version of the model which should make it faster to render in Unreal Engine. Decimation can be performed in Blender and the technique used is a simple Collapse to 10% of the original mesh size.

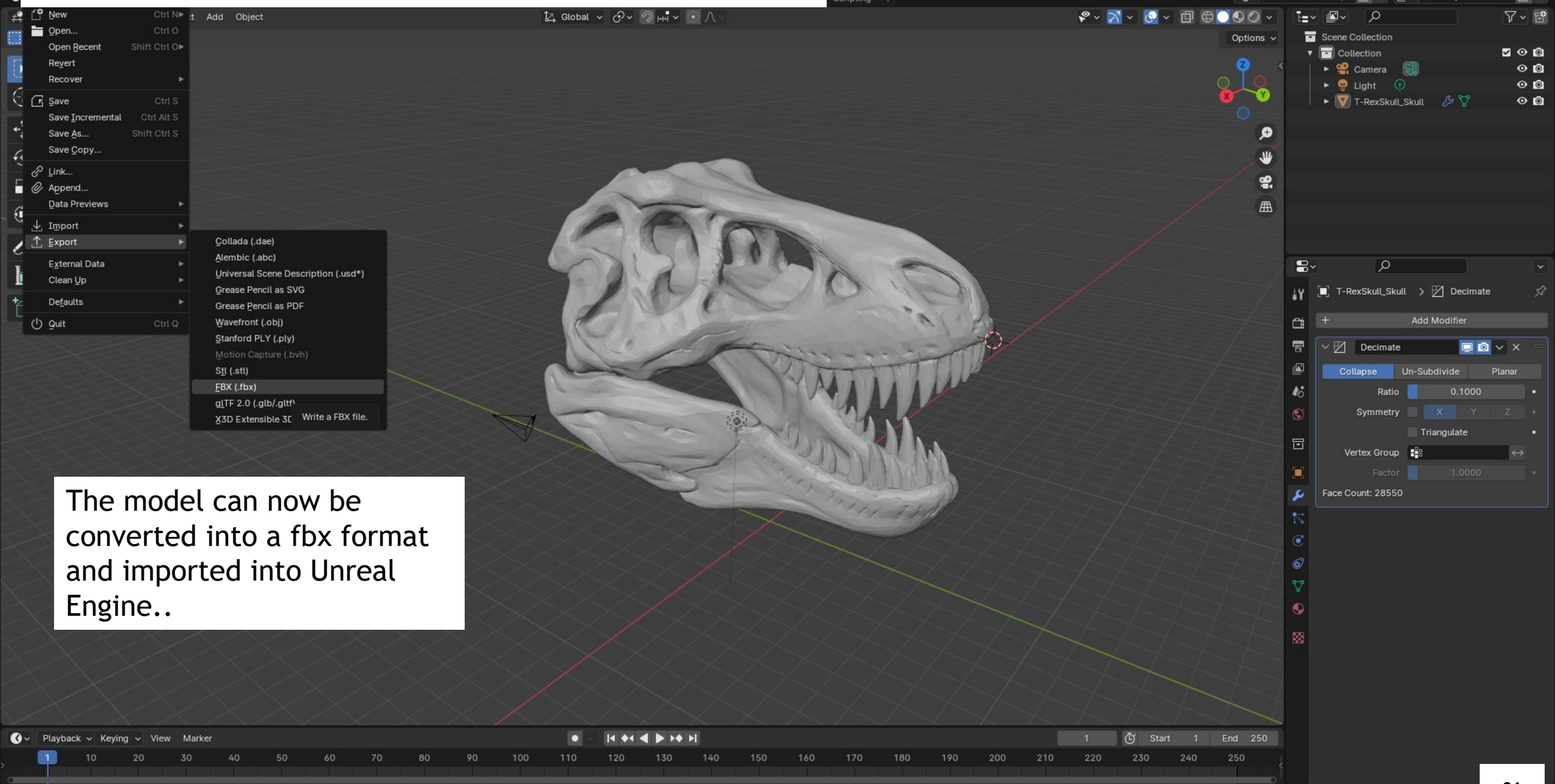


Blender Decimate Modifier Settings:

- Modifier: Decimate
- Ratio: 0.1000
- Symmetry: X
- Vertex Group: (empty)
- Factor: 1.0000
- Face Count: 28550



# Developing VR Content for Unreal Engine - Getting a model



The model can now be converted into a fbx format and imported into Unreal Engine..

# Developing VR Content for Unreal Engine - Getting a model

The screenshot shows the Blender File View interface. On the left, there are panels for Bookmarks, System (with Home, Desktop, Documents, Downloads, Music, Pictures, Videos, Fonts, OneDrive, and submissions (1).zip), Volumes (with OS (C:), DATA (D:), Seagate Expansion Drive (E:), Final Backup B (F:), and DUO LINK 4 (G:)), and Recent (with ue\_download, sL\_download3, Desktop, and My Mountain). The main area displays a file explorer for the path F:\2024\_Ohms\052224\_Ohms\, listing files like VR Class, VR Models, and several T-RexSkull model files. The right panel shows the Transform and Geometry properties, with the Smoothing dropdown menu open, highlighting 'Normals Only'. At the bottom, there is a red bar with the filename 'T-RexSkull Rev10p.fbx' and buttons for 'Export FBX' and 'Cancel'.

Name	Date Modified	Size
VR Class	22 May 2024 16:45	
VR Models	22 May 2024 16:47	
T-RexSkull new.fbx	21 May 2024 17:14	1.8 MiB
T-RexSkull Rev.fbx	21 May 2024 09:43	8.5 MiB
T-RexSkull Rev10p.fbx	21 May 2024 09:43	920 KiB
T-RexSkull RevA.fbx	21 May 2024 09:43	8.5 MiB
T-RexSkull RevA10p.fbx	21 May 2024 09:43	920 KiB
T-RexSkull_Object.FBX	21 May 2024 11:07	6.7 MiB
T-RexSkull_Object_001.FBX	21 May 2024 11:08	5.0 MiB

Unreal engine may complain about the model so you can also apply “face smoothing” to the fbx model and then export it.

Export FBX Cancel

## Developing VR Content for Unreal Engine - Building a model

You can also build your own models using Solidworks, AutoCAD and/or Blender. Since we use Solidworks in our AS degrees this is how you would build models for Unreal Engine.

Step 1 - Design and build your model.

Step 2 - Export model as an STL file from Solidworks.

Should maintain dimensional integrity.

STL files can also be used for 3D printing.

Step 3 - Import STL file into Blender.

Model can be scaled and decimated to reduce the size of the mesh model.

Lower poly models should be your goal, but model complexity should be balanced against how the model looks in VR.

Step 4 - Export STL file from Blender as an FBX.

Blender can support the model but may want to add face smoothing.

Always save your Blend file.

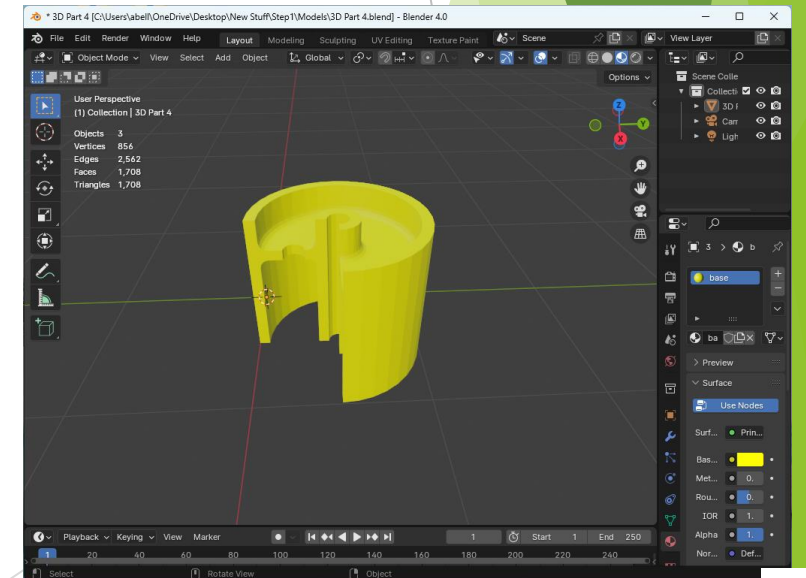
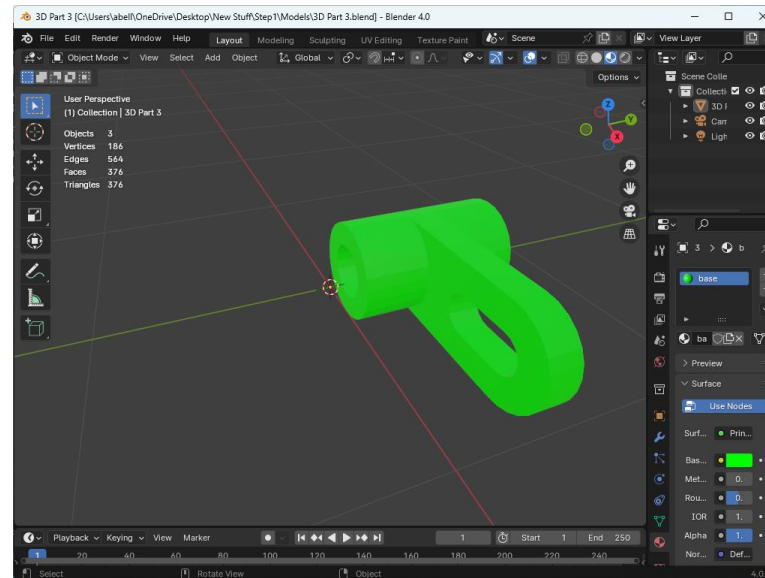
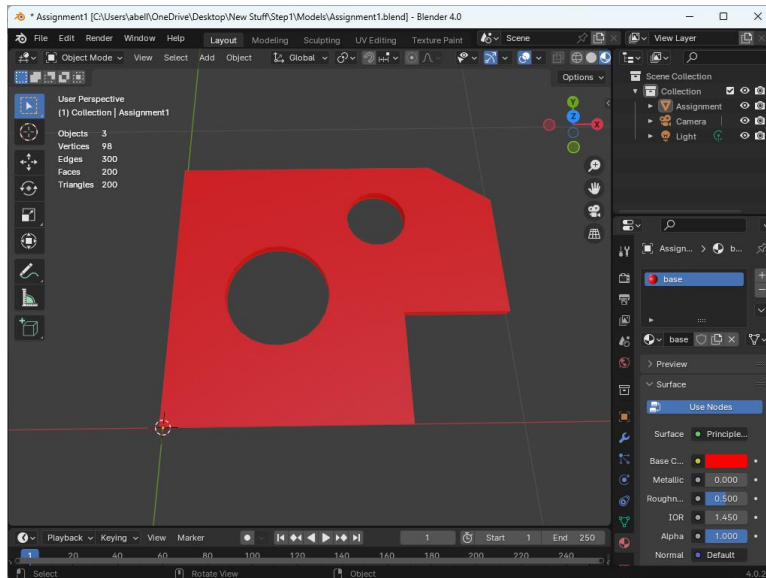
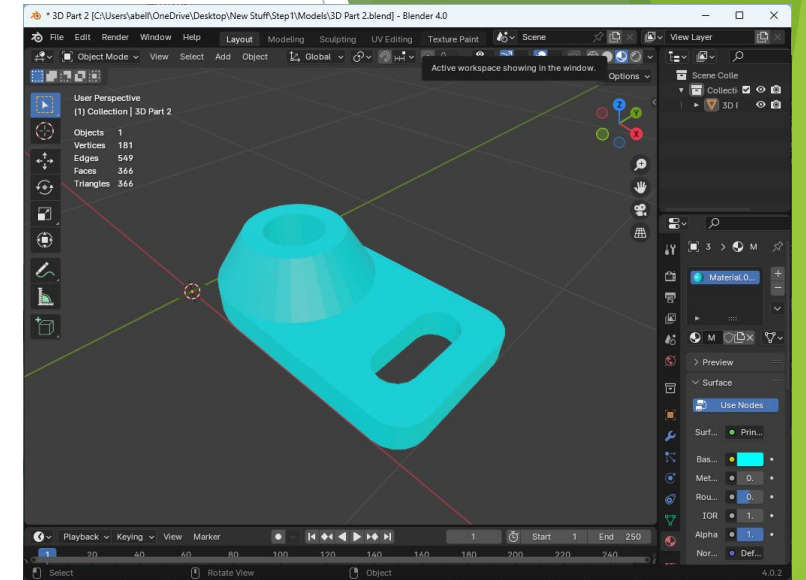
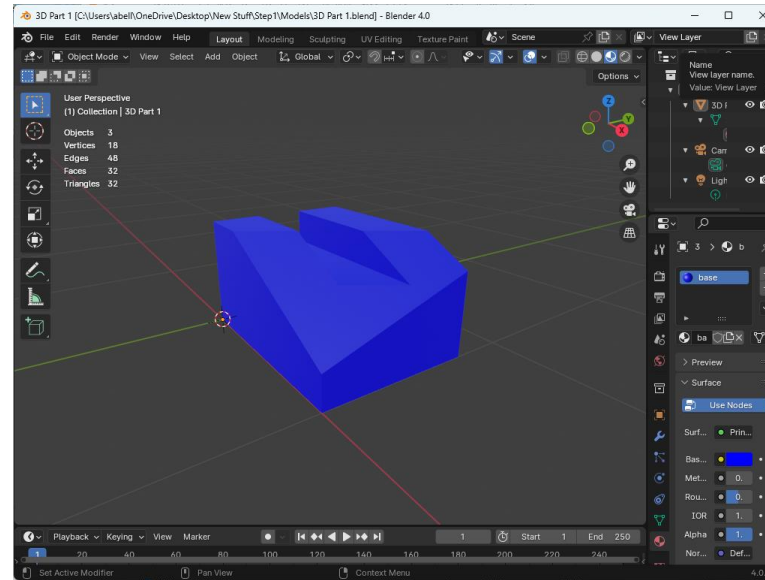
Step 5 - Import FBX file into Unreal Engine.

