Introduction

Thank you for purchasing Tycoon Tile!

This document will outline the core concepts of Tycoon Tile, it's code structure and give some directions on how to extend its functionality. If you have any questions, feel free to contact me on the Unity Forums or via email at support@tycoontile.com .

Overview

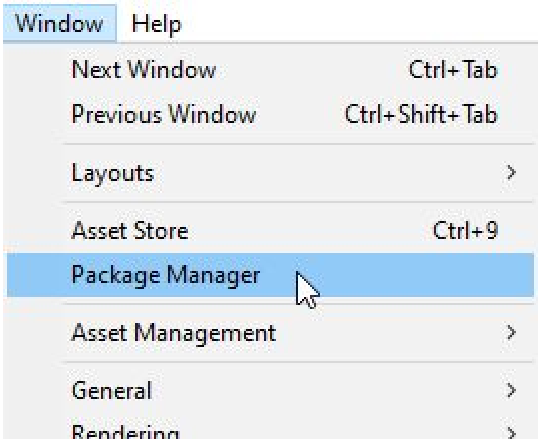
Tycoon Tile is a tile based terrain asset that allows you to create tycoon and city builder games quickly. It can be used in perspective or isometric camera view. Supports intuitive terraforming tools that recreate the classic tycoon game terrain mechanics. Adjust tile height per corner, create rolling hills or steep cliffs. Paint terrain with up to 65536 unique materials. Create water bodies with the click of a button. Kickstart your tycoon game development with Tycoon Tile now!

Features:

- High performance jobified unity job system burst compiled
- Low memory footprint
- Easily extendible with custom operations
- Ready to use out of the box
- Water support included
- Tested for terrain up to 2048x2048
- Supports up to 65536 unique materials
- Includes example runtime terrain generator
- Includes example user interface & terraforming tools
- Create custom terrain operations to adapt to your game's needs.

Installation

Tycoon tile requires a few packages to be installed from the package manager. To install the required packages, navigate to **Window -> Package Manager**



Make sure that the following packages are installed in your project:

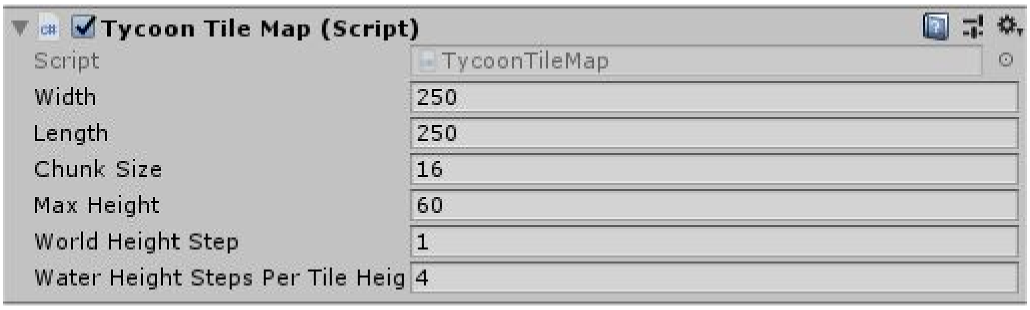
- com.unity.collections (0.0.9 or higher)
- com.unity.mathematics (1.0.1 or higher)

The example scene makes use of the Post-Processing package (2.1.7), so if you want to import and use the example scene, make sure to install the Post-Processing package as well.

Other versions of the packages than the ones mentioned may work, however have not been tested before release. If you are experiencing any issues with more recent package versions then please contact me!

Getting started

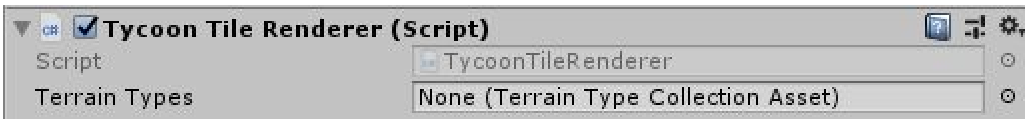
For a basic setup, create a new empty gameobject and attach the Tycoon Tile Map component. Here is a recommended set of values:



Tycoon Tile Map (Script)

Script	TycoonTileMap
Width	250
Length	250
Chunk Size	16
Max Height	60
World Height Step	1
Water Height Steps Per Tile Heig	4

