

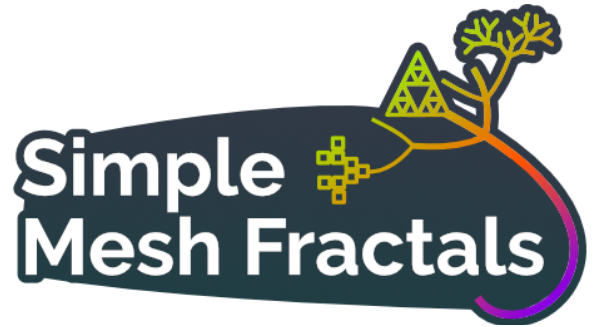
Unreal Engine 4 - Marketplace Asset

Simple Mesh Fractals

Developed by Divivor | Dennis Kastel

Documentation

Last update of this document: 20.03.2019



Description

Simple Mesh Fractals (in short: SMF) can generate interesting tree structures with your static meshes. Customize how many Children it has, how many Branches it can generate and even randomize the generation.

Features:

- Use any static mesh for generation
- Multiple meshes and children supported with own settings
- Gradually change size and rotation of children
- Custom branch amount and spawn conditions
- Branch rotation and offset
- Seed based randomization
- Randomized chance of child branch spawning and branch ending

Contents of this Documentation

Description	1
Contents of this Documentation	2
Important Information. Please read this before using this asset!	4
Getting Started	5
Converting a Fractal into a Static Mesh	12
Using converted Fractals as child elements	14
Parameters and Settings	15
Fractal	15
Fractal Child	16
Generation	17
Randomization	17

Important Information. Please read this before using this asset!

The Simple Mesh Fractals Asset is based on L-Trees and therefore uses a lot of For-Loops. This can directly affect the performance of the Unreal Engine Editor and may even end up crashing or freezing the Unreal Engine itself! The reason for this is that you can't have infinite loops(over ~1M instructions) inside the Engine.

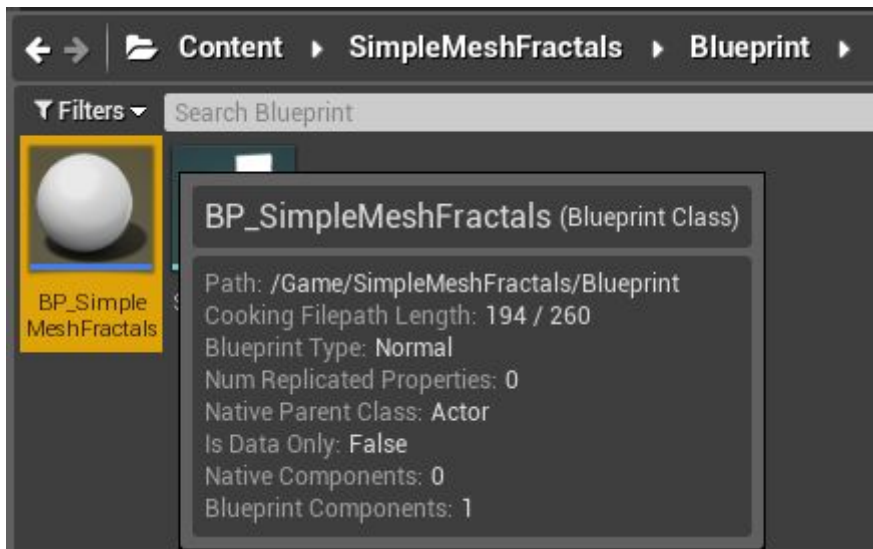
In order to avoid this problem you should make sure to always start with a low object "Amount" in the BP_SimpleMeshFractals and regularly save your project while editing the fractal generators. You can increase the object "Amount" with little steps if everything is working correctly.

Another thing to remember is that this asset is more on the experimental side. In comparison to other Marketplace Assets, the Simple Mesh Fractals isn't straightforward in most cases. Simple Tree like shapes can be done easily, but if you want to have more complex Structures and Fractals you will have to experiment with the parameters a lot. The "SMFDemo" Level has a lot of examples for different types of structures and will help you to understand how they are made.

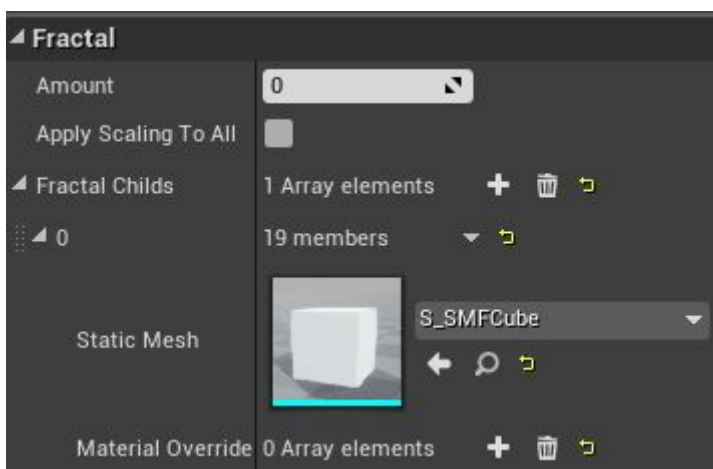
Getting Started

After you have read all the important information that is needed to use this asset correctly, you can start here with making a simple Tree like Fractal from a Static Mesh.

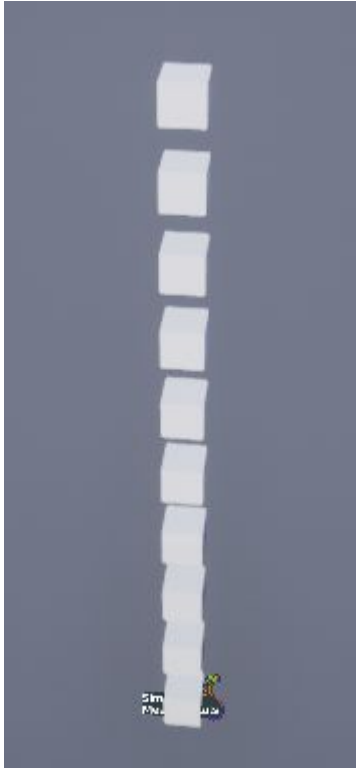
1. Place an Instance of the “BP_SimpleMeshFractals” Blueprint inside your level. It can be found under Content > SimpleMeshFractals > Blueprint > BP_SimpleMeshFractals.