There are more details that could be shared regarding UV Maps, textures and materials using Blender. I would strongly encourage that you learn and use Blender to support developing VR content.



# Developing VR Content for Unreal Engine - Building a model

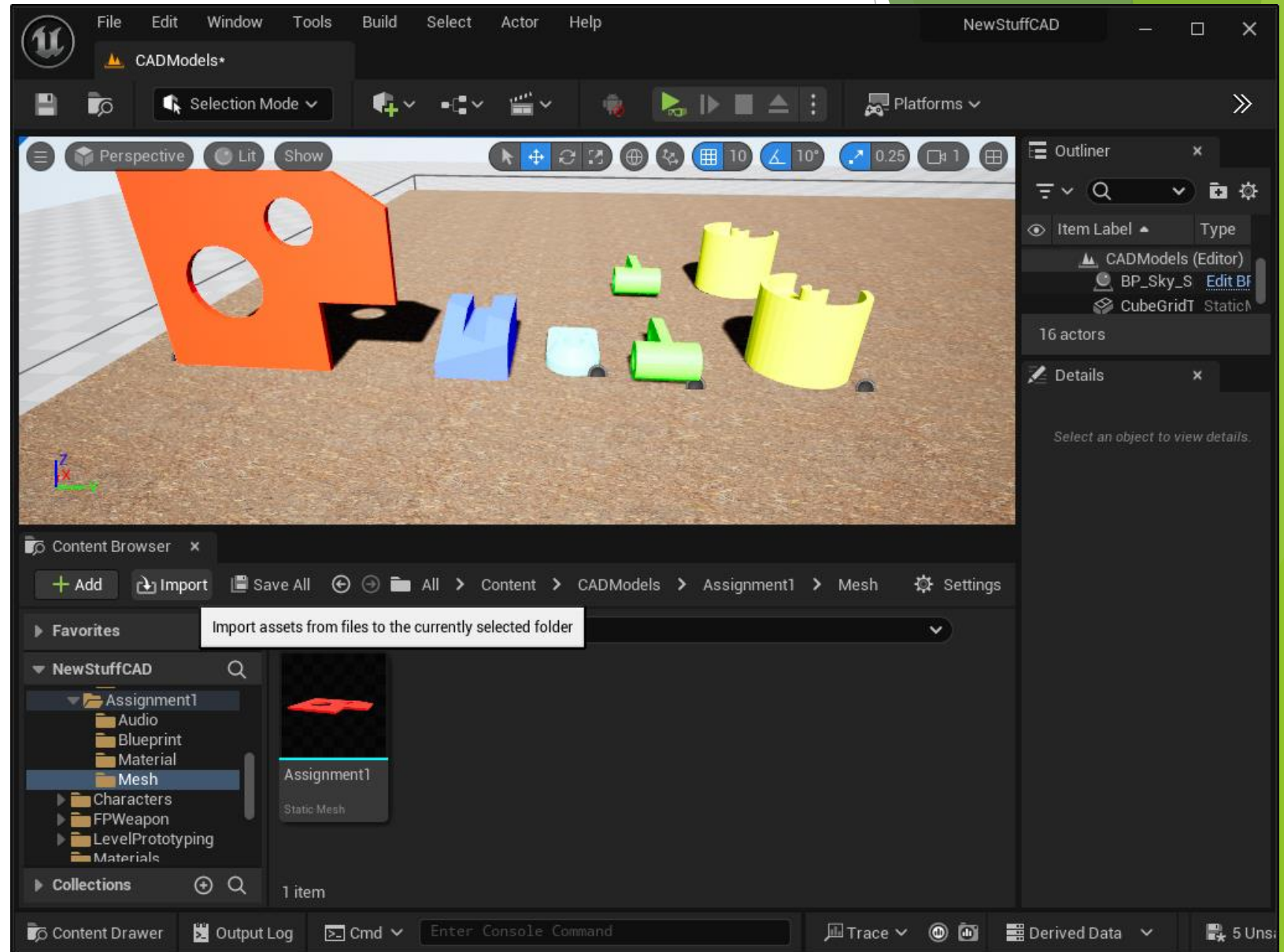
Here are examples of 5 CAD models built with AutoCAD and Solidworks. These models were exported as STL files and then converted to FBX files for Unreal Engine. (Note the number of the “Triangles” is relatively low so these would be low poly models.)



## Developing VR Content for Unreal Engine - Building a model

Within Unreal Engine you should create a folder for each model and import the mesh (FBX file) into the folder you create called “Mesh”. You can also import the material (texture) of the model as well or add the texture within Unreal Engine.

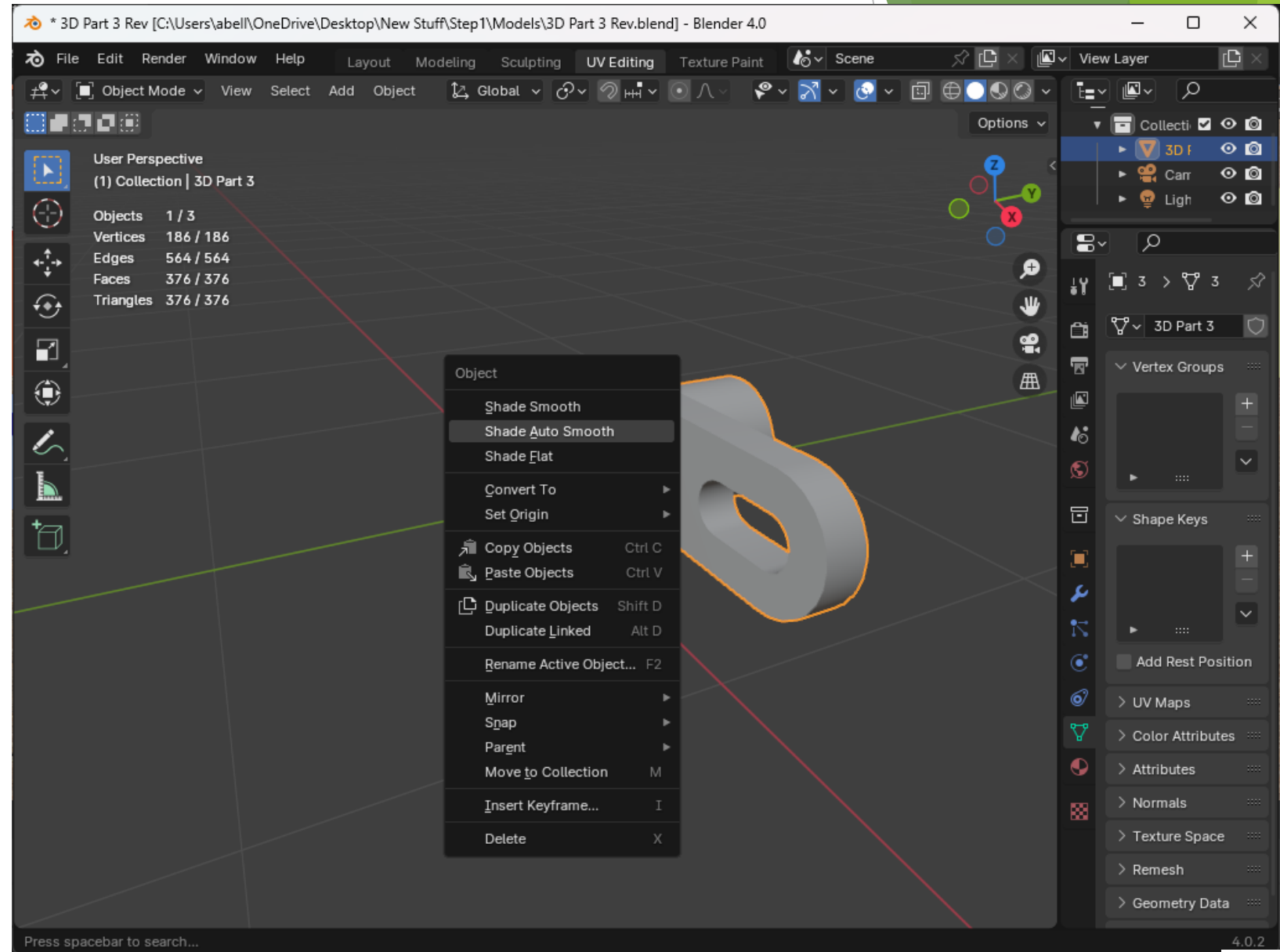
All five models were imported into a folder call CAD Models and then placed within a scenario to view and interact with.