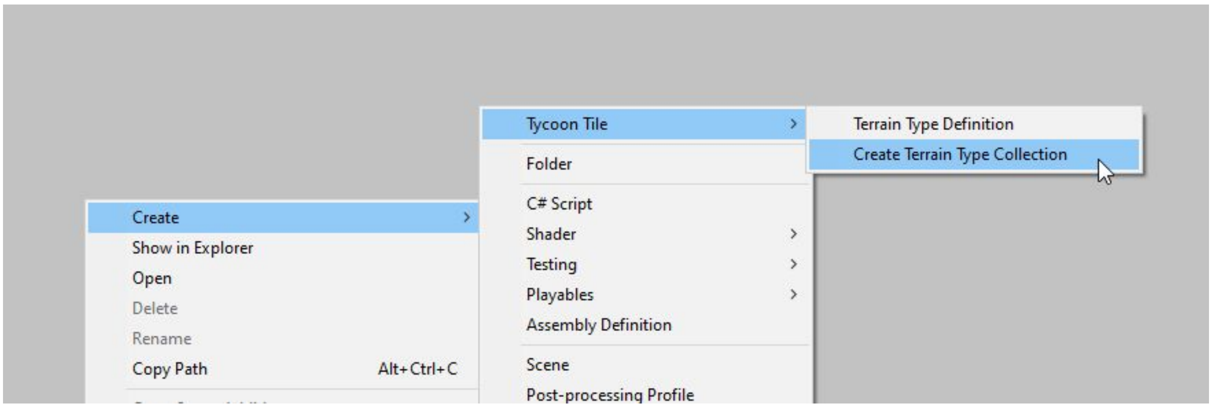
Next, add a Tycoon Tile Renderer component such that the terrain will be rendered.



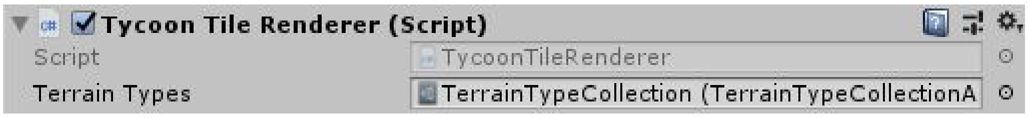
Tycoon Tile Renderer (Script)

Script	TycoonTileRenderer
Terrain Types	None (Terrain Type Collection Asset)

We will need a Terrain Type Collection asset in order to tell the renderer how to render the terrain. You can create one by right clicking in the project window and selecting **Create -> Tycoon Tile -> Terrain Type Collection**



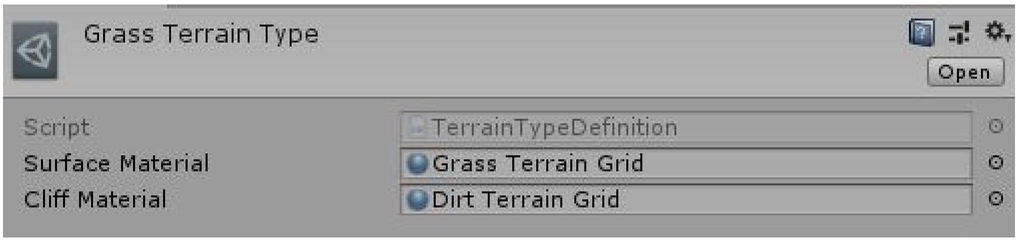
Give the collection a name ('TerrainTypeCollection' in this example) and assign it to the Tycoon Tile Renders' **Terrain Types** field.



Tycoon Tile Renderer (Script)

Script	TycoonTileRenderer
Terrain Types	TerrainTypeCollection (TerrainTypeCollectionA

Now we need to assign some terrain types to the terrain type collection. You can create a new terrain type by right clicking in the project window and selecting **Create -> Tycoon Tile -> Terrain Type Definition**

