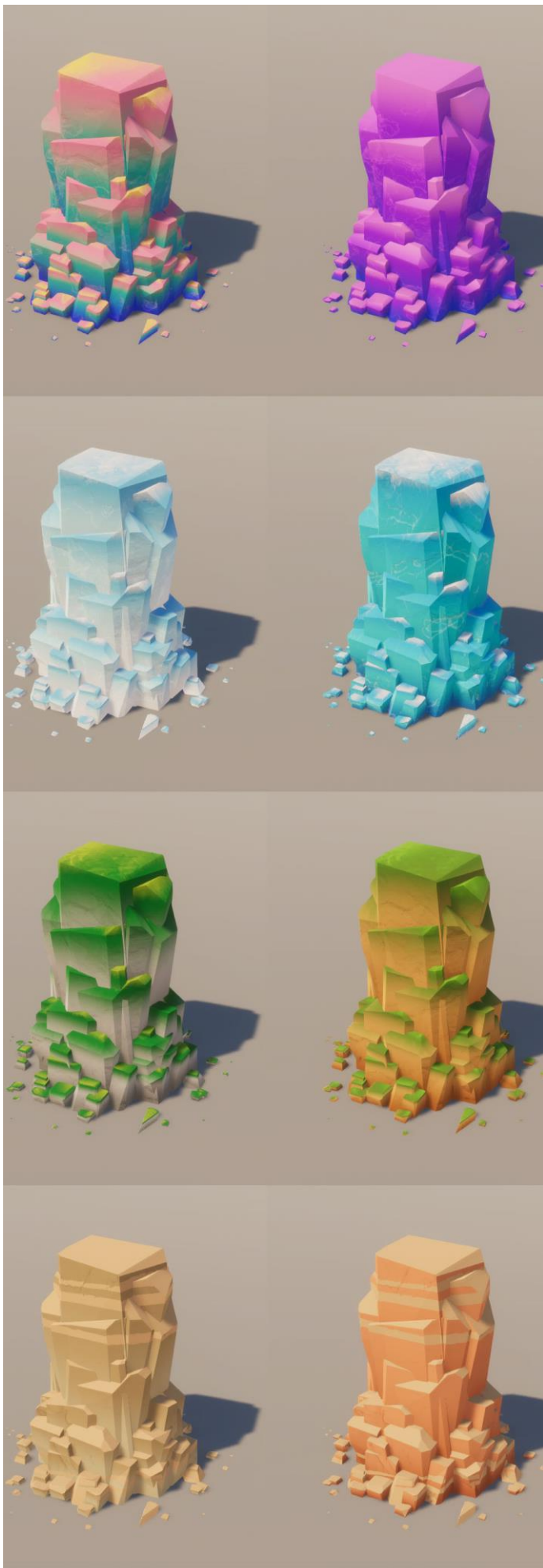




TITAN

ROCK GENERATOR

Documentation (1.2.0)



Contents

Introduction.....	3
How To Use.....	4
Shaders & Materials	7
FAQ.....	8
Contact Us.....	10

Changelog

V (1.2.0)

- Added a new crystal rock type.
- Added support for importing custom meshes.
- Improved the rock generator logic.
- Added random seed number (rocks can be regenerated using the same random seed).

V (1.1.0)

- New rock type to generate new forms.
- Support for the built-in render pipeline.
- Support for runtime rock generation.
- An option to automatically add collider.
- An option to crop the lower part of rocks.
- Meshes are exposed and could be replaced.

INTRODUCTION

Thank you for purchasing the **Titan Rock Generator** asset. This document shows how the tool works and helps you make the best. There is a FAQ section at the end of this document. However, if you have any questions, please contact us at support@dot-products.com. You can also always find us at www.dot-products.com.

What is the Titan Rock Generator?

As you may already know, this is an editor tool for creating stylized rocks inside Unity with a few simple steps. This tool lets you change parameters and see a real-time preview of your changes. Above that, you can select from our custom-made materials or use your materials to accomplish your desired look and feel.

Package contents

Each project has its requirements regarding art style. The main goal of this asset is to allow users to create customized rocks. But like many other assets, we provided several demo scenes that you can look at. They are all nested under a folder called **demo**.

You can delete the demo folder if you want.

Render pipelines

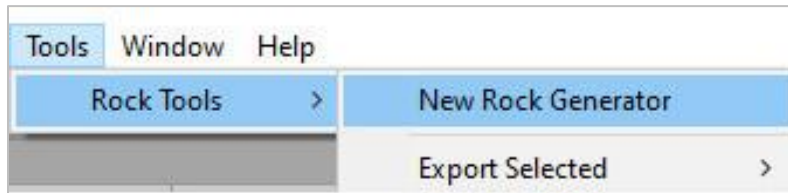
The package includes various shaders and materials designed to be used in all render pipelines. The Standard (built-in) render pipeline, The Universal render pipeline (URP), and the High definition pipeline (HDRP).