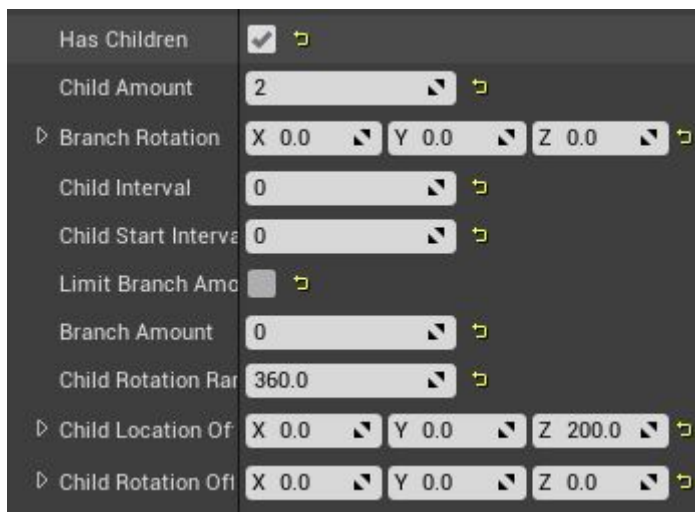
2. Add an element to the “Fractal Childs” Array and assign it a Static Mesh. In this case we are using a simple Cube.



- Let's increase the "Amount" to 10. You will now see a Cube being generated. All 10 of them are placed but invisible because they are overlapping each other. To change this edit the "Offset Location" Z Value to 200.

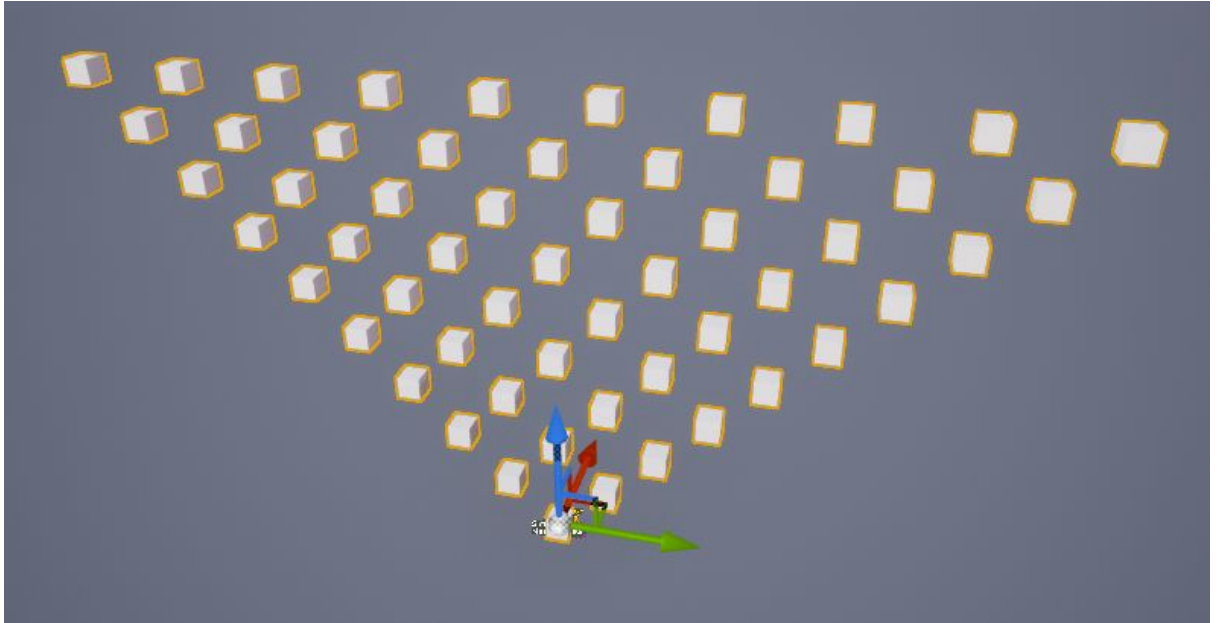


- Now let's add some child components. Enable "Has Children" and increase the "Child Location Offset" Z-Value to 200 in order to see results.

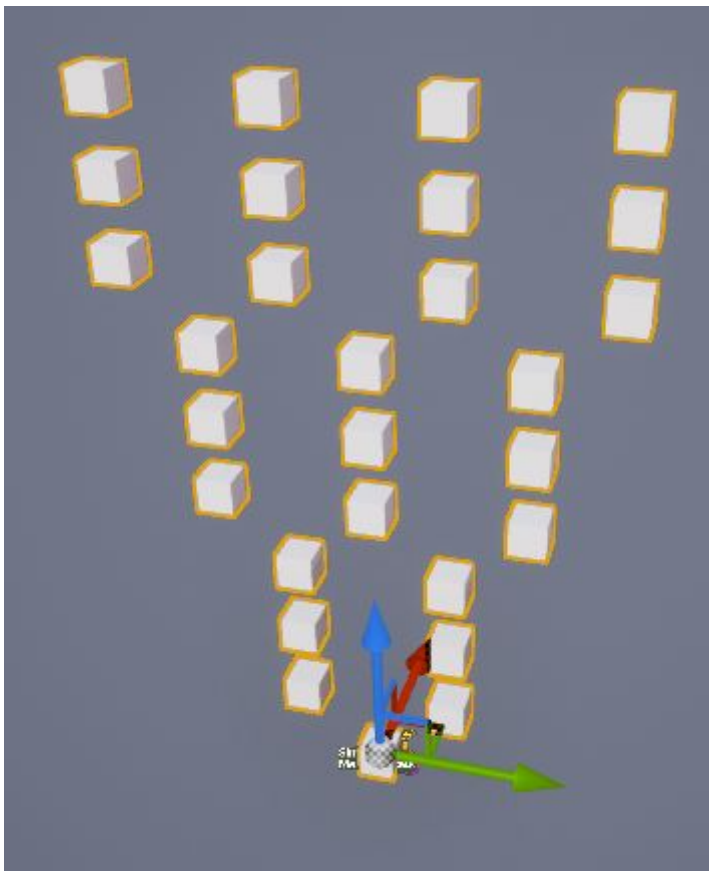


The result should look the same as before.

