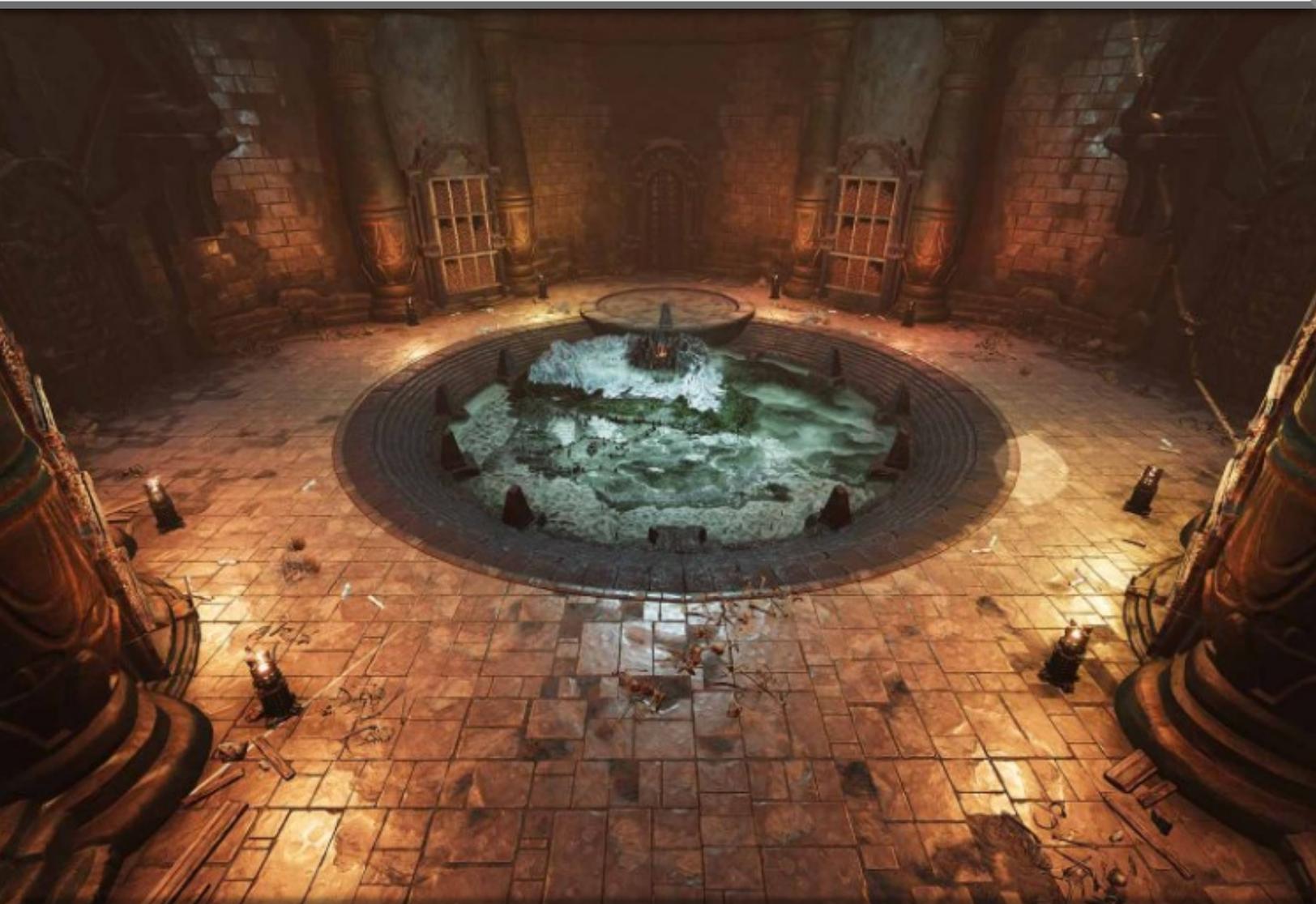


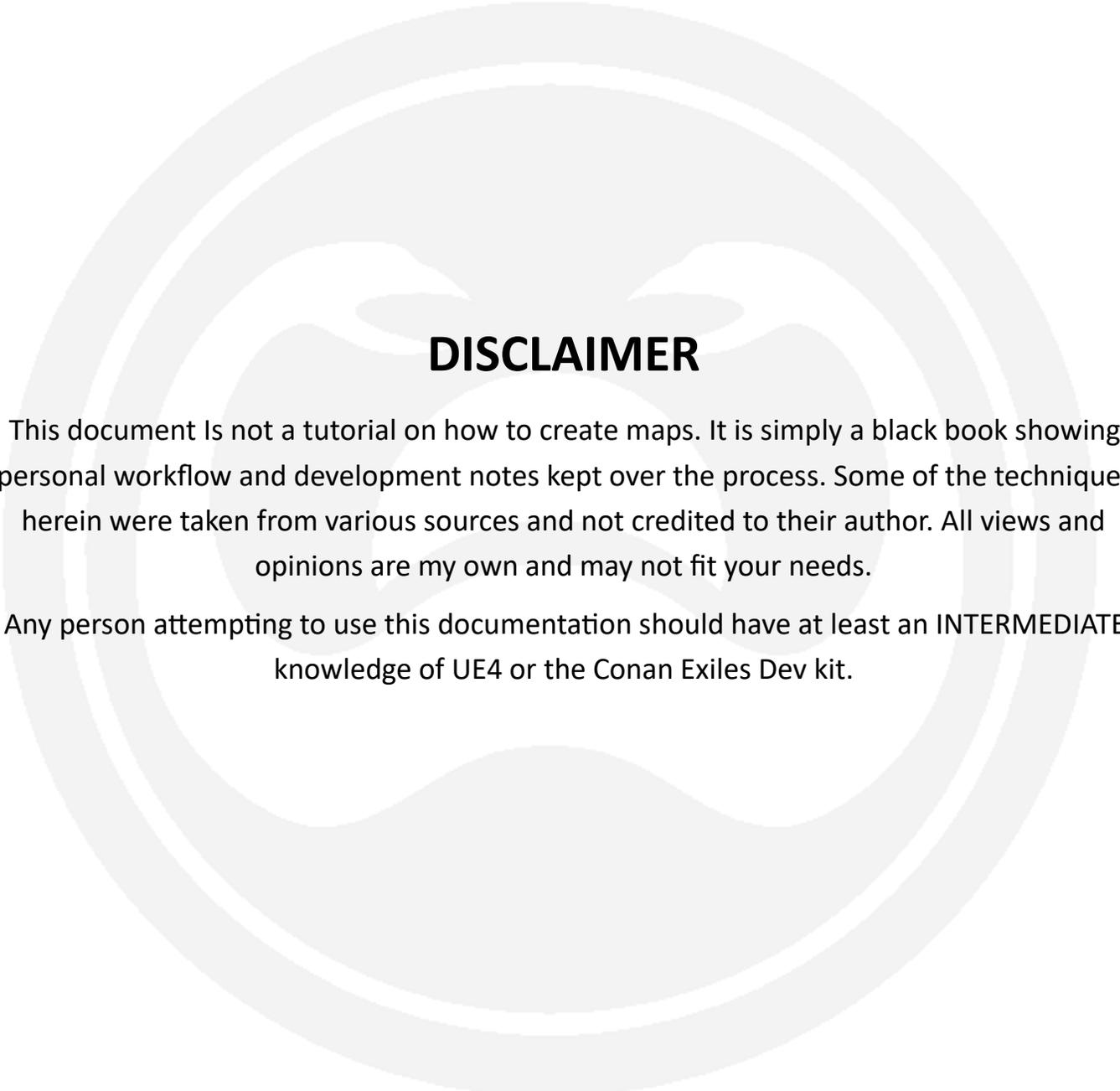
# CONAN EXILES

LITTLE BLACK BOOK OF WORLD DESIGN



FIRST EDITION

WRITTEN BY CONTRASTX



## **DISCLAIMER**

This document is not a tutorial on how to create maps. It is simply a black book showing personal workflow and development notes kept over the process. Some of the techniques herein were taken from various sources and not credited to their author. All views and opinions are my own and may not fit your needs.

Any person attempting to use this documentation should have at least an INTERMEDIATE knowledge of UE4 or the Conan Exiles Dev kit.

# CONAN EXILES

LITTLE BLACK BOOK TO WORLD DESIGN



Written By: ContrastX

Editing: Fia

Reviews: AlrenStorm, Alexandria, Shupaa, Chepi

Additional Write-ups Provided By: Funcom, Derpkitten

# BEFORE STARTING

Before making a map you will need to take into account some technical challenges. These are just a few highlights and best practices.

**MOD Size:** Your mod should NOT be larger than 2gb, and I recommend as close to 1gb as possible. Find ways to optimize. Delete unneeded files and keep directories clean. Try not to import a bunch of custom meshes. *Its best practice to use FC's assets in different ways to get a desired result.*

**Start Small:** Creating a map that is 8x8 at 1010px is a MASSIVE undertaking. Especially if you are the only one working on a project. For your first map, make a 3x3 at 505px. A 505 map will take several weeks to properly complete and get a desired result. This will also help fine tune your skills.

**Deceiving Size:** You will not know it at first but a simple single tile map at 505px is a lot of work. Be sure to play the map in your editor OFTEN. What you may think is a tiny place normally ends up very large. A 505px X 505px, landscape piece is 1656.82 feet squared, OR ONE THIRD OF A MILE! Imagine having to plant trees, rocks, resources, water and grass for one third of a mile!

**Save Often:** Be Sure to save your work often. The more stuff added to your mod will cause longer loading times in UE4 and may on occasion freeze up the dev kit.

**Test Assets:** Some assets show in the dev kit but will not show when your mod goes live. Don't only test in the DEV Kit, but push to steam privately, load it to a server and see what works. All to often you will get a different result and you will need to correct it before the public plays on it.

**Gloss through THIS PDF!:** Before jumping into a map, skim through this PDF first. DON'T try to follow it by the letter and read the whole thing. Its simply a reference and in the order I did things. **This is NOT the end all be all document.** Once you get the gist of its contents you will find one section leads into another.

**Trial and Error:** You will most likely not get map making correct on the first try. It's okay. I spent several weeks just on making heightmaps and trying to get the right feel, only to find myself remaking them again.

**Perseverance:** If your serious about making a map you have to push through the many long hours and months required to do it...and that's for a small one. You WILL spend weeks on the many individual parts of making a map. If you make a playable map in a day, you most likely did something wrong. TAKE YOUR TIME and enjoy your creation! Start small and expand to other regions later.

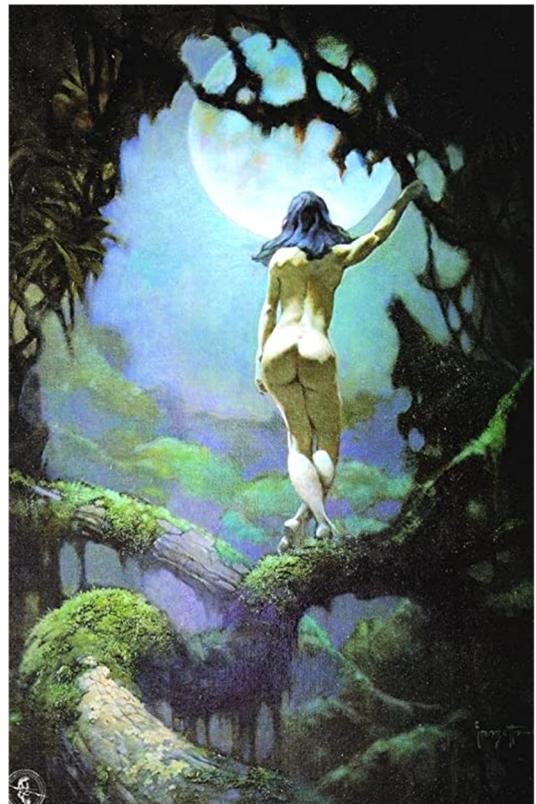
## Play Your Map

One of the most important things to do is PLAY YOUR MAP. You need to see what the player sees. If you put in a massive mountain that you think will give a great climbing experience, CLIMB IT! You may find that the experience is not all that great or could use some cliffs to help with stamina.

**Is It Fun?:** The number one rule is, "IS IT FUN?" A map can have all the cool treasures, creatures and McGuffins in the world, but if the map is not environmentally fun and offer a sense of wonder it will quickly loose its luster. Remember, a map is the "game board" for players and needs to offer playable areas. Be sure not to put in challenges that cant be overcome unless absolutely deemed so. Examples include blockers and anti-climb across the whole map, thus ruining the experience.

HAVE FUN, and Game ON!

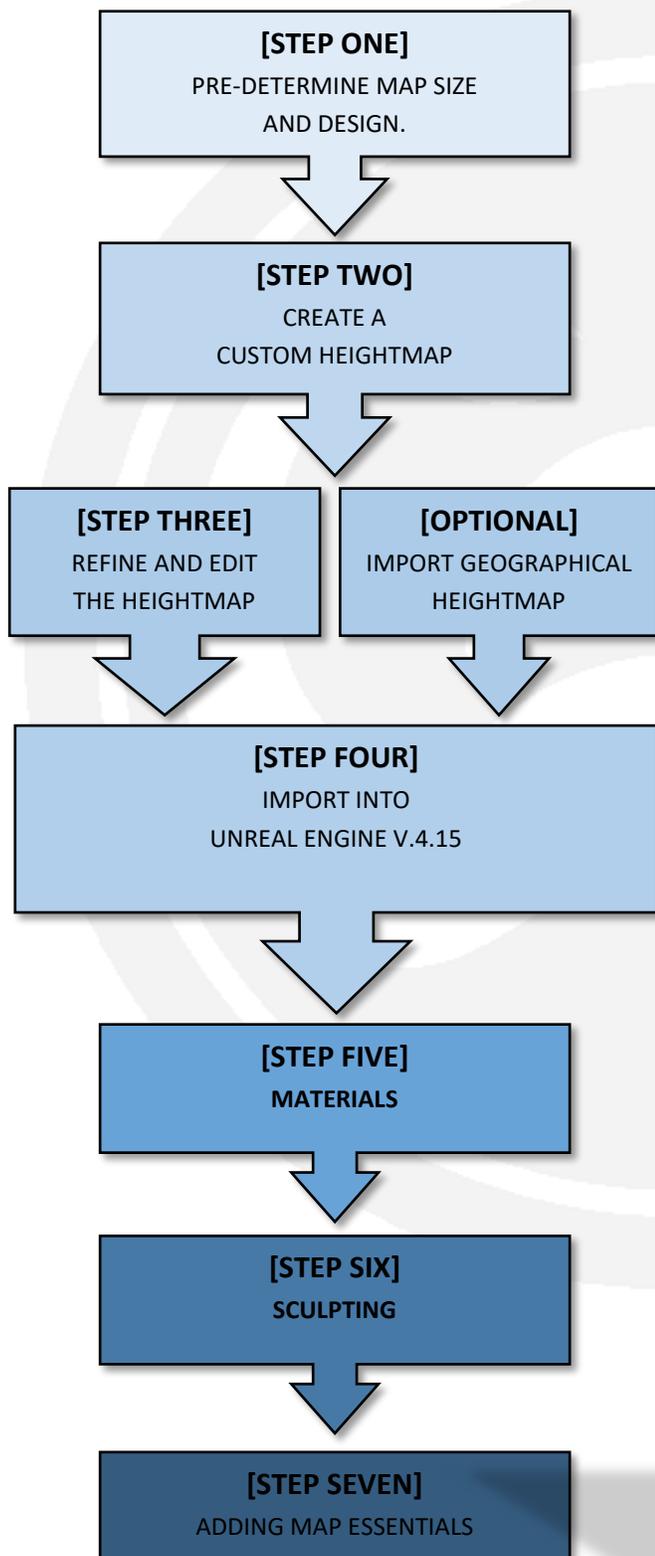
-ContrastX



# FOREWORD

In this document we will be making a playable Conan Map from scratch. Through this document I will not teach the individual use of these programs but it will show the workflow and key points in production. We will be using GIMP, TerreSculptor, Blender and the Conan Exiles Dev Kit.

*Our workflow will look like this:*



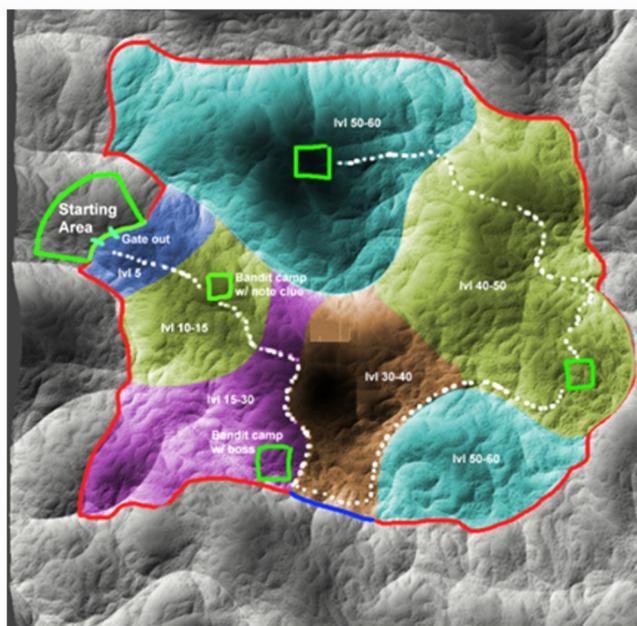
# Map Designing & Flow

You should have a rough idea of what your gameplay will be. What will the goals of the map be? For example, will your players have to survive an arctic region? How about survive the heat of a desert. You will then need to design the flow of the map. Here is an example of flow.

*"To explain the flow here.. The starting area brings players to level 5ish, there's also an area outside to make certain players get to level 5 at least by super-easy mobs etc. I didn't flesh out the story really - all I know is that I want players to traverse in a circular fashion - the reason for doing so is that the map then has a spoke in the middle which becomes prime building real-estate for people of any level, since they can reach all areas if they build on that ridge. At the same time, on a PvP server, those areas would be exposed enough to provide Risk Vs. Reward.*

*The level areas increase in size as levels go on, since most players that play will spend way more time at higher levels than at low levels. I decided to plop down a level 50-60 area in the southeast simply because 1) I had enough for progression already, and 2) because of more time spent at high levels, I decided "hey why not"*

– Robtheswede



# S

## STEP ONE: Determine Map Size

First determine the size of the map you want to design. Several *sections* are sewn together to create a larger map. It's important to note that *sections* are loaded areas the player is currently in. Sections make up components. For the purposes of getting started we care about the number of sections that our map will have.

### ◆ DEFAULT UE4 MAP EXAMPLE

A default Unreal 4 map *section* has the following Specs.

*Resolution: 505*

*Quads: 63*

*Sections: 1*

*Components: 8*

It is important to note that Conan Exiles maps have a higher resolution than the default.

### ◆ EXILED LANDS MAP SIZE EXAMPLE

The Exiled Lands Map has the Following Specs:

(8 Sections wide x 8 Sections high) IMPORTANT: The CE map is approximately 7 x 7 since NOT all the map is playable and contains dungeons, special locations and the NAV mesh which effects NPC AI due to UE4 limitations.

### ◆ SIPTAH MAP SIZE EXAMPLE

The Siptah Map has the Following Specs:

(8 Sections wide x 7 Sections high) IMPORTANT: The Siptah map is approximately 6 x 6 since NOT all the map is playable and contains dungeons, special locations and the NAV mesh which effects NPC AI due to UE4 limitations.

So, what size should you use? The question is relative but think of it like this: Large maps will look more like a miniature golf course while smaller maps will look like a 12-hole golf course. The tradeoff is the overall look. Large maps are more global looking and will give beautiful vista views. Smaller maps can focus localized areas and have very few far-distance views.

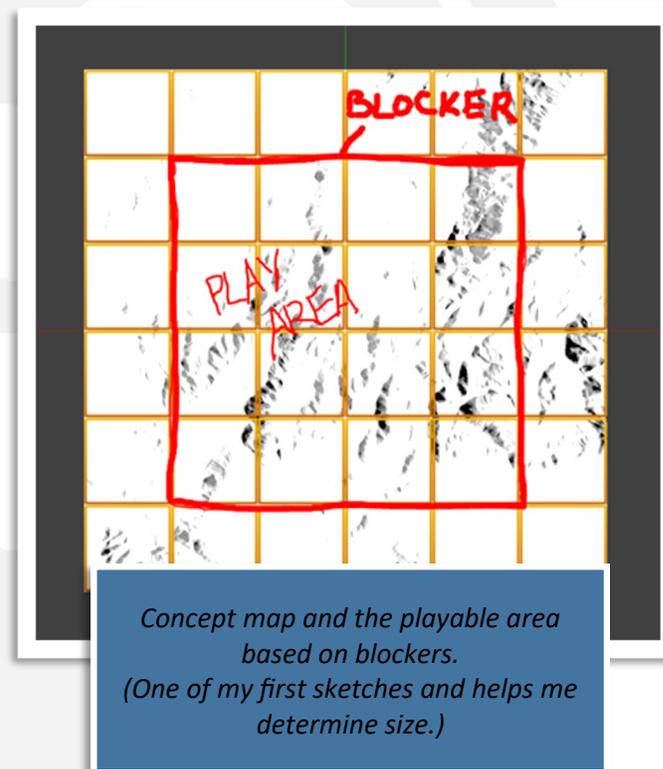
Resolution is also important. You can make a map with a 505 x 505 resolution which is great on file size and helps against file bloating. However a 1010 x 1010 allows more detail in the sculpting but quadruples the file size.

## Considerations on Size

When in the concept phase of map design, think about the players view and access outside the map limits. To make the world more believable you should have at least one tile that renders beyond the "blocker" volumes. This is more of a trick of the eye to make the world feel more infinite.

Try to use geographical blockers such as water, oceans, toxic fume areas, manmade walls or mountains to give your map story.

**FUN FACT:** *1px = 1 Meter in UE4 landscapes. Thus a 505px heightmap = 1656.82 feet. The CE map is roughly 4.3 miles wide.*



To keep things simple, most of our examples and explanations here we will be designing our map with a 505 x 505 resolution. If you want a true Conan Exiles map size use the 1009 x 1009 resolution. The next page will contain info for each heightmap imported should you want a more traditional size. Simply make your mathematical conversions as needed.

## OPTIONAL (ENHANCED RESOLUTION INFO)

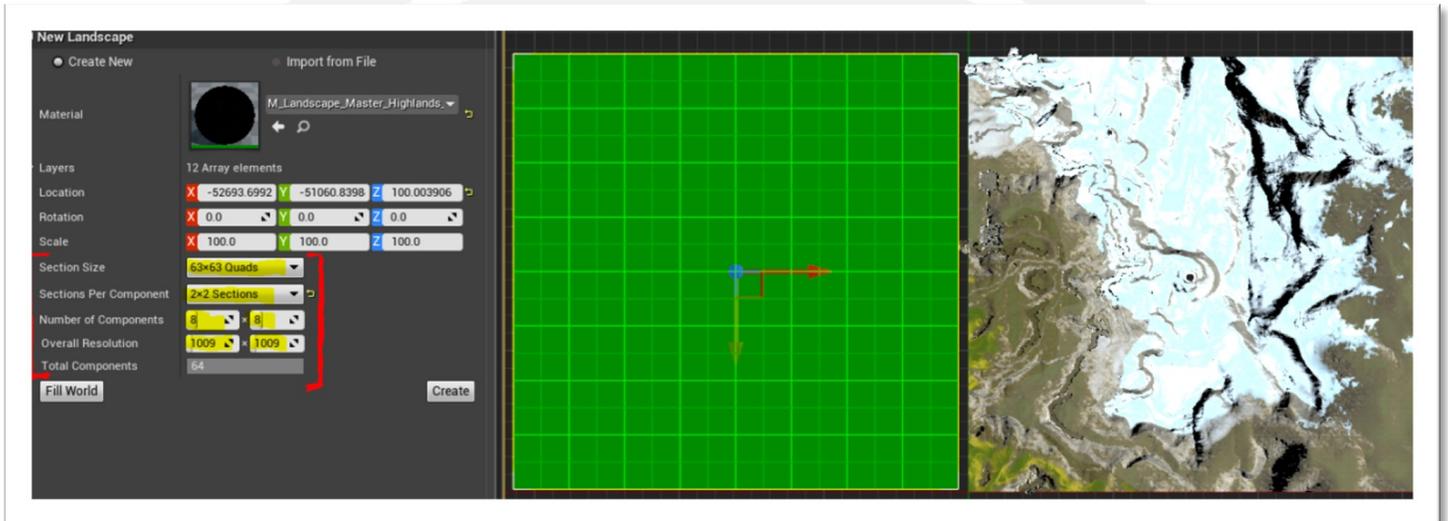
(REFERENCE ONLY! DO NOT USE IF FOLLOWING THIS DOCUMENT AND USING THE 505px RESOLUTION).

If your comfortable with map making and want to make a traditional CE map use the following specs.

### One Section (Tile) in Conan Exiles

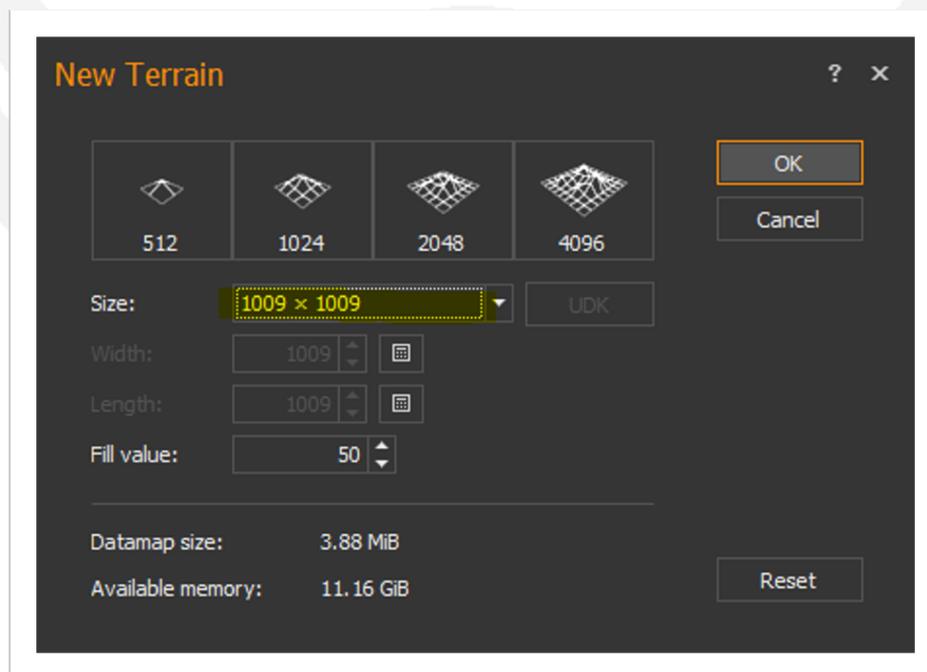
Below are the specs for one tile in Conan Exiles. The Exiles map contains 64 tiles. That's 8 tiles wide and 8 tiles tall to make a complete map.

- ◆ *Section Size: 63 x 63*
- ◆ *Sections Per Component: 2 x 2 Sections*
- ◆ *Number of Components: 8—8*
- ◆ *Overall Resolution: 1009 x 1009*



### TerreSculptor

If using TerreSculptor change the resolution to the below settings. Warning higher resolutions and map sizes will cause program crashes. Save often!



## Performance Considerations

The choice of component size vs the total number of components is a performance trade-off. Smaller component sizes allow quicker LOD transitions and also allows for the occlusion of more terrain, but the smaller size necessitates more components.

Each component has a render-thread CPU processing cost and each section is a draw call, so try to keep these numbers to a minimum. For the largest Landscapes, Epic recommends a maximum of 1024 components.

**REFERENCE:** <https://docs.unrealengine.com/en-US/Engine/Landscape/TechnicalGuide/index.html>

### Recommended Landscape Sizes

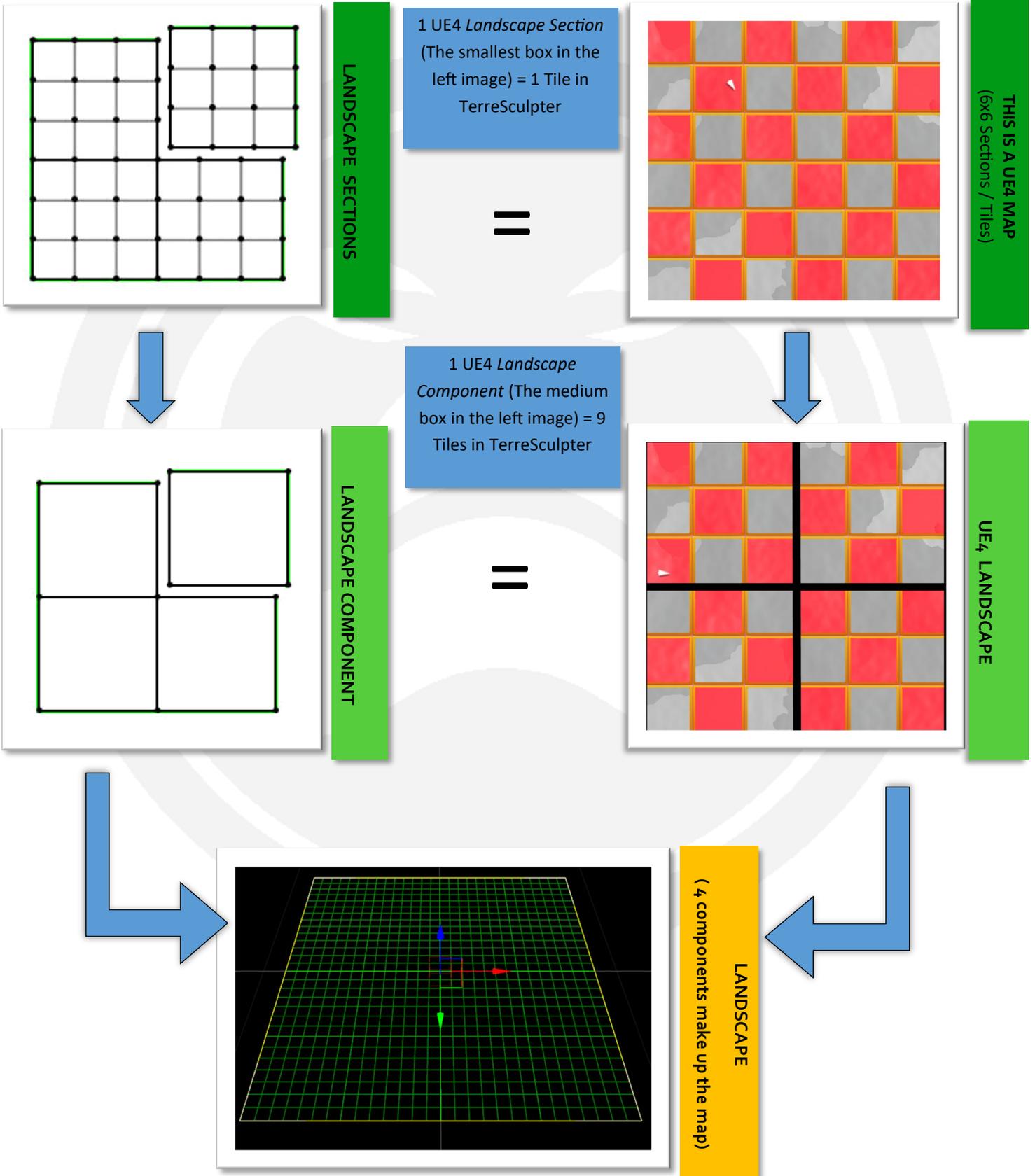
In order to make things easier, here are a number of sizes that maximize the area while minimizing the number of Landscape components.