After importing the titan rock generator package, you should be able to see demo scenes that have been set up for the built-in render pipeline. If you are using URP or HDRP in your project, you should open a file inside the package named URP or HDRP. These packages include demo scenes that has been made especially for each rendering pipeline.

HOW TO USE

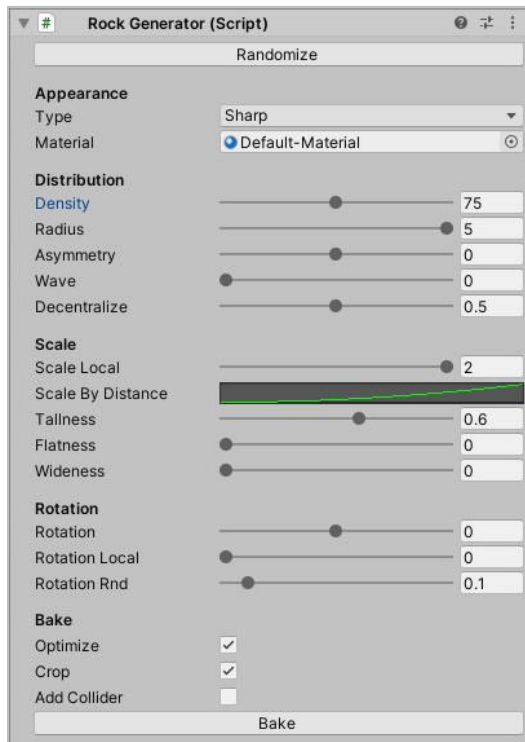
Getting started

To start generating rocks, you need to add a Rock Generator to the scene. So go ahead and add a new one from the tools menu: Tools > Rock Tools > New Rock Generator.



Rock generator parameters

You can customize your rock by changing parameters on Rock Generator's component.



Rock Shapes

Right now, we have three rock types to choose from. The **Cubic**, **Sharp**, and **Crystal** types.



Baking

When satisfied with your rock shape, press the *Bake* button to optimize and save the generated rock mesh. *You have to save the scene before you can bake the rock.*

The baking process shouldn't take longer than a few seconds. It does some operations such as saving the generated rock mesh in a folder next to your scene file.

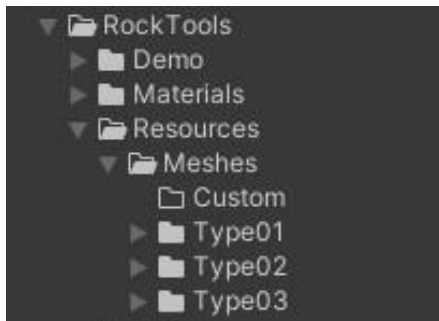
After baking, you can delete the rock generator from the scene if you have done with it.

Important note on deleting the Baked Rock

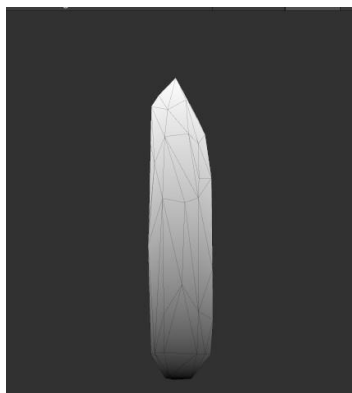
Please note that if you delete a baked rock from the scene, the related mesh remains in the Meshes folder, where the scene is saved. If you do this process several times, there will be several redundant, unused meshes in your project in the Mesh folder. So the best solution is that you first delete the connected mesh and then delete the rock game object in the scene. (Be sure that the mesh file isn't in use in another scene).

Use Custom Meshes

If you want to use custom made rock meshes as input for parts of your rock, you can select the custom type from the inspector while the rock generators object is selected in the scene. With this option you can import your custom meshes to unity and put them in the resources path in `RockTools > Resources > Meshes > Custom`.



To use our rock materials, ensure that your custom meshes have vertex colors that range from black at the bottom (0) to white at the top (1). The colors of the vertices in between should be a gradient blend between black and white, as shown in the example below:



Export the Rock Mesh

After baking the rock, you can export the mesh for further enhancements (OBJ format). You can find it on Tools > Rock Tools > Export Selected.