# Developing VR Content for Unreal Engine - Building a model

For the more complicated models that include circular faces I did a Shade Auto Smooth in Blender. This greatly improved the quality of the model in VR.

Mesh density and lighting will continue to be an issue as you use more complicated models. To get it “right” will take some research and experimentation.





## Developing VR Content for Unreal Engine - Building a model

These are the five CAD models that were uploaded to Second Life. A script was also added (shown below) to the models so they would freely spin. These models are DAE files but were produced using the same Solidworks (STL) models with Blender.



```
default
{
  state_entry()
  {
    llTargetOmega(<1.0,1.0,1.0>*llGetRot(),0.1,0.01);
  }
}
```

# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

To build a model using Unreal Engine requires multiple steps. The first step is to add the “Modeling Tool Editor” plugin. This plugin will help you create models that can be used in VR.

The screenshot displays the Unreal Engine interface in Modeling Mode. The central focus is the 'Plugins' window, which lists various plugins available for installation. The 'Modeling Tools Editor Mode' plugin is highlighted with a checkmark, indicating it is installed. Other visible plugins include 'ML Deformer Nearest Neighbor Model', 'ML Deformer Neural Morph Model', 'ML Deformer Vertex Delta Model', 'Static Mesh Editor Modeling Mode', 'UMG Viewmodel', and 'Wave Function Collapse (Experimental)'. The right side of the interface shows the 'Outliner' and 'Details' panels, with 'MLampA' selected in the Outliner and its properties visible in the Details panel. The bottom of the screen shows the 'Content Browser' and 'Content Drawer' panels.

Plugin Name	Version	Status
ML Deformer Nearest Neighbor Model	Version 0.2	Experimental
ML Deformer Neural Morph Model	Version 0.2	Beta
ML Deformer Vertex Delta Model	Version 0.3	Beta
Modeling Tools Editor Mode	Version 0.1	Beta (Installed)
Static Mesh Editor Modeling Mode	Version 0.1	Beta
UMG Viewmodel	Version 1.0	Beta
Wave Function Collapse (Experimental)	Version 1.0	Experimental



# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

The screenshot displays the Unreal Engine 4.27 interface. On the left, the 'Create' panel shows various primitive shapes like Box, Cylinder, Sphere, Cone, etc. The central viewport shows two custom lamps: one with a red shade and one with a grey shade, both on a grey grid floor. The right-hand side features the 'Outliner' panel listing various actors, with 'SkySpherePC2' selected. Below it, the 'Details' panel shows the properties for 'SkySpherePC2 (Self)', including location, rotation, and scale.

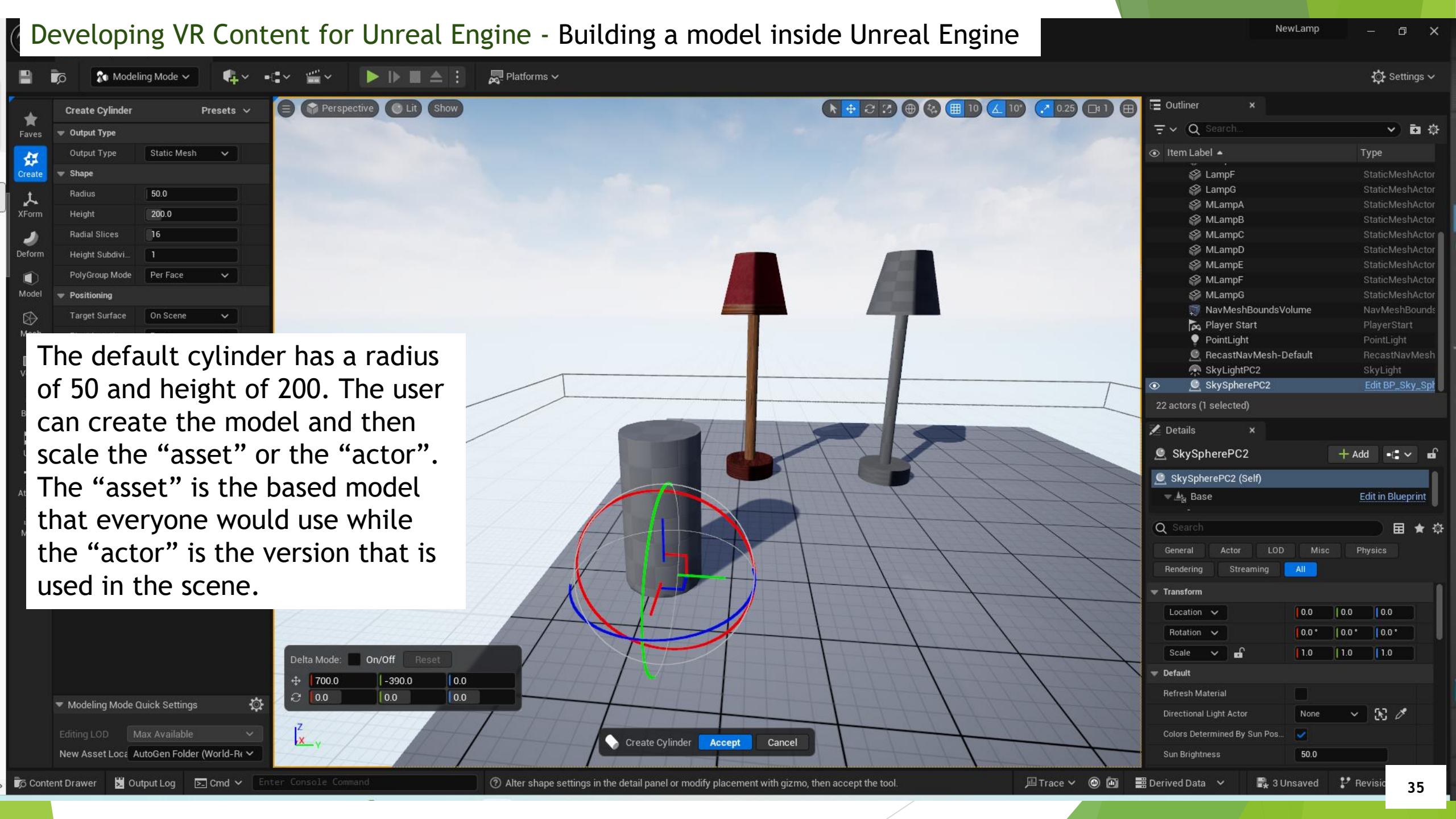
Using the Box, Cylinder and Sphere you can create this lamp(s) in Unreal Engine.

Item Label	Type
LampF	StaticMeshActor
LampG	StaticMeshActor
MLampA	StaticMeshActor
MLampB	StaticMeshActor
MLampC	StaticMeshActor
MLampD	StaticMeshActor
MLampE	StaticMeshActor
MLampF	StaticMeshActor
MLampG	StaticMeshActor
NavMeshBoundsVolume	NavMeshBounds
Player Start	PlayerStart
PointLight	PointLight
RecastNavMesh-Default	RecastNavMesh
SkyLightPC2	SkyLight
SkySpherePC2	StaticMeshActor

Property	Value
Location	0.0   0.0   0.0
Rotation	0.0°   0.0°   0.0°
Scale	1.0   1.0   1.0

# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

The default cylinder has a radius of 50 and height of 200. The user can create the model and then scale the “asset” or the “actor”. The “asset” is the based model that everyone would use while the “actor” is the version that is used in the scene.





# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

If we edit the cylinder used for the base of the lamp ...

Item Label	Type
VRTemplateMap (Editor)	World
DirectionalLight	DirectionalLight
Floor	StaticMeshActor
LampA	StaticMeshActor
LampB	StaticMeshActor
LampC	StaticMeshActor
LampD	StaticMeshActor
LampE	StaticMeshActor
LampF	StaticMeshActor
LampG	StaticMeshActor
MLampA	StaticMeshActor
MLampB	StaticMeshActor
MLampC	StaticMeshActor
MLampD	StaticMeshActor
MLampE	StaticMeshActor

Property	Value
Location	1090.0   0.0   0.0
Rotation	0.0°   0.0°   0.0°
Scale	1.0   1.0   0.1604
Mobility	Static   Station   Movabl
Static Mesh	Cylinder_301



# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

... we can see that this model was not modified, i.e. it maintained the original dimensions and the “actor” was scaled.