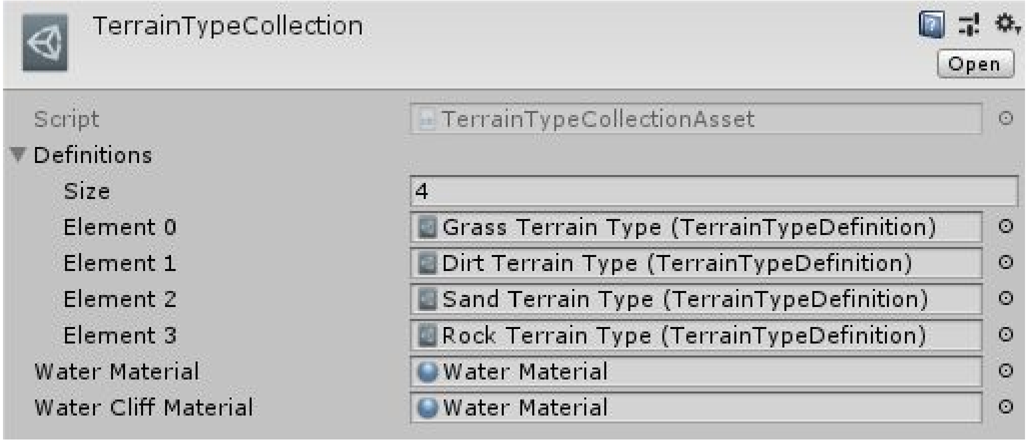


Grass Terrain Type

Script	TerrainTypeDefinition
Surface Material	Grass Terrain Grid
Cliff Material	Dirt Terrain Grid

Every terrain type definition needs to have a surface material and a cliff material assigned. The surface material will be used to render the terrain surface, while the cliff material will be used to render the tile cliff. Tycoon Tile comes with a grid shader that is compatible with the built-in Unity renderer, which was used for the materials in this example.

After creating a terrain type definition, add it to the list of definitions in the terrain type collection. The example scene has its terrain type collection configured as follows:

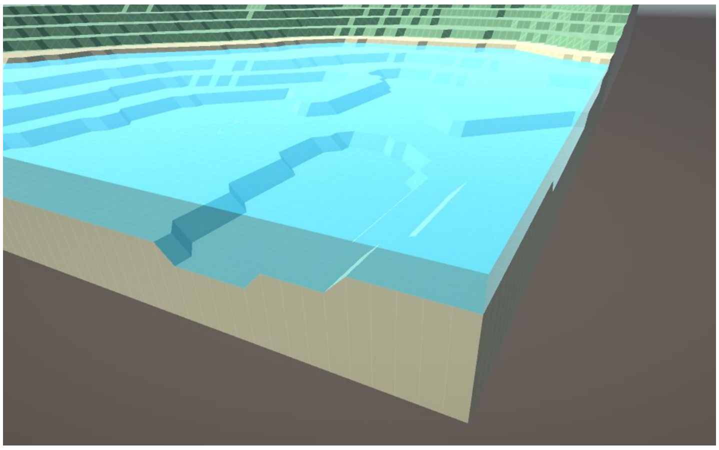


TerrainTypeCollection

Script	TerrainTypeCollectionAsset
▼ Definitions	
Size	4
Element 0	Grass Terrain Type (TerrainTypeDefinition)
Element 1	Dirt Terrain Type (TerrainTypeDefinition)
Element 2	Sand Terrain Type (TerrainTypeDefinition)
Element 3	Rock Terrain Type (TerrainTypeDefinition)
Water Material	Water Material
Water Cliff Material	Water Material

You need at least one terrain type definition in your collection.
Lastly, assign a material to the water material and water cliff material fields. In this example, the same blue transparent material with a standard shader is used.

The water material will be used to render water surfaces. The water cliff material will be used for the vertical water mesh on the map edges.

