Water2D Tool v1.x

Introduction

Water2D Tool is a tool for Unity3D that allows you to quickly create 2D water with dynamic properties for your 2D games.

Water Creation Process

To create a new water object you can access the menu (GameObjects->2D Water) where you can choose between a water with a 2D or a 3D collider, Figure 1, or use the shortcuts (CTRL+W, CTRL+SHIFT+W).
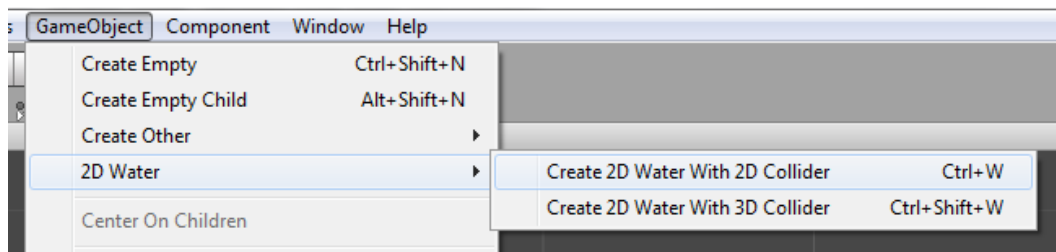

Figure 1 - Water2D menu.

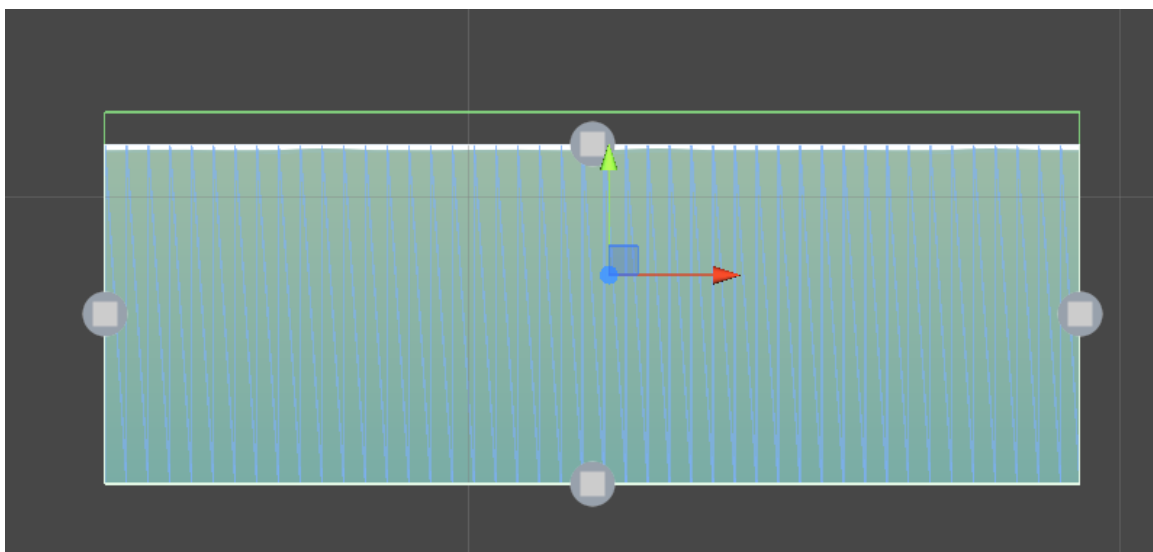After the water is created you can use the 4 handles to change its size and the position of its 4 edges, Figure 2.

Figure 2 – Water size manipulation.

Scripts

When a Water2D object is created it has two scripts, (*Water2D_Tool and Water2D_Simulation*).

Water2D_Tool (Script)

This is the script that handles mesh creation and updating, collider creation and updating, sets the default material. There are a few public variables you can change in the inspector, Figure 3.
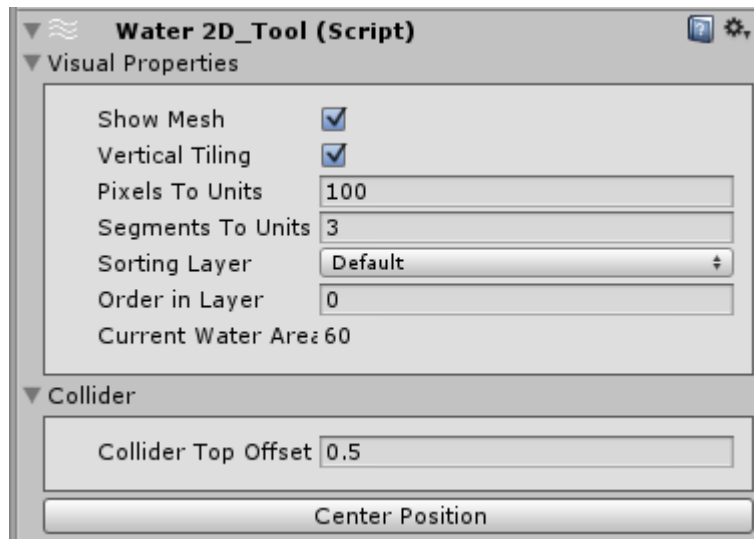

Figure 3 – *Water2D_Tool* public fields.