- Let's make the child elements visible by offsetting the center position of the child elements. You will have to change the X and Y Values of the "Child Location Offset" for this. In this example we will set the Y Value to 200.

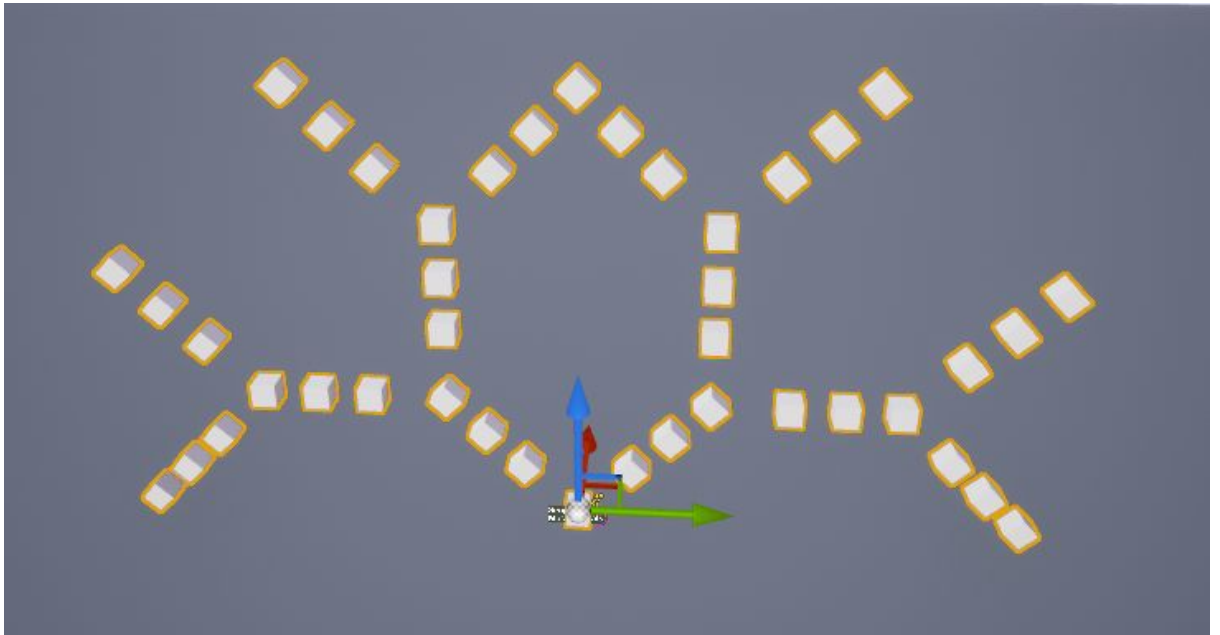


- Let's add a little variation. Set the "Child Interval" Value to 3!

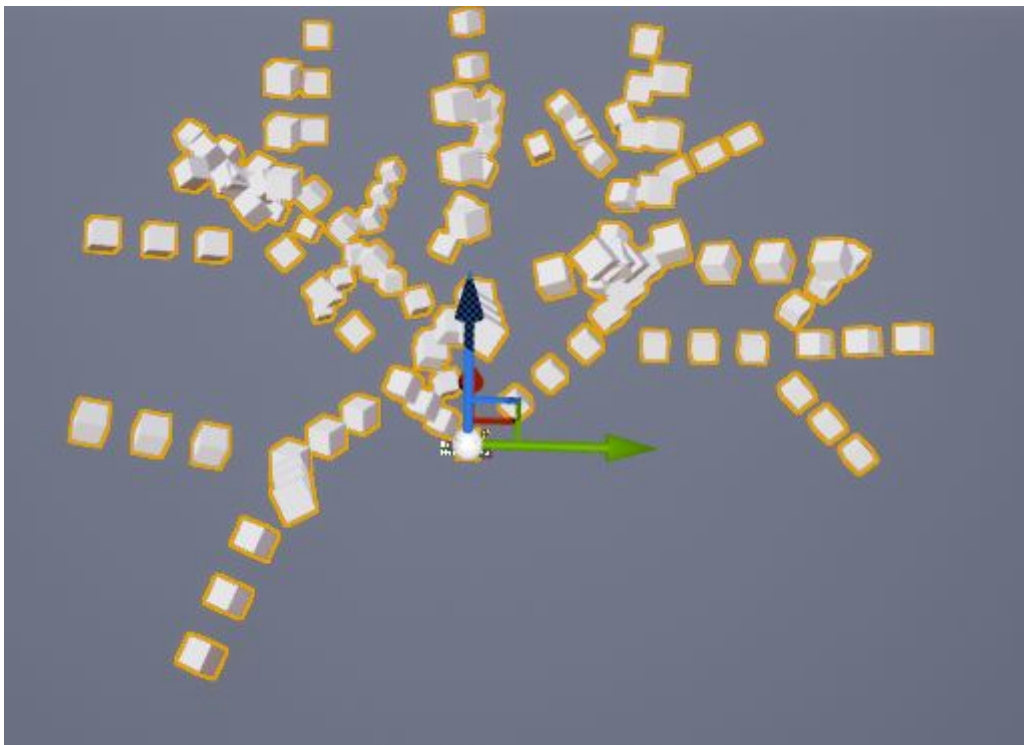


As you can see, this value is responsible for the branch distance.