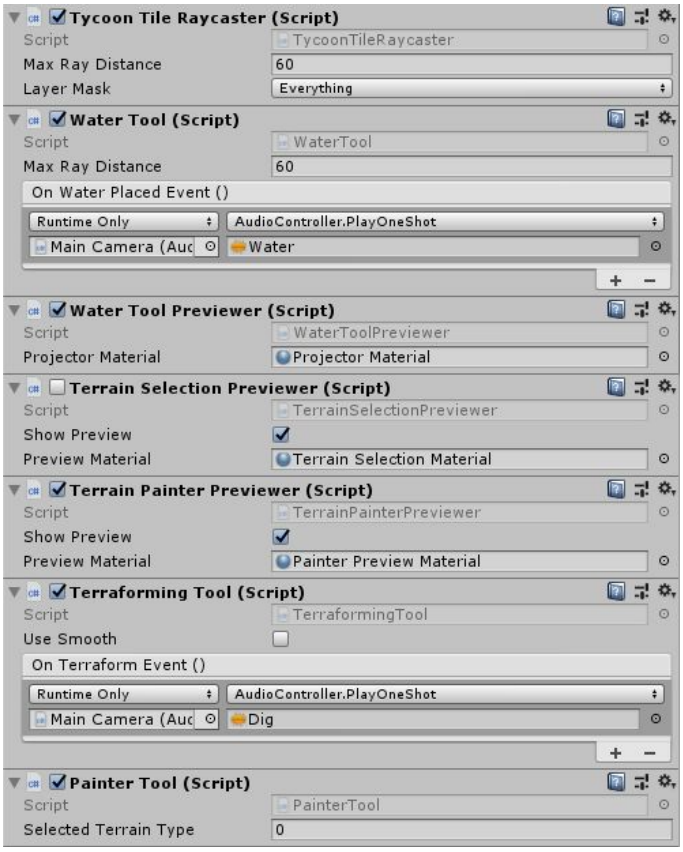


Now the terrain is ready to accept operations scheduled via script and update its visuals. However, we still need to setup the camera and tools such that we can actually send these operations to the terrain. For more information about these tools, see the **Example contents** section.

The terraforming and painting tools both depend on the **TycoonTileRaycaster** component. The TycoonTileRaycaster is responsible for keeping track of terrain selection based on camera position and user input. It then informs the tools and their previewers about any selection changes. Tools such as the terraformer and painter can then use the OnSelectionChanged event to act upon selection changes and schedule operations on the terrain. The water tool and its previewer do not depend on the TycoonTileRaycaster, since they deal with floating point world positions (including vertical position) instead of tile positions in horizontal directions only.

In order to be able to use the terraforming tool, one needs to add the **TycoonTileRaycaster** component to the main camera gameobject of the scene. Then, add a TerraformingTool under **Add Component -> Tycoon Tile -> Tools -> Terraforming Tool**. In order to be able to preview your selection while terraforming, add a **TerrainSelectionPreviewer** under **Add Component -> Tycoon Tile -> Previewers -> Terrain Selection Previewer**. Without the previewer you can schedule terrain operations, but you won't be able to see what your current terrain selection is. For the painting tool, the setup process is similar as described above, using the **PainterTool** and **TerrainPainterPreviewer**.

In the example scene, the camera object is configured as follows:



☑ Tycoon Tile Raycaster (Script)
Script: TycoonTileRaycaster
Max Ray Distance: 60
Layer Mask: Everything

☑ Water Tool (Script)
Script: WaterTool
Max Ray Distance: 60
On Water Placed Event ()
Runtime Only | AudioController.PlayOneShot
Main Camera (Auc | Water

☑ Water Tool Previewer (Script)
Script: WaterToolPreviewer
Projector Material: Projector Material

☐ Terrain Selection Previewer (Script)
Script: TerrainSelectionPreviewer
Show Preview: ☑
Preview Material: Terrain Selection Material

☑ Terrain Painter Previewer (Script)
Script: TerrainPainterPreviewer
Show Preview: ☑
Preview Material: Painter Preview Material

☑ Terraforming Tool (Script)
Script: TerraformingTool
Use Smooth: ☐
On Terraform Event ()
Runtime Only | AudioController.PlayOneShot
Main Camera (Auc | Dig

☑ Painter Tool (Script)
Script: PainterTool
Selected Terrain Type: 0

Core concepts

Terrain/Map
A map or terrain grid is the top level representation of a Tycoon Tile terrain. It contains all the tile data in a flattened 2D array. It has a set width and length

Chunk
A chunk is standard sized partition of a map. It is used as an optimization measure to update the graphical mesh that represents a part of the terrain. By using chunks and only updating meshes of parts of the terrain that have changed, updating all the meshes of the entire terrain is prevented. Update work is distributed across background threads on a per-chunk basis. The chunk size is configurable on the terrain component

Tile
A tile represents a 1 by 1 meter area of the terrain that consists of 4 corners of which the height can be individually adjusted.