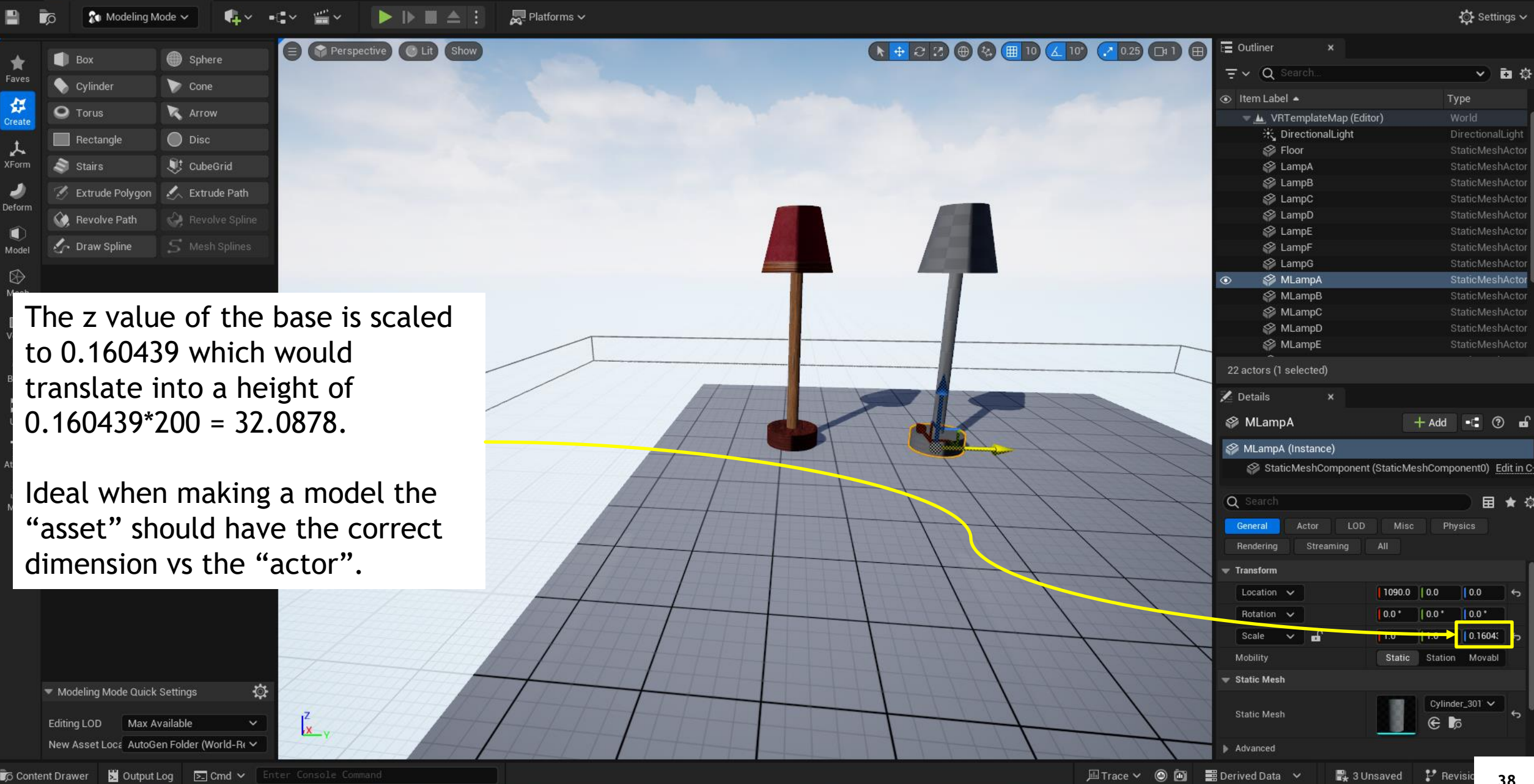
LOD: 0  
Current Screen Size: 0.689318  
Triangles: 128  
Vertices: 132  
UV Channels: 1  
Distance Field: 21x21x42 = 0.00Mb always loaded, 0.03Mb streamed  
Approx. Size: 100x100x200  
Num. Collision Primitives: 0  
Estimated Compressed Disk Size: 0.00 MB (0.00 MB Nanite)

Socket Manager x  
SOCKETS +  
0 sockets  
Select a Socket

Convex Decomp... x  
Hull Count 4  
Max Hull Verts 16  
Hull Precision 100000  
Apply Defaults

Content Drawer Output Log Cmd Enter Console Command 3 Unsaved Revision 37

# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine



The z value of the base is scaled to 0.160439 which would translate into a height of  $0.160439 * 200 = 32.0878$ .

Ideal when making a model the “asset” should have the correct dimension vs the “actor”.

# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

The pole for the lamp has a radius of 11.5 and length of 360. This

Item Label	Type
LampD	StaticMeshActor
LampE	StaticMeshActor
LampF	StaticMeshActor
LampG	StaticMeshActor
MLampA	StaticMeshActor
MLampB	StaticMeshActor
MLampC	StaticMeshActor
MLampD	StaticMeshActor
MLampE	StaticMeshActor
MLampF	StaticMeshActor
MLampG	StaticMeshActor
NavMeshBoundsVolume	NavMeshBounds
Player_Start	PlayerStart
PointLight	PointLight
RecastNavMesh-Default	RecastNavMesh
SkylightPC2	Skylight

Delta Mode:  On/Off

+	1090.0	0.0	0.0
↺	0.0	0.0	0.0

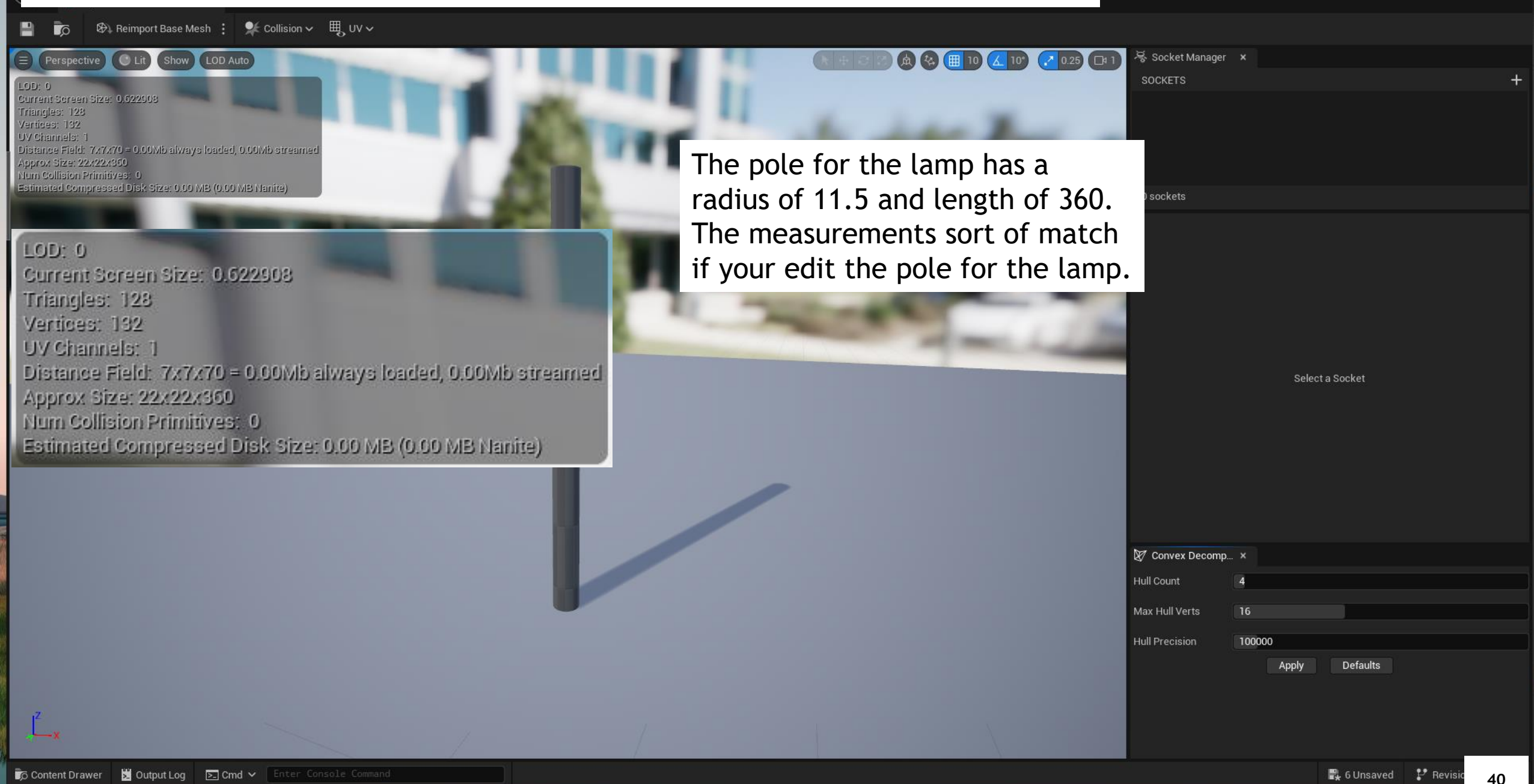
1m

Create Cylinder

Content Drawer | Output Log | Cmd | Enter Console Command | Alter shape settings in the detail panel or modify placement with gizmo, then accept the tool. | Trace | Derived Data | 6 Unsaved | Revision |



# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine





# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

Two versions were created. One without a texture and one with a texture(s). Also, it should be noted that the lamp shade was created using a Boolean operation which subtracts a smaller diameter cylinder from a larger one.



Outliner

Item Label	Type
VRTemplateMap (Editor)	World
DirectionalLight	DirectionalLight
Floor	StaticMeshActor
LampA	StaticMeshActor
LampB	StaticMeshActor
LampC	StaticMeshActor
LampD	StaticMeshActor
LampE	StaticMeshActor
LampF	StaticMeshActor
LampG	StaticMeshActor
MLampA	StaticMeshActor
MLampB	StaticMeshActor
MLampC	StaticMeshActor
MLampD	StaticMeshActor
MLampE	StaticMeshActor

22 actors (1 selected)

Details

Floor

Floor (Instance)

StaticMeshComponent (StaticMeshComponent0)

General

Transform

Location	-550.0	-550.0	-50.0
Rotation	0.0°	0.0°	0.0°
Scale	21.0	11.0	0.5

Static Mesh

SM\_Cube

# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

There are still some issues to clean up with these models. After you “Build” the level you may have issues like overlapping UVs.

- ```
△ Q_Cylinder_3017E9A5 Object has overlapping UVs.
  • Q_Cylinder_3017E9A5 Lightmap UV are overlapping by 50.0%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Box_68B60007 Object has overlapping UVs.
  • Q_Box_68B60007 Lightmap UV are overlapping by 83.3%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Cylinder_0347E005 Object has overlapping UVs.
  • Q_Cylinder_0347E005 Lightmap UV are overlapping by 50.0%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Cylinder_3017E9A5 Object has overlapping UVs.
  • Q_Cylinder_3017E9A5 Lightmap UV are overlapping by 50.0%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Box_68B60007 Object has overlapping UVs.
  • Q_Box_68B60007 Lightmap UV are overlapping by 83.3%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Cylinder_0347E005 Object has overlapping UVs.
  • Q_Cylinder_0347E005 Lightmap UV are overlapping by 50.0%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Sphere_8A353C38 Object has overlapping UVs.
  • Q_Sphere_8A353C38 Lightmap UV are overlapping by 83.3%. Please adjust content - Enable Error Coloring to visualize.
△ Q_Sphere_8A353C38 Object has overlapping UVs.
  • Q_Sphere_8A353C38 Lightmap UV are overlapping by 83.3%. Please adjust content - Enable Error Coloring to visualize.
```