Overall size (vertices)	Quads / section	Sections / component	Component size	Total Components
8129x8129	127	4 (2x2)	254x254	1024 (32x32)
4033x4033	63	4 (2x2)	126x126	1024 (32x32)
2017x2017	63	4 (2x2)	126x126	256 (16x16)
1009x1009	63	4 (2x2)	126x126	64 (8x8)
1009x1009	63	1	63x63	256 (16x16)
505x505	63	4 (2x2)	126x126	16 (4x4)
505x505	63	1	63x63	64 (8x8)
253x253	63	4 (2x2)	126x126	4 (2x2)
253x253	63	1	63x63	16 (4x4)
127x127	63	4 (2x2)	126x126	1
127x127	63	1	63x63	4 (2x2)

# Landscape Parts & Overview

Yellow—Landscape Edge  
Light Green—Component Edge  
Medium Green—Section Edge

\*\*Quads are not shown here, but are tiny squares found in individual sections.



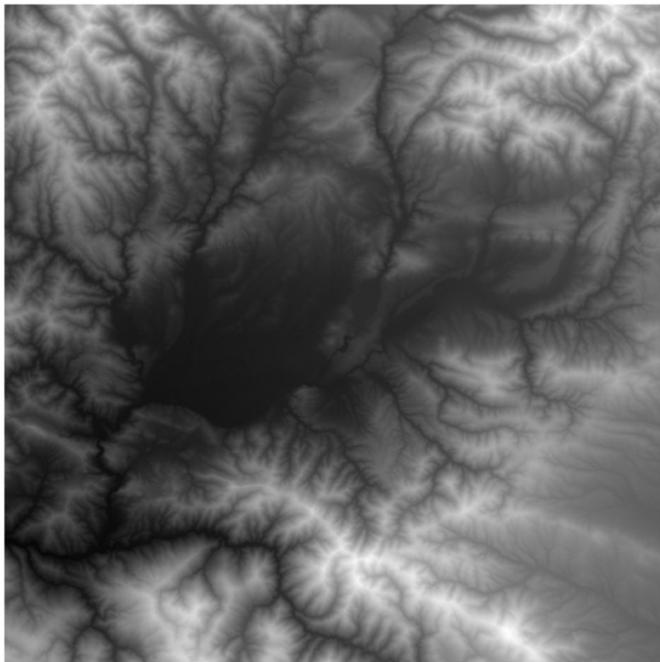
## S STEP TWO: Create Your Height Map

Open your favorite art program and begin sketching out your map design. You will need the following specs when setting up and saving your finished map.

- ◆ Determine the size of your map. To do this multiply the number of sections you want by a resolution of 505. Example: *I want a map that is 3 sections wide and 3 sections high. So, my image size will be 1515px x 1515px. [505 x 3 = 1515px]*

(IMPORTANT): The bigger the map you may want to lower the resolution to avoid your program freezing). I'll be using 300px for a 1515 x 1515 map. (If using the info from pages 7—8, you will have to adjust accordingly).

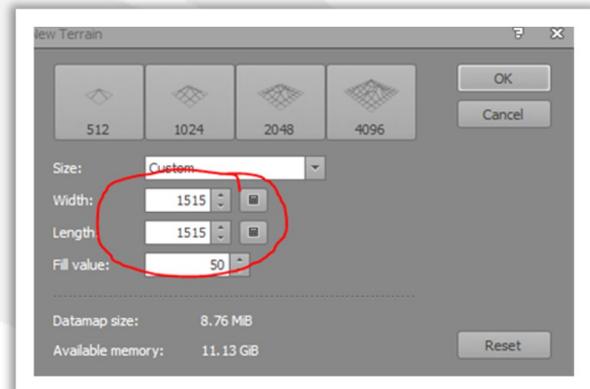
- ◆ Export as: PNG, 16-bit, Greyscale. Be sure to take your time and refine your export to help with details and layout. This will help with the clean up later. Save/Export as a PNG.



## S STEP THREE: Import to TerreSculptor

I will not be doing a tutorial on using this program however a download link can be found HERE: <http://www.demenzunmedia.com/home/>  
TUTORIALS HERE: [https://www.youtube.com/channel/UC\\_8MqFatjX\\_L7QqfuZpyHjg](https://www.youtube.com/channel/UC_8MqFatjX_L7QqfuZpyHjg)

- ◆ Open TerraSculpter
- ◆ Select New Project
- ◆ Adjust the Width to 1515 and Length to 1515 (px)
- ◆ Select OK

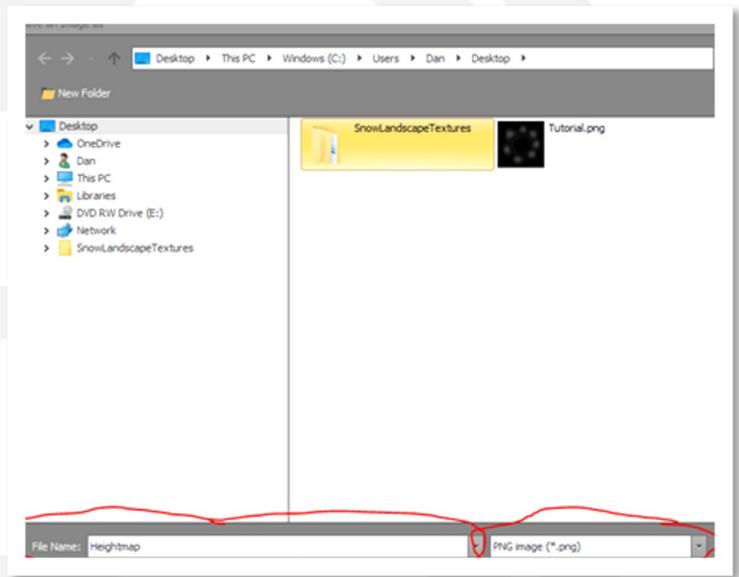
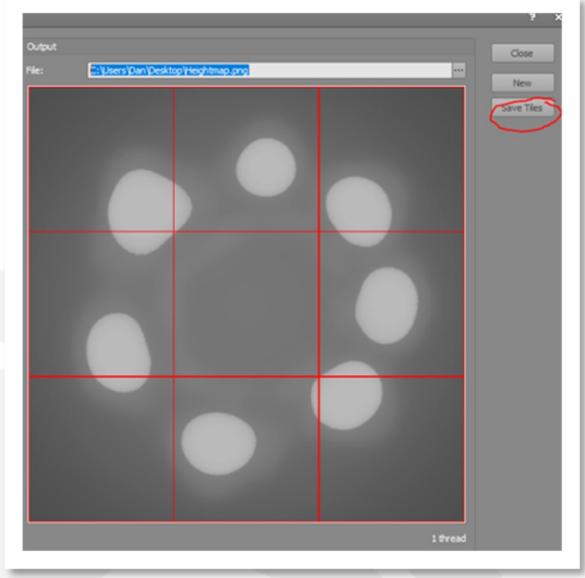
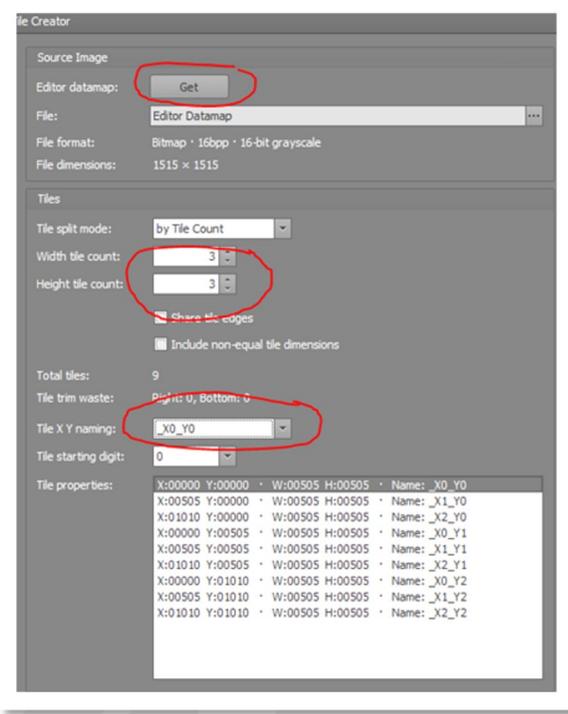


- ◆ Open File at the top left
- ◆ Import Terrain, (select the file type) so your file will show as an option.
- ◆ Select the Heightmap created in GIMP
- ◆ Start making the adjustments to clean up the heightmap. Once you have TerreSculpter, import your map and begin fine tuning its geographical looks. Add erosion, change the altitude and much more. This is where the heightmap cleanup will pay off. However, take more time here for additional details and a secondary refinement for a cleaner look.

**[HINT]:** Go to *Modify > Convolution Filter > to smooth out the lines between heights. You can also use Modify > smooth to clean up artifacts.*

- ◆ Create > Tile Creator > Get in the editor Data map
- ◆ Go to the *Create* tab > *Tile Creator*.
- ◆ Click the *GET* button to automatically import your map.
- ◆ Change the *Width Tile Count* and *Height Tile Count* to the number of *sections* your map will have. In our example it will be 3 x 3.
- ◆ Change the *Tile X, Y Naming* drop down menu to *\_x0\_y0*.

(See next page for image)



- ◆ Use the **SAVE TILES** button to export the height maps to a separate folder. These tiles will be used later to import into UE4.
- ◆ It is important to Save the map as “Heightmap” and as a “.png”

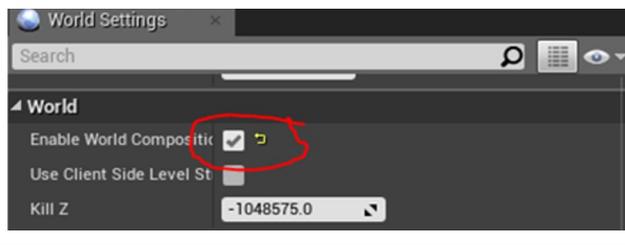
### OPTIONAL: Use A Premade Heightmap

- |  |  |   |
|--|--|---|
| <p>If you do not want to create your own heightmap you can use <a href="http://terrain.party/">http://terrain.party/</a> and import the file in TerreSculptor.</p> <ul style="list-style-type: none"> <li>◆ For a more localized map use 30km screen capture in Terrain Party and download your heightmaps.</li> <li>◆ When you open the heightmaps up we will be using the SRTM3 V4.1 map since it's easier to clean up.</li> <li>◆ Load the SRTM3 map into TerreSculptor</li> <li>◆ Go to Modify &gt; Convolution Filter.</li> </ul> | <p>Change the preset to 6 x 6. Use Erosion.</p> <ul style="list-style-type: none"> <li>◆ Go to modify and use “Altitude Top” to shift the map.</li> <li>◆ Go To “Altitude” and make and make additional altitude adjustments.</li> <li>◆ Make any other adjustments needed.</li> <li>◆ Export and save as a png.</li> <li>◆ Import into your favorite art program and scale the image size. Be sure to use 505 x Sections. We will have 5 sections so 3030 x 3030 (505 x 6 = 3030 x 3030). Save as a .png and reimport into</li> </ul> | <p>TerreSculptor.</p> <ul style="list-style-type: none"> <li>◆ Create &gt; Tile Creator &gt; Get in the editor Data map</li> <li>◆ Go to the <i>Create</i> tab &gt; <i>Tile Creator</i>.</li> <li>◆ Click the <i>GET</i> button to automatically import your map.</li> <li>◆ Change the <i>Width Tile Count</i> and <i>Height Tile Count</i> to the number of <i>sections</i> your map will have. In our example it will be 6 x 6. Change the <i>Tile X, Y Naming</i> drop down menu to <i>_x0_y0</i>.</li> <li>◆ Save Tiles</li> </ul> |
|--|--|---|

## STEP FOUR: Import to UE4

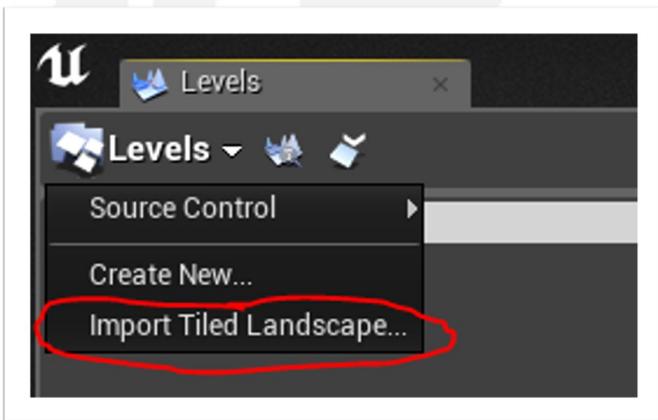
In this section you will be setting up your gameplay area. Open the Conan Exiles Dev Kit and create a new mod.

- ◆ Open UE4 and create a new mod (This will take a little bit to open)
- ◆ File > Create New Level > Select Save Current > Select your mods folder > Add Name > Save
- ◆ Save and name your level. For example, CustomMap. Be sure to select your mod folder when saving.
- ◆ Look in World Settings Panel and Enable World



Composition. This is a very Important step.

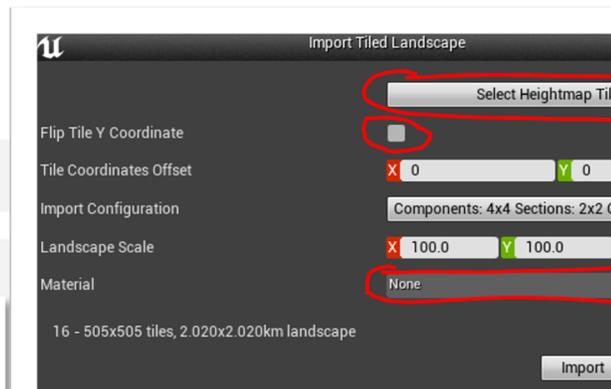
- ◆ Go to Window and open > Levels
- ◆ Import Tiled Landscape in the levels panel.



### SPECIAL NOTE ON CAVES, MATERIALS & HEIGHTMAPS

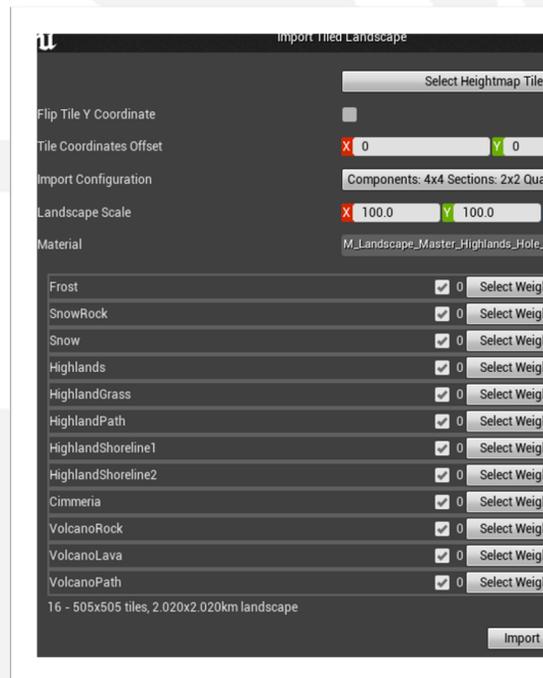
It is important to note if you are importing a heightmap you will need to attach a material before the import process. UE4 does not like to make holes in the heightmap after the fact.

- ◆ In the small window that opens select the files that were created for the Heightmap
- ◆ Uncheck Flip Tile Y Coord
- ◆ Change the Tile Coord offset to -3 for X and -3 for Y.



Without Material Instance.

- ◆ Select Import. The tiles created in TerraSculptor will load in.
- ◆ If you have a Material Instance be sure to select it in the "Material" drop down menu.



With Material Instance Chosen

- ◆ Once everything has been selected click "IMPORT".

### Good Landscape Materials to start with:

*M\_Landscape\_Master7\_Inst*

*M\_Landscape\_Master\_Highlands\_Inst*

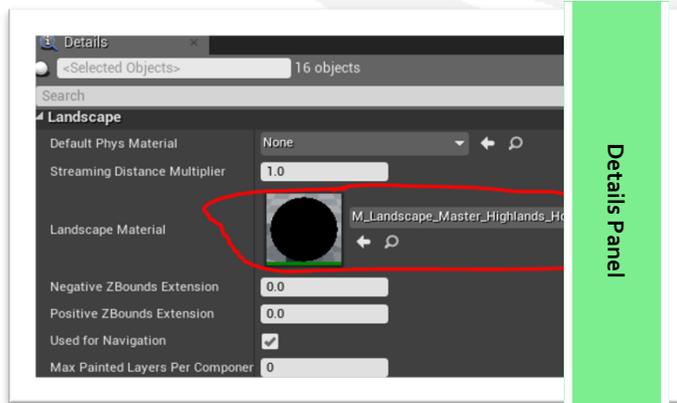
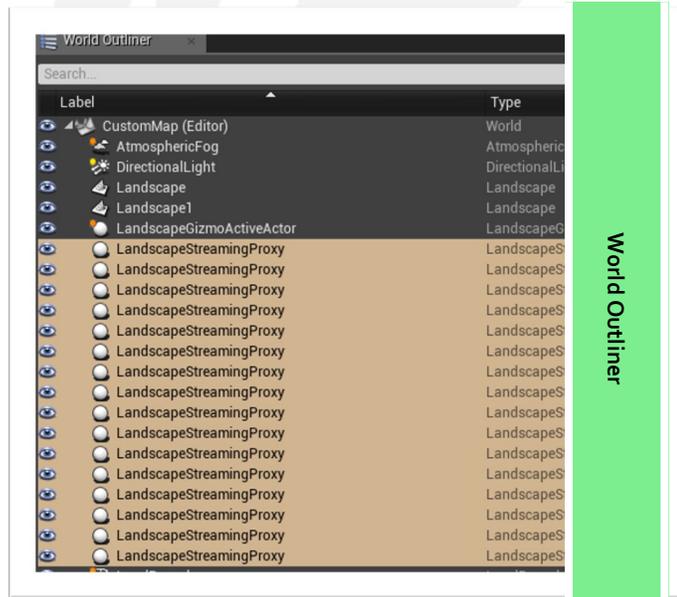
# S STEP FIVE: MATERIALS

## How To Fix A Black Landscape

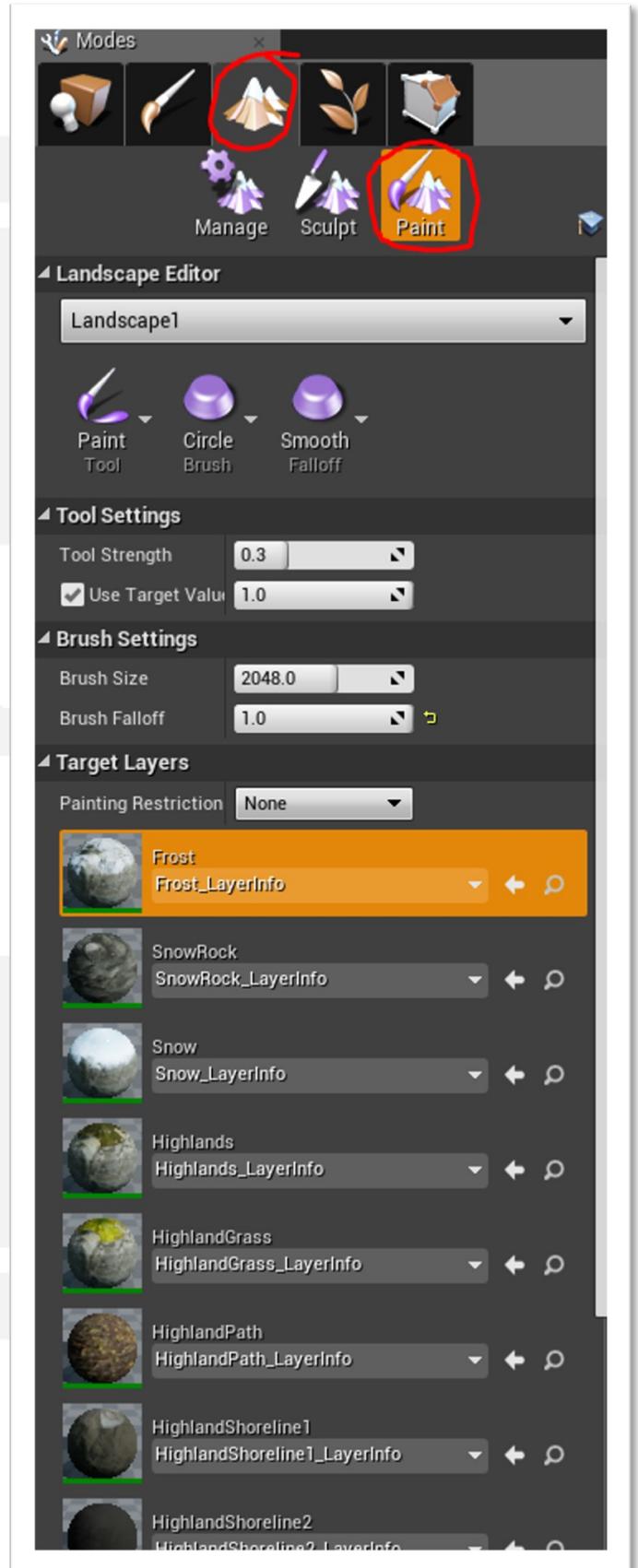
Once your map is loaded it will appear all black. It is important to note this only occurs when you choose a material during the import process.



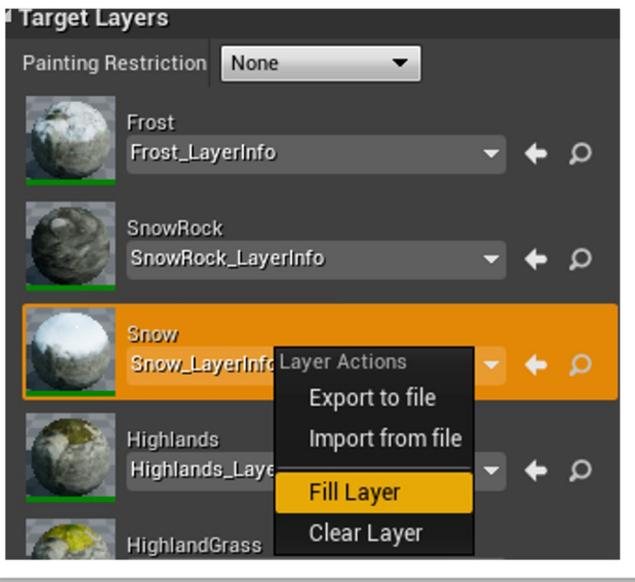
- ◆ In your world Outliner, select all the *LandscapeStreamingProxy*'s and make sure a *Material Instance* is in the *Details* panel.



- ◆ Go to the Modes Panel select "Landscape" then the "Paint" tab.



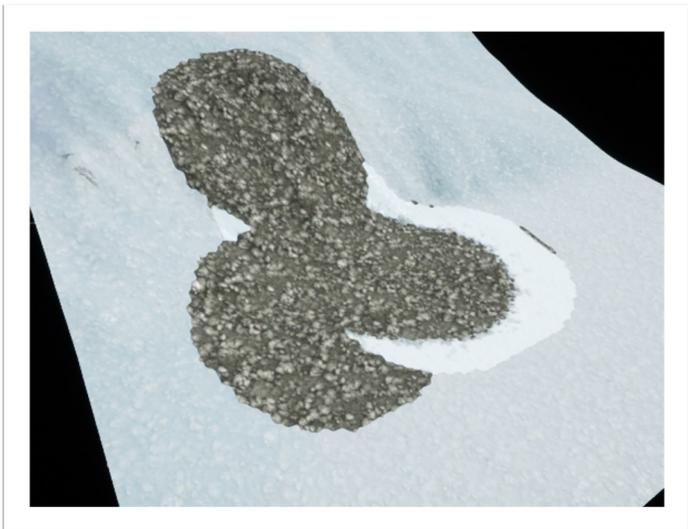
Once you have your landscape paint pallet open you will have two options to get started and get rid of the black.



◆ **FIRST OPTION:**  
Right click on the primary material you wish to use and select "Fill Layer". This will fill the entire map with chosen material.



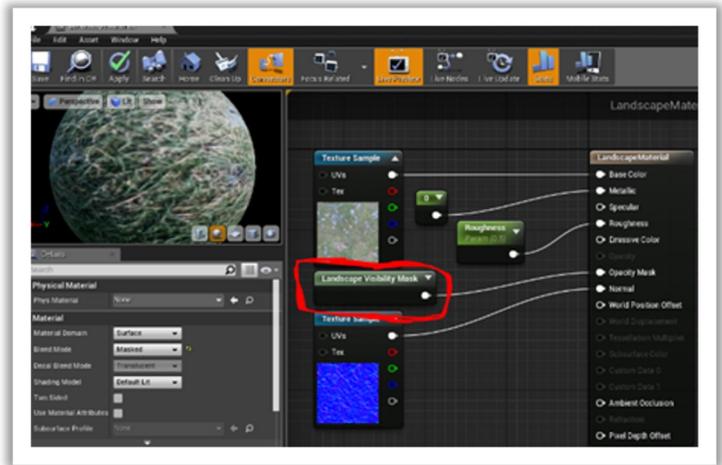
◆ **SECOND OPTION:**  
Highlight your material of choice and simply begin painting.



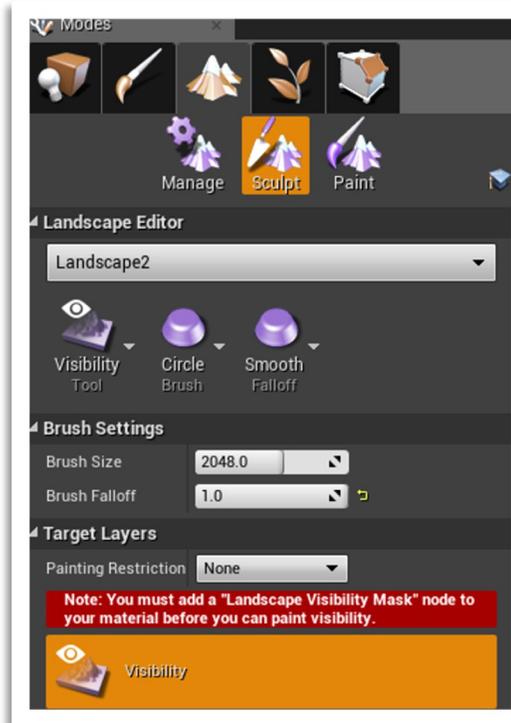
### (Special) About Materials & Caves

This step requires a bit of preplanning if your making your own materials. If you want to be able to have caves on/in your map you will need to have a "Landscape Visibility Mask" node attached to your material. It is highly advised that you have this material prior to tile import. This will allow you to make holes in the mesh for your cave entrances.

- ◆ Be sure to have materials that contain a "Landscape Visibility Mask". This will be done at the same time you import your Tiles. If no material is added at this point in the process you WILL NOT be able to make caves.



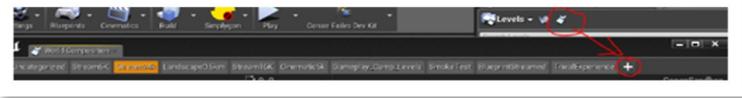
If you do not have a "Landscape Visibility Mask" you will receive this Warning.



## Streaming Distance

For your map to load smoothly you will want to create the “Streaming Distance”. To do this you will need to:

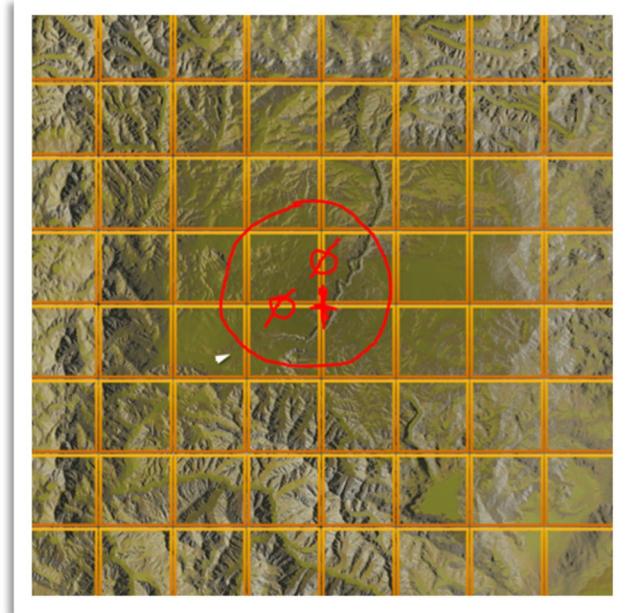
- ◆ Open *World Composition*.
- ◆ Click the “+” and name your tab “Landscape”.



- ◆ Set the streaming distance to about 50,000. ConanSandbox uses these settings:

- Landscape: 50,000
- Gameplay levels: 10,000
- Caves: 25,000

- ◆ Select all your map, right click and “Load” all the tiles.
- ◆ “Right Click” and “Assign Layer”. This will place your map tiles into the new tab.



### [HELPFUL HINT]

*To help with a more rapid map load, create a geographical obstruction that will break up long distance views. Use mountain ranges, walls etc. Take your time checking each view distance from different locations on the map. Consider key locations to obstruct long distance views such as the edge of the map and center of the map.*

### Set your map to ZERO, ZERO!

To center your map to Y0\_X0 open “World Composition”. Select all your map and simply drag the map to the center of “Y0\_X0 while in the “World Composition” panel.



## Levels & Sub-Levels

Once you import your Heightmap you may notice your levels will not appear. You will have to restart Devkit in order for the generated folder containing them to show.

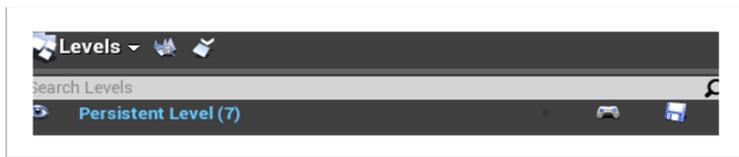
Sub levels serve multiple purposes. If you have a team working on a particular part they can work in a specific level. Also adding sub-levels to levels is good for loading.