Tile corner
Represent a single height sample in one corner of a tile. Height can be adjusted in steps. The world height per step can be configured on the terrain component

Terrain type
A terrain type defines the look of a tile. Tiles can have one terrain type assigned at a given time. The terrain painting tool assigns terrain types to tiles.

Terrain type collection
A terrain type collection holds references to terrain types. A single terrain type collection can be assigned to a map at a given time. A maximum of 65536 terrain types can be referenced in a terrain type collection.

Operations
An operation reads or modifies (part of) the terrain. Any reads or modifications to the terrain should be done via an operation. Operations must keep track of which chunks are modified, such that terrain meshes will update appropriately.

Example contents

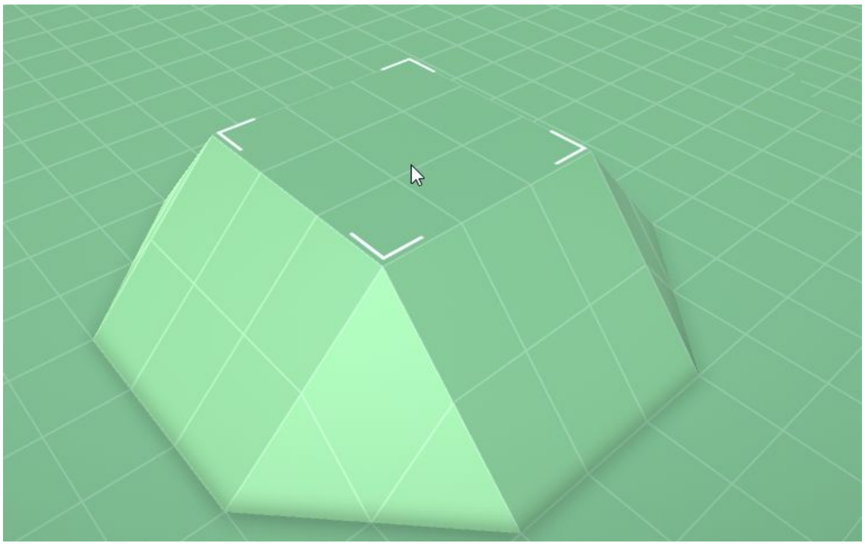
The example folder contains an example scene demonstrating the features of Tycoon Tile. A few scripts and tools are included.

Terraforming tool

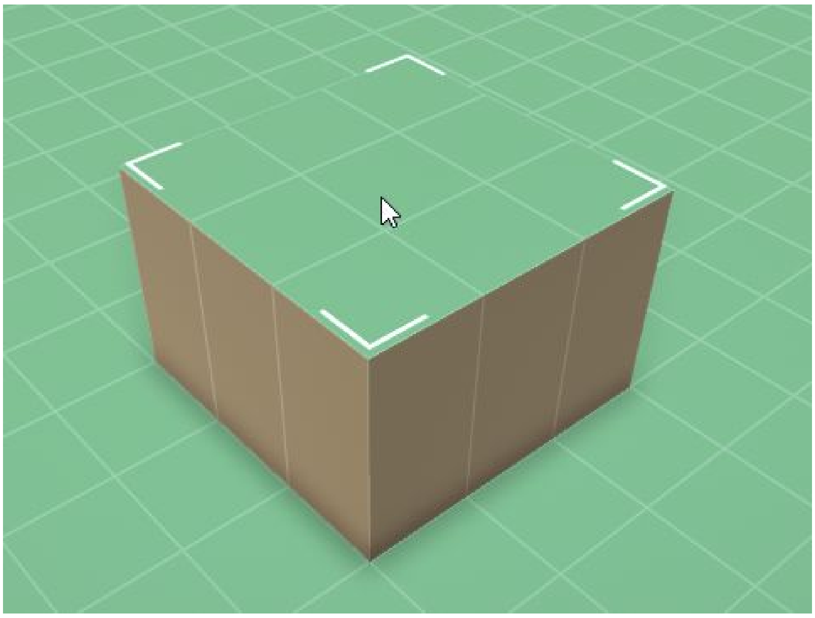
The terraforming tool demonstrates how to schedule terrain operations and what each operation does. Selection size can be adjusted with the scroll wheel or with the UI slider. Edits can be made by holding the left mouse button and dragging the mouse up or down while having tiles selected.