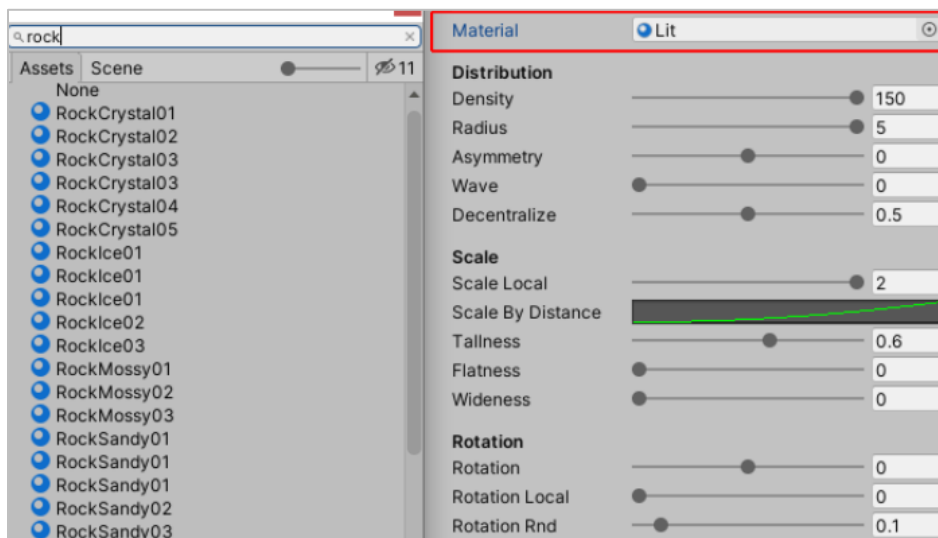
However, please note that exporting the mesh will result in the loss of data stored in the vertex colors (the data used in the rock shader). As a result, the rock shaders will not function correctly on meshes after reimporting them into Unity.

SHADERS & MATERIALS

The package includes seven Base shaders and 19 rock variation materials (mossy rock, sandy rock, snowy rock, Crystal, etc.). You can also change shaders in Shader Graph.

Rock shaders use tri-planar projection instead of UV maps. So you can assign one material to all rocks in the scene, which helps reduce draw calls.

After adding a rock generator to the scene, you can assign a material from the Material field.



Some tips on rock materials

- Before using materials, we recommend using a copy of the primary material. Leave the main material parameters unchanged for future use.
- The shaders are UV-independent. We use the World Space Coordinate system to handle textures.
- We use pre-stored data from rock meshes vertex color in rock shaders. So, rock shaders only work on rock generators.
- All shaders were made in the Shader Graph (for URP & HDRP) and Shader Forge (for built-in render pipeline). So you can make changes in every one of them. But make sure you have a copy before changing.
- All shaders use two types of texture: An ambient occlusion map and a normal map.

Lightmapping & Custom Materials

Unfortunately, we do not support light maps with our custom materials at this time. Currently, the only way to use lightmapping is by using a built-in Unity material, the Standard Shader, or a shader with a Meta Pass for the generated rock. These materials can read baked light maps. In future updates, we might incorporate lightmapping alongside our custom materials.

FAQ

Can it also make highly detailed rocks?

The primary purpose was to generate rocks as simple and low detailed as possible.

Are you able to export the generated rock as a single mesh?

Yes, you can export rock mesh after the mesh Baking process. Currently supports OBJ as export format.

Are your materials built using the shader graph?

Yes. All shaders are developed using the Shader Graph, and the source is also included in the package; and the built-in shaders are developed using Shader Forge.

Do the rock meshes have a UV map or do they use a tri-planar shader?

Our rock shaders use a tri-planar coordinate to handle patterns. However, we generate a UV map for the rock to use for other purposes such as light baking.

Are the rocks compatible with custom shaders/materials?

You can use your custom shader. But we recommend using our tri-planar shaders and changing them in the Shader Graph to desire your look.

Is that a custom height-based shader?

Yes, the shader is custom. We store data in vertex color that is used later in our shaders to find peaks of the rocks.

How does it handle mesh generation? Can I use it in production, or will it hurt performance?

Performance is a critical topic for us. The final rock is the most optimized possible. As this is a stylized rock generator, we used simple low poly meshes as rock parts, and it has no cost in runtime.

Is it an entire mesh generated at runtime or an extensive collection of prefabs that are just transformed?

We have a collection of different shapes of rock meshes. After that, it's just placement and transforming.

Is it possible / will it ever be possible to modify the rock's settings (Density, Radius, etc.) through code, or is it pre-runtime only?

It's possible to generate and even bake the rock; but some functionalities such as generating UV maps will not be available in runtime.

After generating a complicated rock, your algorithm cuts intersect meshes?

At first, we did that. But as we developed this tool, we came across this as a big challenge because cutting intersections add so many vertices to the mesh. The only benefit was a cleaner or more optimized UV map. And as our shaders didn't use UV, we decided not to cut the intersections.

Can we edit the rock in runtime?

We do not have a specific API for editing the rock in runtime, but the codes are clear, and with a bit of programming knowledge, you can perform all operations in runtime.

Can you animate the parameters?

It's not possible right now, but it could be with some changes.

CONTACT US

Thank you for using our asset. If you need any help using the asset or have any comments, we always love to hear from you. Feel free to send us a message, and we will get back to you ASAP.

Email support@dot-products.com

Website www.dot-products.com