Show Mesh – When disabled hides the mesh shape in the Scene View

Vertical Tiling – When enabled, if the water height is greater than the max water height that can be created with the current Pixels To Units value, the texture will be tiled vertically, Figure 4. Here the green and yellow rectangles are a single 512x512px texture.
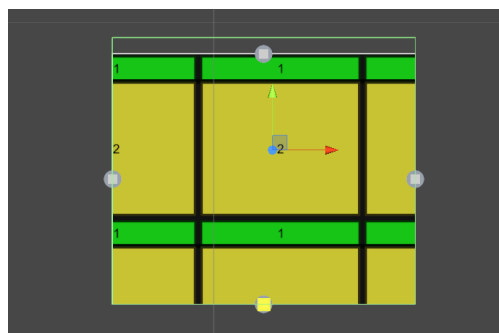

Figure 4 – Vertical Tiling Enabled.

When disabled the texture will be stretched if the water height is greater than the max water height that can be created with the current Pixels To Units value, Figure 5.
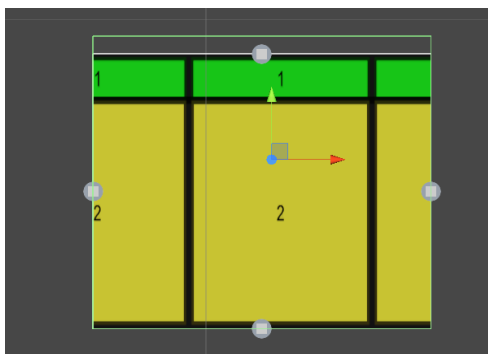

Figure 5 – Vertical Tiling Disabled.

Pixels To Units – This value allows you to specify how many texture pixels should fit in 1 Unity unit.

Segments To Units - This value allows you to specify how many vertical segments (or quads) should fit in 1 Unity unit, Figure 6.
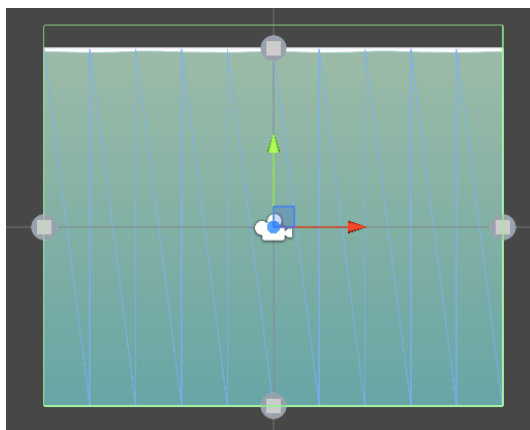
Figure 6 – One vertical segment to one Unity unit.

Sorting Layer and Order in Layer – This two have the same functions as Unity Sprite's attributes.

Current Water Area - The value of this field is not used in any calculations. When creating an animation that animates the area of the water, use this value as a guide to see how the water area changes between two positions.

Box Collider Z Offset – Offsets the center of the box collider on the Z axis.

Box Collider Z Size – The size of the box collider on the Z axis.

3

Center Position – This is a button you can use to reset the Transform Position pivot point of the water object to the center of the mesh.

Water2D_Simulation (Script)

