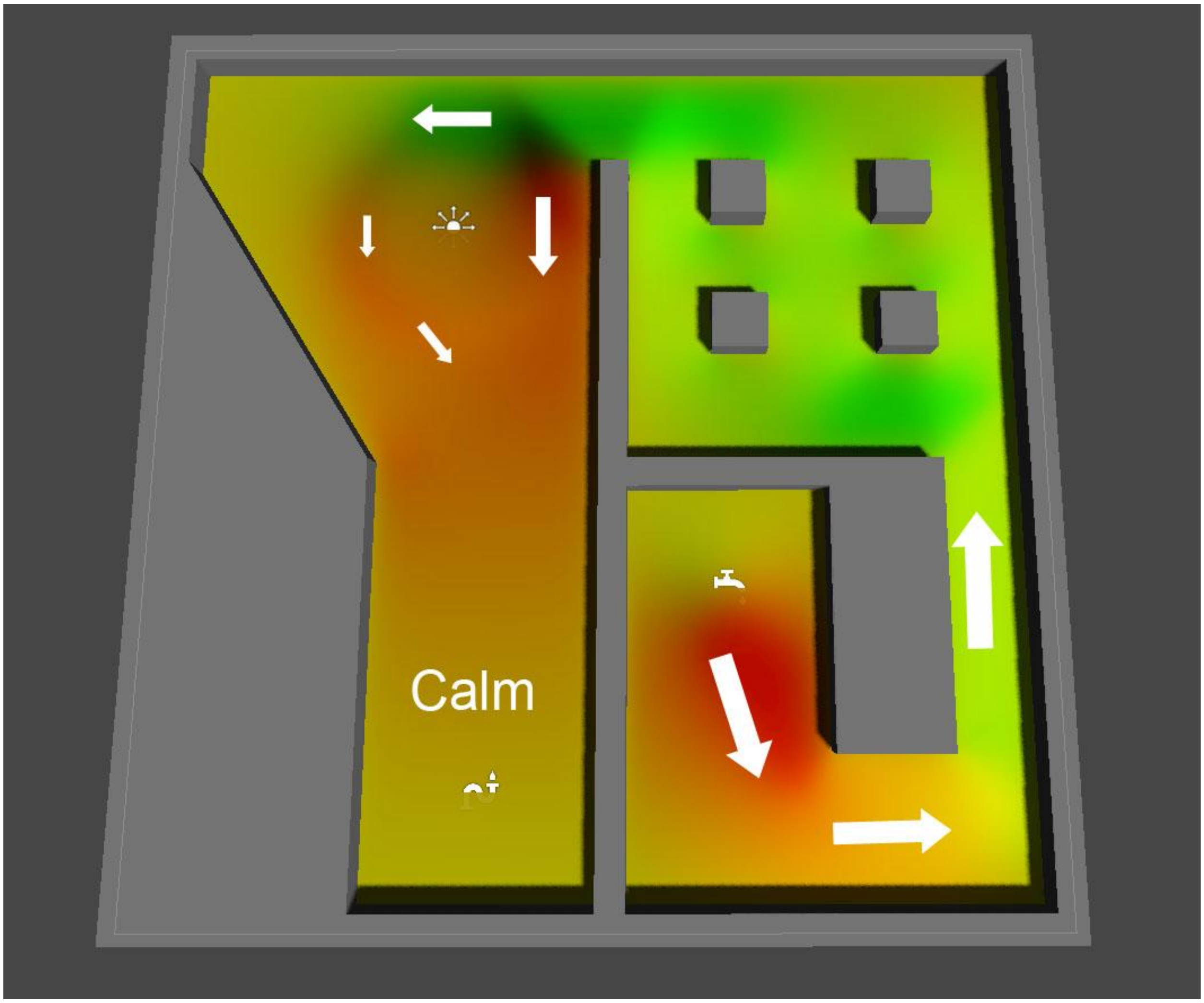
Flowmap Generator Reference

Table of Contents

Flowmap Overview

What is a flowmap?

The appearance of moving water in games is often created by scrolling two normal maps in different directions and adding them together. This works fine for large open bodies of water, but when dealing with rivers and more enclosed areas the effect breaks down. Flowmaps are a way to give the illusion of water that flows in different directions. The underlying idea is the same, the shader scrolls a normal map, but the direction is no longer constant over the water's surface.