- Let's rotate the branches by changing the "Child Rotation Offset" X Value to 45.

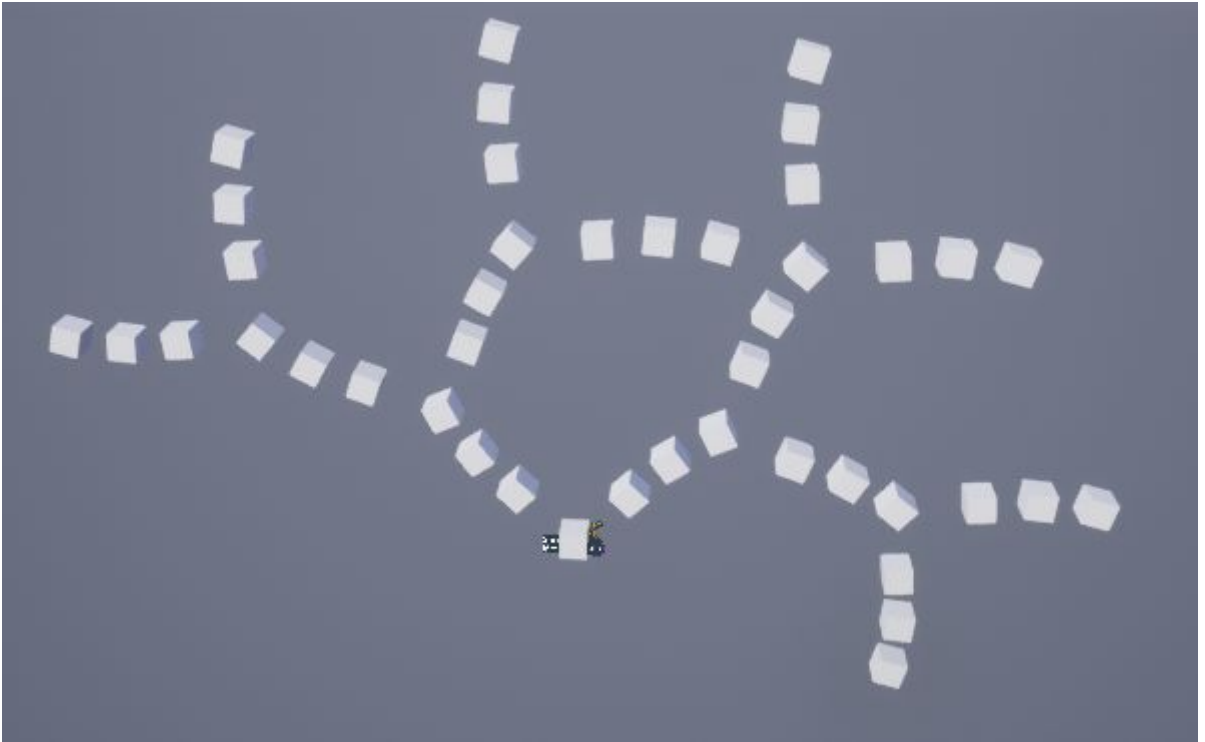


This only changes the orientation of the branches that are generated through childs. Let's change the "Child Amount" to 3 to see more possibilities.

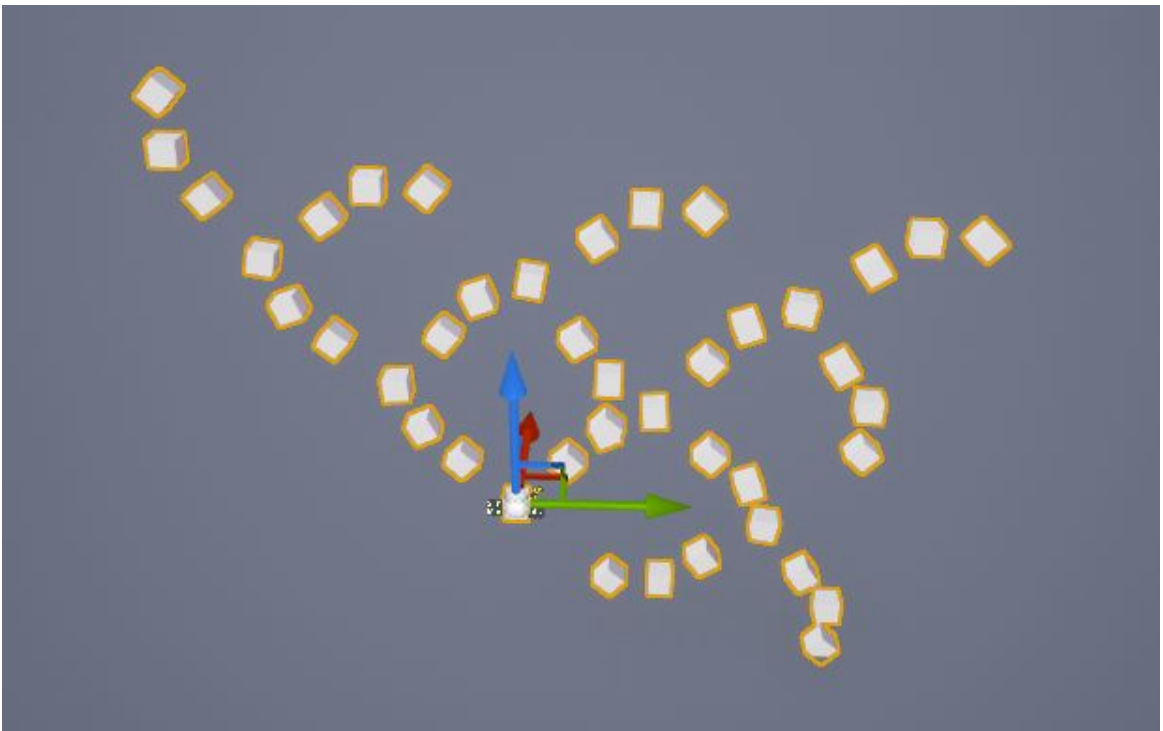


You can now see that it has entered the 3rd dimension. But it doesn't look very good for the moment. Let's change it back for the purpose of this tutorial.