## Quick And Dirty Heightmap Creation (WARNING — THIS INFO NEEDS TO BE TESTED FURTHER!)

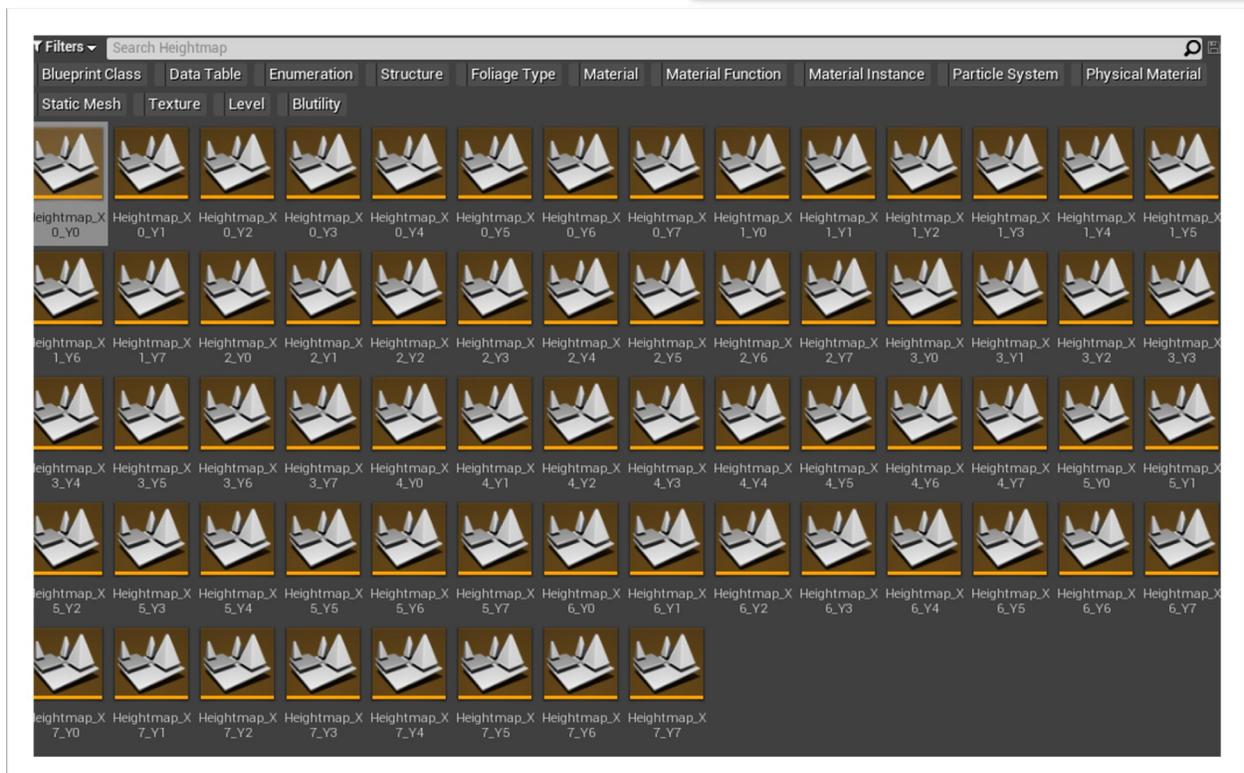
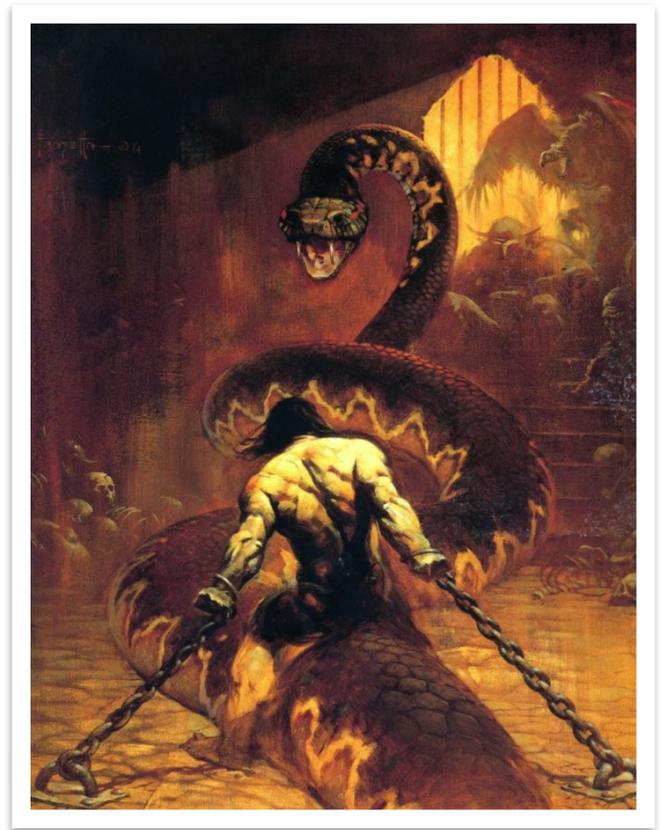
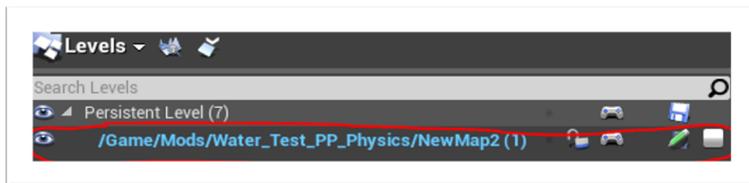
So - you don't want to go through the hassle of customizing and importing heightmaps and tiles. If you decide you want to create them in the Unreal Engine make your Main Game Map and enable "World Composition".

Click on the "Levels" drop down and "Create New". This will create a new level.

Now Create a landscape from the modes panel. Make sure your map is the size and resolution you want. Once your landscape is created click "SAVE". This will generate a new "Sublevel in your content folder.



In your "Levels Tab" make sure "Persistent level" is highlighted and make it "Current".



## STEP SIX: Sculpting

After your map is loaded you may want to add additional geographical features and details. You can also fix some areas where the map edges are not matching up. You can do this by simply smoothing them out using the smoothing tool. Be sure to take some time fixing any artifacts in the meshes.

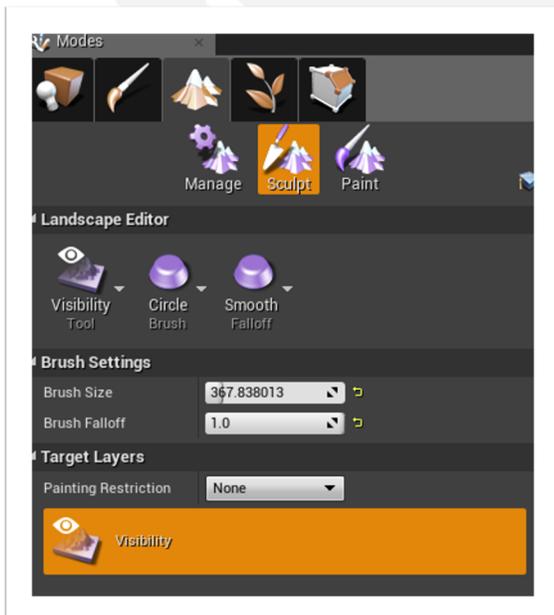
### Sculpt Tools

As part of your cleanup use the sculpt tools in the following order.

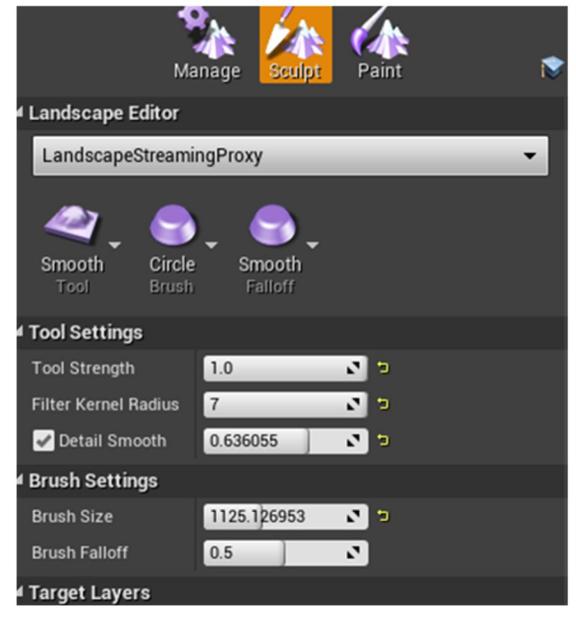
*Thermal Erosion* – Use this first and paint over your map. The settings are as follows



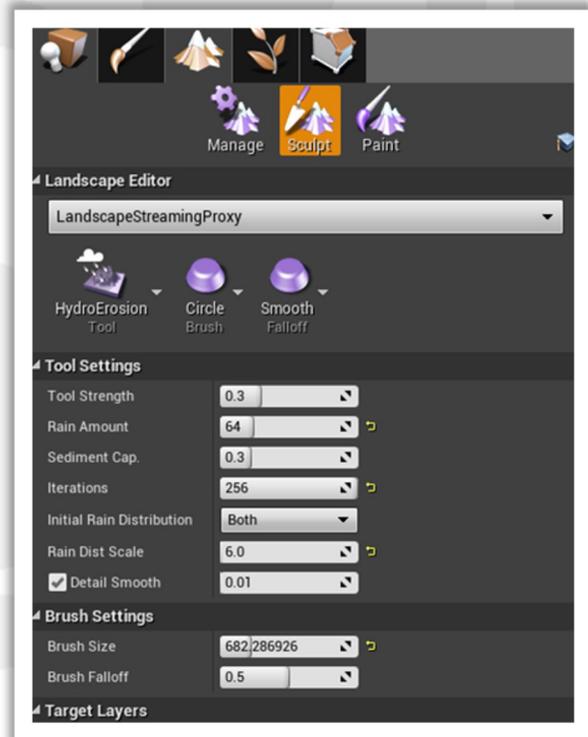
*Visibility Tool*— Use this for caves. This tool will make a hole in your mesh. Use SHIFT + Left click to undo the hole.



*Smooth Tool* – Use the smooth tool to fix any imperfections the Erosion tools give

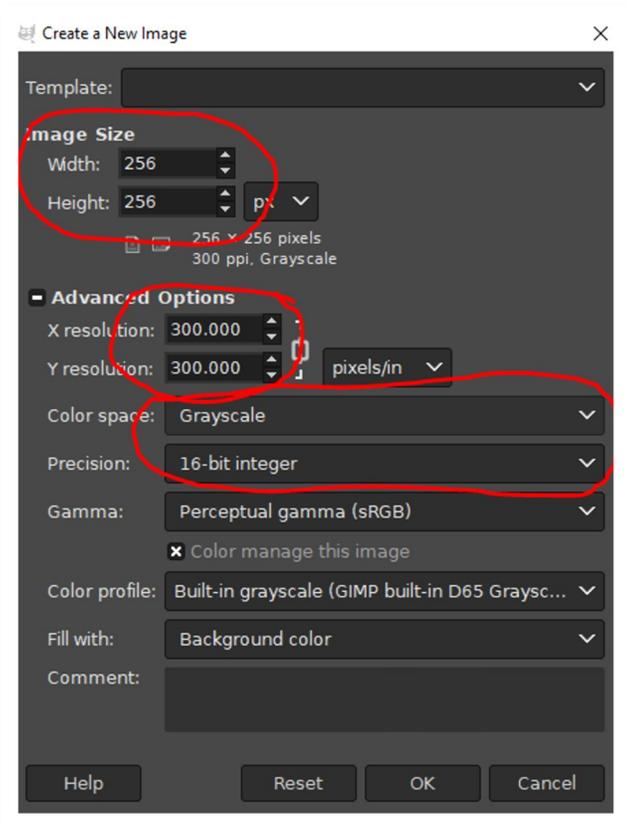


*Hydro Erosion* – After applying Thermal Erosion, apply Hydro Erosion. The settings are as follows. This will help increase the creases between mountain ridge.

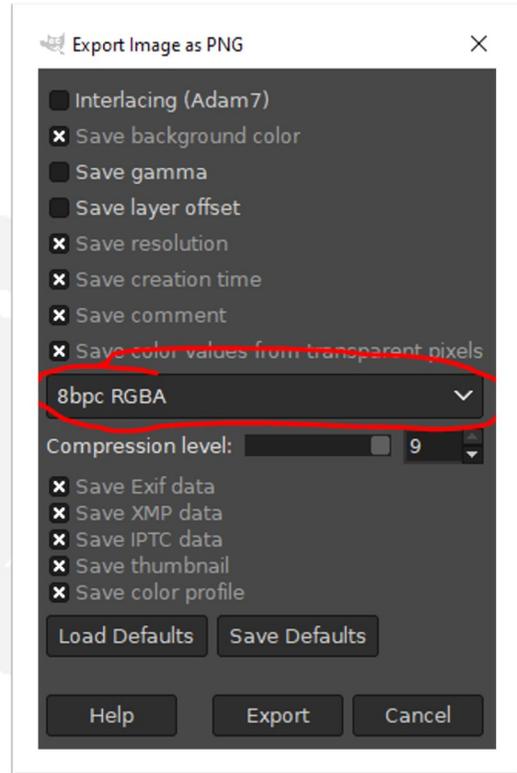


## Creating Custom Sculpting Brushes

To make a custom sculpting brush for your alpha brush open gimp and set your image with the following settings.

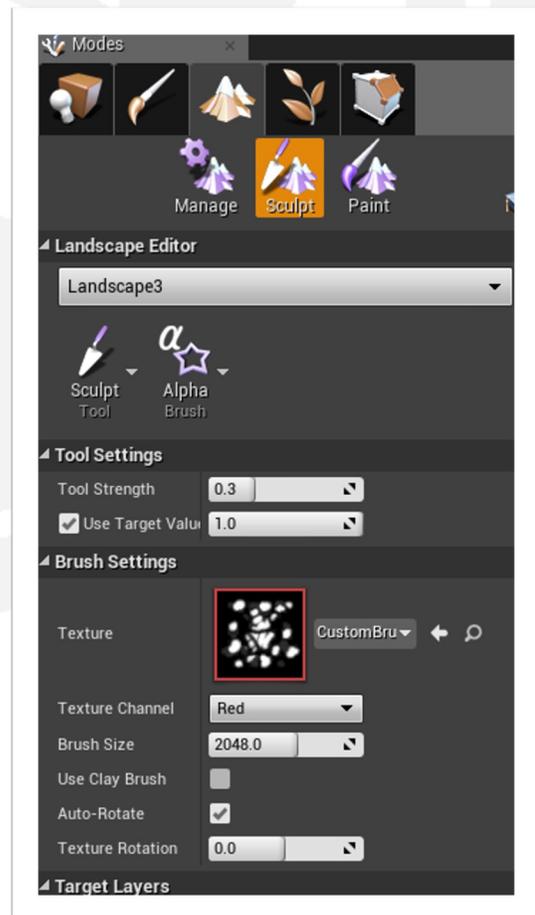
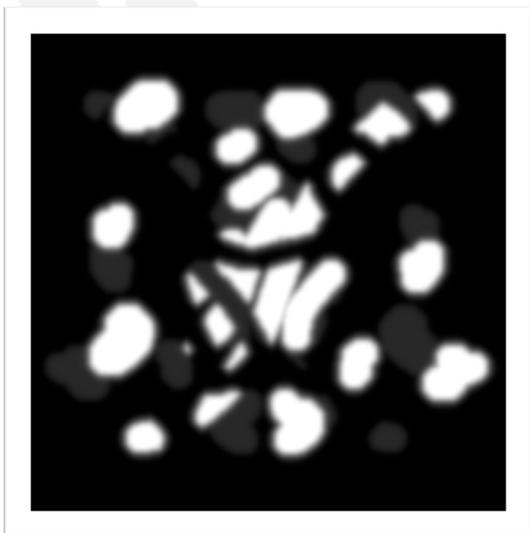


Once you're finished with your custom brush be sure to "Export" as a ".png" with the following settings.



When you're done, Save the brush as a .png. Then import your brush into UE4.

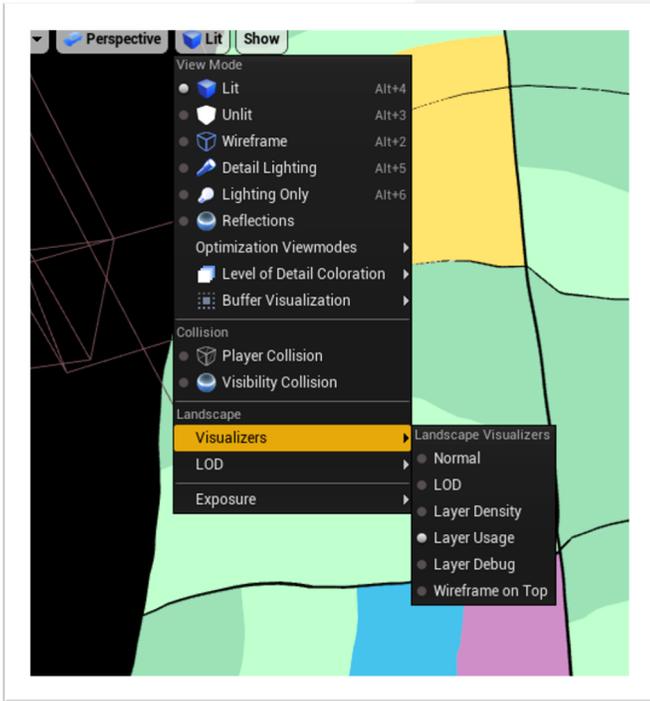
Create your brush using black and white as you would a heightmap.



## Material Landscape Optimization

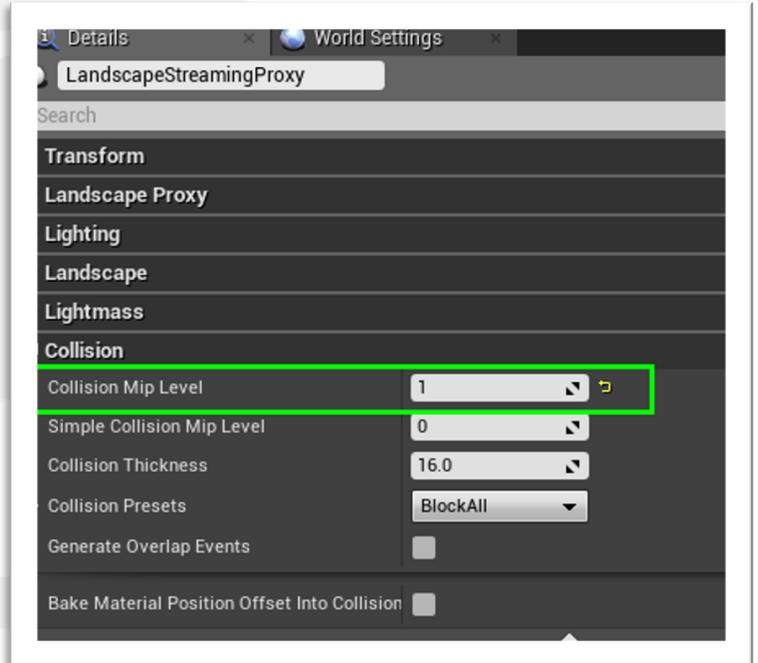
On Exiles & Siptah we use no more than 4 landscape layers per chunk due to a hard lock on PS4 and general performance cost.

- ◆ To see the layers used, make sure you are in Landscape Mode → Paint and find the Landscape visualizers.
- ◆ Use LOD 1 Collision on MIP to save memory/Performance.



The collision will be less accurate since it has less polygons than the visual mesh, in most cases this is not an issue, but there are things to be aware of.

- ◆ The closer the heightmap is to 90 degrees the bigger chance of the player clipping through the landscape.
- ◆ If you cut a hole in the landscape, it will be less accurate around the edges.
- ◆ To help visualize how the LOD1 on the landscape looks like. Use the console command `"r.ForceLOD 1"`



## What is LOD

LOD or "Level of Detail" will be effected when you paint on your mesh with layers of materials. The more materials the more performance is required.

*In computer graphics, **level of detail (LOD)** refers to the complexity of a 3D model representation. LOD can be decreased as the model moves away from the viewer or according to other metrics such as object importance, viewpoint-relative speed or position. LOD techniques increase the efficiency of rendering by decreasing the workload on graphics pipeline stages, usually vertex transformations. The reduced visual quality of the model is often unnoticed because of the small effect on object appearance when distant or moving fast.*

*Although most of the time LOD is applied to geometry detail only, the basic concept can be generalized. Recently, LOD techniques also included shader management to keep control of pixel complexity. A form of level of detail management has been applied to texture maps for years, under the name of mipmapping, also providing higher rendering quality.*

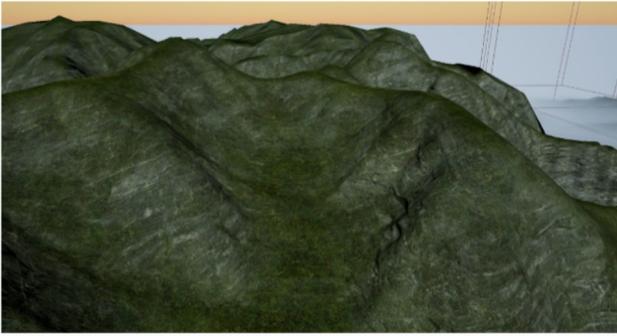
*It is commonplace to say that "an object has been LOD'd" when the object is simplified by the underlying LOD-ing algorithm.*

-Wikipedia

## Cliffing

Cliffing is the art of giving additional play area and blocking out areas of the heightmap to make them more interesting. By simply adding cliff models (e.g. Static Meshes) The mountains will obtain additional interesting shapes, but you'll need to do a lot of what we call "cliffing" to give a feel of depth.

*Original Mountain*



One of the reasons why cliffing is going to be important is gameplay space - if you don't have some flats, mountains eat up a ton of space and a lot of the space becomes unplayable (as in - can't walk and for that reason not have combat) like this:



*Same mountain but "Cliffed" up.*



*Another Example of "Cliffing".*



### [HELPFUL HINTS]

*Try not to scale the cliffs beyond twice their original size. This may cause issues with the material resolution.*

#### **How to fix objects set in the Persistent Level**

If you accidentally place objects in the persistent level simply select them and press CTRL + X. This will make all the selected items disappear ("Cut").

Now select the level and "MAKE CURRENT". Once it is highlighted blue, use CTRL + V ("Paste"). All items selected will reappear and render on the correct level.

This technique can be used for any objects accidentally placed on the wrong level.

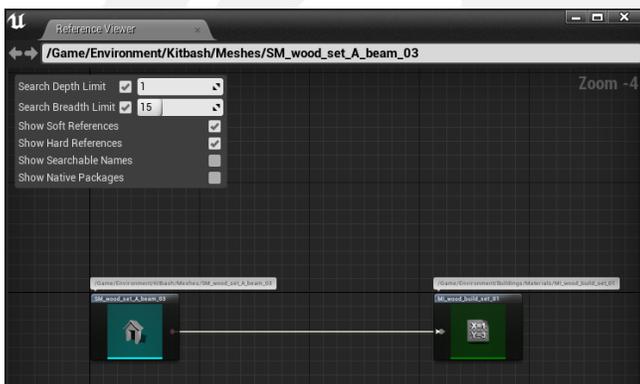


## A few notes on assets

Conan Exiles do not use all the assets that exists in the Devkit. The ones not baked by the Granite process can not be used when making a mod, without copying them and putting them in as custom assets in your mod. The reason for this is that Granite doesn't clobber the textures, which in turn results in the assets, when placed, will not show up in the game when it's run. They DO, however, show up when playing in the editor.

This can be a tricky problem to overcome, since it's not possible to know at a glance if the asset is used or not. But there is a way of finding out if the asset you want to use can be used or not.

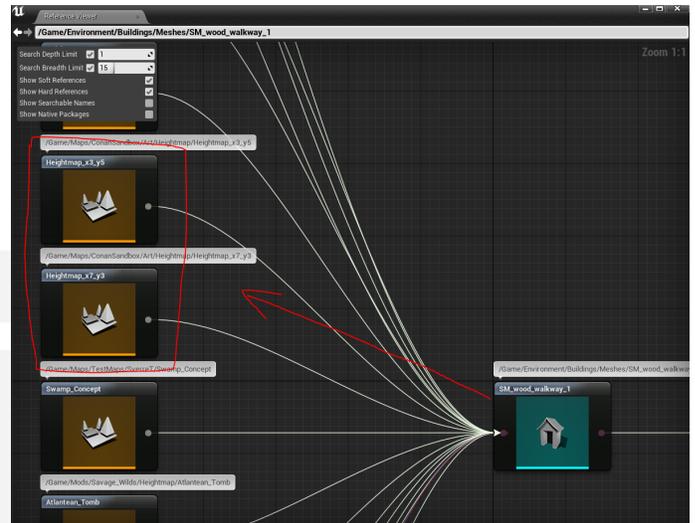
If you right-click an asset and select "Reference Viewer" from the menu, you will see a window appear, such as the one below.



What you're looking for is if the asset is used in the ConanSandbox map or not. If it's not, there's a high risk of the asset not showing up in the game.

In the image above, you can see that the asset is not linked to the ConanSandbox map at all, and so shouldn't be used unless you know that it's used in Siptah.

Another example below:



In this case we can see that the asset is used in multiple ConanSandbox maps, and for that reason, it will show up just fine. It would also be fine if the asset was used by a blueprint that in turn is used as a placeable in the item table or on the map.

This goes for materials, skeletal meshes, static meshes, decals and just about any other assets in the game.

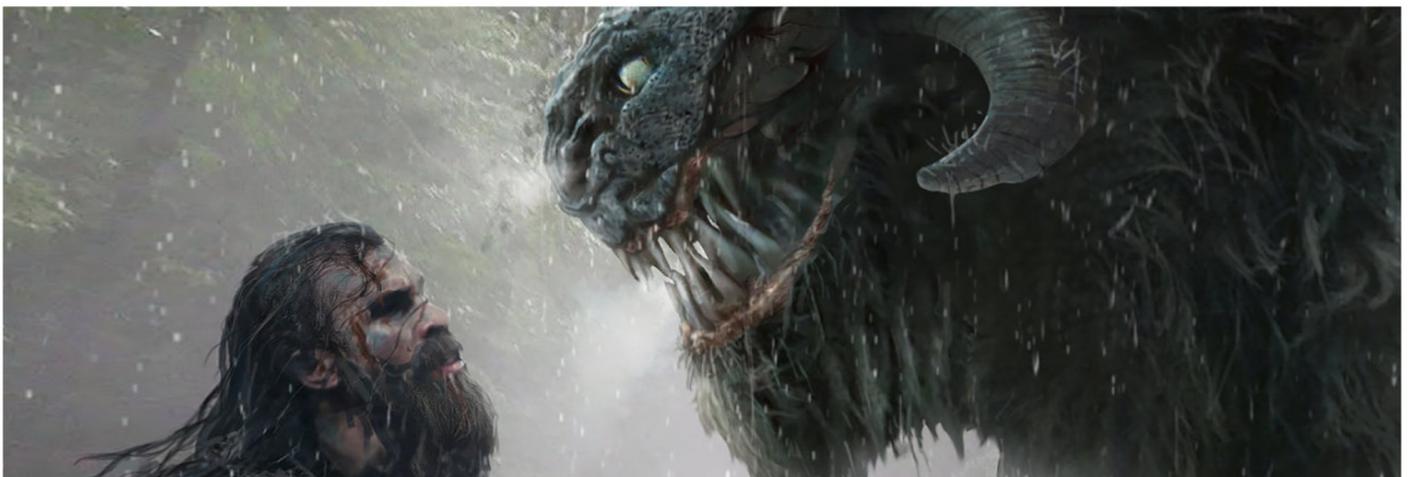
Hunting down instances of this after you've made your mod can be a real hassle, so making sure you're using correct assets to begin with will save you a lot of time later on, as it's not easy to replace every single rock of a specific type across an entire map with one that works.

## Size Comparison

Use the following model to compare your landscape sizes.

Human Size Reference: Use *SM Sizeref Body 01 static*

*mesh* found in filters by typing human.



## How to export your UE4 Height Map

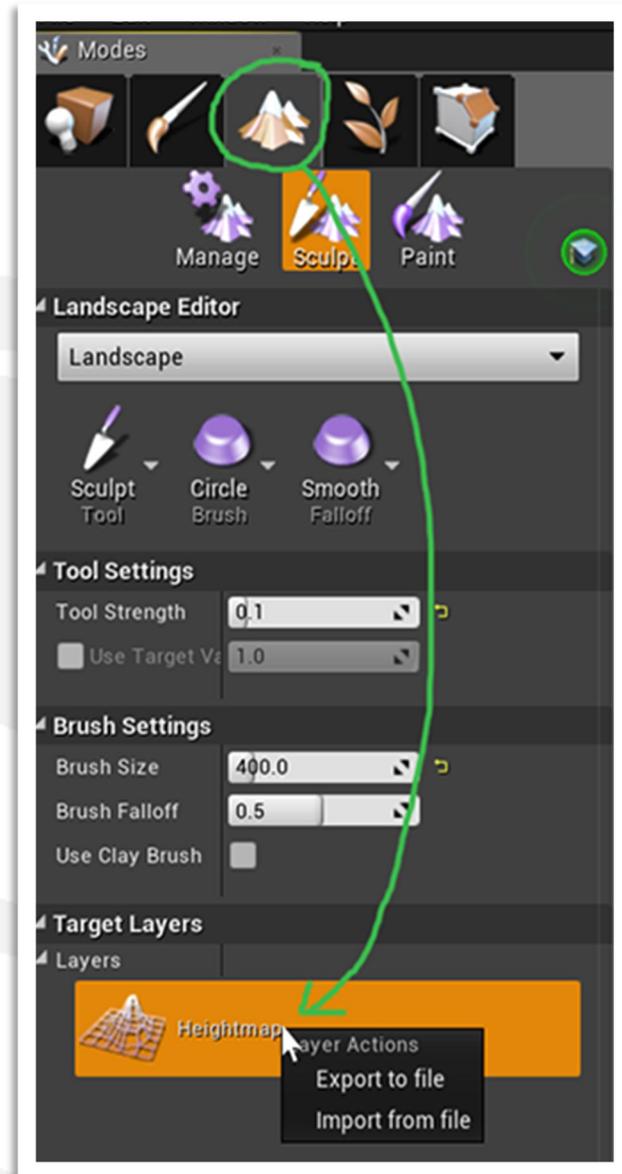
After using your sculpting brushes to refine some of the areas export your heightmap into Terresculpter. This is how you get a Grayscale 16 Bit heightmap. It may also be used to fix custom brush details that appear grainy (because they show as 8 bit not 16 bit).

To export the landscapes heightmap you need to do the following steps:

1. Select the landscape and go to the Landscape Tools on the top left mode/tools panel (this changed in 4.25 to being a mode pull down box above the viewport)
2. In the landscape panel locate where it says "Heightmap" and right click, then "Export to file".

To paint a 16 bit texture onto it you need to do the following:

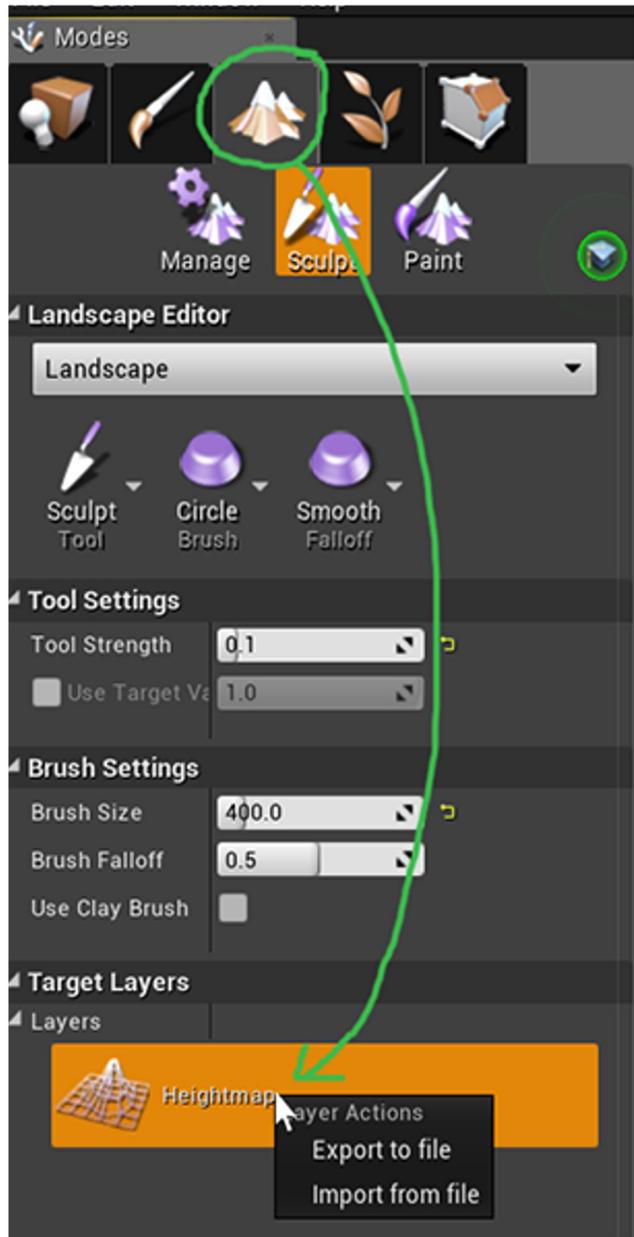
1. Open the heightmap in GIMP
2. Open a 16 bit texture and copy/paste it onto the heightmap.
3. Set the Blending Mode on the Layers Panel to "Screen" or whichever blending mode gives you the best result. Just make sure you aren't changing the textures blending colors or warping the 0 to 1 range of the texture and/or heightmap as it just will not look right. Screen worked for me.
4. Flatten and save the heightmap (possibly save the original heightmap that was exported to preserve the original details as you may need to go back if mistakes are made)
5. In Unreal Engine, right click the same Heightmap section and either choose "Reimport" or "Import from file".



