Terrain Engine 2D

A 2D Block Engine for Unity

Out now on the Unity Asset Store

BUY NOW!

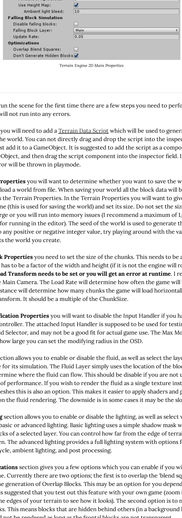
Main Properties

This page explains how to setup all of the Main Properties in the World custom inspector.

Table of Contents

General

The Main Properties tab in the custom inspector holds all the most important properties which you need to have access to in order to setup your terrain. It contains fields from multiple classes connected to the World for easy access.



Terrain Engine 2D Main Properties

Setup

Before you run the scene for the first time there are a few steps you need to perform to ensure you will not run into any errors.

For starters you will need to add a Terrain Data Script which will be used to generate the terrain for the world. You can not directly drag and drop the script into the inspector field, you must first add it to a GameObject. It is suggested to add the script as a component of the World GameObject, and then drag the script component into the inspector field. If this is not added an error will be thrown in playmode.

In **Terrain Properties** you will want to determine whether you want to save the world you generate or load a world from file. When saving your world all the block data will be saved to file as well as the Terrain Properties. In the Terrain Properties you will want to give your world a unique name (this is used for saving the world) and set its size. Do not set the size of the world too large or you will run into memory issues (I recommend a maximum of 1,000,000 blocks total for running in the editor). The seed of the world is used to generate the terrain, it can be set to any positive or negative integer value, try playing around with the values to see how it affects the world you create.

In the **Chunk Properties** you need to set the size of the chunks. This needs to be an even number and has to be a factor of the width and height (if it is not the engine will round it for you). **The Load Transform needs to be set or you will get an error at runtime**. I recommend it be set to the Main Camera. The Load Rate will determine how often the game will load chunks. The Load Distance will determine how many chunks around the load transform will be loaded. The Load Transform. It should be a multiple of the ChunkSize.

In the **Modification Properties** you will want to disable the Input Handler if you have your own input controller. The attached Input Handler is supposed to be used for testing with the OSD and Grid Selector, and may not be a good fit for actual game use. The Max Modify Radius determines how large you can set the modifying radius in the OSD.

The **Fluid** section allows you to enable or disable the fluid, as well as select the layer which fluid will use for its simulation. The Fluid Layer simply uses the location of the blocks in that layer to determine where the fluid can flow. This should be disable if you are not using fluid for the sake of performance. If you wish to render the fluid as a single texture instead of chunks of meshes this is also an option. This makes it easier to apply shaders and perform post processing on the fluid rendering. The downside is in some cases it may be the slower option.

The **Lighting** section allows you to enable or disable the lighting, as well as select whether you wish to use basic or advanced lighting. Basic lighting uses a simple shadow mask which will hide the blocks of a selected layer. You can determine how far from the edge of terrain blocks will be hidden. The advanced lighting provides a full lighting system with options for a day/night cycle, ambient lighting, and post processing.

The **Optimizations** section gives you a few options which you can enable if you wish to speed up your game. Currently there are two options; the first is to overlap the 'blend squares' which optimizes the generation of Overlap Blocks. This may be an option for you depending on your textures. It is suggested that you test out this feature with your own game (zoom in and look closely at the edges of your terrain to see how it looks). The second option is to not generate hidden blocks. This means blocks that are hidden behind others (in a background layer for example) will not be rendered as long as the frontal blocks are not transparent.

Objects

- Toggle OSD Toggle for enabling/disabling the OSD - Accessible at runtime
- OSD Update Rate The rate at which the OSD updates its values (in seconds)
- Terrain Data Script The custom script used for procedurally generating the world
- Toggle Cursor Show or hide the cursor in game - Accessible at runtime

Terrain Properties

- Save World Saves the generated terrain to file in playmode
- Load World Loads terrain from file in playmode
- Select World Directory Opens up menu for selecting the directory of the world for loading
- Width The total width of the world (in block units)
- Height The total height of the world (in block units)
- Seed Integer value used to procedurally generate the world
- Random Seed Randomly generate a seed

Chunk Properties

- Chunk Size The side length of a chunk (in block units)
- Load Transform The Transform of the GameObject where chunks will be loaded
- Load Rate The rate at which chunks are checked and loaded into the scene
- Load Distance The horizontal distance from the object which chunks will load in

Modification Properties

- Enable Input Handler Enable or disable the Input Handler in favour of using your own custom input controller
- Max Modify Radius The max size of the Modify Radius

Fluid

- Disable Fluid Disables the fluid simulation, fluid rendering, and prevents placement of fluid
- Select Fluid Layer Choose the block layer to use for the fluid simulation
- Render Fluid as Texture Select this if you want to render the fluid as a single texture using the Fluid shader (smooths out the edges and gets rid of the blocky look, but may also be slower)

Lighting

- Disable Lighting Disables the light system and prevents light updating
- Select Light Layer Choose the block layer to use for generating the shadow mask
- Illuminated Edge Blocks Selected the number of blocks from the edge of the terrain for light to illuminate
- Pause Time Pause the day/night cycle and movement of time
- Time Factor The factor time is determined by how fast time will go by in game (a factor of 1 is realtime)
- Time of Day The current time of day, used to control the ambient light color
- Sunlight Color The color of the ambient lighting during the day (in ID Mpos)
- Night Color The color of the ambient lighting during the night (in ID Mpos)
- Down Res The amount to scale down the lighting texture (in powers of 2), creates a blurring effect
- Number Blur Passes The number of times the lighting texture will be blurred before it is rendered to screen
- Ambient Light Layer The block layer the ambient lighting will use to map to the terrain
- Amount of light bleed The amount of blocks light will bleed into from the edge of the terrain
- Use Height Map Whether to use a height map to generate the ambient lighting (blocks above the surface of the terrain will be illuminated uniformly to mimick sunlight)
- Ambient light bleed The amount of blocks that ambient light can bleed through vertically into the terrain (only used when the Height Map is enabled)

Falling Block Simulation

- Disable falling blocks Disables the falling block simulation, no blocks will fall with gravity
- Falling Block Layer This is the layer which can contain falling blocks
- Update Rate The rate at which the falling Block Simulation will run (higher rate means slower update time)

Optimizations

- Overlap Blend Squares Allows the option to overlap the 'blend squares' (used when generating Overlap Blocks) over the block edges. By default the blend squares replace the block edge, but this adds a lot more vertices and triangles to the generated mesh
- Don't Generate Hidden Blocks If this option is selected blocks that are hidden behind other layers will not be rendered

Copyright © 2015 Matthew Wilson. All Rights Reserved.