The terraforming tool has two modes: Smooth edit and Cliff edit.

When smooth edit is selected, the terraforming tool will raise or lower tiles while keeping connected tile surfaces uninterrupted. It will not create new cliffs. Tiles outside the selection will thus also be affected.



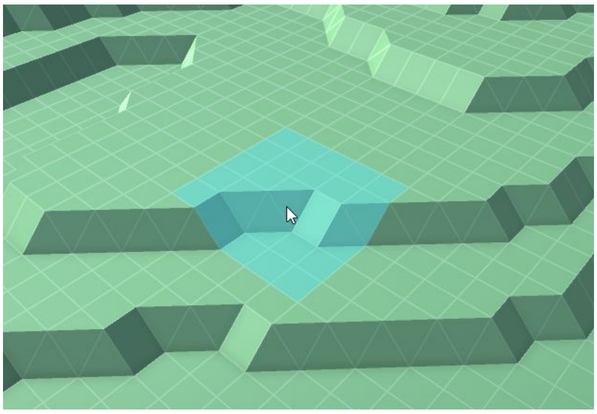
When cliff edit is selected, only the tiles within the selection will be edited and new cliffs will be generated on the edge of the selection.

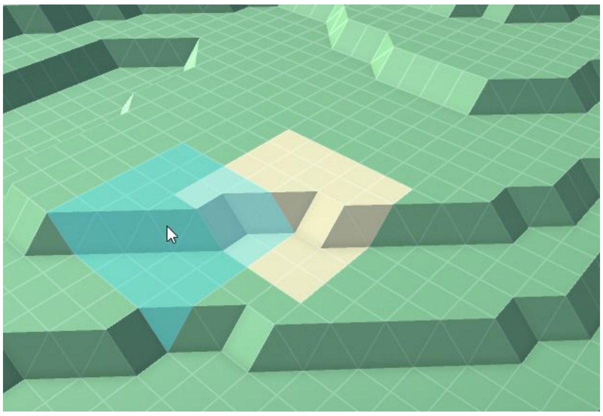


Note that when only a single corner is selected, the terraforming tool will always behave as if it is in cliff mode.

Painting tool

The painting tool demonstrates assignment of terrain types to tiles using a brush. Brush size can be adjusted with the scroll wheel, and a terrain type can be selected. Terrain types are painting on the terrain by holding the left mouse button and dragging over the terrain.