Here's a flowmap created with Flowmap Generator. The texture contains directional vectors encoded in the red and green channels, similar to how normal maps store normal vectors. In this case a color of (255, 128, 0) results in a vector pointing to the right, with a magnitude of 1. The closer a color is to (128,128,0) the smaller the magnitude and therefore the slower the normal map will scroll.

Using a flowmap in a shader

A good overview of how to use flowmaps in a shader, along with some sample shader code, can be found here: http://graphicsrunner.blogspot.se/2010/08/water-using-flow-maps.html Valve have used flowmaps in several of their games and have created some interesting presentations detailing their approach. One of them can be found here: http://www.valvesoftware.com/publications/2011/gdc_2011_grimes_nonstandard_textures.pdf

The general idea is that the flowmap vector is added to the scrolling normal map's uv coordinate. This scrolls the normal map in the direction that the flowmap is pointing. After a short amount of time however, the uvs would be quite distorted. To prevent this the amount of distortion is reset after a set amount of time. This results in a popping effect when the time cycle is restarted. The way to fix this is to fade in a second normal map. This second normal map is also being distorted by the flowmap, but the timeline is offset by half a cycle.

I've included several flowmap shaders in the Examples folder in the Flowmap Generator package. These can be freely reused and edited. A large number of tutorials exist for creating flowmap shaders in different game engines and there are several packages of flowmap shaders available on Unity's Asset Store.

Performance

Flowmap shaders are slightly more performance heavy than shaders that just scroll two normal maps over each other but are still useable in most situations. Most mobile hardware support flowmap shaders.

Creating flowmaps by hand

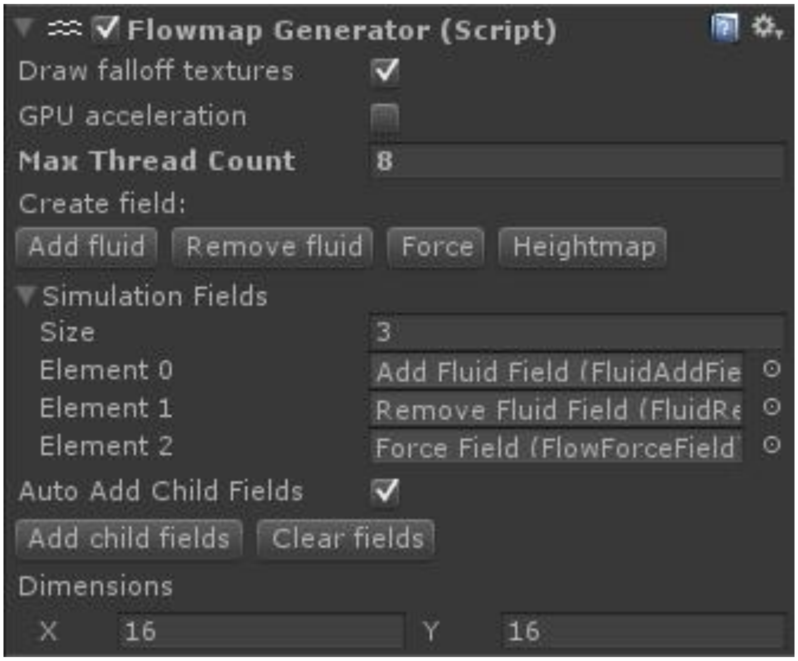
Flowmaps can be difficult to create. A flowmap represents a vector field and hand painting vectors in an image editing application rarely gives good results. Valve use Houdini to paint vectors and several programs exist for painting vectors that follow a brush stroke. There are several drawbacks to this approach. If the scene changes it is necessary to repaint the flowmap in that area. Painting vectors for a water plane that covers an entire level can be quite time consuming.

Creating flowmaps using Flowmap Generator

Flowmap Generator uses a fluid simulation to create the flowmap texture. This ensures that vectors are created across the entire flowmap and allows quick resimulating if the scene changes. The simulation is controlled by fields that act on the simulation and can push the fluid around to create velocity vectors. Simulating a flowmap using GPU acceleration can take from just a few seconds to simulate. Without GPU acceleration Flowmap Generator uses a multithreaded approach on the CPU, so most simulations take less than a few minutes. This allows fast iteration times and using fields allows the flowmap vectors to still be user controlled.