8. We are now going to rotate the elements itself. Change the "Offset Rotation" X Value to 10 to get the following result.



We can increase the curvature with the travelled distance by setting an additional offset scaling to the "Offset Rotation Change" Value. We will change the X Value of this to 0.3.

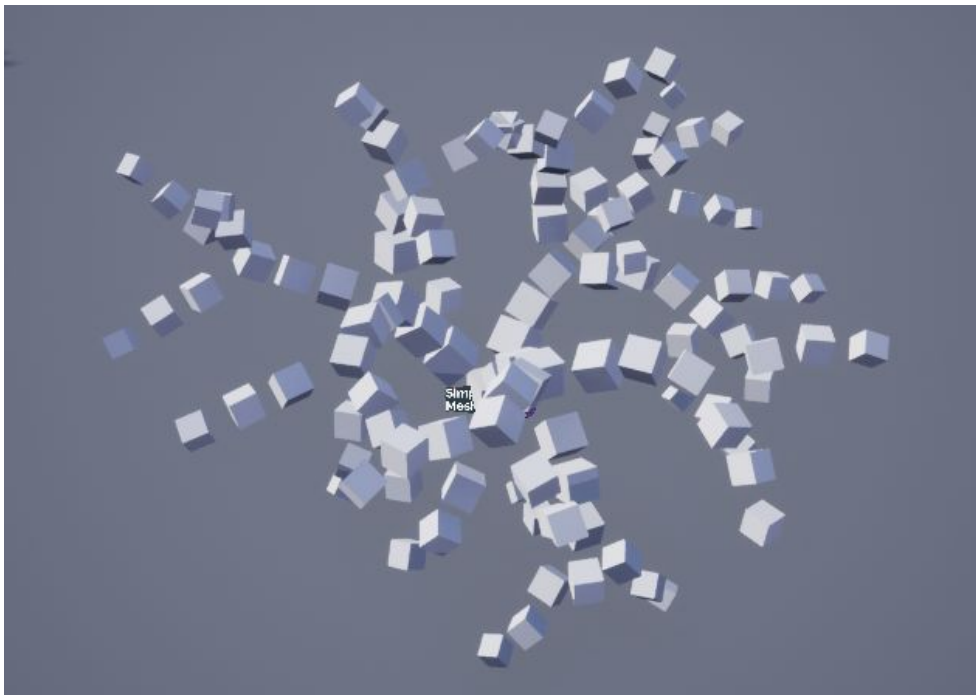


You can do this for location offsets as well.

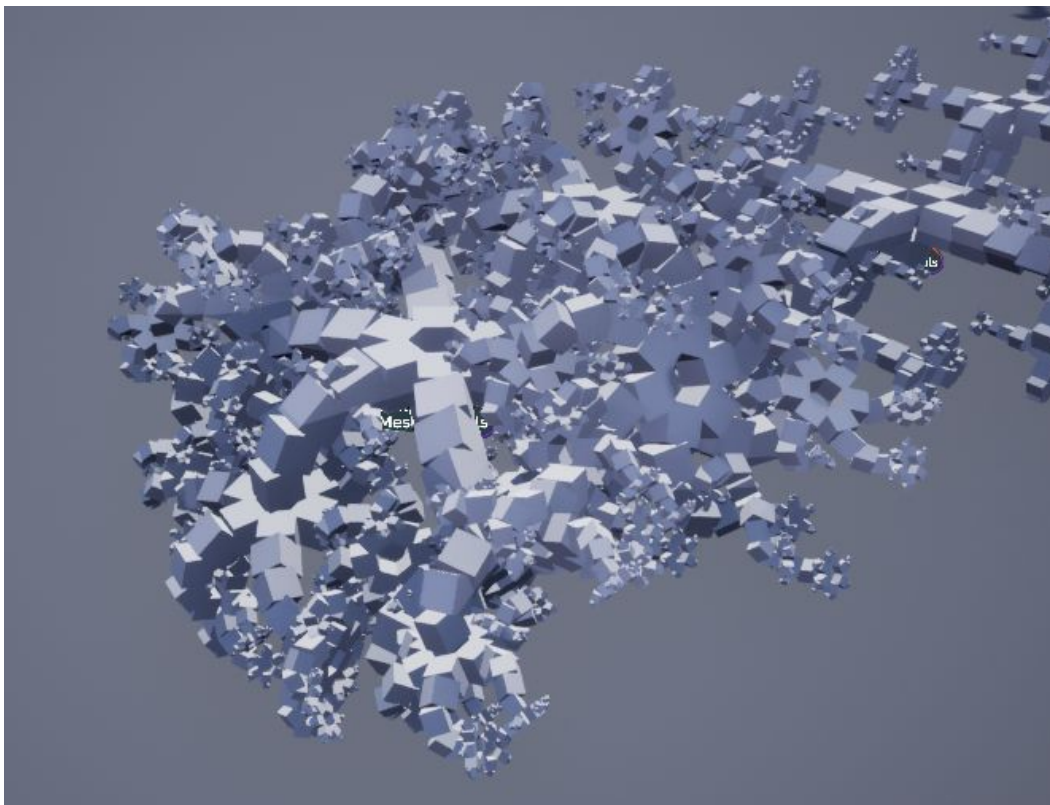
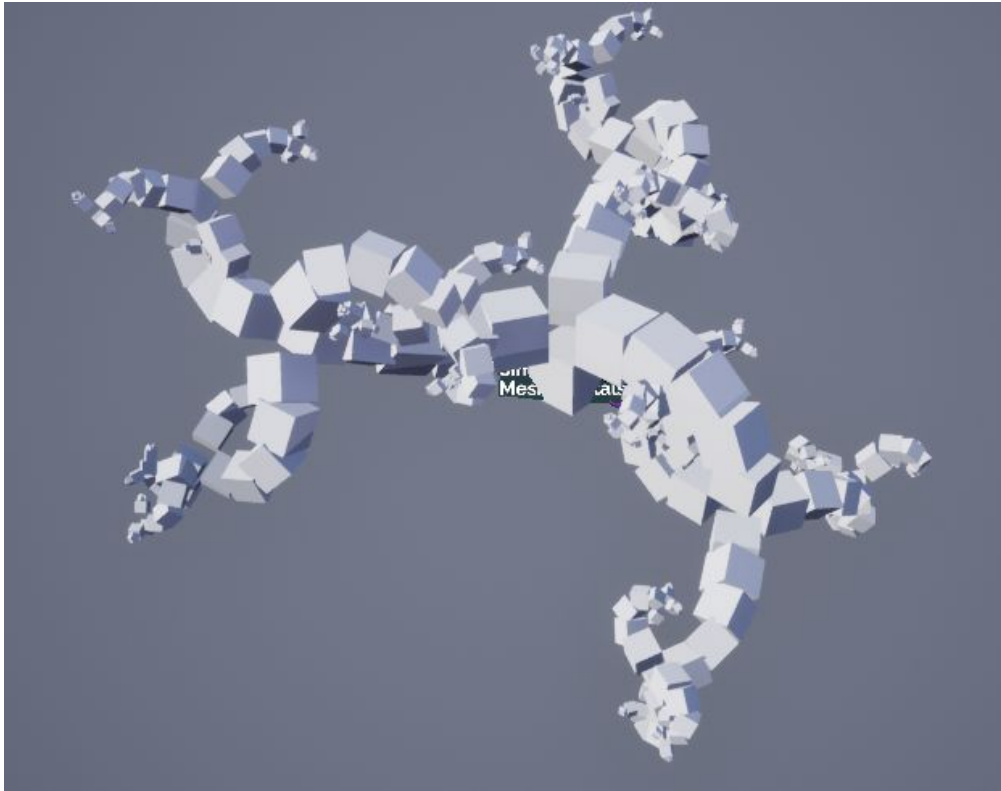
- At last we are going to change the scaling over distance with "Scale Change". Set the X Y and Z Value to -0.05.