## How to Quick Fix Tears in Your Mesh

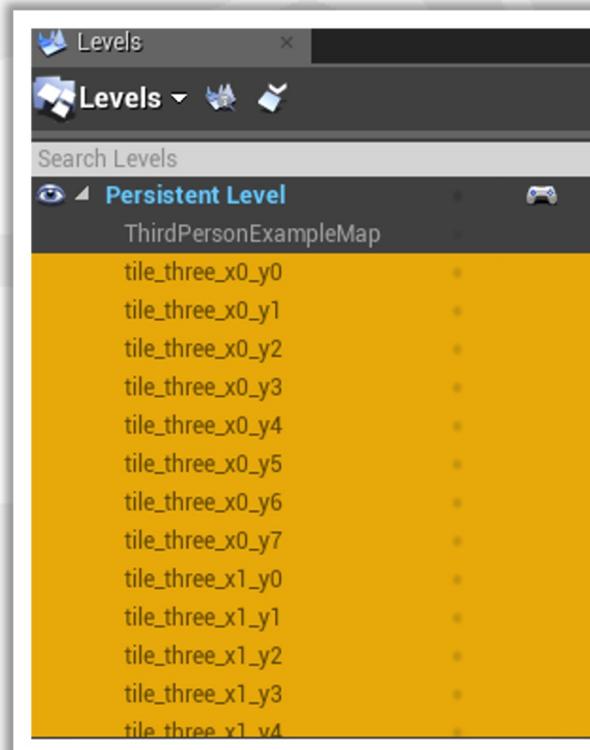
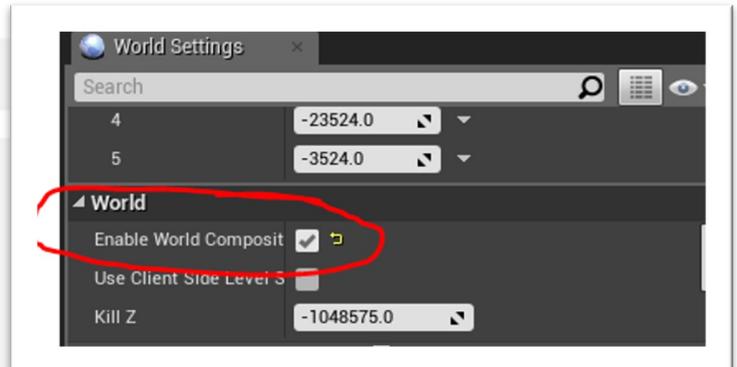
You may find when you import your heightmap not all the edges will align. One way to fix this issue is to use the smoothing tool and lightly "rub" the bad areas out.

The quickest and second way is to export your heightmap, save it on your desktop and reimport it.



## How to delete levels under the persistent level?

You can disable "Enable World Composition" in the World settings. This will remove all levels under Persistent Level. Then you may go on your disk drive and delete all height maps under the Content folder.



SHIFT + LEFT CLICK TO SELECT ALL

## A note about levels

To clarify what we talk about when we talk about levels, here's a helpful image.

### Persistent level

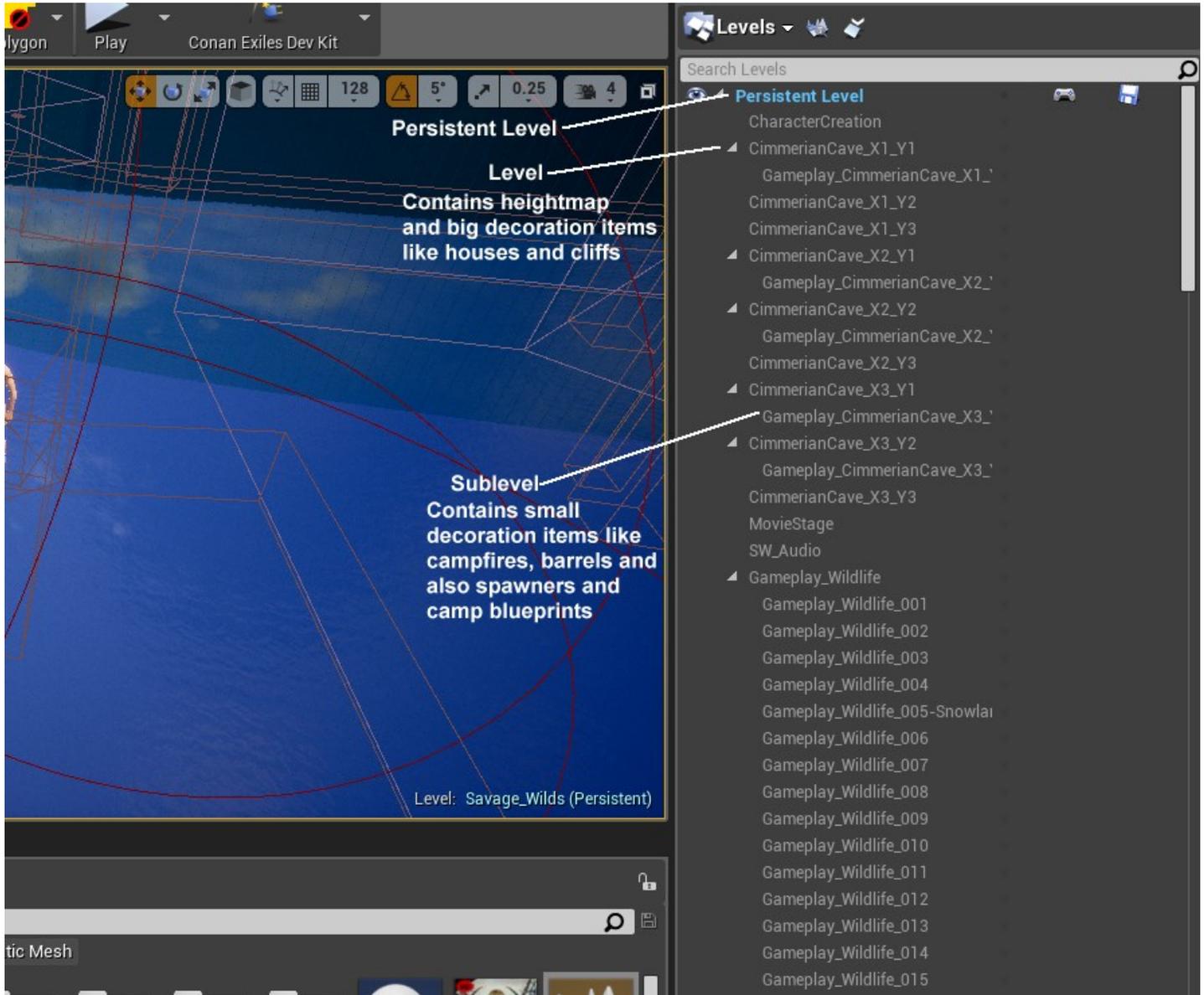
This should contain things that need to be loaded all the time for the map to work, such as weather systems, post processing volumes linked to by UltraDynamic Sky, and things of this nature.

### Levels, a.k.a "Main levels"

These contain heightmaps, large decoration objects, cliffs and the like—things that players need to see at a distance.

### Sublevels, a.k.a "Gameplay levels"

These contain camp actors and decorations for specific camps.

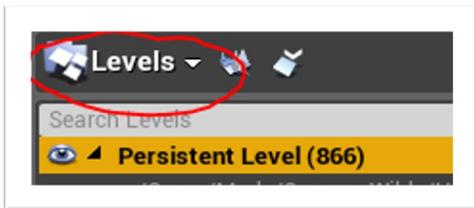


## Importing Sub-Levels and Dungeons

To help with loading put your dungeons on sub-levels. This will help with server performance and address other performance issues because it's possible to set up streaming distances individually.

### STEP 1

Go to Levels and "Make New" level.



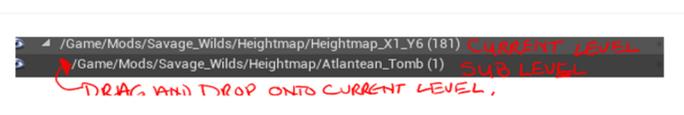
### STEP 2

Load the level that you will be attaching the sub-level to.

### STEP 3

Load the new sub-level you just created.

### STEP 4

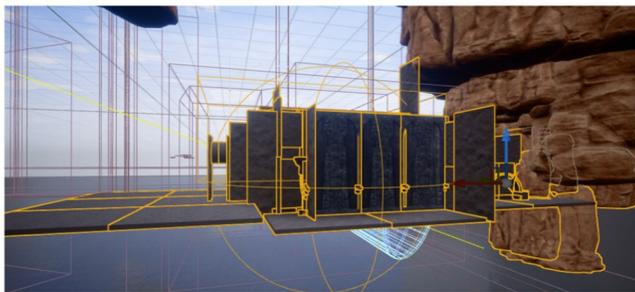


Drag the newly created sub-level onto the current level.

**So you now need to move your dungeon to the sub-level!**

If you created your dungeon on the original map you can simply transfer it to your newly created sublevel.

### STEP 1



Select all your dungeon pieces in your "World Outliner"

### STEP 2

Use CTRL-X. Be sure to exclude your "Post Processing Volumes" in the persistent level.

NOTE: When using CTRL-X, you will notice your items disappear. DON'T PANIC!

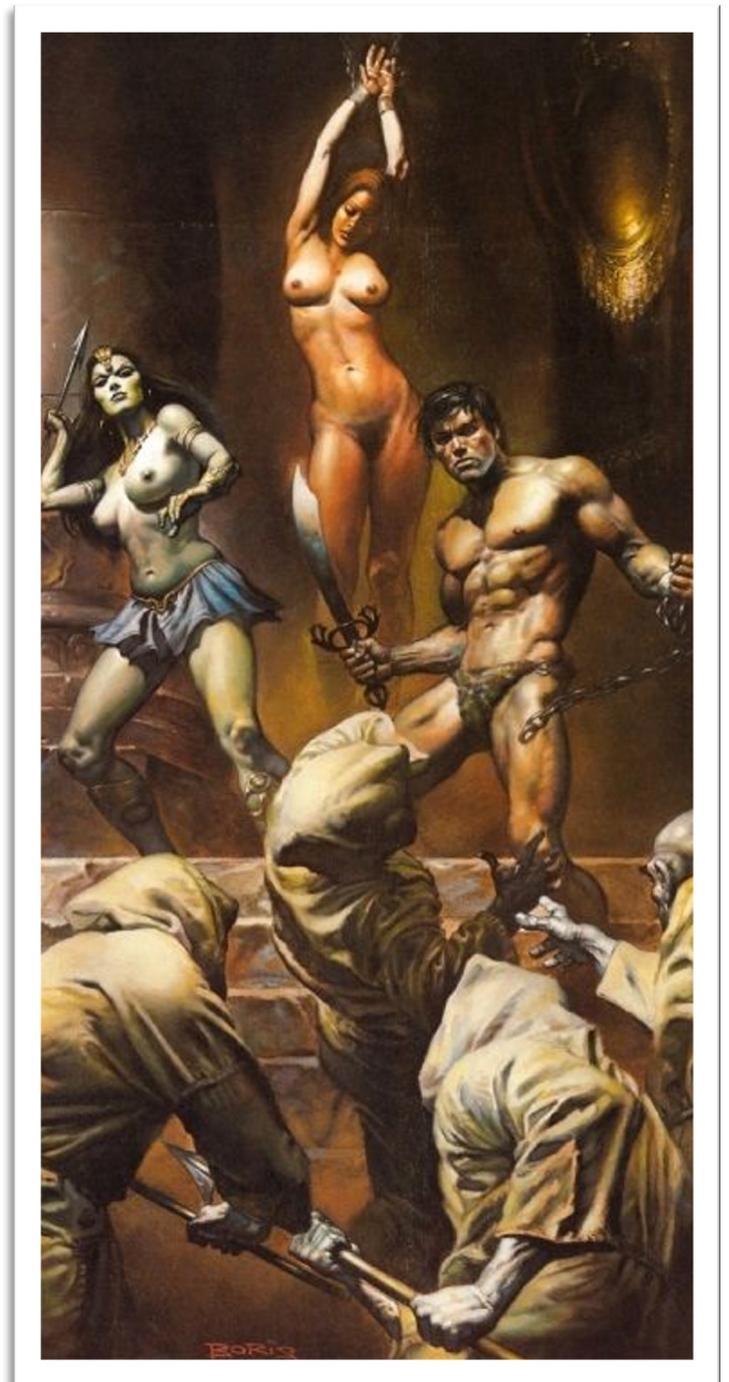
### STEP 3



Make sure your New Sub-Level is "Made Current".

### STEP 4

Use CTRL-V to paste in your dungeon to your new sub-level.

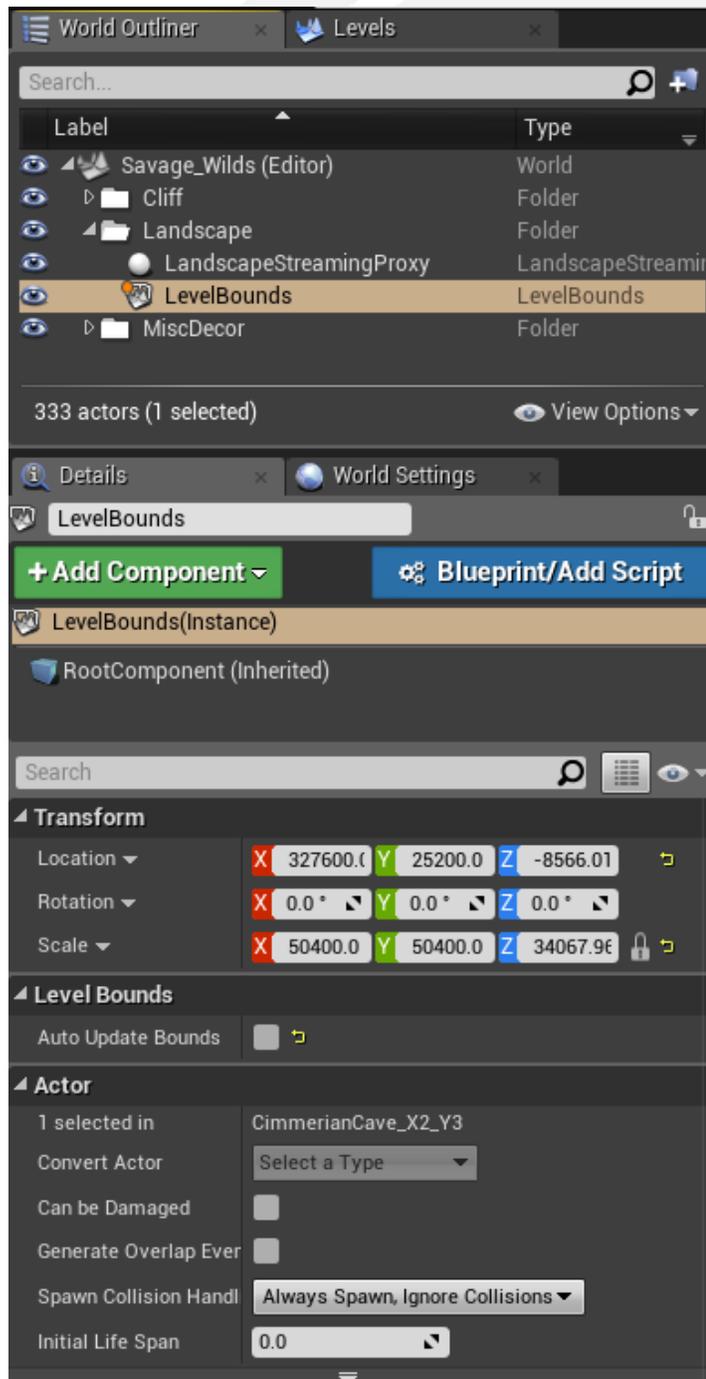


## What are Level bounds and why are they important?

Level bounds determine, together with Streaming Distance, the range where your level is loaded in comparison to player positions.

If your level bounds are very small, the level might not load correctly even if players get near them and if it's too big, the level is loaded even when it's far away.

Very big level-bounds have a severe effect on server and client-performance, and so should be as correct as possible.



## Some helpful tips for dealing with level bounds

Let's say that we have a heightmap that is 50400x50400 in size. If level bounds are set to AutoAdjust (which they are by default) and you put an actor or cliff outside these bounds, the level bound will take this into consideration and automatically expand to contain this actor. This has a number of adverse effects—not only does it start loading from a further away distance than what it needs to, it also leads to a very messy World Composition layout and is an organizational nightmare.

I recommend that you set the levelbounds for ANY AND ALL LEVELS THAT CONTAIN HEIGHTMAPS to not AutoAdjust the second you make them. This will save you a large amount of hassle later on.

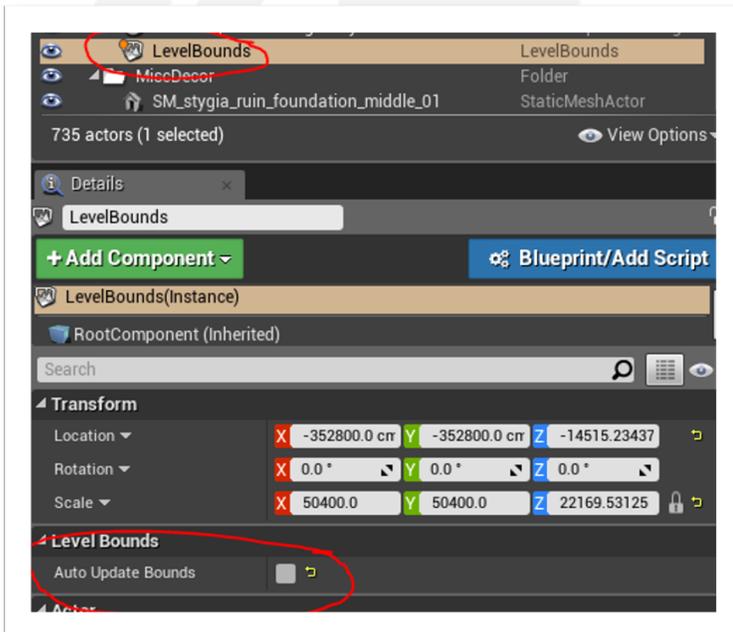
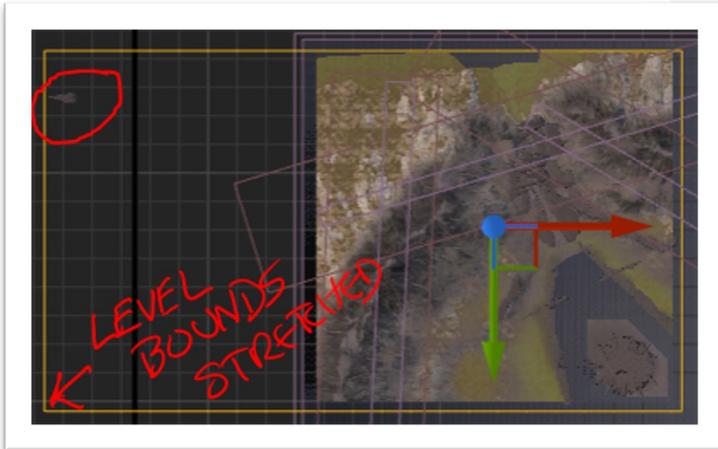
For Sub-levels, you may want to start off by using the AutoAdjust—but be aware that some blueprints, particles, lights and other custom actors for some reason make Unreal hiccup and extend the level bounds in very strange ways.

If this happens, do this:

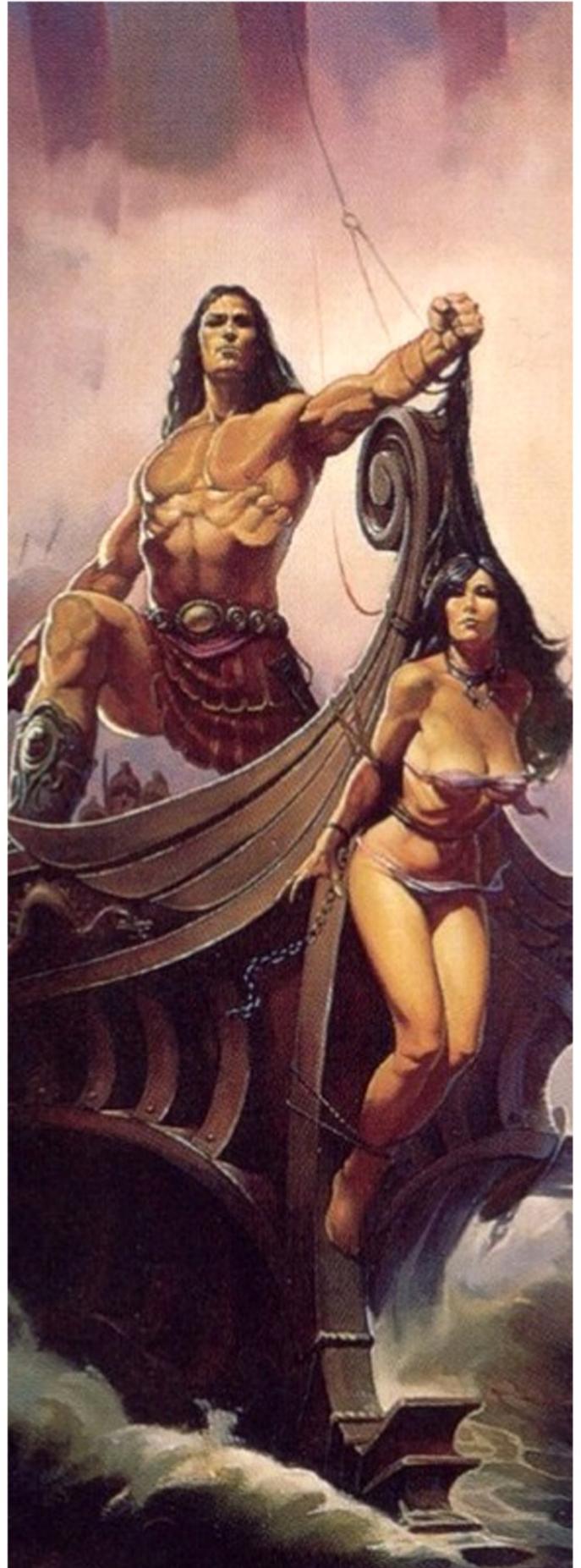
1. Make sure that AutoAdjust is ON
2. Mark all non-static meshes in your level, CTRL-X to cut them
3. Turn AutoAdjust OFF
4. CTRL-V to paste in your misbehaving assets

## Correcting Level Bounds

You may find that your level bounds are stretched. You can fix this by simply loading the map > select the “Level Bounds” and uncheck “Auto Update Bounds”. If this does not work look for lights, blueprints or some other actor that may be causing the issue, cut them out, adjust the bounds and then paste them back in.



Be sure to uncheck each heightmaps “Auto Update Bounds”. This feature does not work well in UE4 4.15 and seems to work best if unchecked and saved for each level.



## CITIES , CAMPS AND RUINS

If you want to add cities and ruins you will need to keep track of your actor count. A recommendation is to keep your actor count below 1500 per square area. If you go over a 1500 actor count try to keep the adjoining square tiles with a lower count. This will help with loading.



## What is Considered a Actor?

For the purposes of map building, all objects like barrels, tables , rugs and other decorative objects are actors. The actual buildings themselves are also actors. To help with loading all the city and its deco, put them on a “Sub-Level”

## Tricks of the Trade

When it comes to rendering buildings, You can add large buildings to the level and not the Sub-level. This will serve three purposes.

- 1) Large buildings will load from a distance giving the illusion a large city is off in the distance.
- 2) Allows you to split up interior decorations from long distance loading. You don't want all the interior deco to load from a mile away and potentially causing issues.
- 3) Try to use the same asset as much as possible in different ways . Using multiple assets in one area and trying to detail out a area too much can/ will cause issues with loading.

## NPC Camps

NPC camps are simply smaller locations not as large as cities towns and villages. These locations are much smaller but will also need their own “Sub-Levels”

A guide for making population in camps can be found here:

<https://www.dropbox.com/s/qp0u3bj9ork0v93/AllGuides.zip>



## POI (Points of Interest)

Points of Interests on a map can be thought of as being many things. For us, a Point of Interest can meet any (or several) of the following criteria:

1. It is a camp
2. It is an interesting visually distinct area
3. It is a location for a dungeon or cave
4. It is a location that has special functionality
5. It is a location that has specific resources in it

Points of interests can be seen easily by players by adding the location of them to the Map Marker data table. In the Map Marker data table, you can name these POIs, assign icons and also the discovery distance and any potential journey-step unlock.

Visual POIs do not always need to be marked on the map. If you have created a small area that you want to keep a secret, or if you feel that, while interesting, isn't enough to warrant a discovery entry, feel free to leave it out of the table.

## An example of a good POI



And to the right, a very bad example of a POI



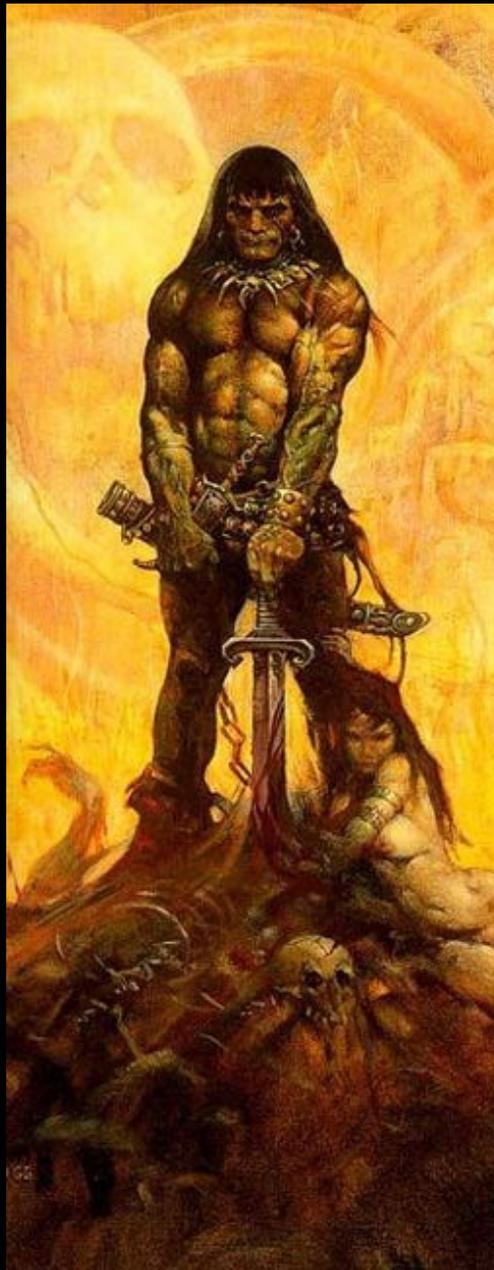
## The purpose of POIs

Points of Interests have many purposes. An ideal POI is a navigational tool as well as something the players see as a unique location. A perfect example of this is the central Tower structure on the Isle of Siptah. It can be seen from a very far distance, and you know exactly where you're headed if you take aim for it.

Another purpose of POIs is to sell the map. If your idea of a POI is "a nice looking rock", players will scratch their heads in confusion. If, however, your rock is thousands of feet tall, with a skull that has a lava waterfall coming out from each of its eyes.. Now.. NOW we're talking a POI people will remember.

# ESSENTIALS

ADDITIONAL MAP PARTS



# S STEP SEVEN: ESSENTIALS

Essentials will cover a wide variety of finishing topics. Once your map is fleshed out you will be able to start rounding it out with the following:

- ◆ SunLight / Moonlight
- ◆ Exponential Height Fog
- ◆ BP\_SplineBuildLimit & Building Blockers
- ◆ Blocking Volumes
- ◆ Climbing Blockers
- ◆ Anti-Undermesh Volume
- ◆ Water
- ◆ NAV Mesh
- ◆ Post Process Volume (Water & Environment)
- ◆ Temperature
- ◆ Time of Day Manager
- ◆ Ultra Dynamic Sky BP
- ◆ Weather
- ◆ Wind direction
- ◆ Sound

## HINT: Copy / Paste Your Settings

If you want to use current settings from the Exiles map simply load up the Conansandbox. Find the object you want to copy (example: Ultra\_Dynamic-Sky). Right click and copy it. Then open your custom map and paste it into your World Outliner.

## Exponential Height fog

This is optional but can give some additional environmental depth to your map. Simply apply it to your map and make the appropriate changes in the details panel.

## Atmospheric Fog (SPECIAL NOTE)

Its important to note that Atmospheric Fog may cause issues with off distance rendering. This may be due to Funcoms custom ultra Dynamic Sky. You will have to test this on a case by case setting.

## Nav Mesh Bounds Volume

Used for NPC AI

## BP\_SplineBuildLimit

If you want to limit the areas players can build: Create a BP\_SplineBuildLimit that covers the playable build area and tag "Allows inside". Anything outside this will be non-buildable unless you create more splines with "Allows inside". Create no build areas within this using additional volumes and splines.

- ◆ For complex shapes use: /Game/Systems/ BP\_SplineBuildLimit (Often referred to as a BuildingBlockerSpline). It can be inverted, allowing players to only build inside and not outside it. Mainly used on the BuildingBlockerSpline covering the entire build area on the map. Building splines have no height restrictions, it will go infinite up/down.

- ◆ Keep overly complex splines to a minimum.

Select and ALT-drag a point will create a new point on a clockwise order in the spline

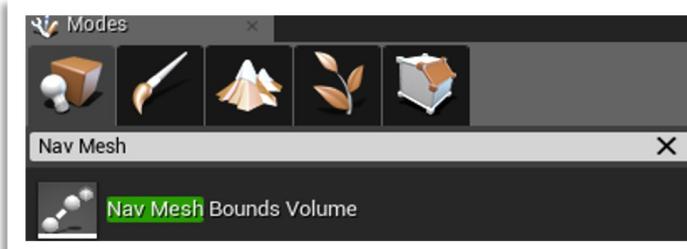
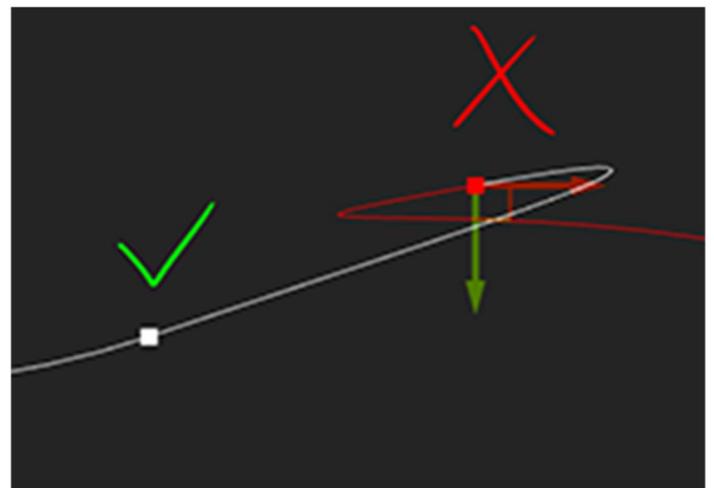
Be wary of twisted rotation of the spline tangents as this will cause random gaps in the functionality.

- ◆ For simple shapes (Box) use: /Game/Systems Building/ BuildingBlockerVolume

Great for caves and anywhere you don't want the limit to extend infinite up/down

- ◆ Don't use scale transform top alter size, change the "Box Extent values instead.

Building Blocker volumes and splines need to be in the "Persistent Level" to ensure they are always loaded and active even if you teleport.



## Ultra Dynamic Sky

Ultra\_Dynamic\_Sky has a ton of options that need to be adjusted. You can manually put in these options or Copy/Paste from the Conan Exiles Sandbox map.

