Terrain generation

Overview

Generation of the simulation’s game world and its terrain is handled through a dedicated MapGenerator class. At the start of the game the class generates the ground surface, a road through the area, bordering mountains to block access to outside of the play area and vegetation for the player to scan and see in the simulation. The class uses a perlin noise -based algorithm to procedurally create the height variations of the world and borders. The class has a seed parameter that can be changed in the Unity editor to change the layout of the generated world. The class has a variety of other parameters that can be changed in the code which affect the generated world size, borders, road and form. By default the values are set to a reasonable level and shouldn’t require changing unless specifically wanted. The MapGenerator class also uses VegetationGenerator, MapDisplay, MeshGenerator and TextureGenerator classes to divide parts of the terrain generation process.

MapGenerator key features

Terrain Generation:

- Uses Perlin noise to generate a height map for the terrain.
- Supports configurable parameters like mapWidth, mapHeight, noiseScale, heightMultiplier, and more.
- Includes a cubic easing function to smooth transitions near the map's borders.

Road Generation:

- Creates a simple road running vertically through the map.
- Smoothly blends the road into the terrain using easing functions.
- Marks road positions on a roadMap for further processing.

Boundaries Generation:

- Automatically generates walls around the map to act as boundaries.
- Walls are created or updated dynamically based on the map's dimensions.

Rendering Modes:

- Supports two rendering modes: NoiseMap (visualizes the noise map as a texture) and Mesh (renders the terrain as a 3D mesh).
- Dynamically loads materials for terrain, roads, and mountains.

Vegetation Integration:

- Integrates with a VegetationGenerator to add vegetation to the terrain.
- Clears and regenerates vegetation when the map is updated.

Validation:

- Ensures that parameters like mapWidth, mapHeight, noiseScale, and others are clamped to valid ranges using the OnValidate method.

Key Methods

GenerateMap():

- Main method for generating the terrain.
- Creates the noise map, applies height adjustments, generates roads, and renders the terrain.

GenerateRoad():

- Generates a vertical road on the terrain.
- Smoothly transitions the road into the surrounding terrain using easing functions.

CreateBoundaries():

- Creates walls around the map to act as boundaries.
- Dynamically adjusts wall size and position based on the map dimensions.
- Uses helper function CreateWall to make individual walls

SetDefaults():

- Resets all parameters to default values.

OnValidate():

- Ensures parameter values are valid and triggers map generation if autoUpdate is enabled.

Fields and Properties

Public Fields:

- seed: Random seed for noise generation.
- autoUpdate: Automatically updates the map when parameters are changed.

Private Fields:

- renderMode: Determines the rendering mode (NoiseMap or Mesh).
- lastWarningTime and warningCooldown: Prevent spamming warnings in play mode.

Inspector Fields (Hidden):

- Terrain dimensions: mapWidth, mapHeight, borderWidth, roadWidth.
- Noise parameters: noiseScale, octaves, persistence, lacunarity.
- Materials: terrainMaterial, roadMaterial, mountainMaterial.
- Vegetation: vegetationGenerator.

Usage

- Attach the MapGenerator script to a GameObject in Unity.
- Configure the parameters in the Inspector or use the SetDefaults method.
- Call GenerateMap() to create the terrain.

Summary

The MapGenerator class is a robust and flexible tool for procedural terrain generation in Unity. It combines noise-based algorithms, road generation, and boundary creation to produce detailed and customizable terrains. Its integration with vegetation and rendering options makes it suitable for a wide range of applications, such as games or simulations.