For the last finishing touches we will decrease the offset locations to 100 and set the "Child Amount" back to 3. Also enable "Apply Scaling To All" for the best result in this case.



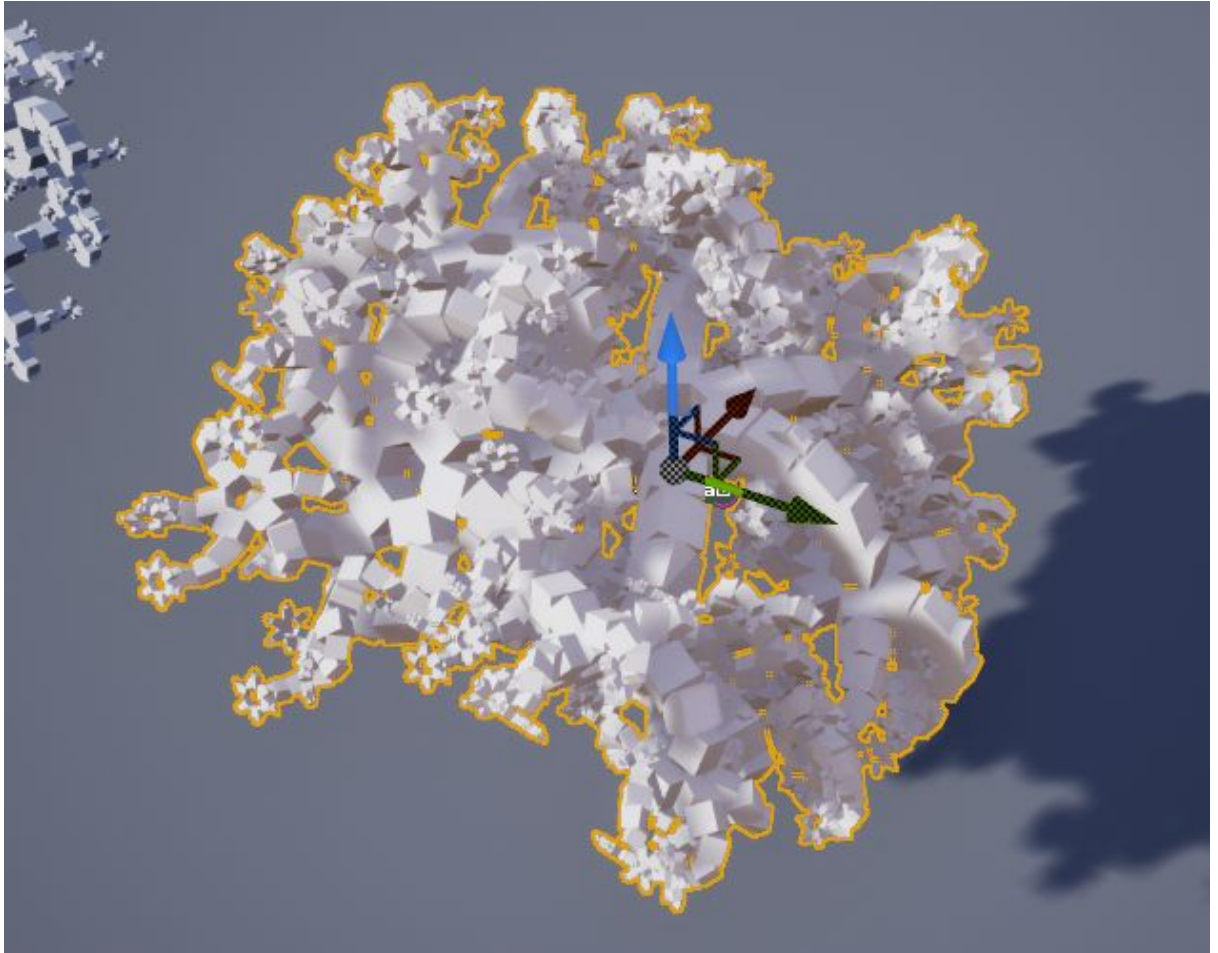
You now know the basics! Start experimenting with all the values and parameters and try to see what kind of structures you can generate with this.