### RECAP: Copy / Paste Settings

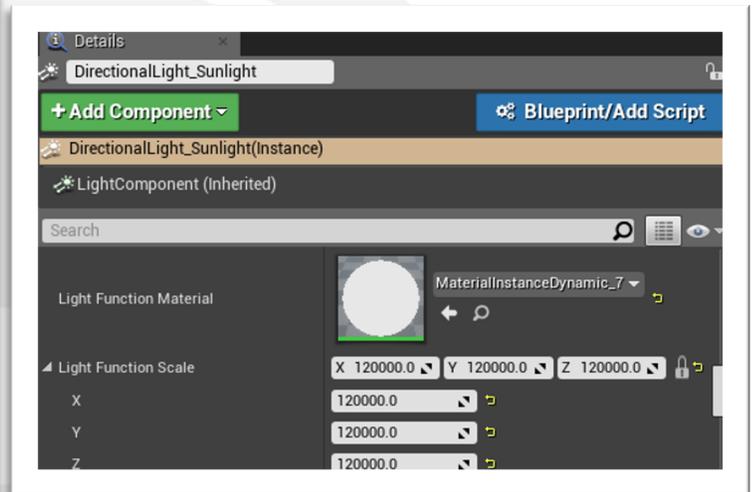
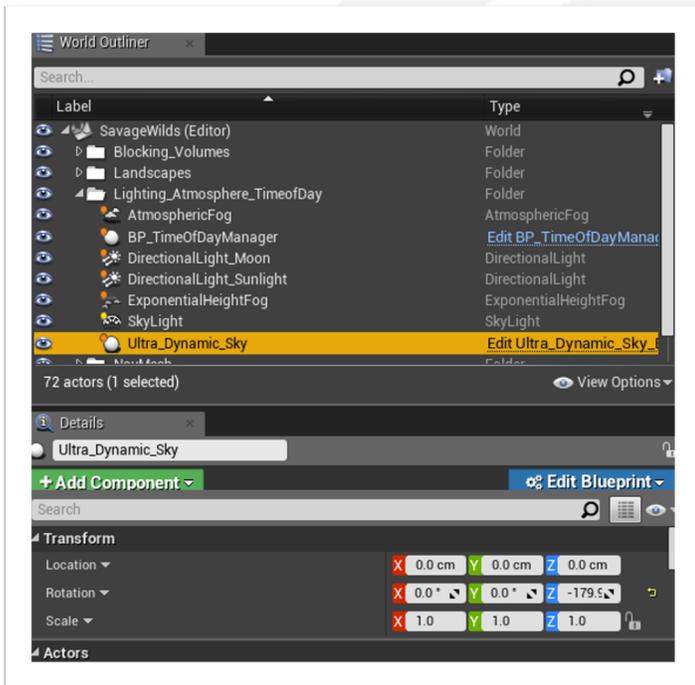
If you want to use current settings from the Exiles map simply load up the Conansandbox. Find the object you want to copy (example: Ultra\_Dynamic-Sky). Right click and copy it. Then open your custom map and paste it into your World Outliner.

## Light Actor (Sun and Moon)

### RECAP: Copy / Paste Settings

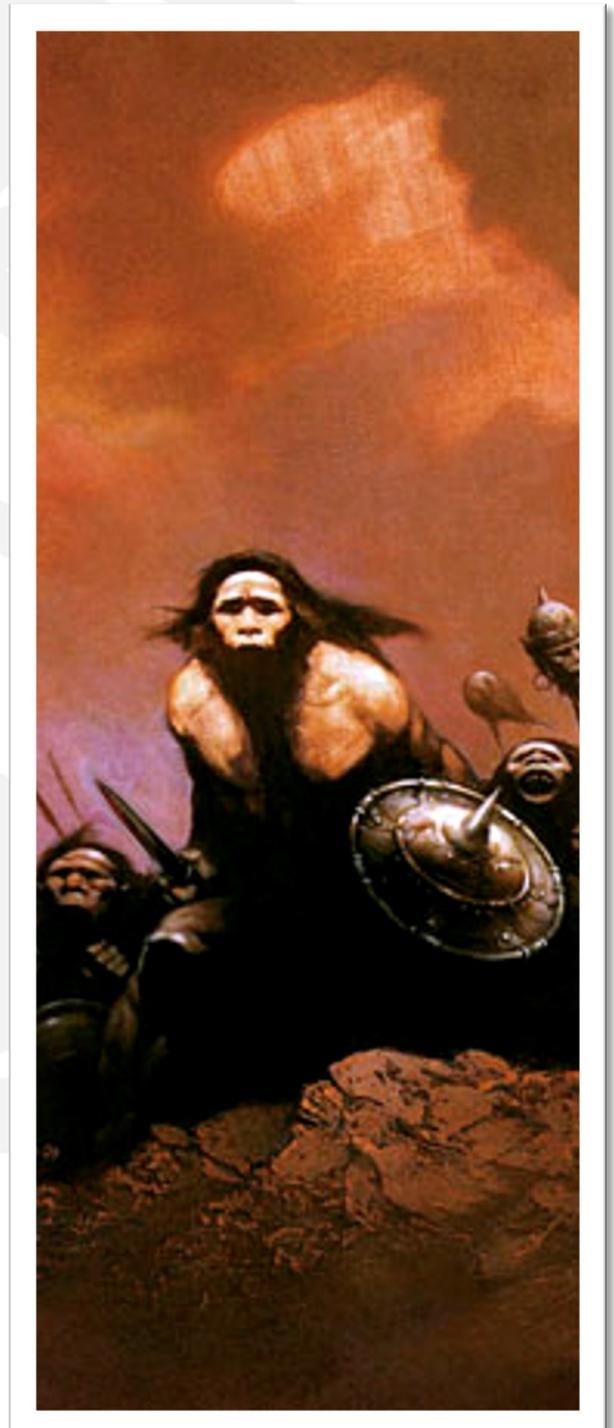
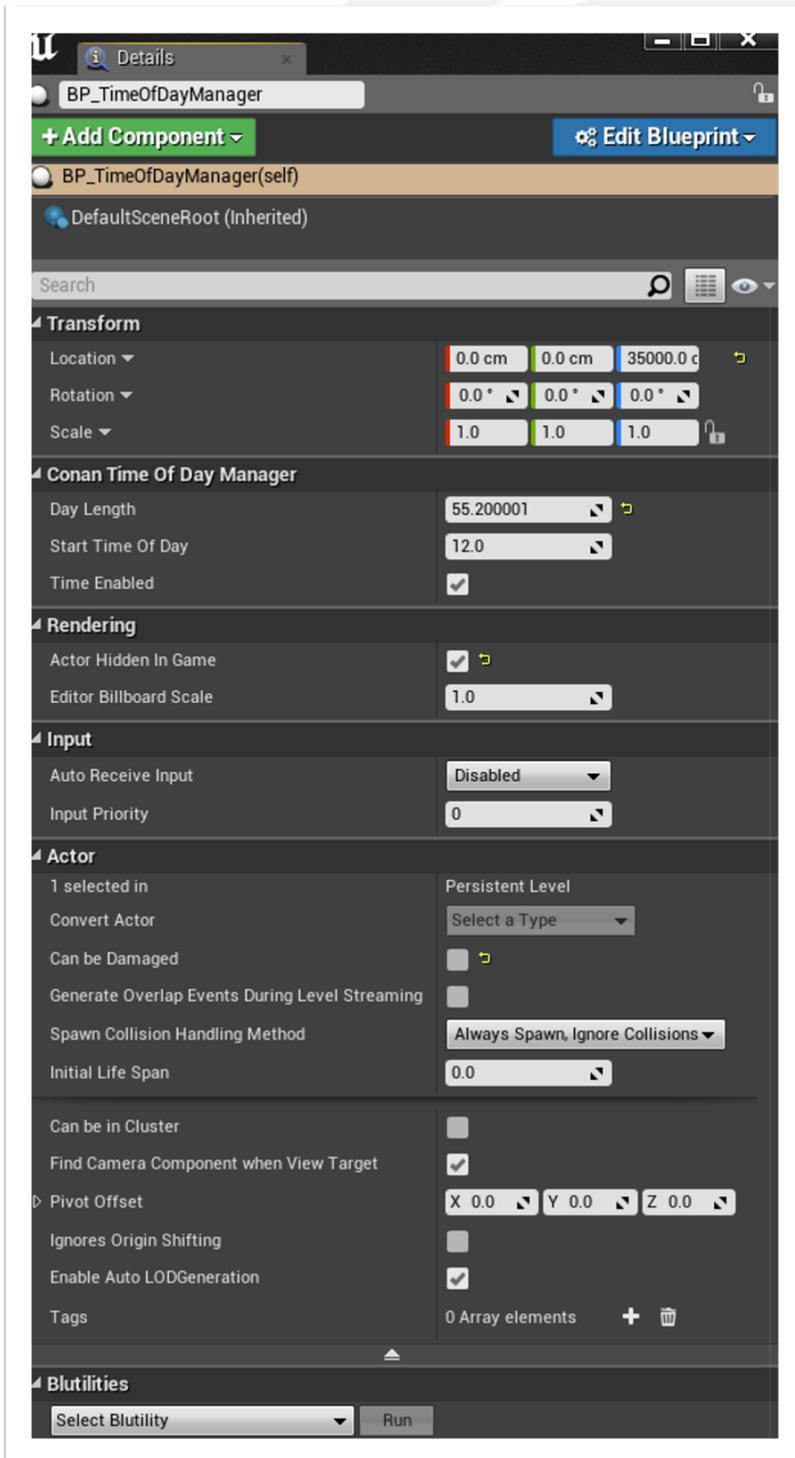
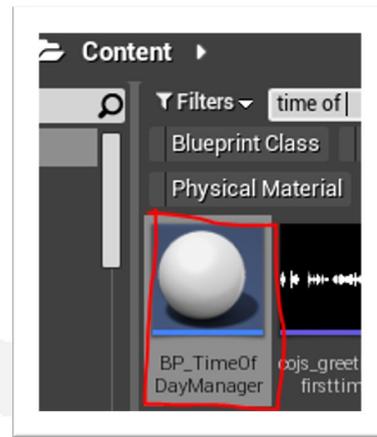
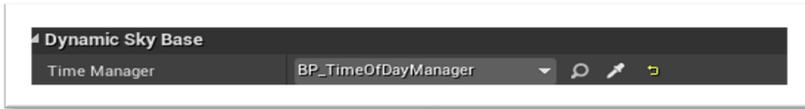
If you want to use current settings from the Exiles map simply load up the Conansandbox. Find the object you want to copy (example: MoonLight & Sunlight). Right click and copy it. Then open your custom map and paste it into your World Outliner.

You may want to copy paste these directional light sources From the Conan Exiles Sandbox due to limited access to the *Light Function Material*.



## TIME OF DAY MANAGER

The "Time of Day Manager controls the day / night cycles. You can either copy paste this from the CE game map or drop the blueprint into your map. Its important that you add the Dynamic Sky Base to "BP\_TimeOfDayManager" in order for it to work properly.



## WORLD SETTINGS REFERENCE SHEET

This is just a quick reference sheet to compare your mods world settings to the Conan Exiles issued settings. Its important to note you may change some of these settings such as the *Spawn Probability Bands*. You will often use the *World Composition*

check box. Especially during the process of building your map. In the initial stages of designing your map try not to touch too much in this section without consulting the guide or testing the change first.



World Settings

Search

**Game Mode**

GameMode Override: None

Selected GameMode: None

Default Pawn Class: None

HUD Class: None

Player Controller Class: None

Game State Class: None

Player State Class: None

Spectator Class: None

**Land Claim**

Building Grid Size: 800

Foliage Grid Size: 800

**Lightmass**

Lightmass Settings

Static Lighting Level Scale: 1.0

Num Indirect Lighting Bounces: 3

Indirect Lighting Quality: 1.0

Indirect Lighting Smoothness: 1.0

Environment Color

R: 255

G: 143

B: 90

A: 0

Environment Intensity: 1.0

Diffuse Boost: 1.0

Use Ambient Occlusion:

Generate Ambient Occlusion Material Mask:

Direct Illumination Occlusion Fraction: 0.5

Indirect Illumination Occlusion Fraction: 1.0

Occlusion Exponent: 1.0

Fully Occluded Samples Fraction: 1.0

Max Occlusion Distance: 200.0

AI

AILODBoundary: 400.0

AILODMax Raycast Distance: 400.0

Building Stability Decrease: 25

Static Navigation: SN\_Convo

Spawn Probability Bands: 3 Array elem

0: 2 members

Radius: 10000.0

Weight: 25

1: 2 members

Radius: 30000.0

Weight: 70

2: 2 members

Radius: -1.0

Weight: 5

Spawn Probability Bands Listen Server: 3 Array elem

0: 2 members

Radius: 10000.0

Weight: 25

1: 2 members

Radius: 20000.0

Weight: 75

2: 2 members

Radius: -1.0

Weight: 0

Conan World Settings

Spawn Desert Bounds: 6 Array elem

0: -118496.0

1: 281504.0

2: 272060.0

3: 372060.0

4: -23524.0

5: -3524.0

World

Enable World Composition:

Use Client Side Level Streaming Volumes:

Kill Z: -1048575.0

Enable World Bounds Checks:

Enable Navigation System:

Enable AISystem:

Enable World Origin Rebasing:

Kill ZDamage Type: [Dropdown]

World Composition Class: BP\_C [Dropdown]

Minimize BSPSections:

Default Color Scale: X 1.0, Y 1.0, Z 1.0

Physics

Override World Gravity:

Global Gravity Z: 0.0

Default Physics Volume Class: DefaultPh [Dropdown]

Physics Collision Handler Class: [Dropdown]

Visualize Material Diffuse:

Visualize Ambient Occlusion:

Volume Light Sample Placement Scale: 1.0

Compress Lightmaps:

Granite Light Map Compression: Unreal Equivalent

Lightmaps: 0 Lightmap(s)

Packed Light and Shadow Map Texture Size: 1024

Force No Precomputed Lighting:

Environment

Foliage Physics Range Client: 1500.0

Foliage Physics Range Server: 800.0

Rendering

Default Max DistanceField Occlusion Distance: 600.0

Global DistanceField View Distance: 20000.0

Dynamic Indirect Shadows Self Shadowing Inten: 0.8

Precomputed Visibility

Precompute Visibility:

Place Cells Only Along Camera Tracks:

Visibility Cell Size: 200

Visibility Aggressiveness: VIS Least Aggress

Audio

Default Reverb Settings

Apply Reverb:

Reverb Effect: None

Volume: 0.5

Fade Time: 2.0

Default Ambient Zone Settings

Exterior Volume: 1.0

Exterior Time: 0.5

Exterior LPF: 20000.0

Exterior LPFTIME: 1.0

Interior Volume: 0.5

Interior Time: 0.5

Interior LPF: 20000.0

Interior LPFTIME: 0.5

Default Base Sound Mix: None

LODSystem

VR

World to Meters: 100.0

Mono Culling Distance: 750.0

Tick

Allow Tick Before Begin Play:

Min Global Time Dilation: 0.0001

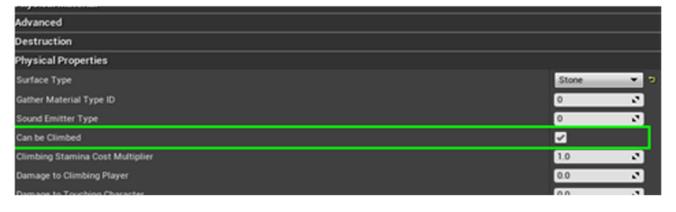
Max Global Time Dilation: 20.0

Min Undilated Frame Time: 0.0005

Max Undilated Frame Time: 0.4

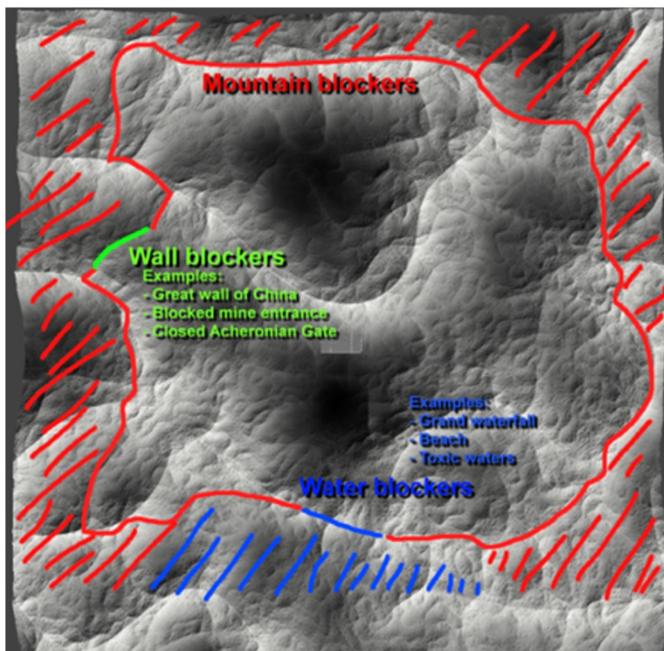
## Climbing Blockers

To stop the player from climbing in an area:  
Place out the blueprint /Game/Systems/Building/  
BP\_ClimbingBlockerVolume and adjust the size. Make sure  
the climbing blocker is in the "Persistent Level" to ensure  
they are always loaded and active even if you teleport.  
By default, all physical materials used have the "Can be  
climbed = ON", if you want a specific material to not be  
climbable (like ice) then you can assign a new physical  
material with it turned off.

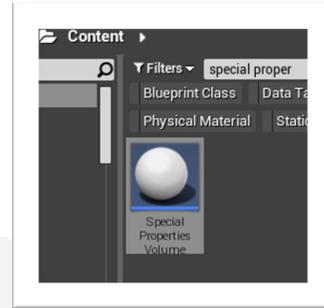


## Example of Natural Blockers

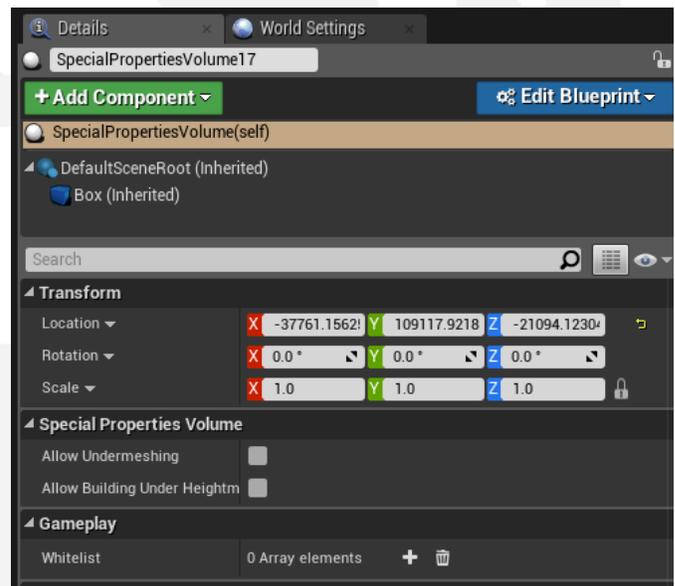
Try to use geographical blockers such as water, oceans,  
toxic fume area, man-made walls or mountains.  
Geographical blockers help keep a player's immersion and  
is another tool to prevent your map from just ending into  
oblivion.



## Anti Undermesh Volume



Because of the undermeshing exploits players have been  
using, an anti undermesh system has been created to stop  
players from being able to build and access areas they are  
not intended to. This system will detect if players are  
underneath or inside terrain or static meshes and will then  
kill players after a certain amount of time has passed. To  
prevent players from being killed inside caves, which  
normally are located under the landscape, a special  
properties volume is necessary. The special properties  
volume must cover all parts of the cave and allow  
undermeshing must be ticked on for the volume to work:



## IMPORTANT NOTE!

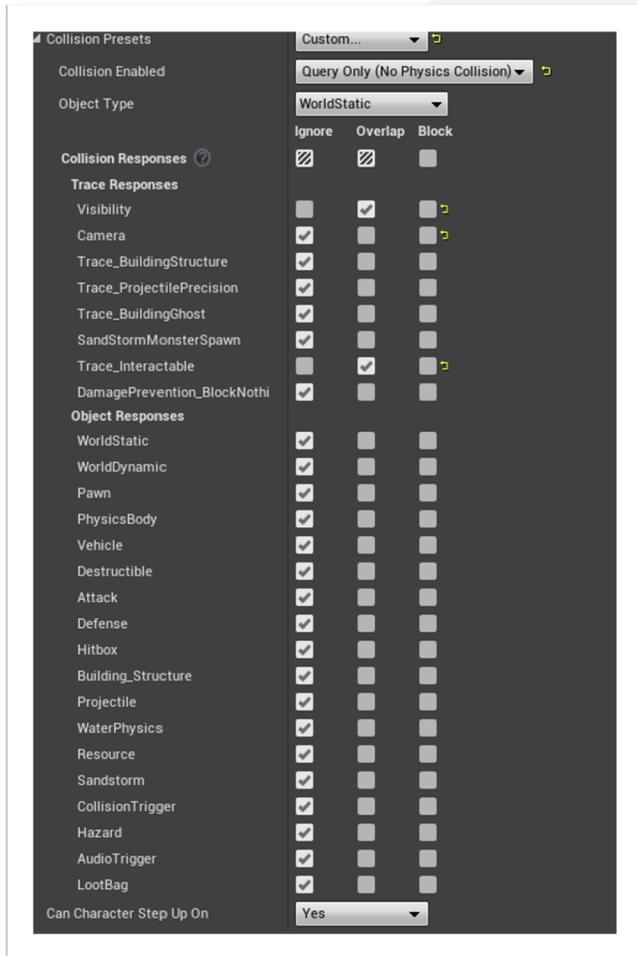
The "Allow Undermeshing" volume MUST be ticked for  
creatures to be able to fight under heightmaps as well.

## Water

It is often easier to just copy this from previous made map, but this will explain the components needed.

To be able to drink water:

- ◆ The Mesh must have Actor tag “Water”
- ◆ The mesh must have collision so the interaction raycast have something to “hit”
- ◆ Collision settings (See Image Below):



## Water Basic Info

- 1) Water Static meshes are all held on the persistent level..unless a dungeon.
- 3) Ocean MI is NOT used (Except in Dagon Dungeon)
- 4) The largest mesh size is 25 x 25 x 25 however closer to the play area the mesh size gets smaller. The smallest recorded size 2.75 closest to any shoreline.
- 5) Majority of material used is MI\_RiverNodFlow which is used in rivers and large bodies of water.
- 6) Rivers that lead into larger bodies of water, the river mesh may run under the larger body of water or directly connect to it. Waterfalls can also blend the bodies together.

## Creating River Meshes

Funcoms Forum tutorial has little info on this topic. The correct steps are as follows.

- 1) use Spline\_BluePrints/master\_river\_spline
- 2) Shape out your river
- 3) Select all the spline segments
- 4) Go to Window > Developer Tools > Merge actors
- 5) A static mesh will be created, assign it to your mod folder.
- 6) Right click on the stat mesh, ASSET ACTIONS > EXPORT FBX
- 7) Import FBX to Blender.
- 8) Clean up mesh to determine speed of river flow
- 9) Export as FBX
- 10) Reimport into UE4 (Right click on existing mesh in UE4 and Reimport modified river mesh)

**SPECIAL NOTE:** Controlling Direction of the River Flow: Use a Negative or positive value in the "Water Tiling" or "Water Tiling Horizontal"

## Adding Ripples

- ◆ For the water mesh to receive ripple effects, Rendering → “Height output reference water” = ON
- ◆ Example Material with ripples set up: /Game/DLC\_EXT/DLC\_Siptah/Environment/Materials/MI\_river\_siptah\_refraction\_Stillwater.
- ◆ Example texture used in the material to mask out where ripples should be: /Game/DLC/DLC\_Siptah/Environment/Landscape/Textures/T\_IsleOfSiptah\_WaterRipples.

## Swimming:

- ◆ Add a Physics volume with Collision preset set as “WaterPhysicsVolume”
- ◆ In the Physic Volume, Character Movement set Water Volume = On
- ◆ Post process Volume is needed for underwater camera affect. Create your own or Use an existing one from Exiles/Isle of Siptah.

## Good practices:

- ◆ Make rivers with “/Game/ProceduralNaturePack/Spline\_BluePrints/master\_river\_spline”, export it to your preferred 3d software and tweak the UV-mapping to make it go fast/slow in areas.
- ◆ It is prudent to keep rivers flat and use waterfalls to give it elevation. Adjusting all the Volumes (Postprocess, Physics & Nav) to correctly follow a sloping meandering river is a headache you would want to avoid.

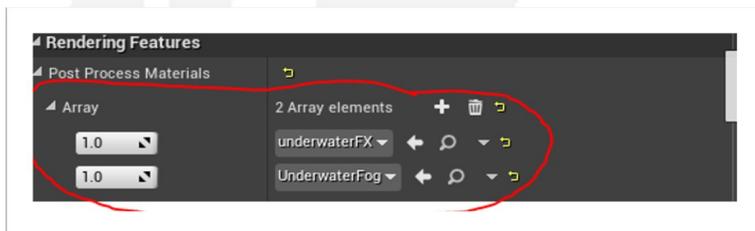
## Aligning Water Physics & Post Processing Volume

To properly align your physics volume you will need to use the following steps. It will require a little math to figure out.

1. Place your water plane. If the "Z" transform reads something like 12345.788974 cm make it 12345.00. Rounding out the number will make calculating easier.
2. Place your physics volume and copy paste the X and Y cords from the water plane into the physics volume X and Y. **DO NOT TOUCH THE Z CORDS.**
3. Subtract the volumes Z from the water planes Z.  
Example: *My water plane sits at Z: 500cm. My Physics Volume sits at Z: 300cm. I subtract 500cm from 300 and get a result of 200cm*
4. Now look for the physics volumes brush shape. Its set at default 200. Now add 200 to the default brush setting. **(500cm — 300cm = 200cm + 200cm = Z 400cm)**
5. Repeat this process for the Water post processing Volume.

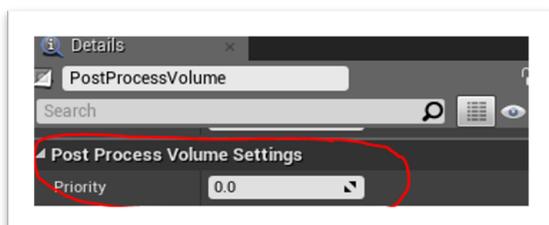
## Adding Underwater Post Processing FX

To give your water a "foggy" look you will need to add two arrays. Look under Rendering Features and add two slots.



## Post Processing In Other Places (Setting Priority)

Post processing will also be used inside caves, dungeons and even different biomes. However it is important to set the priority of your post process volumes. The higher the number the more important it is when players enter the area.



## Considerations

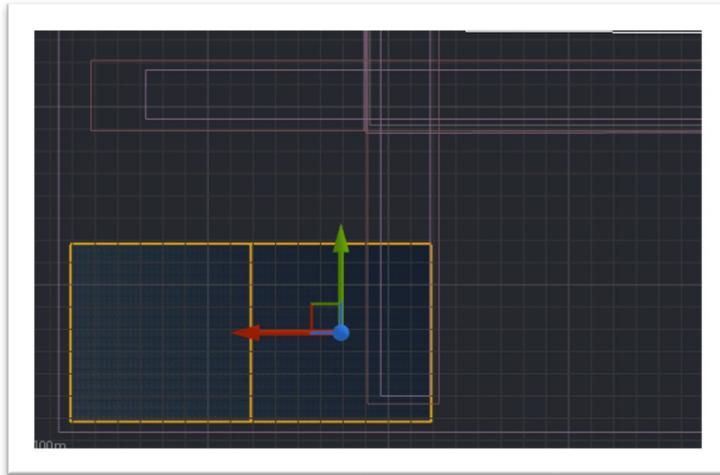
Underwater post processing in CE has been created with different effects through out the map. It is strongly recommended that you go check out the original CE map and its different post processing. Just about every dungeon and water area has a slight change or two to give the player a feel of environmental emersion. I would also suggest studying real world water. Examples include divers in the Caribbean, Artic, Atlantic and the Pacific Oceans and the clarity of those locations.

With a little research on post processing in the Unreal Documentation you can create your own special FX with relative ease.

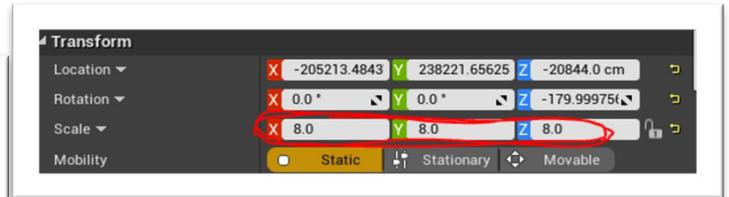


## Tiling Water

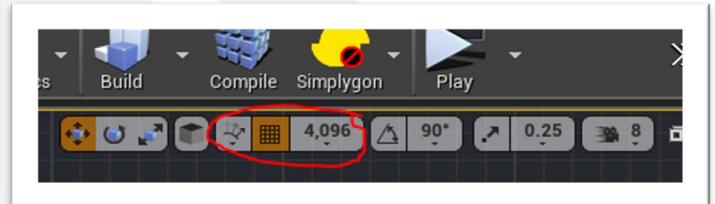
This can be a tedious task. So here is a quick way to make your water tiles in a matter of moments.



5) Make sure your mesh is the “Power of 2”. In the image below we put the scale at 8 x 8 x 8.



5) Enable Snapping and set the position Grid Snap Value to 4096.

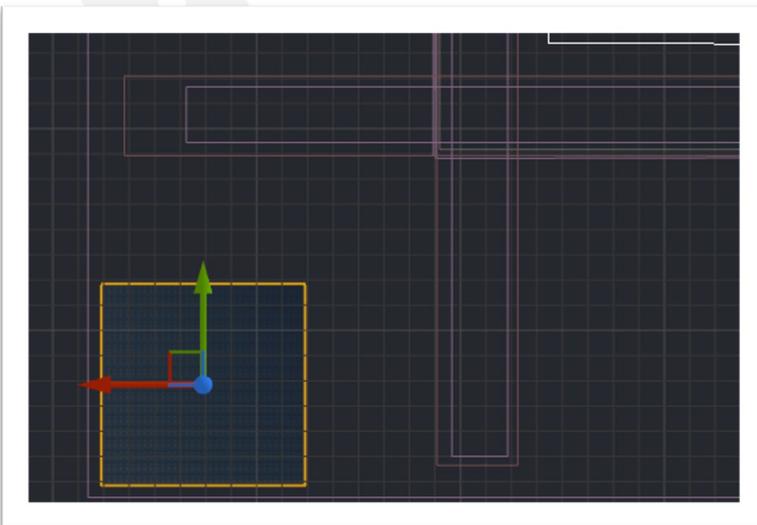


- 1) Go to Edit > Edit Preferences
- 2) Search for “ Use Power of Two Snap Size”. Check the box.

6) With the Static mesh water still selected “hold ALT and drag off the original water plane.



- 3) Search for “SM\_water\_01”. Hover over the Static mesh and check its Approx. Size. (its should read 4096x4096)
- 4) Drag the static mesh in your level. Select / highlight it.