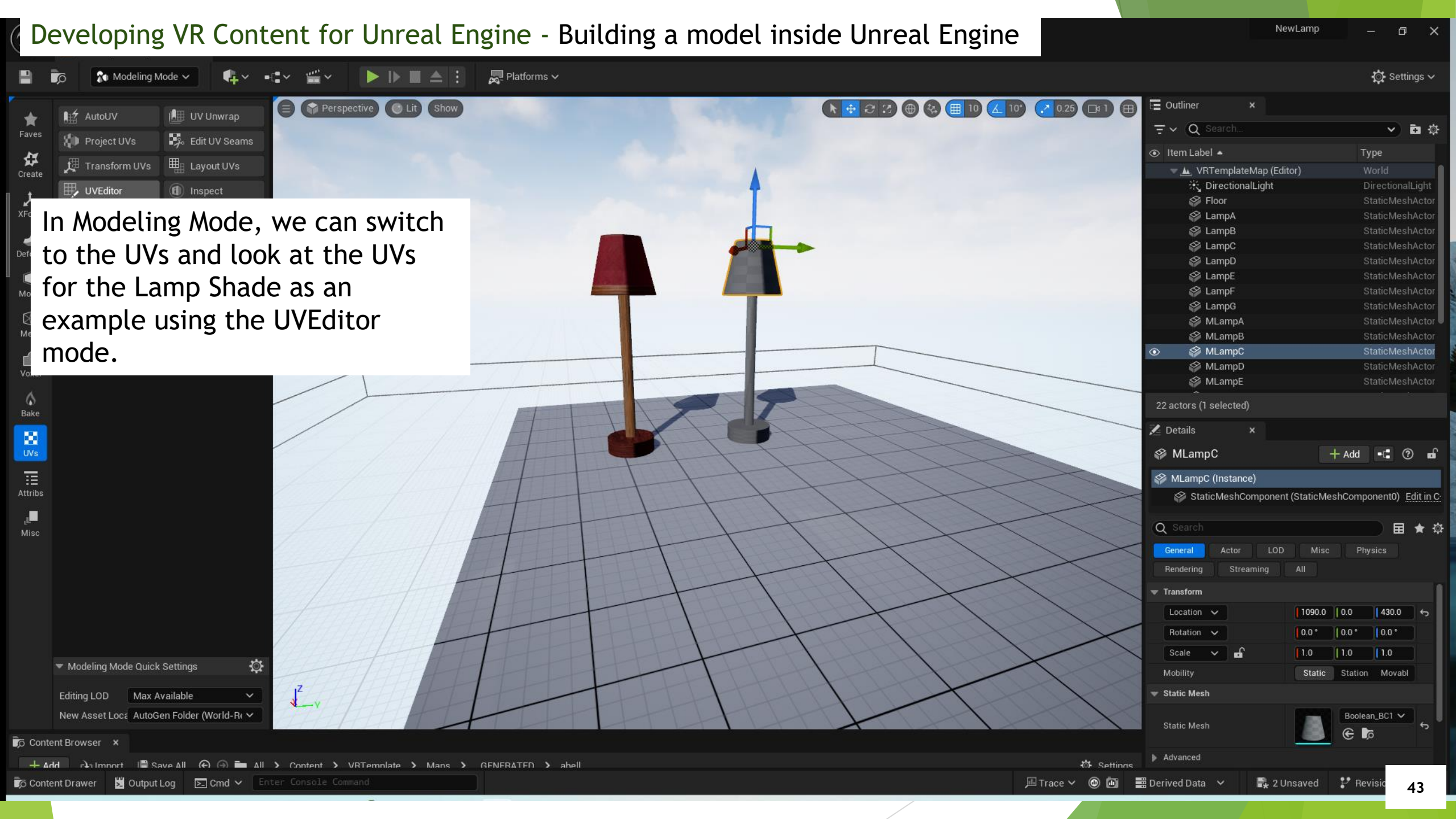
Lighting Build - Aug 5, 2024, 11:34:35 AM

CLEAR

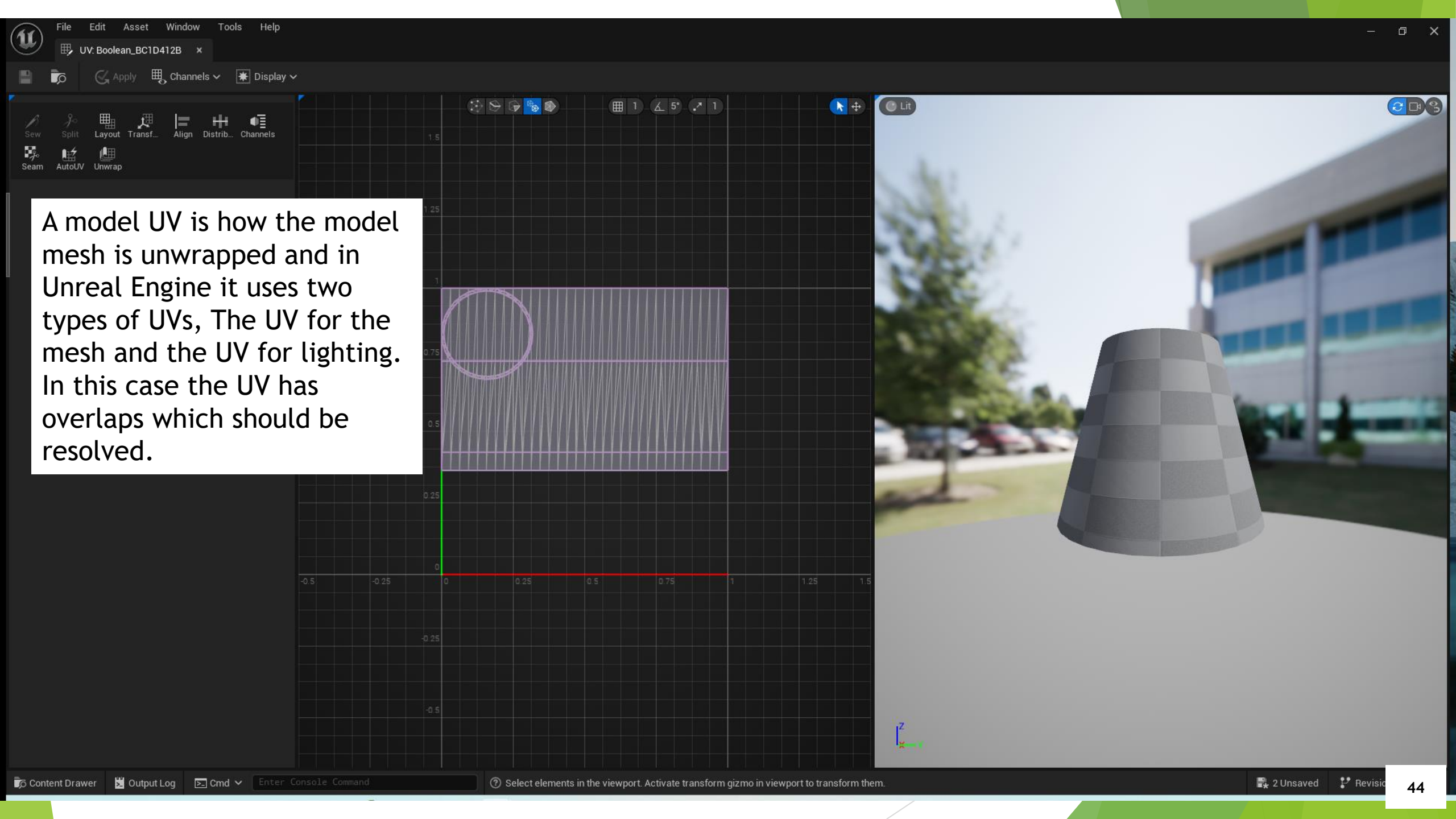


# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

In Modeling Mode, we can switch to the UVs and look at the UVs for the Lamp Shade as an example using the UVEditor mode.