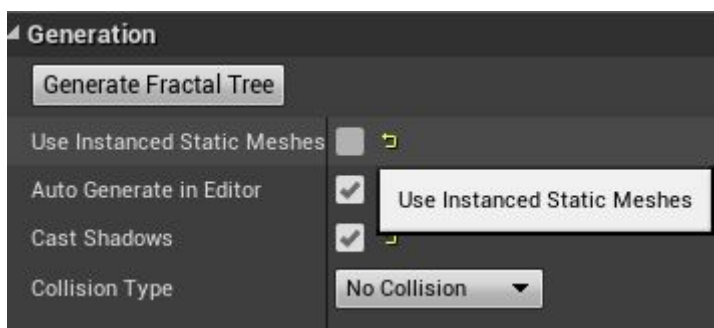
Converting a Fractal into a Static Mesh

The Simple Mesh Fractals Asset can also be converted into a Static Mesh. This tutorial describes how you can do it.

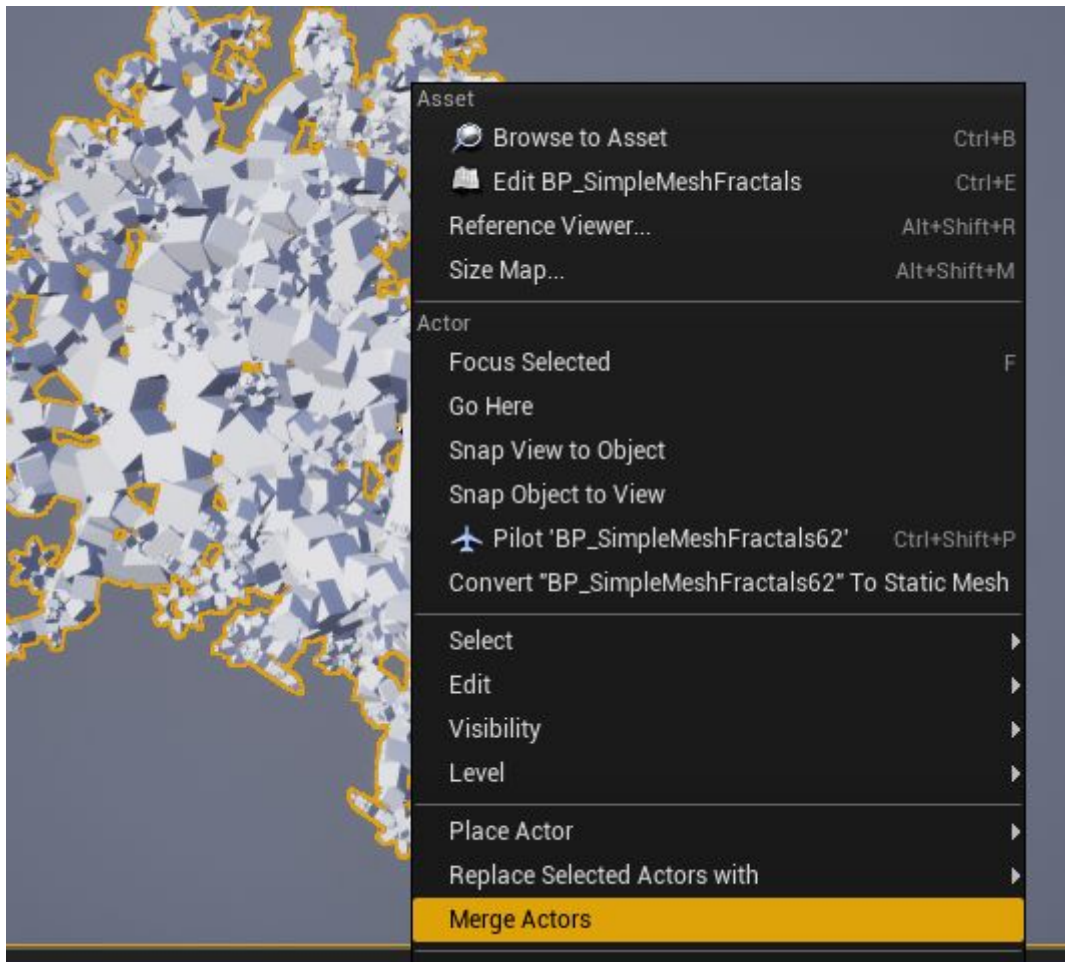
1. Select the BP_SimpleMeshFractal Instance that you want to convert.



2. Disable “Use Instanced Static Meshes” under “Generation” in the Details Panel of the actor. Please note that this action generates Static Mesh Components instead of using the Instanced Static Mesh Component. This can decrease your overall performance.



3. Right-Click your Actor in the Level and select "Merge Actors".



You should now see window with a list of Mesh Components.