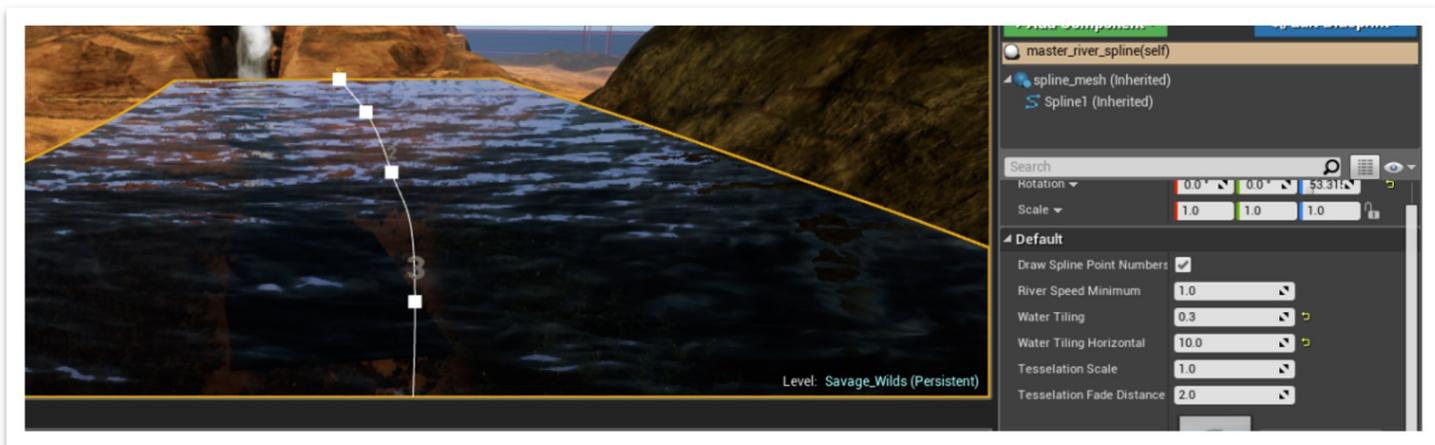


## Tiling Water (ADDITIONAL NOTES)

This info was provided by Fia

*“To adjust the water tiling:*

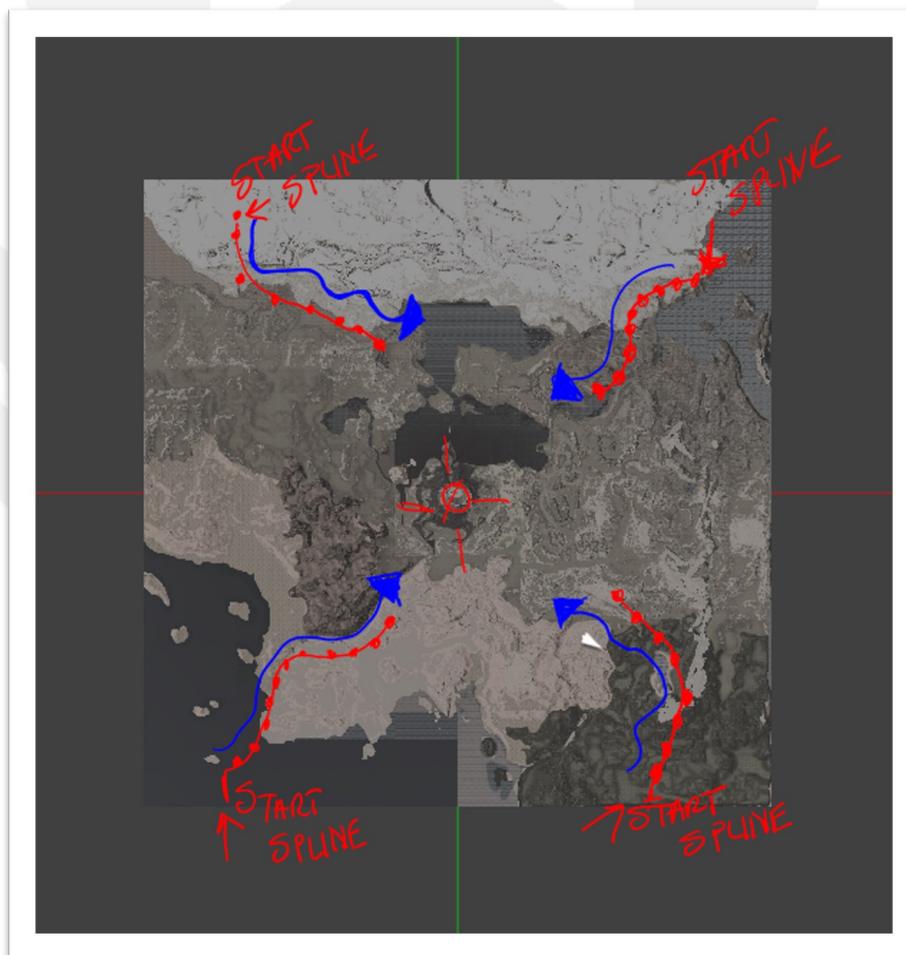
- ◆ *Water Tiling does better with a smaller number*
- ◆ *Water Tiling Horizontal should be a larger number.*
- ◆ *It's best not to go over 40 spline points for one river.*



## River Waterflow Direction

When making rivers flow correctly you will want to start your River Spline on the outside and work inwards. The image below gives an example where to start your splines and work towards X, Y, Z ZERO.

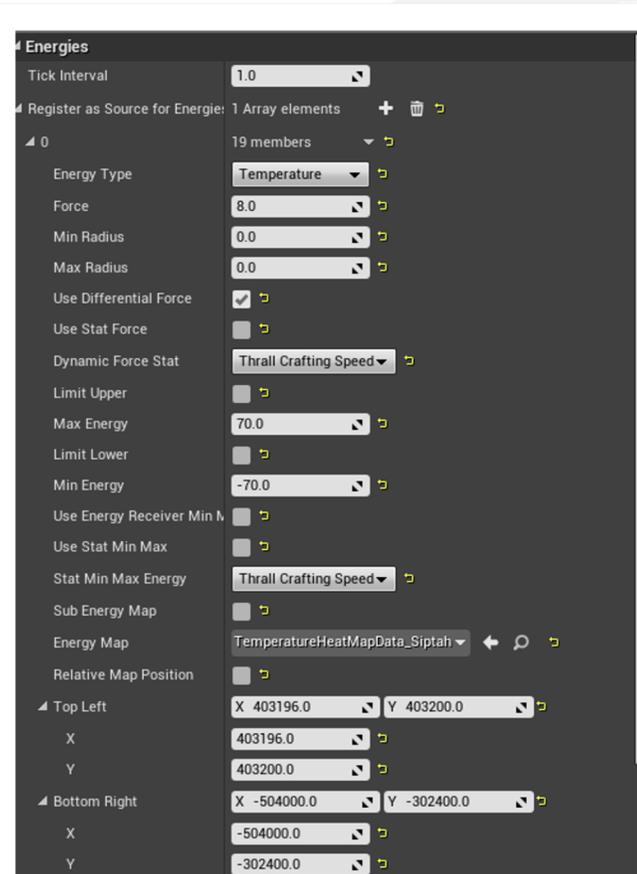
It is important to note that the water will show flowing in the right direction, however when you export the mesh into Blender 3D and reimport it the river will flow the wrong way. To fix this start your splines on the out side working in.



## Temperature

Temperature for a map is provided by a grayscale texture, with pure black being the coldest, and pure white being the hottest temperature. For reference, see T\_TemperatureMap. Somewhere in your level you will need to place an actor with an EnergySource component on it. This allows you to set some values for how Temperature is managed in your map.

Here are the values for Siptah for reference



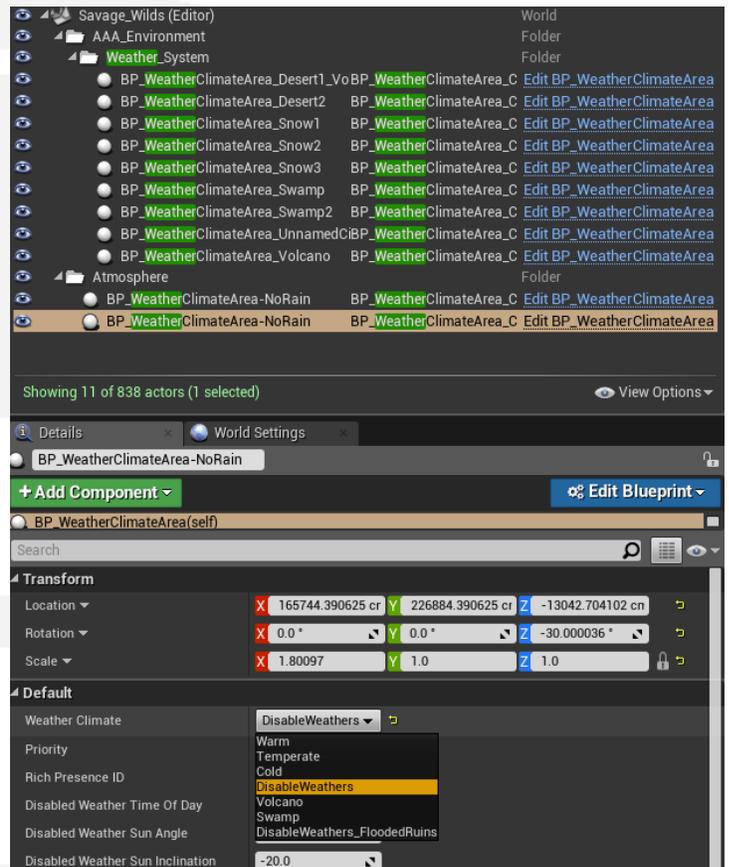
The important values here are the Energy Map, which defines the texture we discussed earlier, and the Top Left, Bottom Right values. An important note here is that Top Left needs to have positive values, and Bottom Right needs to have negative values. This may mean, depending on how your map is laid out in world coordinates, that you may need to rotate your texture 180 degrees so that it is upside down, and allows these values to invert it.

You can check your temperature map in game by pressing ] when the map UI is open. You may need to zoom in and back out once to get the temperature image to scale to the right aspect ratio.

## Snow & Rain

If you want snow and rain in certain areas of your map, you will need to implement the following:

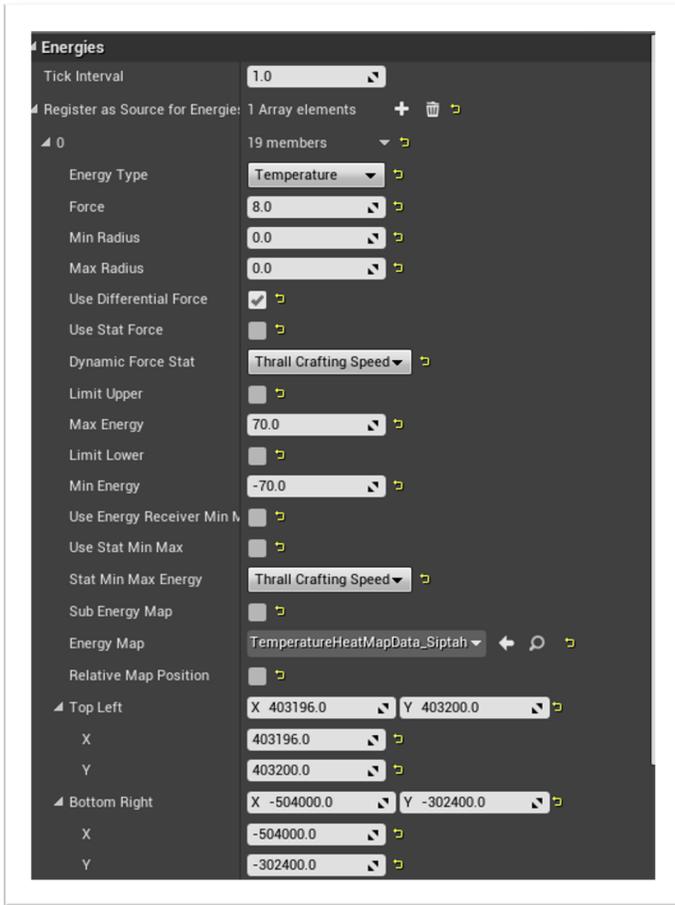
First, find the blueprint named BP\_WeatherClimateArea and drag it into the world. Adjust the area of the blueprint so that it covers the region where you want snow or rain (or exclude weathers like for caves)



Remember to set up priorities for these areas—this is especially important for excluding rain and snow in caves.

## Temperature / Corruption

Temperature for a map is provided by a grayscale texture, with pure black being the coldest, and pure white being the hottest temperature. For reference, see T\_TemperatureMap.

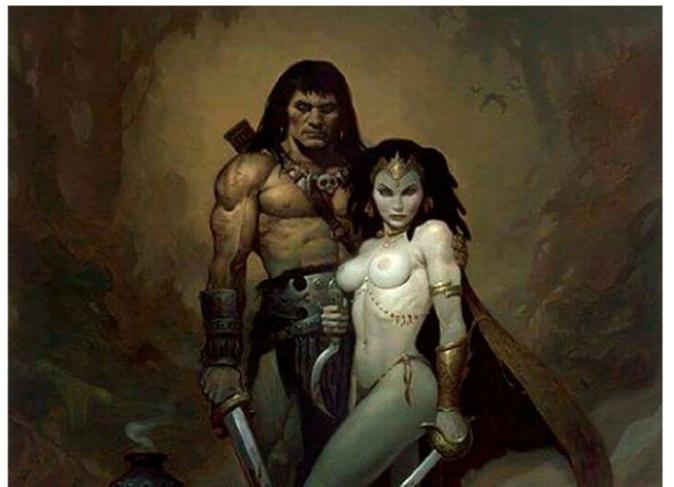
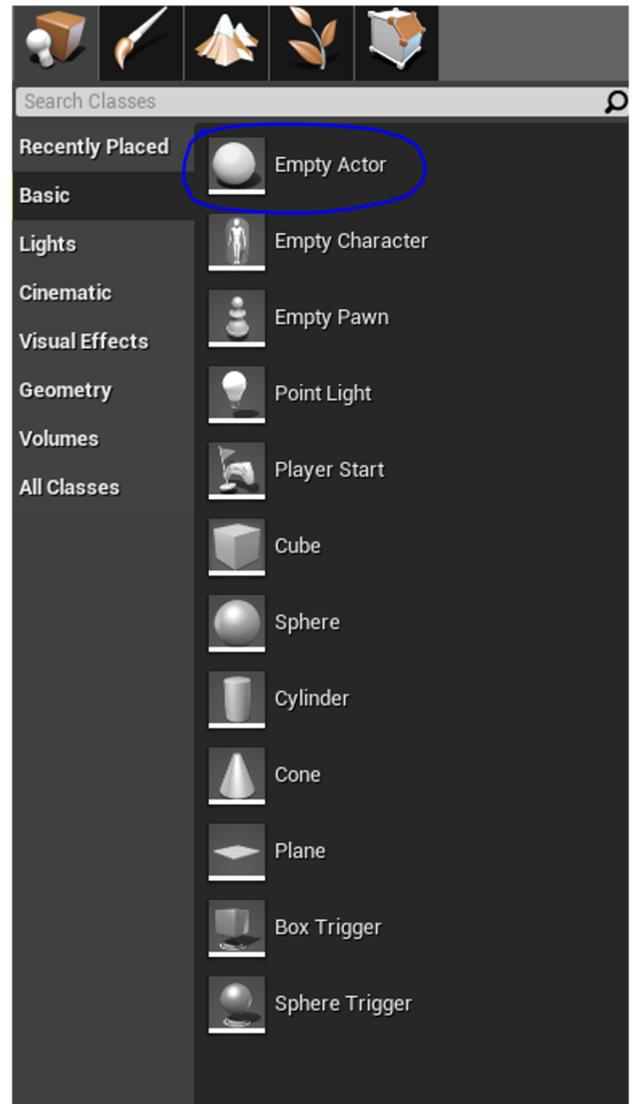


Somewhere in your level you will need to place an actor with an EnergySource component on it. This allows you to set some values for how Temperature is managed in your map.

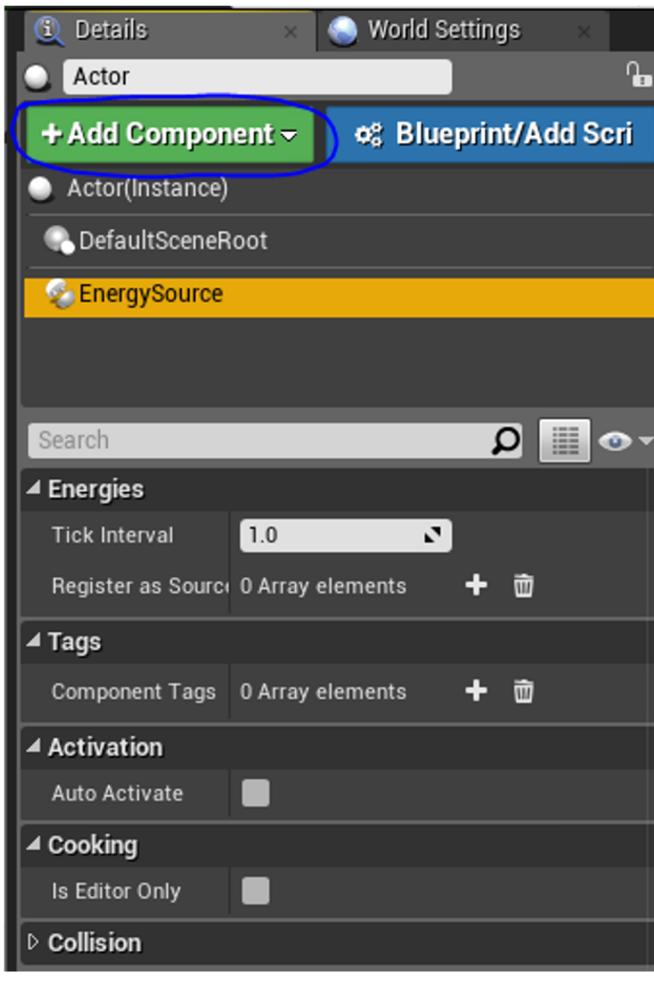
Here are the values for Siptah for reference. The important values here are the Energy Map, which defines the texture we discussed earlier, and the Top Left, Bottom Right values. An important note here is that Top Left needs to have positive values, and Bottom Right needs to have negative values. This may mean, depending on how your map is laid out in world coordinates, that you may need to rotate your texture 180 degrees so that it is upside down, and allows these values to invert it.

You can check your temperature map in game by pressing ] when the map UI is open. You may need to zoom in and back out once to get the temperature image to scale to the right aspect ratio.

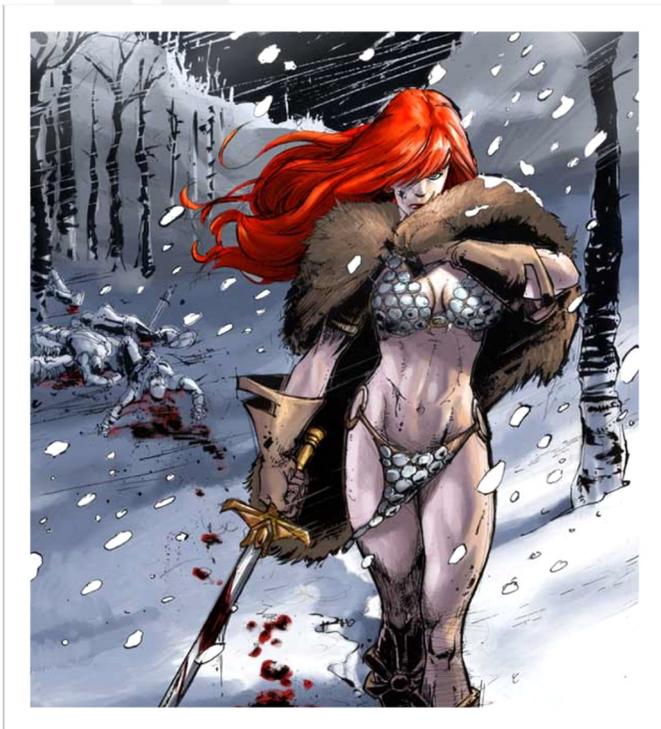
The temperature system is similar to how the character creation works, only we do not need to add a tag to the actor. We will begin by adding an empty actor to the persistent world.



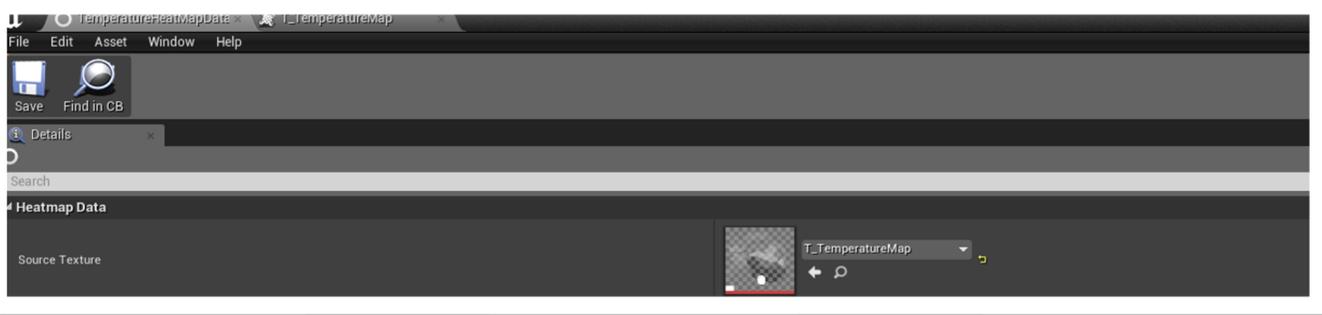
In the details panel of this new empty actor we will click “Add Component” and add “Energy Source” as a component.



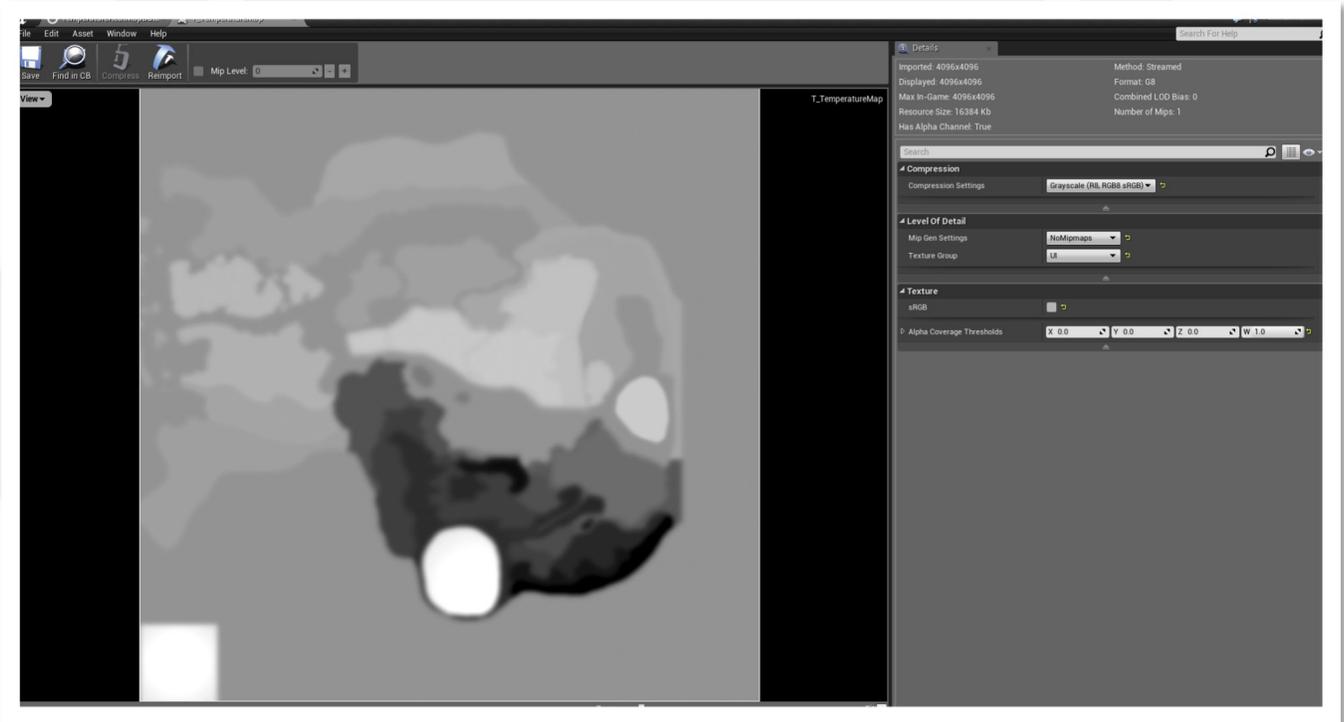
Now with the EnergySource selected, we will click the “+” sign on the “Register as Source for Energies” in the details panel. This will add a child where we can plug in some settings to tweak temperature or corruption for our specific map. The values for temperature on ConanSandbox can be seen in the next image. You may tweak from these numbers if you like. Please take note of the “Energy Map” field in the details panel and that its pointed to “TemperatureHeatMapData”. You may search this in the Content Browser and save a copy of it into your mod folder. Another thing we need to note is the “Top Left” and “Bottom Right” fields in the image below. Those are the coordinates for the two corner extents of the map. There is no Z value as temperature data has no Z value.



Now as you explore the TemperatureHeatMapData, a few things to notice: Its transparent and rotated 180 degrees.



I've filtered the texture by modified properties to aid making your own texture for a temperature map. This image is a Grayscale image, that's 4096x4096.



White on this map is hot and black is cold. Anything between will average out based upon the temperature from the settings of the EnergySource, most notably the "Min Energy" and "Max Energy". Once you've created your own temperature texture map, you'll plug it into the copied TemperatureHeatMapData you saved in your mod folder, then point it to your texture and now plug it into the Energy Source of the actor we added to the persistent world.

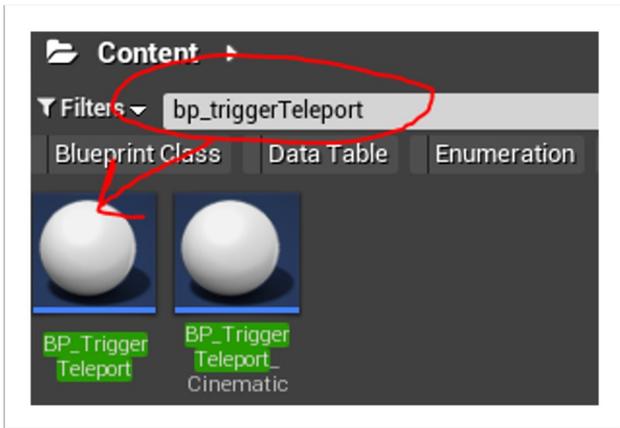
This same process will work for a corruption map too. You just adjust the "Energy Type" to "Corruption" on another Empty Actor.

### OPTION 1: Custom Teleporters

To create custom teleporters for cave or dungeon areas follow these steps.

#### STEP 1

look for *BP\_TriggerTeleport* in the Content Browser.



#### STEP 2

Copy the *BP\_TriggerTeleport* into your mod folder.

#### STEP 3

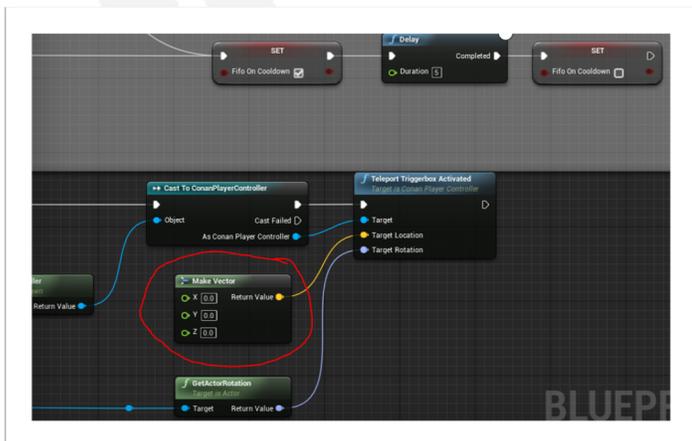
Be sure to rename the blueprint.

#### STEP 4

Open your Blueprint. Once open go to Construction Script tab.

#### STEP 5

Once open, add a Make Vector node. Add your destination cords.

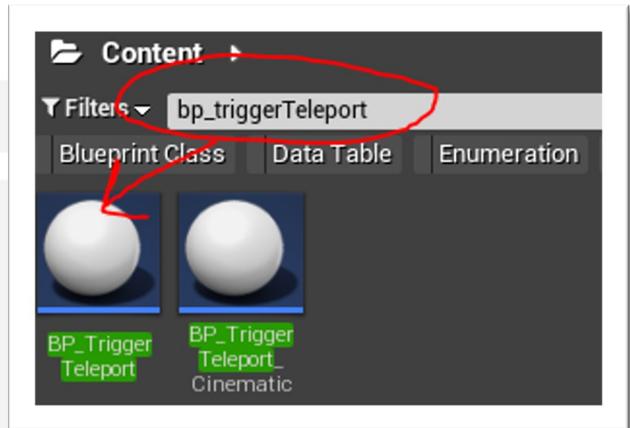


### OPTION 2: Standard Teleporters

This is the easiest option and will prevent multiple players from teleporting on top of one another.

#### STEP 1

look for *BP\_TriggerTeleport* in the Content Browser.

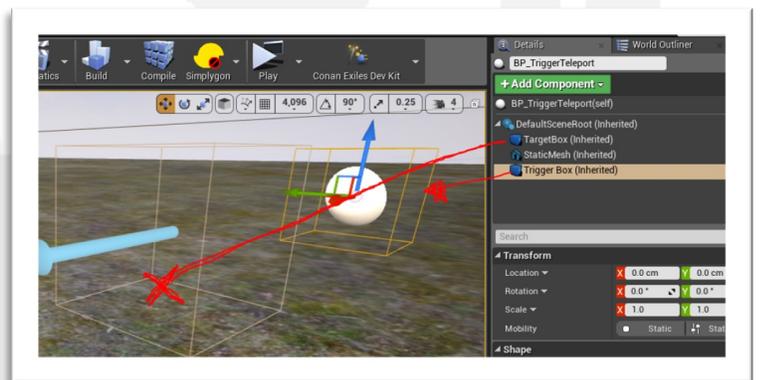


#### STEP 2

In the "Components" window of the placed blueprint, find the Triggerbox and adjust the size of it - this will be the trigger that, when players touch it, teleports them.

#### Step 3

In the "Components window of the placed blueprint, find the TargetBox and drag it to the location in the world you want players to teleport to and adjust the angle and size of it.



**NOTE:** The "Trigger Box" is what activates the teleport and the "Target Box" is the player destination.

## FOLIAGE CHECKOFF SHEET

The following three pages are resources that should be in/on every map. If these are NOT available players may not be able to complete recipes.