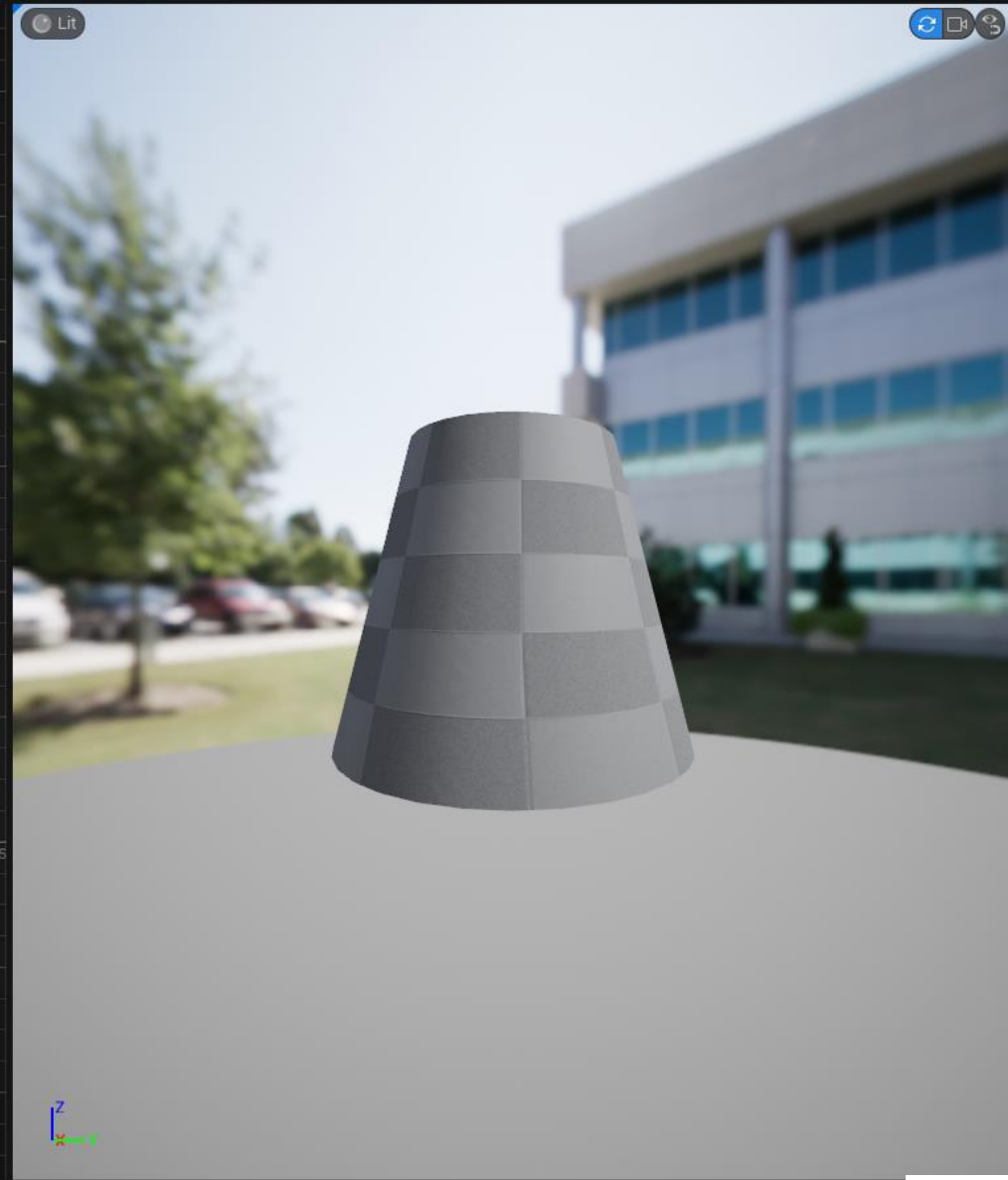
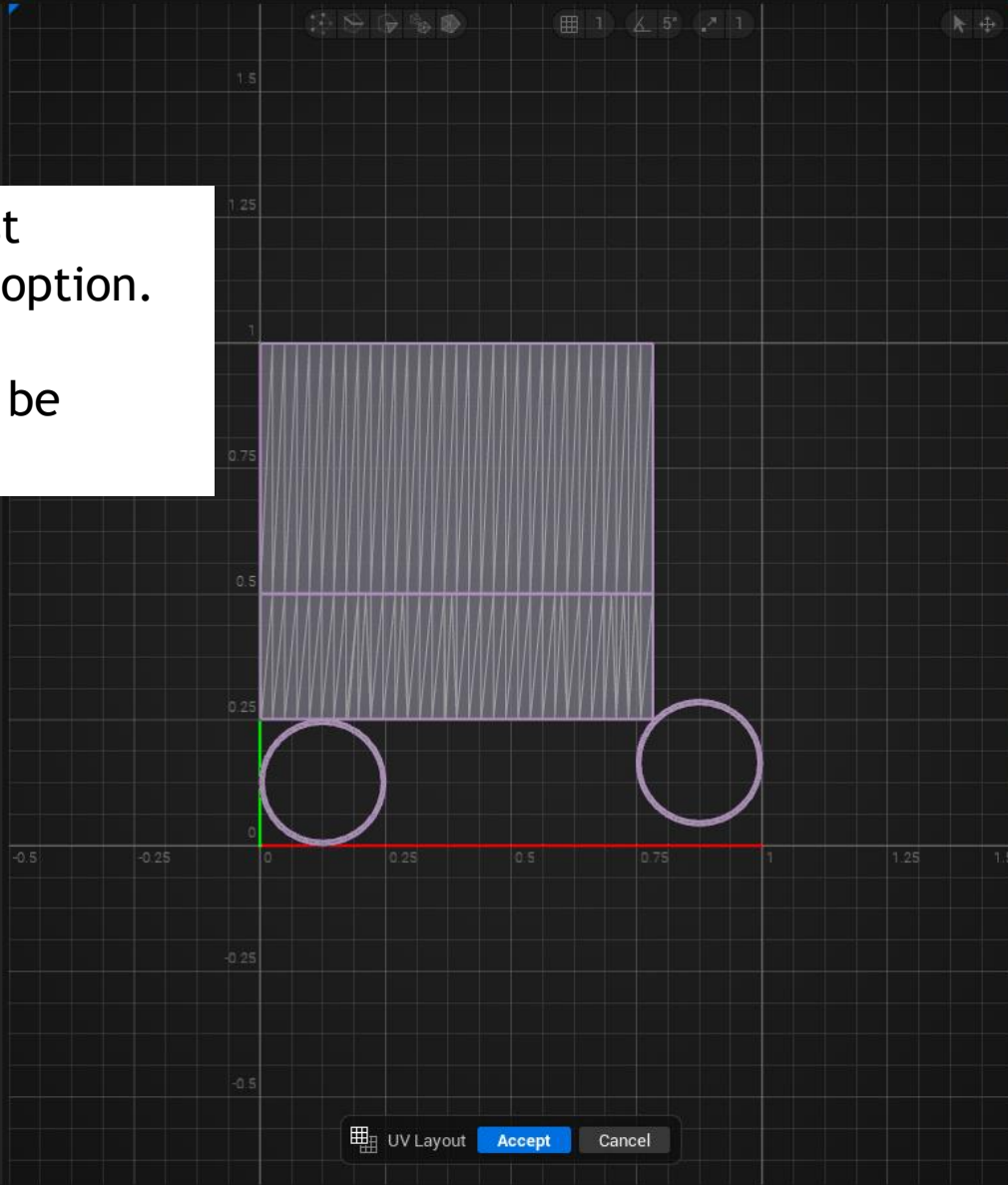






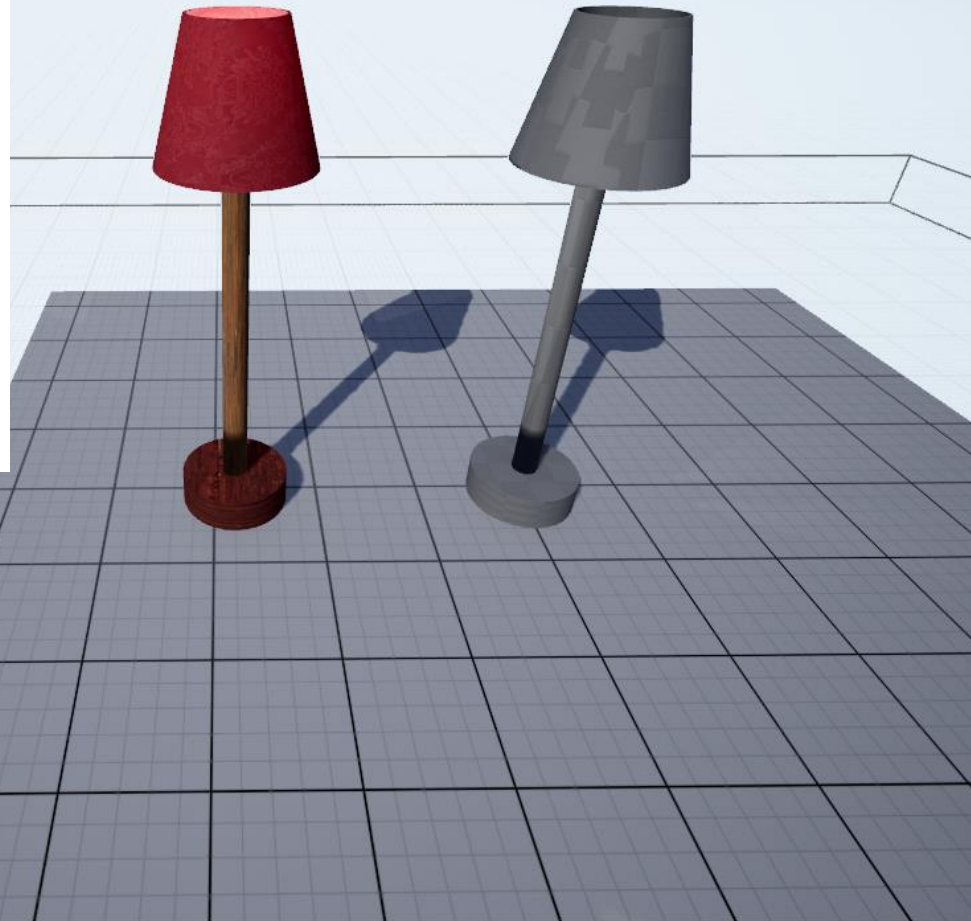
A model UV is how the model mesh is unwrapped and in Unreal Engine it uses two types of UVs, The UV for the mesh and the UV for lighting. In this case the UV has overlaps which should be resolved.

This was done by just selection the layout option. Once corrected the overlapping UVs will be eliminated.



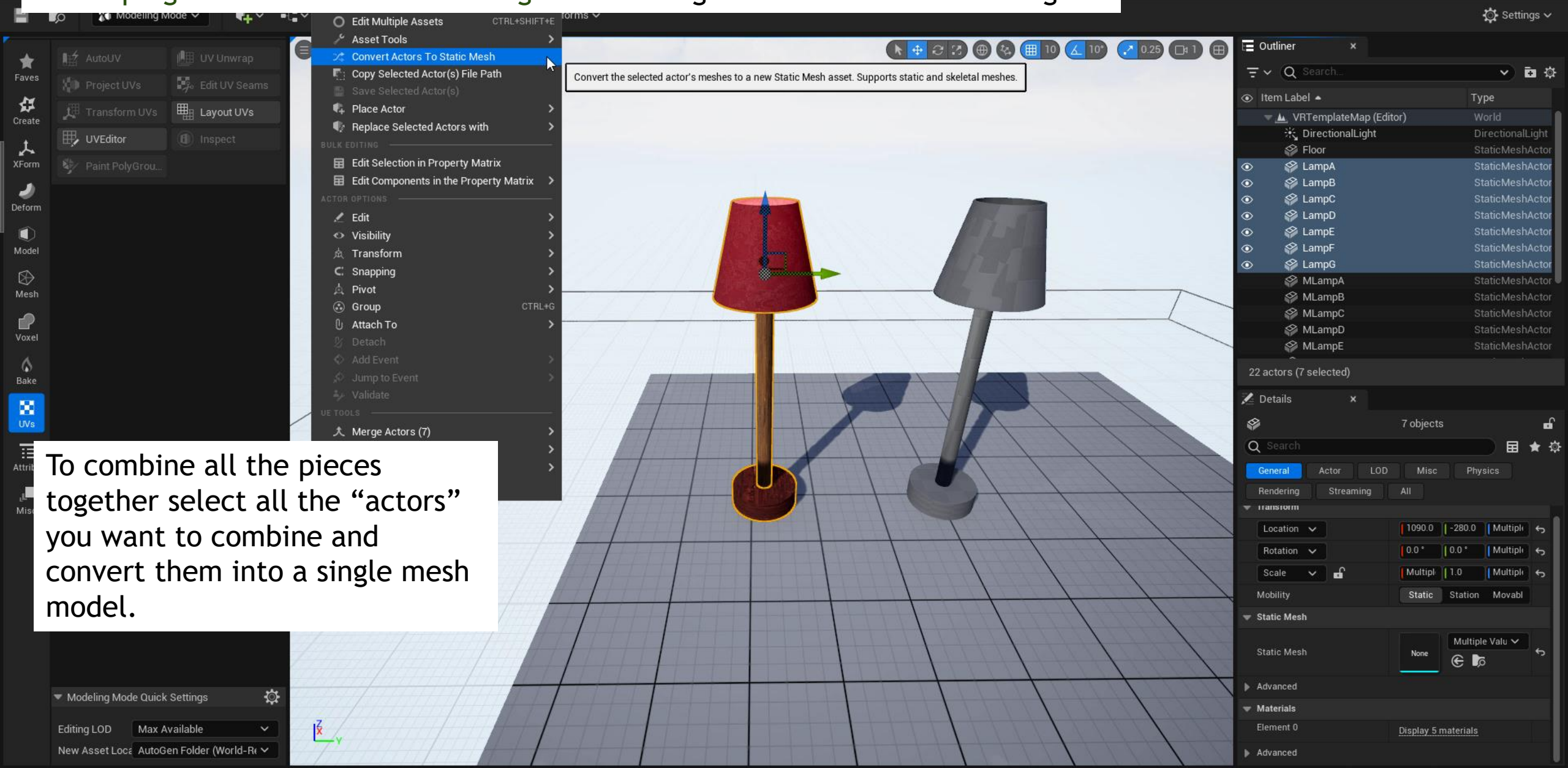
# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