This script contains the code that controls the behavior of the water at Run time. There are a lot of public variables you can change in the inspector, that allow you to change the behavior of the water. The variables are arranged in 4 groups: *Spring, Buoyancy, Animation* and *Miscellaneous*. Figure 7.
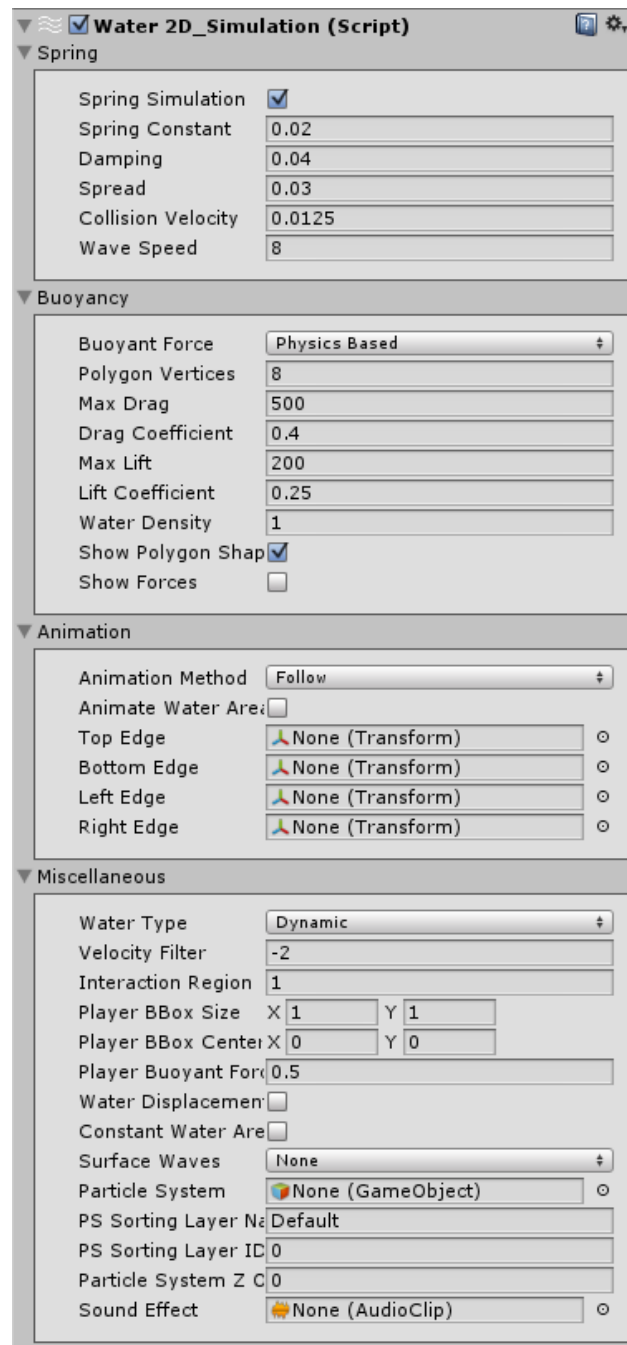

Figure 7 – *Water2D_Simulation* public fields.

SPRING:

This group contains the variables that control the way the water reacts to a falling object.

- **Spring Simulation** - Enables the simulation of springs. This what makes the surface of the water react to objects.
- **Spring Constant** - This value controls the stiffness of the springs. A low spring constant will make the springs loose. This means a force will cause large waves that oscillate slowly. A high spring constant will increase the tension in the spring. Forces will create small waves that oscillate quickly.
- **Damping** - The damping slows down the oscillation of the springs. A high dampening value will make the water look thick like molasses, while a low value will allow the waves to oscillate for a long time.
- **Spread** - Controls how fast the waves spread.
- **Collision Velocity** - Limits the velocity a spring will receive from a falling object.
- **Wave Speed** - Another variable to control the spread speed of the waves.
- **Overlap Sphere Radius** - The radius of a sphere that will be used to check if there is a 3D collider near a surface vertex.

BUOYANCY:

This group contains the variables that control the way the objects behave in the water.

- **Buoyant Force**
  a) **Physics Based** – This method uses Archimedes Principles to calculate the behavior of the objects in the water. It produces the most realistic simulation as it takes into account the shape of the object that is floating in the water.
     1. **Polygon Vertices** - When an object with a circleCollider2D is detected an imaginary regular polygon collider is created based on its radius and position. Use this to set the number of vertices the regular polygon collider should have.
     2. **Max Drag** - The max drag force that should be applied to an object.
     3. **Drag Coefficient** - Determines how much drag force should be applied to an object.
     4. **Max Lift** - The max lift force that should be applied to an object.
     5. **Lift Coefficient** - Determines how much lift force should be applied to an object.

6. **Water Density** - Sets the water density. In a water with low density the objects will submerge faster and come to the surface slower. If the water density is great the objects will stay more at the surface of the water and will submerge slower.

7. **Show Polygon Shape** - When enabled will show in the Scene View the shape of the polygon that is below the waterline.

8. **Show Forces** - When enabled will show in the Scene View the velocity direction, drag direction, lift direction and the normal of a leading edge.

b) **Linear** - This method uses the objects position to simulate the Buoyant Force. It does not take into account the shape of the object so the result is not very realistic. If you want better performance use this method as it is less expensive.