REQUIRED RESOURCES	USE THESE FOLIAGE TYPE	LOCATION
Aloe	Aloe_Vera_Desktop_1_FoliageType Aloe_Vera_Desktop_2_FoliageType Aloe_Vera_Desktop_3_FoliageType	Along beaches and water areas
Amanita mushroom	SM_Plant_MushroomHighland01 SM_Plant_MushroomHighland02 SM_Plant_MushroomHighland03 SM_Plant_MushroomHighland04 SM_Plant_MushroomHighland05	Anywhere green or highlands, typically forests
Amanita mushrooms (glowing)	FT_Plant_Mushroom_01_Glow FT_Plant_Mushroom_01_Glow_02 FT_Plant_Mushroom_04_Glow	Swamp/Highland caves
Asura's Glory	SM_Highlands_IvanTea01 SM_Highlands_IvanTea02	Green lands
Black Ice	Highland_Black_Ice_01 Highland_Black_Ice_02 Highland_Black_Ice_03	Northern ice areas
Black Lotus	Lotus_plant_Grey	Swamps
Bone	SM_Skeleton_Bone1_FoliageType SM_Skeleton_Bone2_FoliageType SM_Skeleton_Bone3_FoliageType SM_Skeleton_Pelvis_FoliageType SM_Skeleton_Ribs1_FoliageType SM_Skeleton_Ribs2_FoliageType SM_Skeleton_Skull1_FoliageType SM_Skeleton_Skull2_FoliageType	Wherever it is suitable
Brimstone Pickaxe Stalagmite	Cave_saltstalactite_01_FoliageType	Poisonous areas with higher than normal
Brimstone pickups	Brimstone_rock_small01_FoliageType Brimstone_rock_small02_FoliageType Brimstone_rock_small03_FoliageType	Cave, Poisonous areas with higher than normal heat
Clam	Highland_Clam	Beaches, oceans
Coal	Stone_Resource_Coal_01	Universally spread out but with areas of high density
Crimson Lotus	Lotus_plant_Crimson	
Dead Trees	SM_dead_tree_2_FoliageType SM_dead_tree_3_FoliageType SM_dead_tree_4_FoliageType	Areas of extreme heat and blight. Rarely in forests
Eggs	Food_Egg	Only on nests of creatures

## FOLIAGE CHECKOFF SHEET

REQUIRED RESOURCES	USE THESE FOLIAGE TYPE	LOCATION
False Mandrake	Elderberry_Fruit_FoliageType	Green lands
Feather	SM_Vulture_feather_FoliageType	Only on nests of creatures
Fire Lotus	Lotus_plant_Fire	Former battlefields, aligned with
Frost Lotus	Lotus_plant_Ice	Northern frostlands
Glowing Goop	SM_AlgaeB_01_glow_FoliageType	Underwater
Golden Lotus	Lotus_plant_Golden	Volcanic areas and places where me-
Grubs	Food_Grubs	Anywhere there are grass-lands with
Ice	Highland_White_Ice_01 Highland_White_Ice_02 Highland_White_Ice_03	Icy highlands
Ice floe	Highland_Ice_Sheet_Cracked	Northern frostlands with water
Iron	Stone_Resource_Iron_01 Stone_Resource_Iron_02	Universally spread out but with areas of high density
Lupine	Elderberry_Flowers_01_FoliageType	Green lands
Midnight Blue flower	SM_Highlands_MidnightBlue01	Green lands
Obsidian	Obsidian_01 Obsidian_02 Obsidian_03	Volcanic areas and places where me- teors have hit
Orange Phykos	FT_Underwater_RedAlgaeA FT_Underwater_RedAlgaeB FT_Underwater_RedAlgaeC	Underwater



## FOLIAGE CHECKOFF SHEET

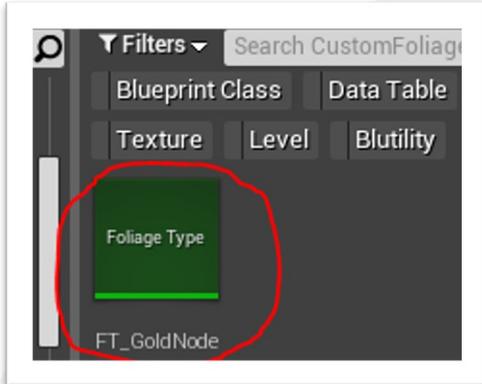
REQUIRED RESOURCES	USE THESE FOLIAGE TYPE	LOCATION
Pickup Crystal	Cave_Crystal_small_01_FoliageType Cave_Crystal_small_02_FoliageType Cave_Crystal_small_03_FoliageType Cave_Crystal_small_04_FoliageType Cave_Crystal_small_05_FoliageType Cave_Crystal_small_06_FoliageType	Caves
Pickup Crystal big cluster	Cave_Crystal_01_FoliageType Cave_Crystal_02_FoliageType	Caves
Pickup Crystal blue	Cave_Crystal_01_green_FoliageType Cave_Crystal_02_green_FoliageType	Caves
Puffball mushroom	SM_Plant_MushroomDesert01 SM_Plant_MushroomDesert02 SM_Plant_MushroomDesert03 SM_Plant_MushroomDesert04 SM_Plant_MushroomDesert05	Deserts near water
Purple Lotus	Lotus_plant_Purple	Swamp
Silver	Stone_Resource_Silver_01 Stone_Resource_Silver_02	Deep caves
True Indigo	Dog_Rose_01_FoliageType	Green lands
Volcano Flowers	SM_volcano_flower_01_FoliageType SM_volcano_flower_02_FoliageType SM_volcano_flower_03_FoliageType	Volcanic areas and places where meteors have hit
Yellow Lotus	Flower_Lotus_plant	Along rivers
Gossamer	PickupGossamer	Where spiders live, caves
Leavening	SM_Highlands_Leavening01 SM_Highlands_Leavening02	Highland areas, usually flatter terrain
Hops	SM_Highlands_Hops01	Highland areas, usually flatter terrain



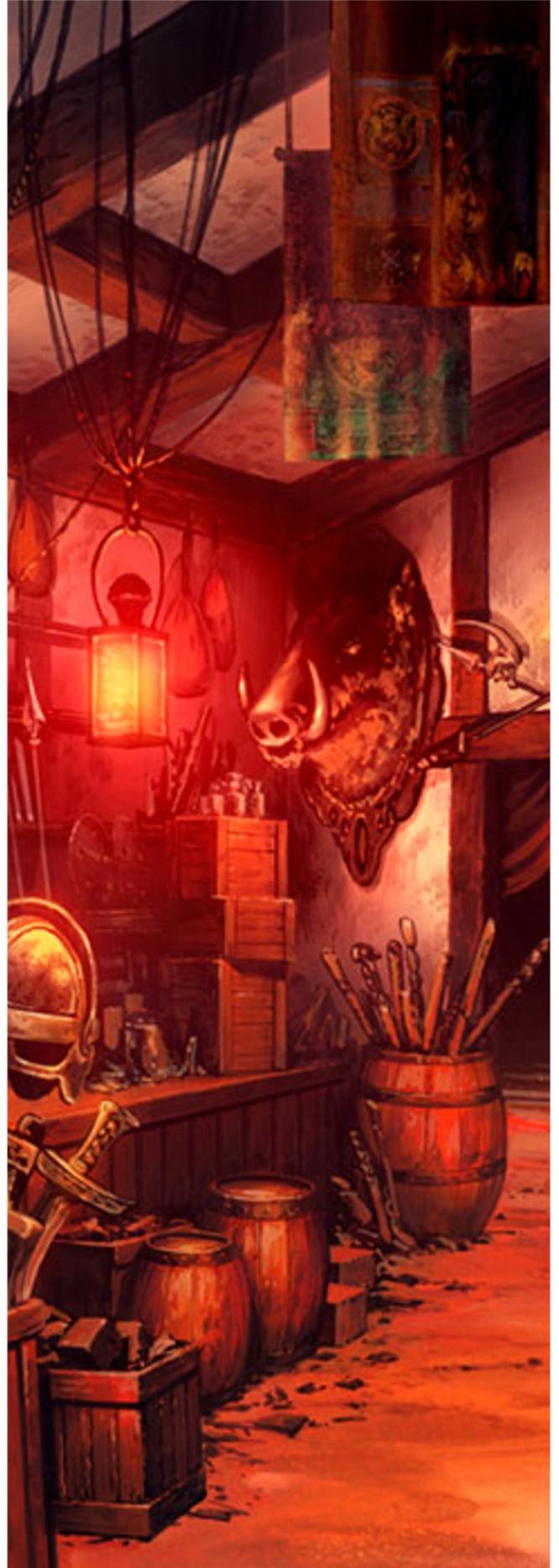
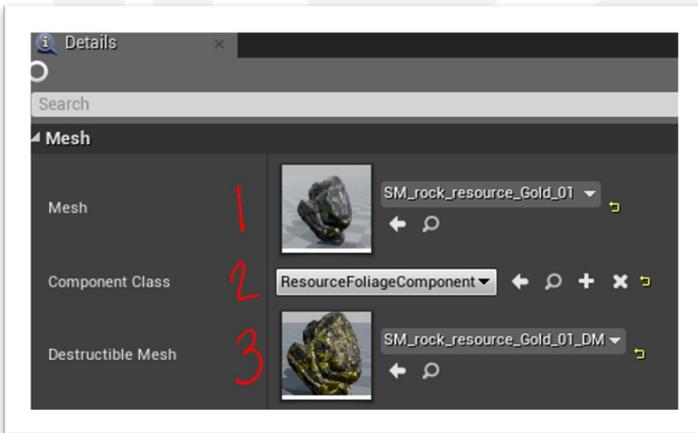
## Creating Foliage

So you find a resource in the devkit that is not available. Here is how to implement a new foliage type. In this example we will be adding a Gold Node which not available.

- Right Click in your *Content Browser* > *Miscellaneous* > *Foliage Type* (Name your Foliage type)



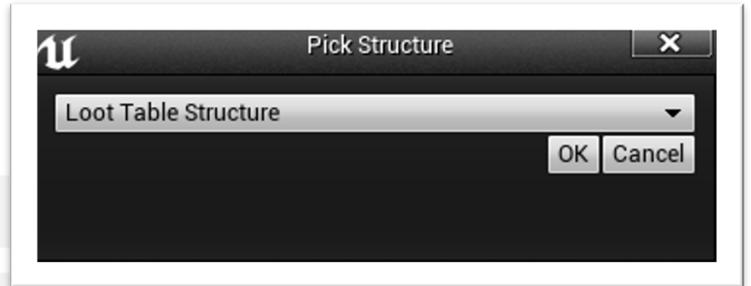
- Open the new *Foliage Type* and add the (1) Mesh, (2) Component Class and (3) Destructible Mesh. Its important to note that if the item is a pick up not something you mine use the *PickupFoliageComponent* instead.



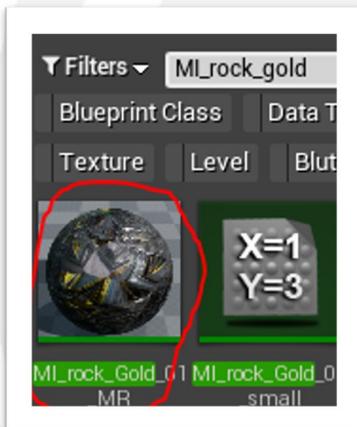
## Assign a Physical Material

The physical material is an entry that determines how the material behaves and links to what resources are granted to the player when harvesting the resource node.

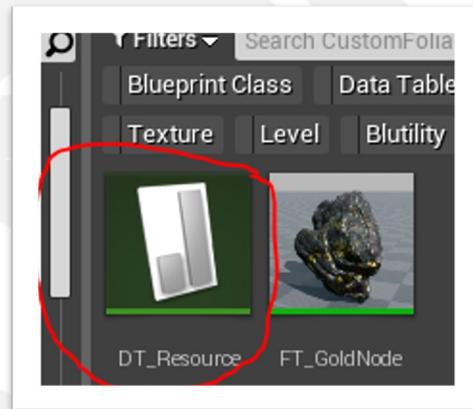
- Make a new Data Table and use the *Loot Table Structure*.



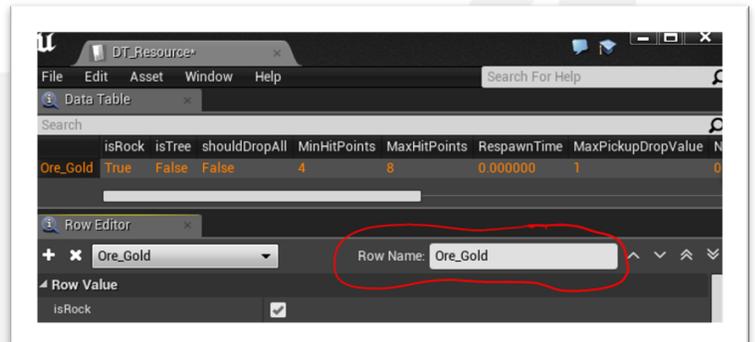
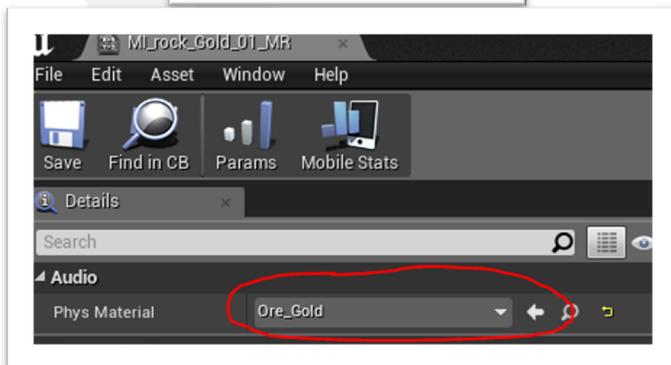
- If you double-click on your material, you can see that it is currently linked to the *Ore\_Gold* physical material.



- Name your new Data Table and open it up.



- In the row name be sure to name it EXACTLY the same as the Physical Material. In our case it will be *Ore\_Gold*.



Now that we have physical materials for both our nodes your work is almost done - the final thing we need to do is to map our new material to a new table.

## Adding resource to loot-tables (CONT.)

Open the "LootTable\_Resource" table and create a new entry here. This entry must have the same row-name as our created physical material.

It may be worth going over this table in detail to explain what is going on here.

**isRock** - this should be clicked if the resource is harvested like a rock

**isTree** - likewise, if the resource is a tree, this should be clicked

**shouldDropAll** - Uncertain what this is, we've never used it

**MinHitPoints** - the minimum amount of hits that the resource can take before depleted

**MaxHitPoints** - the maximum amount of hits that the resource can take before depleted

**RespawnTime** - the time it takes for the node to respawn

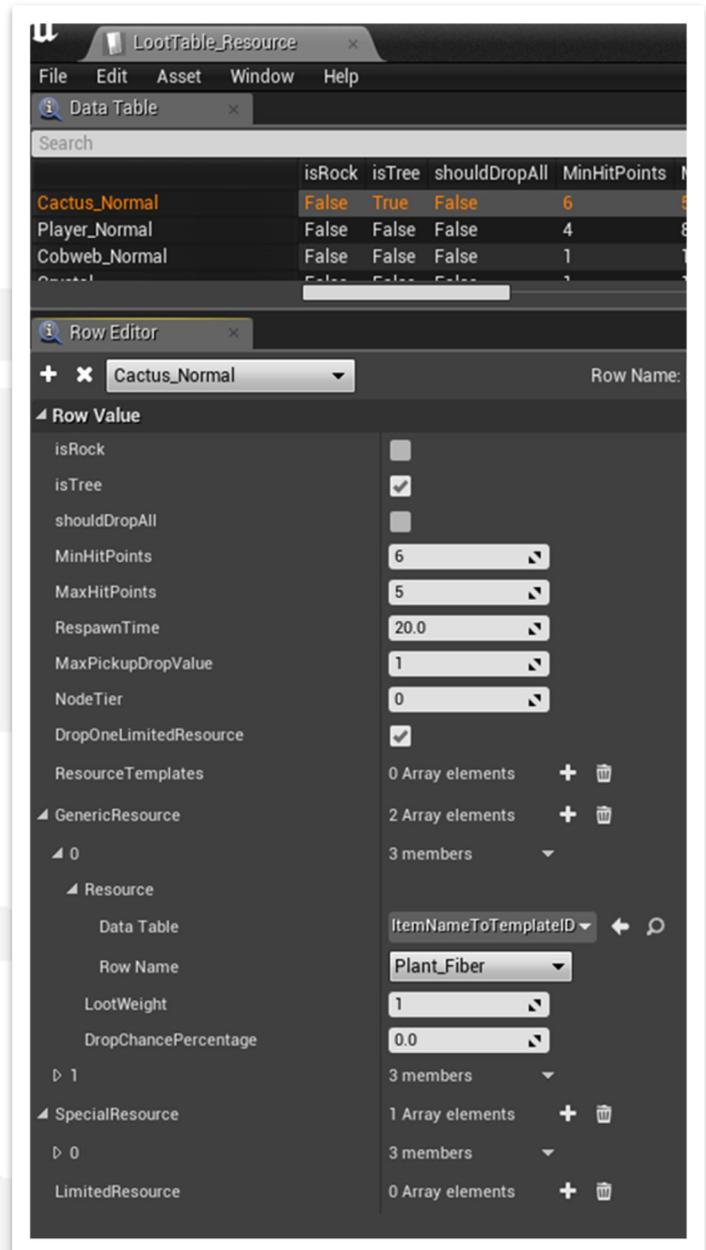
**MaxPickupDropValue** - leave this as is, it's not used

**NodeTier** - defaults to 0, which allows any tier of any tool to harvest it, we'll get into this later. In the actual game, we don't do anything with this except for star metal.

**DropOneLimitedResource** - If this is checked the system will make certain that at least one limited resource drops, regardless of the dropchance for it

A quick note about **ResourceTemplate/** - The resource template dropdown is optional and for our purposes we don't need it. It is used to set up global templates for certain monster types so that it is easier to assign loot to new monsters.

We won't need to edit this for our purposes right now but if you are creating a new monster and have a new item that you would like to use the same template as, say, spiders, you can select the spider template from this and your monster will drop all the things defined in that template (in addition to the extra items you assign below).



### GenericResource/

This part of the loot-table defines which resources are given out for every chop on the resource. The LootWeight and the DropChancePercentage entries here doesn't matter and you can safely leave it at 1/0.

Note that all the loot-tables go through the ItemNameToTemplateID table, which means that if you have created a new item, you will need to create entries for these in that table and add it through the mod-controller.

### SpecialResource/

This part of the loot-table is a standard loot-table. You can enter as many resources here as you like, and assign a DropChancePercentage to them.

### LimitedResource/

As with SpecialResource, this allows you to select any amount of resources to drop here, but - they are limited by the MaxDropAmounts value. Typically things like unique items or heads are set to MaxDropAmounts=1 here.

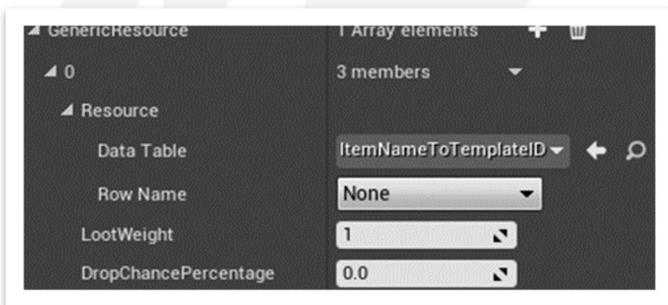
**TableRow** - This is the unique entry for the item. This needs to match the PhysicalMaterial.

**CorrectToolType** - This determines what tools are allowed for the resource. Note that you may have multiple tools with access to the resource

**ResourceTier** - This is used if you want the resource to only be available of tools of a specific Item Tier (see below)

**blsToolExclusive** - Check this if you want the resource to only be allowed for the specific tools listed in CorrectToolType. Note: There have been some issues with this functionality, which led to the implementation of a second table called "WeaponTypeResourceStatTable", more about this table below.

**blsTierExclusive** - Check this if you want the resource to only be harvestable by tools of a specific Item Tier. Item tier can be set up in the ItemTable. Usually the tier list goes like this: 0)Any tier, 1)Iron, 2)Steel, 3)Anything above steel.



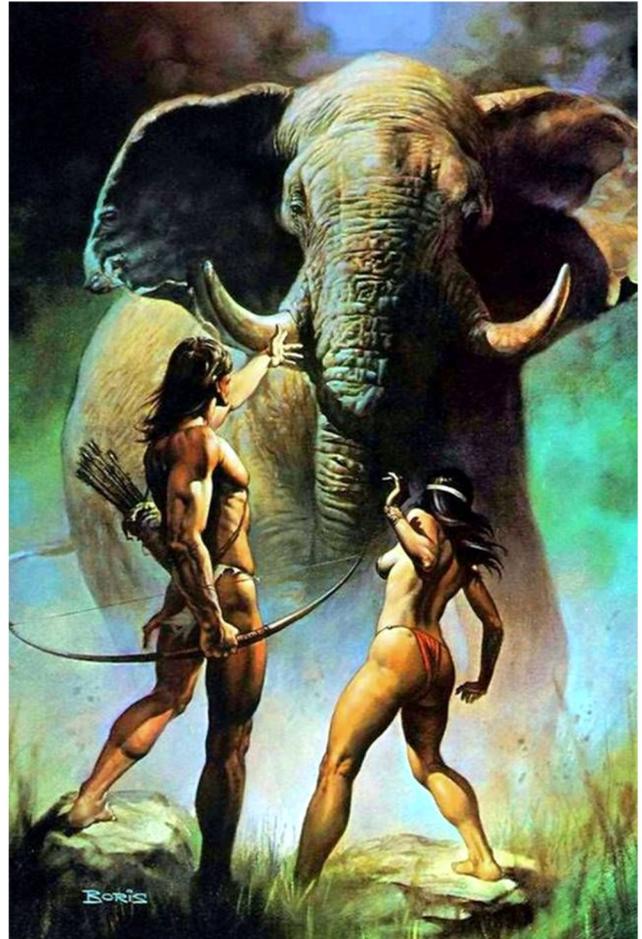
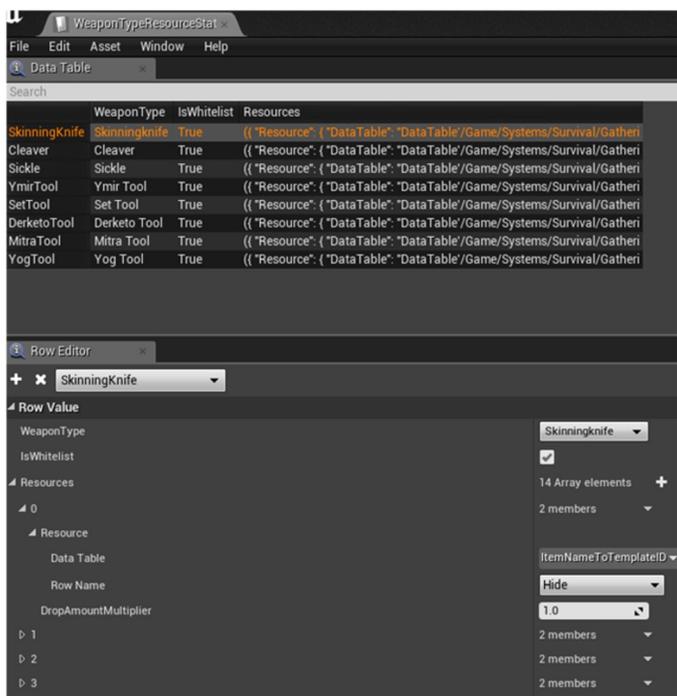
### Dis/Allowing specific harvesting tools

You may want only specific tools to have access to your new resource. In our case, we want to only allow pickaxes to be used on the Jade node. For this, you will need to add a new entry to the ResourceStatTable. This table should be somewhat self-explanatory, but let's go through the options available in this table.

Resource	CorrectToolType	ResourceTier	blsToolExclusive	blsTierExclusive
Stone	(Pickaxe,Multitool)	1	False	False
Wood	(Hatchet,Multitool,Ymir Tool)	1	False	False
Bark	(Pickaxe,Multitool)	1	True	False
Branch	(Hatchet,Multitool,Ymir Tool)	1	True	False
Coal	(Pickaxe,Multitool)	1	False	False
Iron_Ore	(Pickaxe,Multitool)	2	False	False
Ore_Starmetal	(Pickaxe,Multitool)	3	False	True
Crystal	(Pickaxe,Multitool)	0	False	False

## Whitelisting resources

In addition to making tools exclusive, you may also want to whitelist resources. This can be done in the WeaponTypeResourceStatTable. Here you can set up a whitelist for each tool, should you need to.



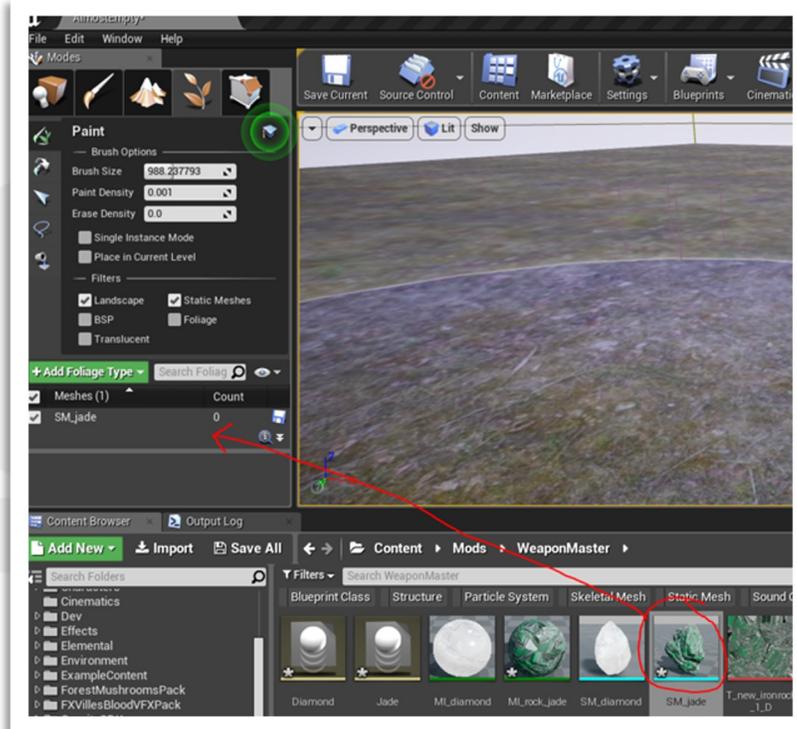
## Painting Resources

Now that we have set up the resources, it's time to actually paint them into the world.

In the Modes tab, select the foliage brush (see image on the right). With this brush, you can paint foliage actors in the world. You will need to drag your foliage mesh into the "Meshes" section of the foliage and make sure that you have it selected.

Also worth noting is that you may want to first unselect any other potential brushes in the list before painting. This guide won't detail the functionality of Foliage painting as such, but there is a comprehensive guide to Foliage in Unreal here:

<https://docs.unrealengine.com/en-us/Engine/Foliage>

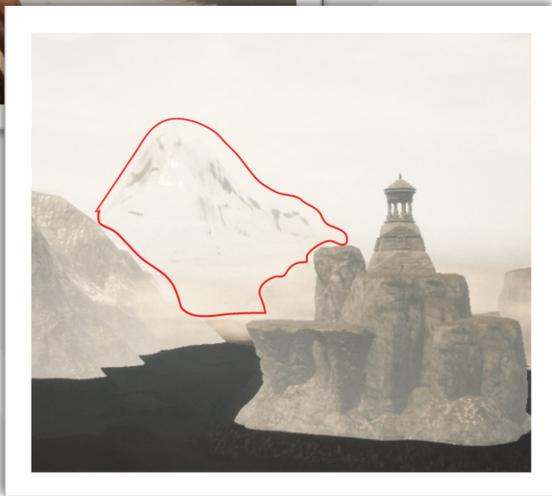
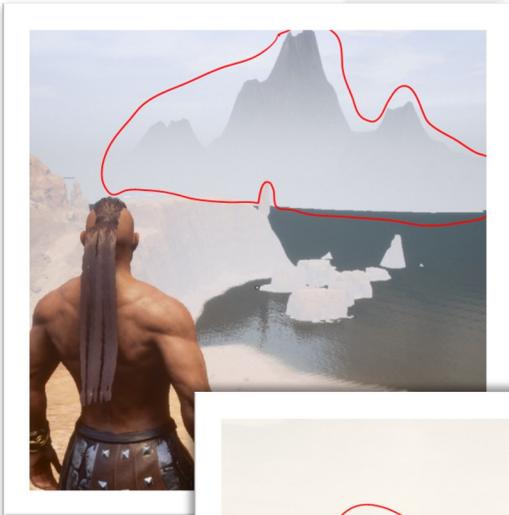


## Illusions of off Distant Vistas

If you have issues with a map that drops off to infinity you can trick the players eye and give the illusion that some land masses exist off in the distance. Use

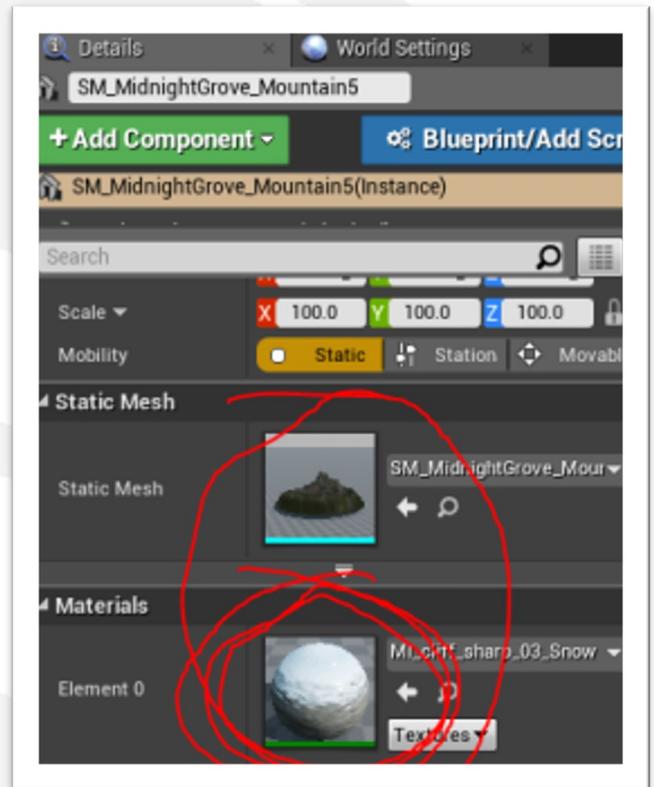
*SM\_MidnightGrove\_Mountain* static mesh.

### Examples of the mesh being used:

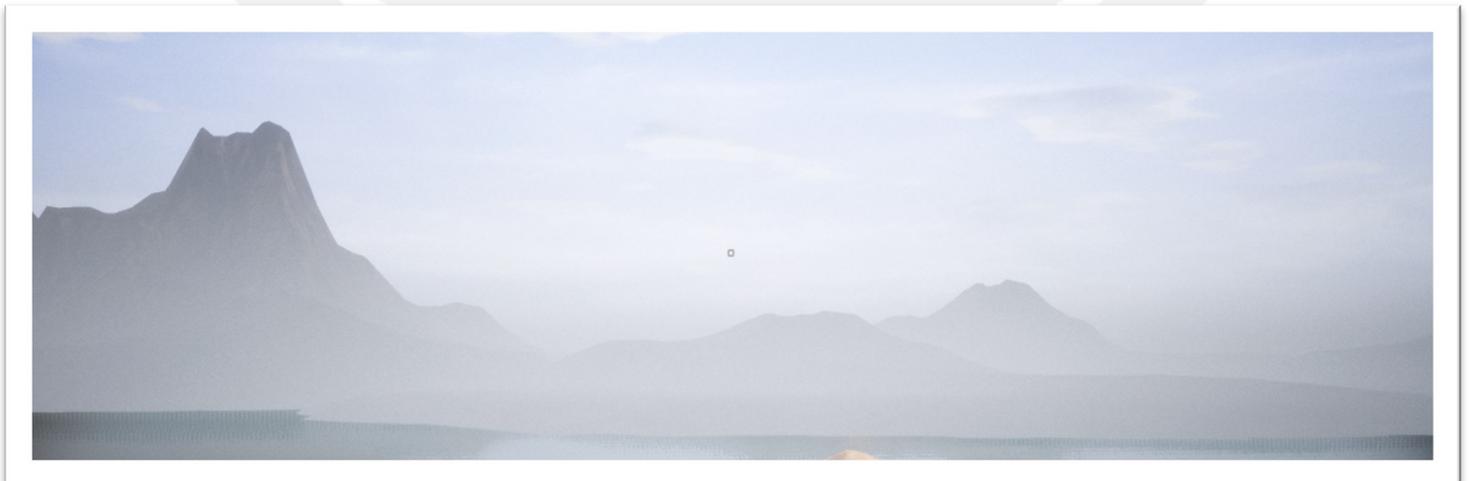


Some options for loading or rendering these meshes would be to put them on a streaming level, persistent level or even on their own level or sublevel. This will be determined by the layout of your map and what the map requires for illusion placement.

You can also use different materials to get different environmental results. These meshes are very small (Approx 248KB) so a ton of these can give wonderful off distance vistas.