Initially my plan was to show you how to export a textured model, but it is not that easy because the UVmaps for the various parts of the lamp are sort of needed to provide the level of detail shown. However, a non-textured version can be created and used and the textures for each section can be downloaded and used to build the model.





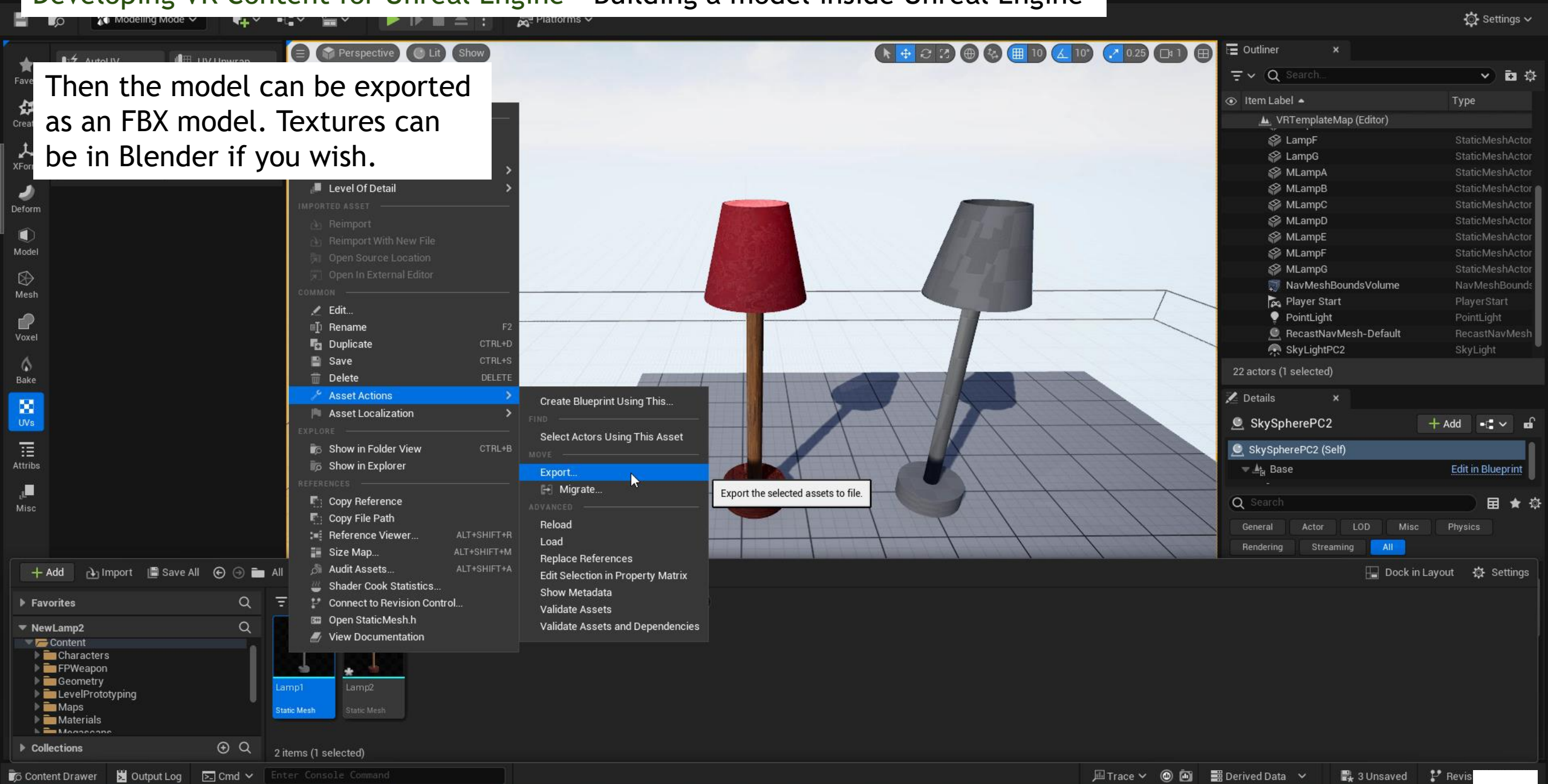
# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine



To combine all the pieces together select all the “actors” you want to combine and convert them into a single mesh model.

# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

Then the model can be exported as an FBX model. Textures can be in Blender if you wish.