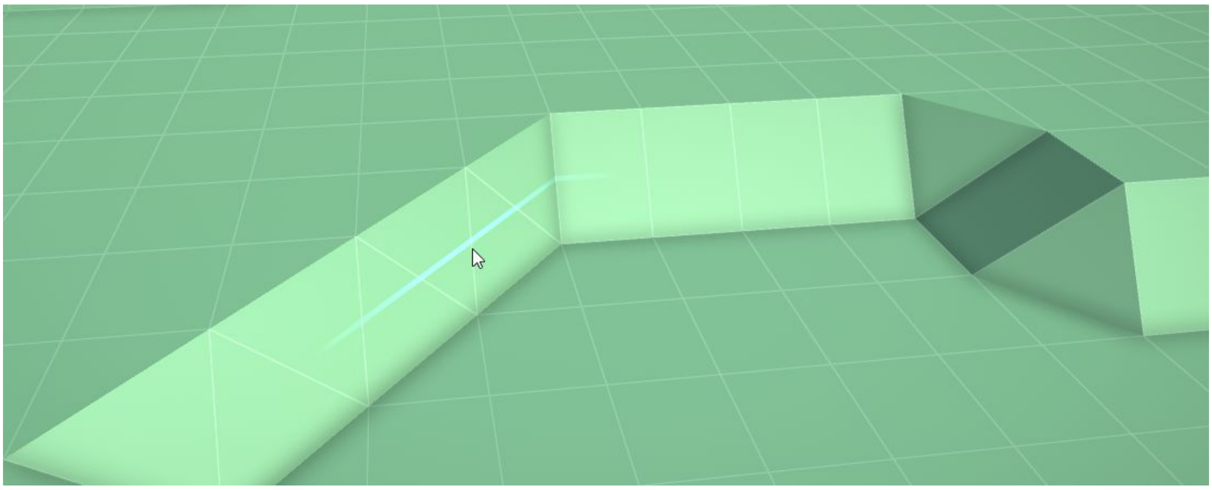


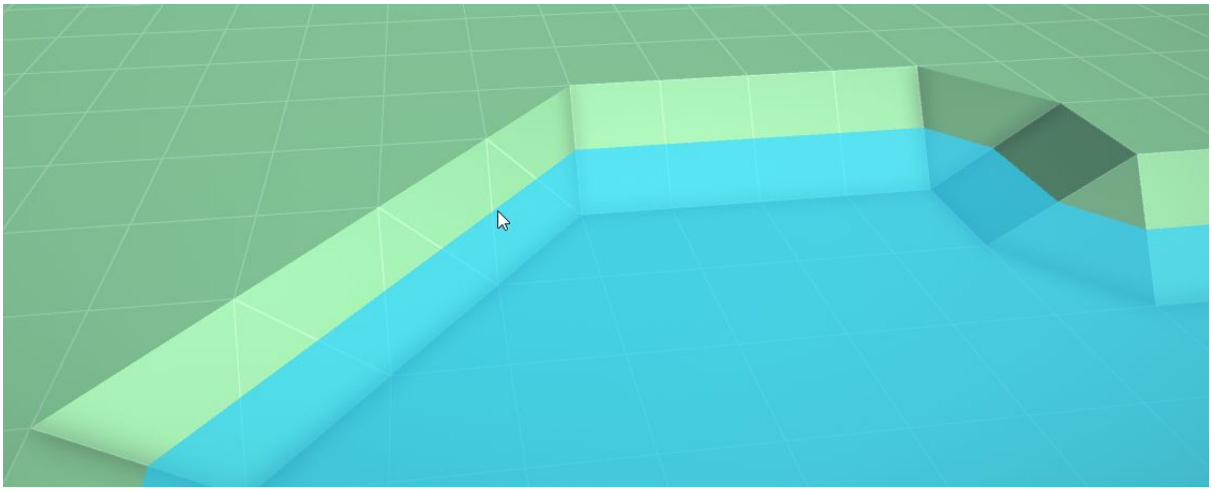


Water tool

The water tool demonstrates adding, adjusting and removing water on the terrain. There are 3 supported actions:

- Creating water body / raising water level: Position the mouse cursor on a cliff or sloped part of the terrain that is not submerged in water. When the left mouse button is clicked, water level will be set to the current mouse cursors vertical position.
- Lowering water level: Position the mouse cursor on a cliff or sloped part of the terrain that is below that the tile water level. When the left mouse button is clicked, water level will be set to the current mouse cursors vertical position. For any tiles where the new water level will be below the terrain, water will be removed for that tile.
- Clearing water body: Position the mouse cursor on the surface of a flat tile that is submerged. When the left mouse button is clicked, water will be cleared for all connected water tiles.





Camera

The demo scene comes with a camera script that allows you to easily navigate around in the scene, in a similar way the editor camera works.

Use WASD to move around
Hold shift to use the boost function which speeds up the camera movement.
Hold the right mouse button to pan and tilt the camera.

Creating new terrain types

Tycoon Tile supports up to 65536 unique terrain types per map. You can add a new terrain type by creating a new TerrainTypeDefinition asset by right clicking in the project window Create -> Tycoon Tile -> Terrain Type Definition. You can then add this definition to your terrain type collection.

Creating custom terrain operations

When creating your game, specials needs may arise to facilitate specific game mechanics that interact with Tycoon Tile terrain. In this case you may need to implement custom terrain operations to support these mechanics.

These operations can then be scheduled to be executed by calling Schedule() on the tycoon tile component and passing an instance of your custom operation as its parameter.

Creating custom previewers

Previewers give feedback to the user about the operation that is about to be performed. Tycoon Tile comes included with three example previewers. The TerrainPainterPreviewer, TerrainSelectionPreviewer and the WaterToolPreviewer. Each of them demonstrates a unique way to show the user what is going to happen when the mouse button is clicked.

One can implement custom previewers to change the way the operation is previewed. In order to implement a custom previewer, the class must hold a reference to the TCRaycaster component and register a function for its OnSelectionChangedEvent. Whenever the selected terrain changes, the previewer will get a callback via this event, so you can update the visualisation to fit your needs.