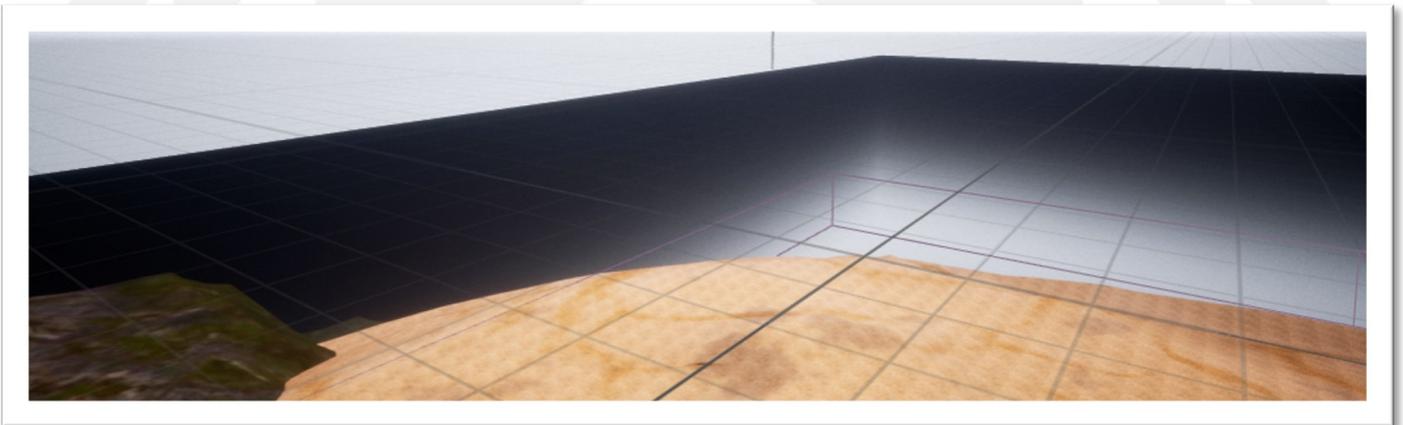
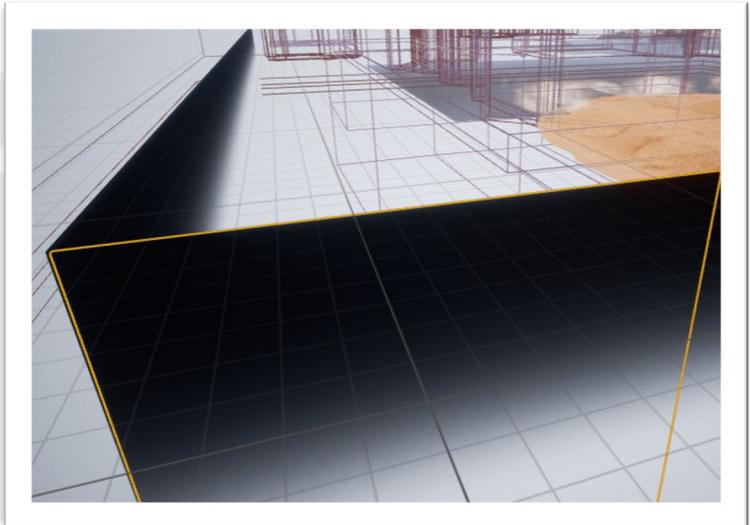
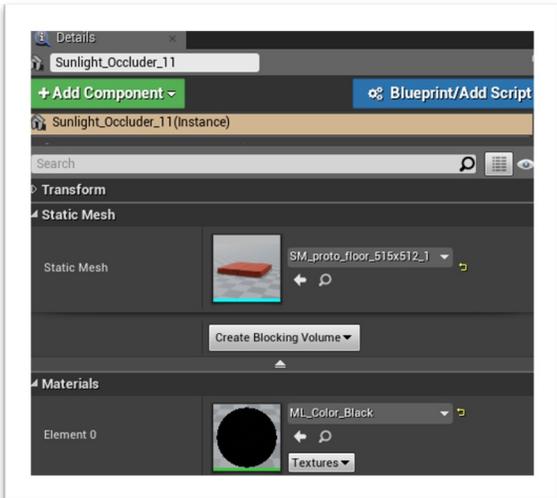
(BELOW) This vista scene is all a illusion and less than 500KB.



## Light Occluder Walls

Light Occluders will help stop sunlight distortions when the sun sets or rises. It also helps with reflections on the water and will give a better look to your maps lighting. These are just a series of “walls” placed on the sides of the map where the sun meets the horizon.

If you load the Exiled Lands you can reference how Funcom built and placed their Occluders. You can also Copy / Paste the occluders from Funcoms map to yours.

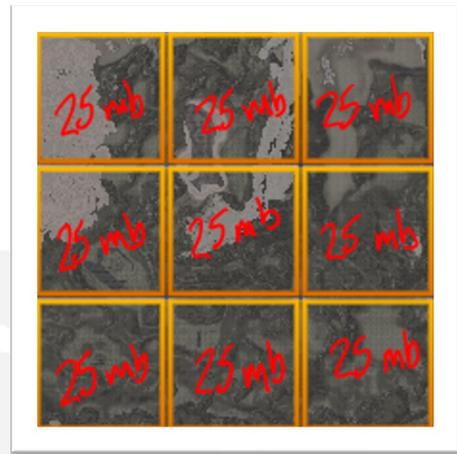


## Optimizing Files

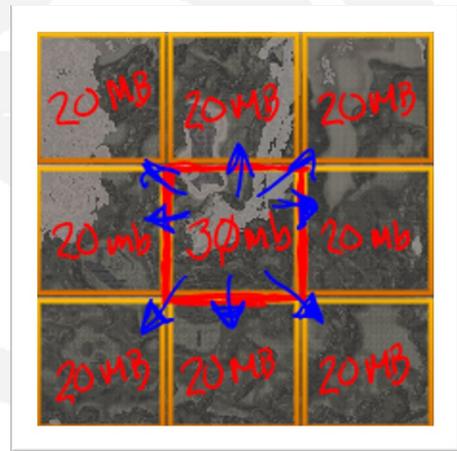
Some optimizations such as your mod size will make your map more playable. I will go a bit deeper on cleaning up some internal work such as BuildData files. These files are not necessary when packing your mod and can reduce download times.



EXAMPLE A



EXAMPLE B



Something to be VERY aware of is individual Heightmap size. A typical Heightmap in the Exiled Lands is not over 100mb. For each actor placed in the heightmap increases that maps size. This causes more loading time and may cause lag when a player enters a region. (Remember, a EL heightmap is 1010 x 1010 or 1009 x 1009 UE4 units. = 100mb)

	Heightmap_1F_Border.umap	12/13/202...	UMAP File	2,194 KB
	Heightmap_1G_Border.umap	12/13/202...	UMAP File	2,457 KB
	Heightmap_1H_Border.umap	12/13/202...	UMAP File	2,504 KB
	Heightmap_1I_Border.umap	12/13/202...	UMAP File	2,672 KB
	Heightmap_2B_Border.umap	12/13/202...	UMAP File	2,116 KB
	Heightmap_2C_Border.umap	12/13/202...	UMAP File	2,051 KB
	Heightmap_2D_Border.umap	12/13/202...	UMAP File	2,677 KB
	Heightmap_2D_Border_BuiltData.uasset	12/13/202...	UASSET File	2 KB

In our example map of 505 x 505 px our individual heightmaps are 1/4 the size of the Exiled lands. That means our file sizes should be no larger than 25mb per heightmap.

If you absolutely have to go over your heightmap size be sure to reduce the adjacent heightmaps to account for loading.



## **Paper Dolls**

How to make the character paperdoll appear in your map:

1. Create a new level named "Paperdoll" (name probably doesn't matter)
2. Go into ConanSandbox and load "Paperdoll" level
3. Mark all the actors in that level
4. Press CTRL-C
5. Load up your map and activate the paperdoll level
6. Press CTRL-V
7. Done



## CHECK OFF SHEET

Use a check off sheet to help set goals and keep track of your work. Organization is key to keep track of your progress. These are just some of the things that will need to be addressed when making a map. Its not a all inclusive tracker so be sure to DYOR (Do Your Own Research).

MAP CHECK OFF SHEET			
Parts To Complete	Completed	Parts To Complete	Completed
Create Heightmap (Option A)	<input type="checkbox"/>	Nav Mesh volume	<input type="checkbox"/>
Sculpt Heightmap (Option B)	<input type="checkbox"/>	Lore (Optional)	<input type="checkbox"/>
Import Heightmap into UE4 4.15 (Option A)	<input type="checkbox"/>	Water Physics Volumes	<input type="checkbox"/>
Add Base Materials (Color)	<input type="checkbox"/>	Dungeon PostProcessing	<input type="checkbox"/>
Set Streaming Distances	<input type="checkbox"/>	Blutility NPC Spawners	<input type="checkbox"/>
Create Sublevels	<input type="checkbox"/>	Player In-Game Map (UI)	<input type="checkbox"/>
Create Custom Brushes (Optional)	<input type="checkbox"/>	Foliage Painting	<input type="checkbox"/>
Check Material Landscape Optimizations	<input type="checkbox"/>	Blocker Volumes (Invis Wall)	<input type="checkbox"/>
Cliffing	<input type="checkbox"/>	Fog Planes	<input type="checkbox"/>
Check / Fix Mesh Tears	<input type="checkbox"/>	Character Creation	<input type="checkbox"/>
Add Cities (Optional)	<input type="checkbox"/>	Safetynet Volume	<input type="checkbox"/>
Add Towns (Optional)	<input type="checkbox"/>	Light Occluders	<input type="checkbox"/>
Add Villages (Optional)	<input type="checkbox"/>	Paperdoll	<input type="checkbox"/>
Add NPC Camps (Optional)	<input type="checkbox"/>	Off Distance Vista Meshes	<input type="checkbox"/>
Add POI (Points of Interest)	<input type="checkbox"/>	Mod-Controller	<input type="checkbox"/>
Add Sunlight / Moonlight	<input type="checkbox"/>	Data Tables	<input type="checkbox"/>
Add Exponential Height Fog (Optional)	<input type="checkbox"/>	Splash Screen	<input type="checkbox"/>
Building Blockers (Spline)	<input type="checkbox"/>		<input type="checkbox"/>
Climbing Blockers	<input type="checkbox"/>		<input type="checkbox"/>
Anti-Undermesh Volumes	<input type="checkbox"/>		<input type="checkbox"/>
Waterplanes (Mesh)	<input type="checkbox"/>		<input type="checkbox"/>
Rivers (Mesh)	<input type="checkbox"/>		<input type="checkbox"/>
Water PostProcessing	<input type="checkbox"/>		<input type="checkbox"/>
River Post Processing	<input type="checkbox"/>		<input type="checkbox"/>
World Post Processing	<input type="checkbox"/>		<input type="checkbox"/>
Temperature (Heat Map)	<input type="checkbox"/>		<input type="checkbox"/>
Time of Day Manager	<input type="checkbox"/>		<input type="checkbox"/>
Ultra Dynamic Sky BP	<input type="checkbox"/>		<input type="checkbox"/>
Weather	<input type="checkbox"/>		<input type="checkbox"/>
Wind Direction	<input type="checkbox"/>		<input type="checkbox"/>
Sound	<input type="checkbox"/>		<input type="checkbox"/>
Resource Painting (Harvestable)	<input type="checkbox"/>		<input type="checkbox"/>
Player Spawn (Starting points)	<input type="checkbox"/>		<input type="checkbox"/>
Dungeons	<input type="checkbox"/>		<input type="checkbox"/>



# 1. Decide on what needs to be optimized

It's easy to look at a region with a problem (low FPS or memory) and start loading the camp or heightmap level and jump into instancing and moving meshes from one heightmap to another.

Doing this is a mistake because the region with the problem isn't necessarily the region that has the problem. Adjoining heightmaps or camps within streaming distance other than the region may be the culprits.

Identifying where the problem lies can most easily be done by doing the following:

## Load the heightmap where the region is loaded

Load all the gameplay levels that intersect with the heightmap and ALL adjoining heightmap

For each of the levels loaded, we can now start looking into the world outliner to identify problems with the heightmap. This can take quite some time in order to do it correctly and meticulously enough to actually solve the problem, so in order to do this properly, we'll follow a few general guidelines.

*Important Note: Study the situation closely BEFORE doing optimizations. Read the steps below before actually jumping in and doing optimization so that you know ahead of time what to do and why.*

## Level-bounds and Streaming Distances

Level-bounds and Streaming distances are tightly linked and it's a good idea to do these two steps at the same time. You should also re-check these AFTER you have done the optimizations to make sure that these (in particular the level-bounds) stays intact.

### Check Level-bounds

The first step (and easiest) should be to check the level-bounds in the level. While doing this step, having the world composition window open is mandatory so you can get visual feedback for changes done.

*Note: Level-bounds for heightmap levels should never be set to "auto-adjust"*

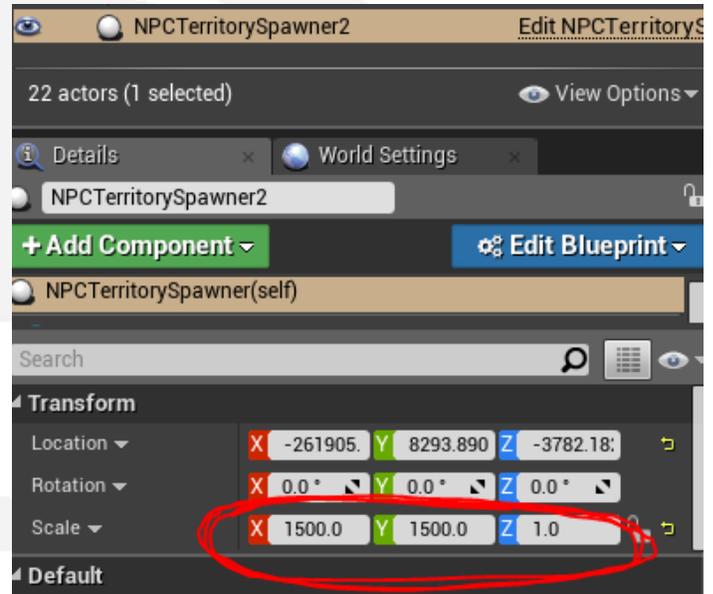
If the level-bounds has been set to NOT have "Auto-adjust", tick the checkbox and see if the level-bound is reduced in the World Outliner.

If the level-bound is not reduced, it may be that either the level-bound is fine but it might also be that something called "NPCTerritorySpawner" has been set up to be too large (more about this below)

If the level-bounds increase in size (especially if the change is significant), it might be that there are actors outside the level-bounds or Unreal thinks that there are.

## Common problem: Territory Managers are huge

This is fairly straight-forward to fix - in the level, search for "NPCTerritorySpawner" and inspect the Scale value. If this is a huge number (higher than 100x100x1), this might require adjustments. Typically, a territoryspawner should not have to be huge - generally speaking, it should cover the camp/wildlife/ whatever it spawns, but no more.



For Camplevels, I recommend reducing these numbers to 25x25x1 and moving the NPCTerritorySpawner to the middle of the camp.

For Wildlife levels where the wildlife is clustered, I recommend reducing these to 100x100x1 and moving the spawner to the center of the wildlife level.

For Wildlife levels where the wildlife is scattered, I recommend making this as big as it needs to be to cover the spawns - typically 500x500x1 should be enough.

### **Common problem: I made the territoryspawner small and now nothing spawns**

Territoryspawners do deactivate at a certain range called Network Relevancy range. This is why I mention in the above example for Wildlife levels that it's a good idea to make it cover the spawns. For camp-levels this should ideally not be a problem, because camps shouldn't be as large as to require this - however, it might be the case that they are, in which case, you should simply increase the size for the TerritorySpawner until things start spawning correctly.

### **Common problem: Auto-Adjust causes the level-bounds to expand**

A typical example of when this happens is when certain blueprints placed in the level, empty HISMA actors or actors that are simply outside the level when they shouldn't be.

#### **Identify the actors that cause this problem**

Mark actors in the level, CTRL-X them and see if the level-bounds adjust. If the bounds does not adjust itself, UNDO the cut function (do NOT paste the actors back in! See the Warning below) and redo this procedure until you have found the offending actors. Once the actors have been found, you can temporarily remove them, unclick "Auto-adjust" in the level-bounds and then paste the actors back in.

If the actor is a static mesh or skeletal mesh that is very far away from the level-bounds, it's almost certain that this mesh can simply be deleted, since it will never have been rendered in the game. However - you will need to evaluate if this asset should then be placed into a level where it seems to belong instead.

**Warning! If the offending actors are blueprints, odds are that these are hooked up to other blueprints. IF the offending actor is a blueprint, you should NOT CTRL-X and CTRL-V these assets. That will break the link. Instead - follow this procedure:**

1. CTRL-X the blueprints
2. Turn off auto-adjust
3. Copy the Location and scale for the level-bounds into notepad
4. Use the "Undo" function until the actors are back in the level
5. Turn off auto-adjust for the levelbounds
6. Copy the location and scale from notepad into the levelbounds

### **Common problem: The camp level is huge**

If the camp-level is very big, it might be necessary to split it up into multiple levels. When splitting up a camp-level, the following questions should be taken into consideration:

Does the camp consist of specific, separate smaller camps?

- If so, making individual camp-levels for each of these camps would be a good approach

Does the camp consist of separate sections that can be split up by themselves?

- If so, making individual camp-levels for each of these sections would be a good approach

Does the camp have huge wall sections or other occluders that prevents players outside the walls looking into the camp itself?

- If so, making a separate camplevel for these occluders specifically is a great approach. In addition, the camp might require a section split, but doing this will make the work a lot easier
- Make the camp-level that contains the walls/occluders have a higher streaming distance and the contents of the camp/ sections have a much smaller streaming distance. See "Check Streaming Distance" below

## Check Streaming Distance

Streaming distance is measured from the edge of a level's level-bounds to the player. If the player is within that range, the level loads. If the Streaming distance is very large, it means that the level, despite perhaps having a very small level-bounds, has a very long streaming distance when it doesn't need to have it.

Hovering over the level in the World Composition will give you the Streaming Distance of a level, like so:



Most commonly, you will see that camp-levels have been set to 10 000. This is what we have decided in the past is the optimal streaming distance for gameplay levels. Heightmap levels typically have been set to a higher value (50 000 or 100 000 for Siptah) to make sure that players do not see a sharp 'end of world' cut.

To find the correct streaming distance for a level, you should check the following:

### Is the camp-level split up into sub-levels?

- If so, is there one of these camp-levels that contain walls/occluders? If this is the case, this level should have a higher streaming distance than the camp inside it. This is the ideal situation and I recommend setting up the smaller levels inside the walls to 5 000 streaming range, and the outer wall levels to a higher band, such as 10 000 or 15 000 or even 25 000 if the walls should be seen from very far away.
- If not, can it be? If so, split the level up.
  - If it cannot be split up, which is a relatively common problem, optimization likely has very little to do with streaming distance, and instead you will need to look into other venues of optimization.

### Common problem: The camp "pops" into view when approaching it

This should be possible to solve by following the above solution where the camp is split up into an outer hull and inner camp levels and then setting the streaming distances for the hull level to be higher. In the case where a hull-level cannot be made, this camp will need to have the streaming distance increased. However - this will cause a memory hog, and the level should be looked at for other, strong optimization passes instead.

### Common problem: Lights do not show up from far away

A good solution to this is to move lights that are not part of blueprints into the heightmap level. If there are no such lights, the solution should be to split out these lights specifically to a level with a higher streaming distance. Something to keep in mind here is that moving camp-assets (BP\_CA\_xxxx actors) to a separate level also means that this new level should have a "CampOwner" and "NPCTerritorySpawner" blueprints in it. Ask a designer for these things or use the CampBlutility to hook them up if you know how.

### Common problem: There isn't a streaming distance preset that makes sense for this level

Create a new streaming distance in the World Composition view and assign the level to this new streaming distance.

## Moving assets between camp and heightmap levels

Assets that are placed in a heightmap level are loaded into memory from far away, given that the streaming distance of such levels are much higher than the streaming distance of gameplay levels.

Here are the guidelines for what heightmap levels should contain

- Cliffs and visual occluders such as acheronian ruin walls and other separators that affects the players visibility range.
- Fogplanes and regional particles that can be seen from very far away
- Water (and related water volumes, navmesh modifiers, etc) that does not span multiple heightmaps
- Heightmap+Foliage (and foliage blocker volumes)
- Blocker/Climb/Water/Audio volumes
- Anything else should be inside camp-levels or at least separate levels from the heightmap.

## “Hull-levels” - a definition

In the text going forward, I will be referring to a concept I call “Hull-levels”. I define a Hull-level as a level that contains a visual barrier/occluder between the player and a camp. An example of this would be a large wall surrounding an encampment, or a series of combined, smaller assets that provide a visual clutter that also works as a camp-delineation. Hull-levels should contain a minimum amount of assets but enough of them to provide a visual barrier for the player to prevent rendering most/all of the inside of a camp.

## Checking assets that are in the heightmap and camp-levels

Assets that do not match the list above should be considered for moving to a camp level, or at the very least, a level separated from the heightmap. It pays to understand why this is done:

A single asset placed in the heightmap will be loaded while in the heightmap as well as while the heightmap is streamed to a player. This means that the more unique assets we have in the heightmap, the more memory will be consumed by these assets.

Conversely, if you have a single asset in the heightmap, you may as well have all the other assets of the same type in the heightmap as well. This is very much an all-or-nothing system, and this is also the reason why I recommend splitting up camp-levels into Hull levels and Inner levels instead of putting the assets into the heightmap. That way, there’s no confusion about what goes where.

**Important note: There are a variety of other issues that might appear, detailed in the list below. Because some of these issues are not always easy to detect, I recommend going through each and every asset placed in the world to see if they match these issues and deal with them individually.**

- Actor is not visible because it’s set to be invisible
  - Sometimes, actors are set to be invisible (or hidden in game) to make use of collision. Assets like this should use cubes or other low-poly meshes. If the mesh is not a cube, cylinder or other primitive, this asset should be replaced with such
- Actor is not visible because it’s inside a mountain, under the ground or otherwise poorly placed
  - This actor should be deleted
- Actor is a skeletal mesh
  - This actor should ideally be placed in the camp-levels and also make sure that it truly is a required asset.
- Actor is a HISMA that is badly instanced
  - This actor should be de-instanced and properly re-instanced
- Actor is a DM (Destructible mesh)
  - This actor should be replaced with the SM (Static Mesh) version of itself
- Actor is a blueprint
  - A common mistake is to split levels and move some (but not all) blueprints from one level to another. This should not be the case. If an asset is a Loot-chest, spawn-point, waypoint, campowner, territoryspawner or a BP\_CA\_xxxx asset it should NOT be in a heightmap level. In addition, you should check the level if these have been correctly hooked up to a campowner/territoryspawner because without these hookups, the actor will not spawn (or in a best-case scenario, only spawn once)

## BP\_CA\_xxxx assets

When the Exiled Lands was created, a number of blueprint actors that contain multiple meshes, particles, sounds and the like were created. Some of these are very useful, whereas others should not be used anymore (at all).

The only BP\_CA assets we should use are the assets that have sounds, animations, functionality and/or particles attached to them. Decorative blueprints should NOT be used - they can't be instanced, and they tick when they don't need to.

In short - inspect the BP\_CA Assets in the Details Pane and if they only consist of static meshes and skeletal meshes, these should be replaced with their static mesh and skeletal mesh counterparts.

You can convert a blueprint into another asset by simply right-clicking the blueprint in the viewport and select "Replace selected actor with..."

### **Common problem: There are non-heightmap assets in the heightmap level but no camp-level for them**

These assets should likely be moved to their own levels. If the area is a unique location, such as a city, a set of ruins or otherwise contain large occluders, these are perfect assets to move to such levels. Check for these things:

- Can players climb tall cliffs nearby and look into the level? If this is the case, the assets should stay in the heightmap level but only if it feels strange that it cannot be seen.
- Is this asset providing a visual occluder to a camp? If so, this asset should either be part of the heightmap level or split into a level that acts like a hull around a camp to allow the camp level to have a smaller streaming distance.
- Is the asset very small? If so, it's very likely that the asset should be in the camp level instead of the heightmap level.
- Is the asset very heavy in terms of memory (for whatever reason such as loading a lot of textures)? If so, the asset should either be art-optimized and/or moved to the camp-level or removed completely. Typical assets that have a very high memory footprint are:
  - Weapons
  - Armors
  - Skeletal meshes for monsters
  - Crafting Stations
  - Marketplace assets

### **Common problem: There are non-heightmap assets in the heightmap level that provide visual occluders**

When it comes to large pieces that are used for this purpose such as the acheronian walls that surround the Unnamed City in the Exiled Lands, these assets can stay in the heightmap.

### **Common problem: There are assets of the same kind in both camp-levels and heightmap levels**

This is very common and ultimately, you will need to make a judgement call on which assets need to exist where. Ideally, the solution is to have all these assets in the same level, be it the heightmap or camp level.

If the assets are small, they should not be in the heightmap level and should be moved

On the other hand, if the asset is big, the asset should be moved from the camp-levels and into the heightmap level or a separate level that has a high streaming distance.

If the assets are occluders, they should be either in the heightmap level OR in a separate level that has a high streaming distance. Ideally, a separate level works best here.

### **Common problem: The asset is a unique asset**

Assets such as this require a judgement call. If the asset can be seen from very far away (For example, the Tower of Siptah) then it's probably best to split it into a specific level rather than linking it to the heightmap. If the asset is very small, it should definitely be in a camp-level. Other factors like occlusion and memory efficiency should also be considered.

### **Common problem: The actor is small or medium sized but placed in such a way that it can be seen from far away**

This happens a lot with crates, flags, banners, spikes and other assets that define the outer edge of a camp. These assets should be moved to a Hull-level if at all possible but no matter what, should not be part of the heightmap.

# **A WORD ABOUT PROCESSES**

You've now read through some/most/all of this document.

This won't necessarily help you much if you are not willing to experiment and—especially important—fail repeatedly. You shouldn't be afraid of experimentation and failure; this is one of the best methods to learn—trite but true.

One of the best resources you could hope to find is the Exiled Lands map. Copy settings from it—copy entire blueprints from it. Copy the post processing volumes, the Ultra-Dynamic Sky, and anything else you feel uncomfortable with.

It took us several days to figure out multiple small fine details about almost all aspects of map-making, and documenting everything we've learned is nigh impossible, so we heartily recommend learning as many shortcuts as possible—copying settings and then toying with them on your map will make you understand the links much better than any amount of text could ever do.

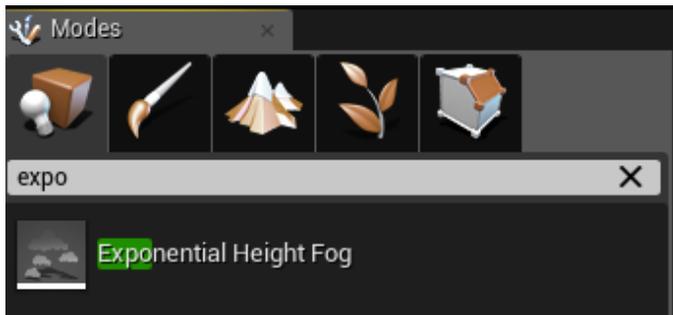


## **APPENDIX**

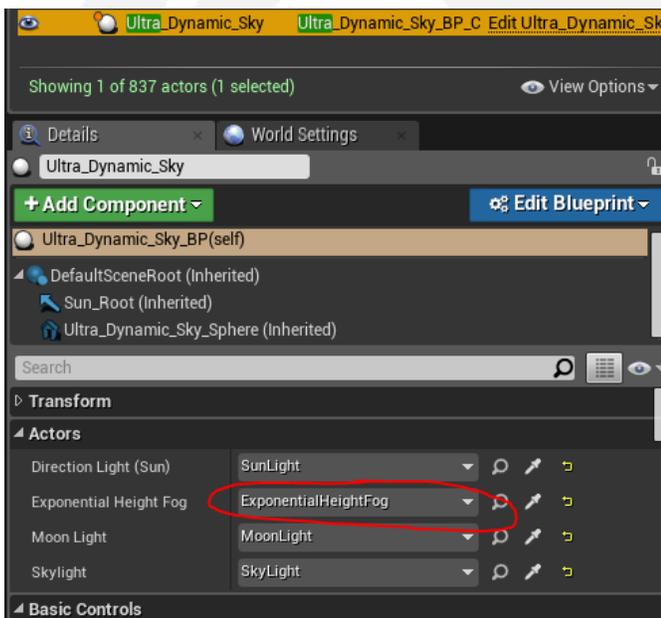
This appendix will cover some of the aspects of map-making that have only briefly been covered before. Please note that it is not very detailed because extensively detailing all functions and concepts of map-making will take a very long time, and a combination of research, online documentation and experimentation will likely teach you just as much.

## Exponential Height Fog

The easiest way to add exponential height fog to your map is to drag in an Exponential Height Fog actor from the modes panel:



This will then need to be linked to your Ultra-Dynamic Sky blueprint, as such:



Do note that the settings that are not exposed in the screenshot above are default.

To the right, you can find the settings we use on the Savage Wilds map, although you might want to set these settings up to suit your own map better.

You can read more about Exponential height fog here:

<https://docs.unrealengine.com/en-US/BuildingWorlds/FogEffects/HeightFog/index.html>

## Post processing areas

When making a map you will want certain areas to have different post processing effects, such as caves or even regions.

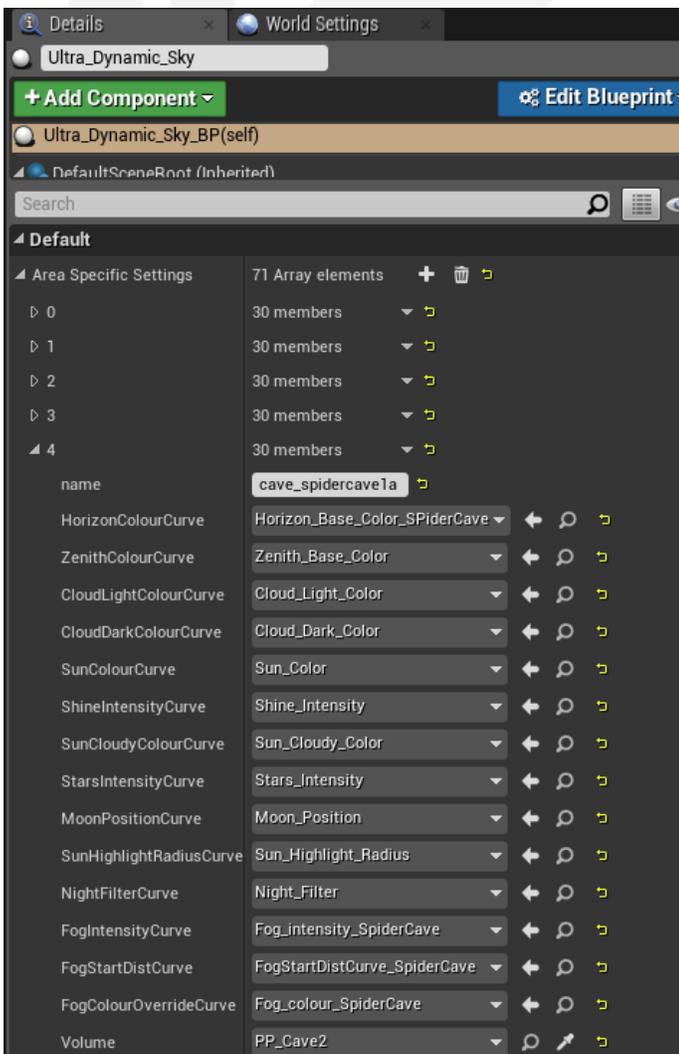
The Ultra-Dynamic Sky blueprint allows you to link post processing areas to post processing settings.

First, create a post processing volume in your Persistent Level (this is also where the Ultra-dynamic sky blueprint should reside). There is no need to set this volume up with settings for light and dark, etc—that's the next step. You will need to shape the volume to match the area though.

**You can read up more about working with volumes here:**

<https://docs.unrealengine.com/en-US/Basics/Actors/Volumes/index.html>

Next, find Ultra-Dynamic sky and in the properties of the blueprint, find the section called “Default” - this is an array of post processing volumes and each volume and looks like this:



If you are just starting up your map, and have copy-pasted the Ultra-Dynamic Sky from the Exiled Lands into your map, you will have a lot of freebies here. Each of these array elements correspond to a place in the Exiled Lands, so if you want the same settings as the Highlands, simply assign your post processing volume to the Highlands entry.

If you have multiple areas that each need settings that you've already used, or if you need more areas, you will need to set up new array elements to link to.

But even then—a lot of freebies can be had here. There are a lot of preset-settings for sky-color, fog-distance and so forth in this folder:

**/Game/Marketplace/UltraDynamicSky/Materials/**

You can use these presets to make your area look like caves, highland areas, the Unnamed City or any combination of the above.



## **ADDITIONAL RESOURCES**

Here are some helpful links to various resources online where you can get additional information, or resources, for map-making.

**A tutorial and an excellent set of heightmap brushes:**

<https://www.youtube.com/watch?v=1gZxjkl5YIA>