1. **Float Height** - Determines how much force should be applied to an object submerged in the water. A value of 3 means that 3 m under the water the force applied to an object will be 2 times greater than the force applied at the surface of the water.

2. **Bounce Damping** - Slows down the vertical oscillation of the object.

3. **Drag Coefficient** - Determines how much drag force should be applied to an object.

4. **Angular Drag Coefficient** - Slow down the angular rotation of the object.

5. **Force Scale** - A value of 1 will make an object with the mass of 1kg float at the surface of the water and an object with the mass of 2kg float 3m below the water surface if Float Height is set to 3m.

6. **Force Position Offset** - By default the force will be applied at the center of the object. Use this to offset the position where the force will be applied to an object.

ANIMATION:

This group contains the variables that control the animation of the water.

- **Animation Method**
  1. **Follow** - The animated handle follows the movement of a referenced object.
  2. **Snap** - The animated handle's global position is set to that of a referenced object.
- **Animate Water Area** - Enable this if you want to animate the increase or decrease of the total water area.
- **Top Edge** - Place here an animated object you want the water line (the top of the water) to follow.
- **Bottom Edge** - Place here an animated object you want the bottom edge of the water to follow.

- **Left Edge** - Place here an animated object you want the left edge of the water to follow.
- **Right Edge** - Place here an animated object you want the right edge of the water to follow.
- **Top Edge Y Offset** - The offset on the Y axis from the position of a referenced object.
- **Bottom Edge Y Offset** - The offset on the Y axis from the position of a referenced object.
- **Left Edge X Offset** - The offset on the X axis from the position of a referenced object.
- **Right Edge X Offset** - The offset on the X axis from the position of a referenced object.

MISCELLANEOUS:

This group contains the variables that control different behaviors of the water.

- **Water Type**
  a) **Dynamic** – All the functions of the water are enabled.
  b) **Decorative** – The water collider is disabled and the water will not react to objects, but it can be animated.
- **Velocity Filter** - An object with a velocity on the Y axis greater than the value of Velocity Filter will not create splashes.
- **Interaction Region -** The bottom region of a colliders bounding box. This value is used to limit the ability of the objects with big bounding boxes to affect the velocity of the surface vertices. A value of 1 means that only the first 1m of the bottom of the bounding box will affect the velocity of the surface vertices.
- **Player BBox Size -** The size for the players bounding box. In most cases the player character will have more than one collider. So to simplify the things, Water2D uses this variable to set the size for an imaginary bounding box that will be used when applying buoyant force.
- **Player BBox Center -** By default the center of the bounding box will be the *transform.position* of the object. Use this variable to offset the players bounding box center.
- **Player Buoyant Force Scale -** Depending on what character controller you are using, you may have a big character that must have a small mass. As a result the Player will not submerge in the water because of its low mass that results in low density. To resolve this problem use this variable to scale down the buoyant force applied to the Player.
- **Water Displacement -** Floating objects will influence the final water area.
- **Constant Water Area** - If the width of the water changes, the height will change too, to keep a constant water Area.

- **Surface Waves -** List of methods to generate surface waves.
  - a) **None –** No surface waves will be generated.
  - b) **Random -** Generates small random splashes.
    1. **Wave Time Step -** The time between splashes.
    2. **Max Velocity -** The constant is used to generate a random velocity between a Max and a Min.
    3. **Min Velocity -** The constant is used to generate a random velocity between a Max and a Min.
    4. **Neighbor Vertex Velocity Scale -** Will scale down (up) the velocity that is applied to the neighbor vertices when RandomWave method is called.
  - c) **Sine Waves -** Overlaps a number of sine waves to get a final wave that changes the velocity of the surface vertices.
    1. **Sine Waves Number -** The number of individual sine waves.
    2. **Max and Min Amplitude -** The variables are used to generate a random amplitude value between a Max and a Min.
    3. **Max and Min Stretch -** The variables are used to generate a random sine wave stretch value between a Max and a Min.
    4. **Max and Min Phase Offset -** The variables are used to generate a random phase offset value between a Max and a Min.
    5. **Sine Wave Velocity Scale** - Will scale down (up) the velocity that is applied to a vertex from a sine wave.
- **Particle System** - A particle system prefab used to simulate the water splash effect.
- **PS Sorting Layer Name** - The sorting layer name for the particle system.
- **PS Order In Layer** - The order in layer for the particle system.
- **Particle System Z Offset** - Offsets the position where the particle systems are created on the Z axis.
- **Sound Effect** - A sound effect generated when an object hits the water surface.