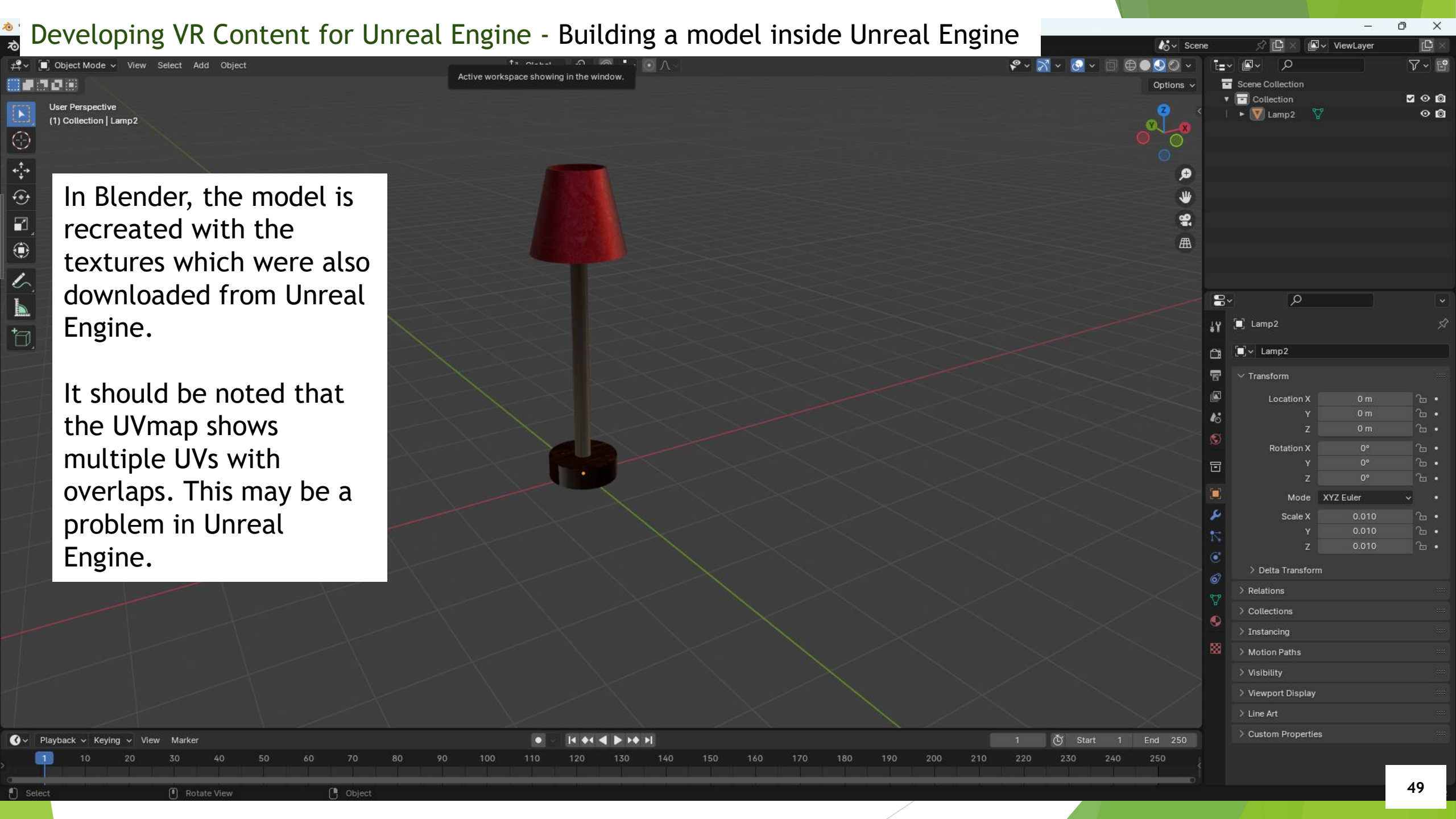




# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

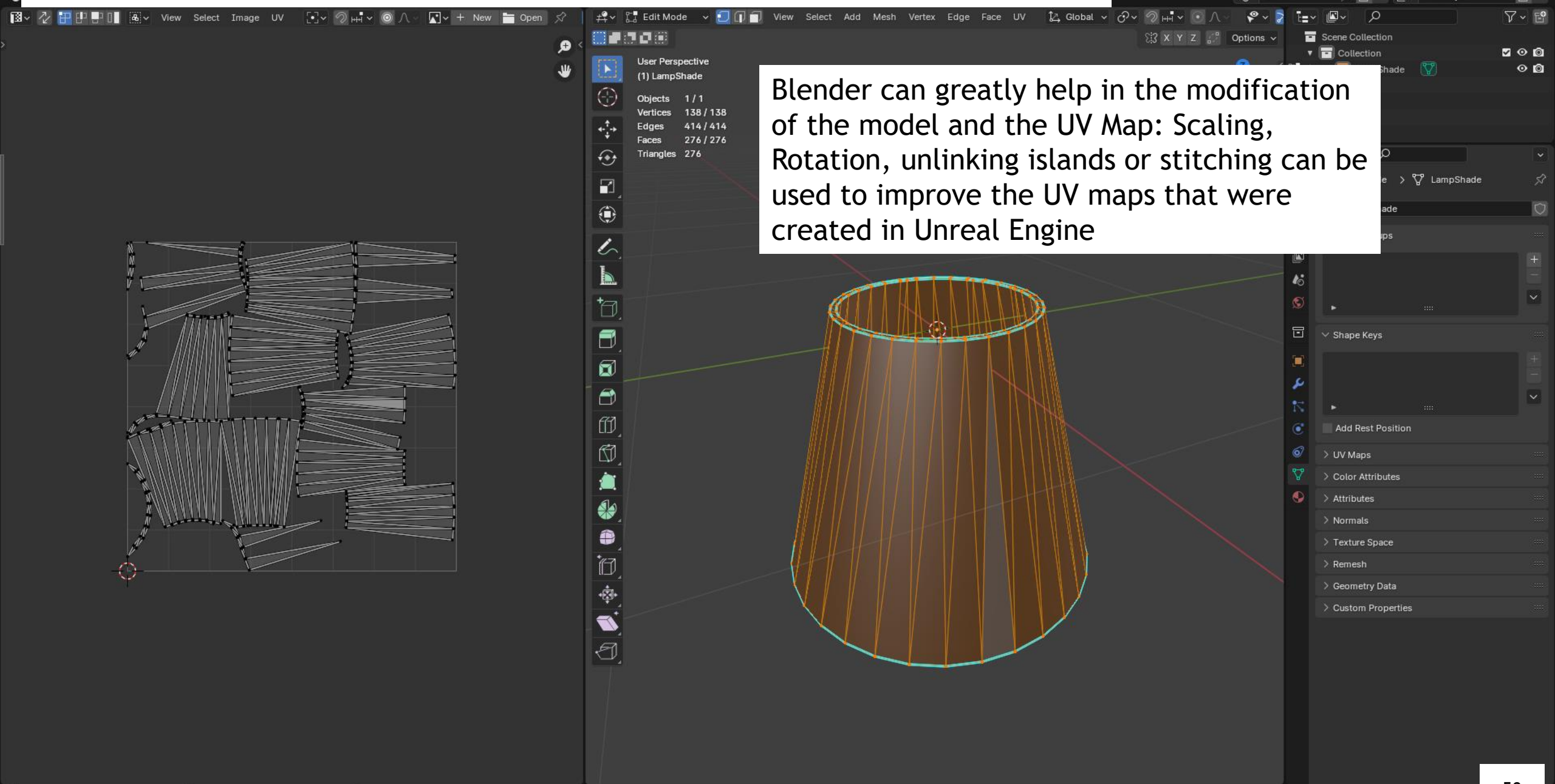
In Blender, the model is recreated with the textures which were also downloaded from Unreal Engine.

It should be noted that the UVmap shows multiple UVs with overlaps. This may be a problem in Unreal Engine.





# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine

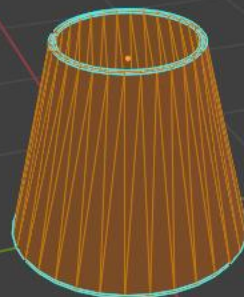


# Developing VR Content for Unreal Engine - Building a model inside Unreal Engine



User Perspective  
(1) LampShade

|           |           |
|-----------|-----------|
| Objects   | 1 / 1     |
| Vertices  | 128 / 128 |
| Edges     | 384 / 384 |
| Faces     | 256 / 256 |
| Triangles | 256       |



As can be seen the UV Map is better than the one was created in Unreal Engine. OBTW, the model was also improved by reducing the number of vertices. Tris were reduced by 20 but the geometry was improved with the elimination of poor geometry.

Scene Collection  
Collection  
LampShade  
LampShade  
MI\_Old\_Fabric\_Wallpaper

MI\_Old\_Fabric\_Wallpaper

Assign Select Deselect

Preview

Surface

Use Nodes

Surface Principled BSDF

Base Color ShadeColor

ShadeColor

Linear

Flat

Repeat

Single Image

Color Space sRGB

Alpha Straight

Vector Default

Metallic 1.000

Roughness 0.553

IOR 1.450

Alpha 1.000

Normal Normal/Map

Subsurface

Specular

Transmission

Coat

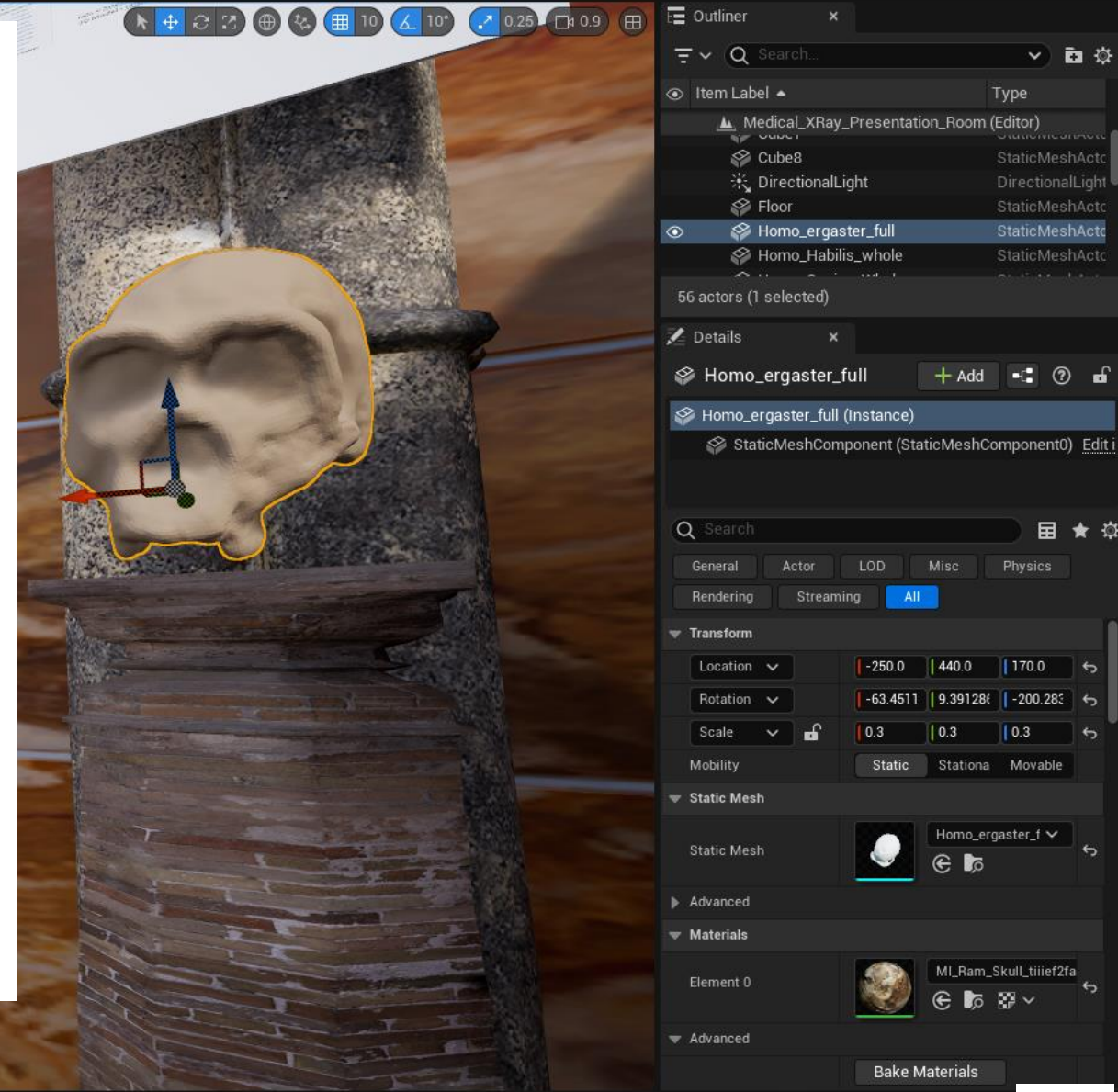


There are many other aspects to using Unreal Engine for the creation of apps and models for your Oculus Quest VR headsets.

Lighting, lightmaps, textures, materials and blueprints are all significant topics to consider and use to build VR apps.

It is also important that you learn some of the VR vocabulary so that you can ask questions to others. YouTube is a great resource and you can search for Discord Servers and ask questions in the Unreal Engine Forum. I would also suggest that you learn Blender and maybe a CAD tool like Solidworks.

Finally, the IT support you get will make your life easier when developing and supporting the use of VR headsets. If you can get admin rights on your development PC this will make it easier for you to build VR Content.





# Developing VR Content for Unreal Engine

## References

- 1.) This website contains many of the models and apps presented at the workshop.  
[https://www.ivytech-mems.org/VR\\_Models.html](https://www.ivytech-mems.org/VR_Models.html)
- 2.) My faculty webpage contains links to some previous presentation on VR.  
<https://www.ivytechengineering.com/abell118/>
- 3.) Many STL models can be downloaded from the web and used in Unreal Engine  
<https://www.thingiverse.com/>
- 4.) This another 3D model website that I use  
<https://grabcad.com/library>
- 5.) Get the latest version of Blender for free (Lots of help on YouTube for it)  
<https://www.blender.org/>
- 6.) You may find help with the Unreal Engine Forum  
<https://forums.unrealengine.com/categories?tag=unreal-engine>
- 7.) Epic Games Installer <https://www.unrealengine.com/en-US/download>
- 8.) Oculus - <https://www.meta.com/quest/setup/>
- 9.) Meta Quest Developer Hub - <https://developer.oculus.com/downloads/package/oculus-developer-hub-win/>
- 10.) Android Studio - <https://developer.android.com/studio/archive>
- 11.) Visual Studio (Community Version) - <https://visualstudio.microsoft.com/vs/community/>

# Questions



Andy Bell  
Department Chair – Engineering  
Ivy Tech Community College – Northeast  
Phone: 260-481-2288 : Fax: 260-480-2052  
abell118@ivytech.edu  
SDKB Technology Center, Room TC1240R,  
3800 N. Anthony Blvd.,  
Fort Wayne, IN 46805  
SL avatar = ivytechengineer