4. Click on the "Merge Actors" Button in this new window and select a place to save your Mesh.



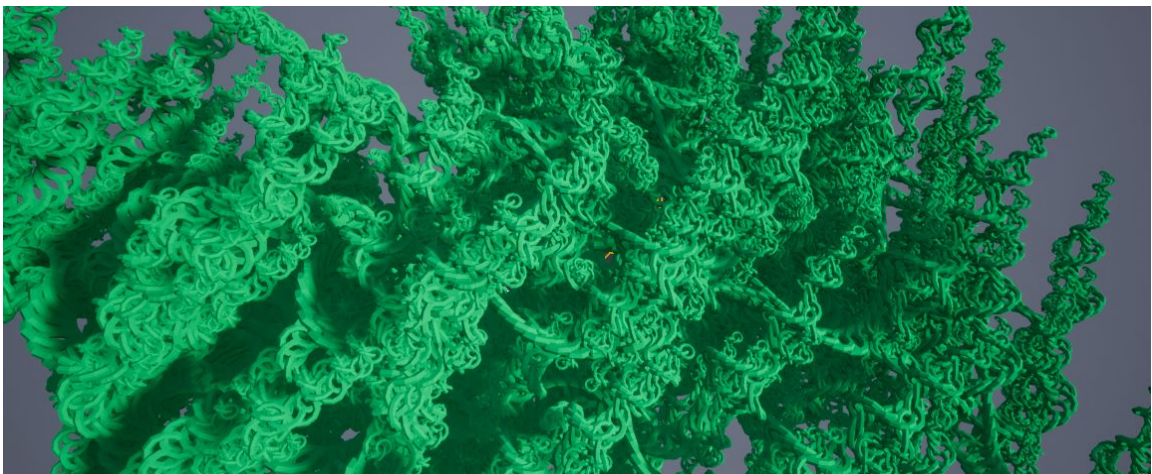
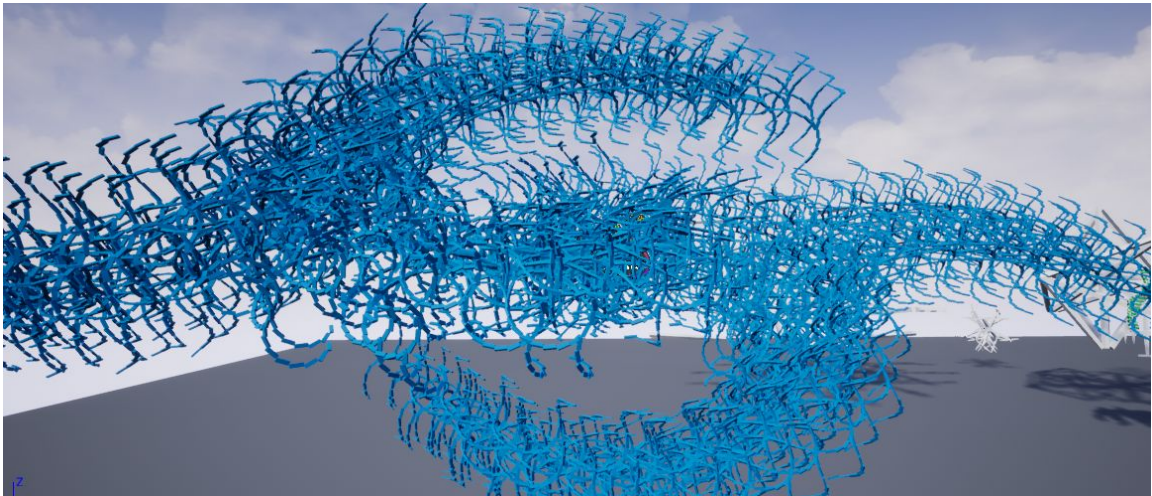
This action can take up several minutes depending on the size and complexity of your Fractal.

For more information on Actor Merging, please visit the Unreal Engine Documentation at <https://docs.unrealengine.com/en-us/Engine/Actors/Merging>

5. You can now close the Actor Merging Window.
Your new Static Mesh made out of the Fractals should now be saved and available for use in your levels.

Using converted Fractals as child elements

Here is a little trick to help you increase your Fractal complexity and size.
After converting a Fractal into a static mesh, you can use this Static Mesh inside the
BP_SimpleMeshFractals!



Parameters and Settings

Fractal

- Amount
Amount of object levels. Example: If the value is set to 10, you will be spawning 10 child elements. If those child elements have their own children, the spawned amount will multiply with each branch.
- Apply Scaling To All
Apply changes of object scaling to location and rotation offset values.
- Fractal Childs
An array of elements that contain data for fractal generation.

Fractal Child

- Static Mesh
The Static Mesh that is being used for Fractal generation.
- Material Override
An array of Materials to override the Static Mesh Materials.
- Rotation
Default rotation of the Mesh.
- Scale
Default scale of the Mesh.
- Scale Change
Additional scale multiplier. This gets added with each iteration and fractal level.
- Offset Location
Default offset location between objects.
- Offset Location Change
Additional offset location multiplier. This gets added with each iteration and fractal level.

- **Offset Rotation**
Default offset rotation between objects.
- **Offset Rotation Change**
Additional offset rotation multiplier. This gets added with each iteration and fractal level.
- **Has Children**
Enables the branch/child spawning of this element.
- **Child Amount**
Amount of children/branches of this element.
- **Branch Rotation**
Additional rotation of the branch.
- **Child Interval**
Distance interval for child spawning of this element. Example: If the value is set to 5, every fifth element generates child branches.
- **Child Start Interval**
The start interval of the child branches. This option can delay the generation of branches.
- **Limit Branch Amount**
This option limits the amount of child branches in a row. This option is relative to the interval parameters.
- **Branch Amount**
The amount of branches in a row. See "Limit Branch Amount".
- **Child Rotation Range**
The range in which the child branches get spread.
- **Child Location Offset**
Similar to "Offset Location" but for generated child branches of this element.
- **Child Rotation Offset**
Similar to "Offset Rotation" but for generated child branches of this element.

Generation

- **Generate Fractal Tree - Button**
A Button to generate the mesh fractals. This is useful if you disable "Auto Generate in Editor".
- **Use Instanced Static Meshes**
Enabled by default. If enabled the SMF generates the fractals with Instanced Static Meshes, which can improve performance. When disabled the SMF will use Static Mesh Components, which can also be converted into a single static mesh.
- **Auto Generate in Editor**
Enable this to auto generate the mesh fractal trees while editing or moving.
- **Cast Shadows**
Enable this to cast shadows from the meshes.
- **Collision Type**
The collision type of the meshes. By default deactivated to improve performance.

Randomization

- **Random Seed**
A generation seed if you are using randomization.
- **Chance Of Child Spawning**
The chance of Child Spawning of elements in percentage from 0 to 1.
- **Change Of Branch Ending**
The chance of Branch Ending of elements in percentage from 0 to 1.