Flowmap Generator Reference

Flowmap Generator Component



☑ Flowmap Generator (Script)

Draw falloff textures ☑

GPU acceleration □

Max Thread Count 8

Create field:

Simulation Fields

Size 3

Element 0 Add Fluid Field (FluidAddFie

Element 1 Remove Fluid Field (FluidRe

Element 2 Force Field (FlowForceField

Auto Add Child Fields ☑

Dimensions

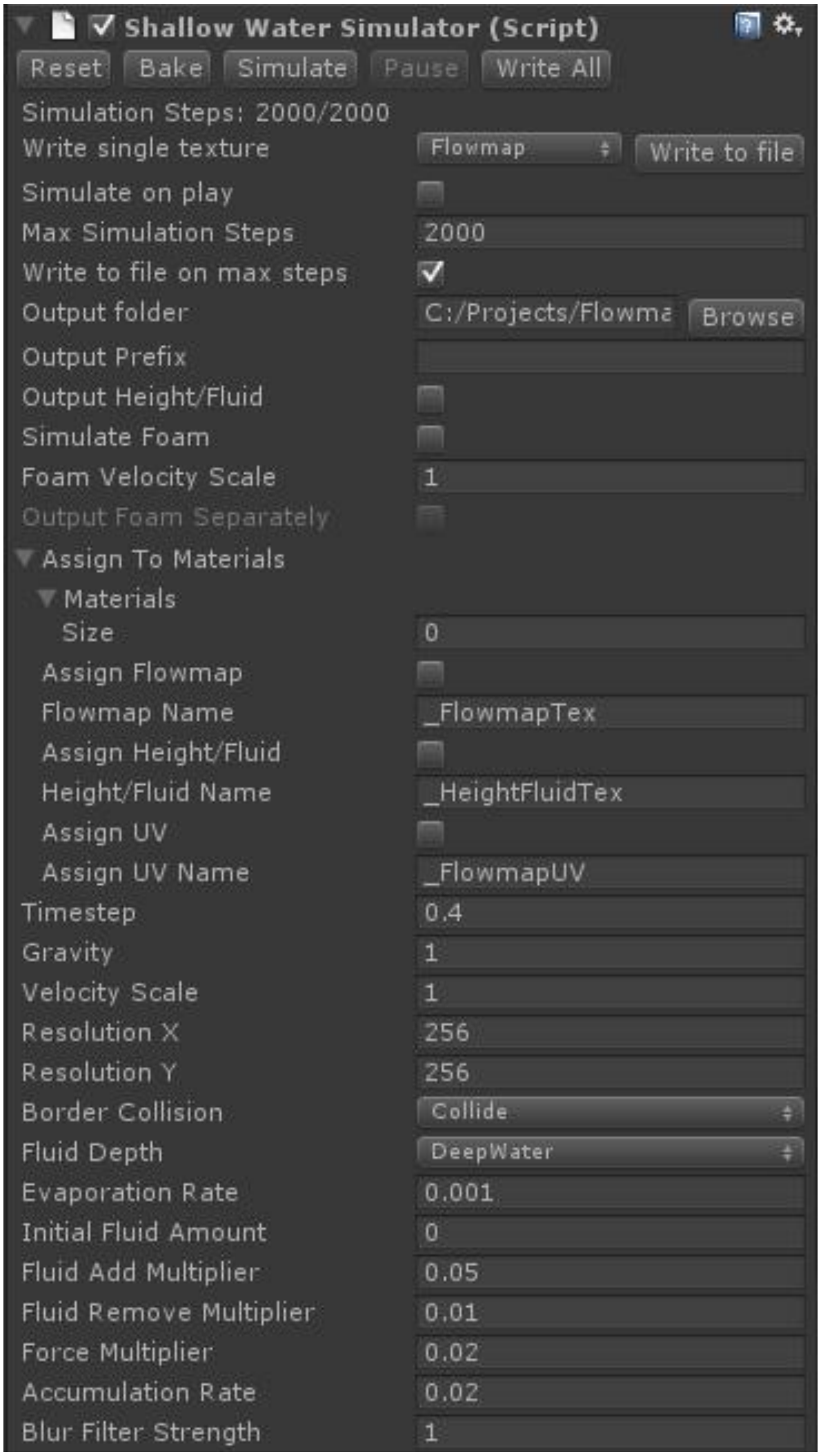
X 16 Y 16

Flowmap Generator is a component that can be attached to a GameObject. A generator can have a flow simulator component and a heightmap component attached. The generator can be used to bake a flowmap to a texture in the editor. The generator can also continuously simulate while your game is playing, by assigning the flowmap being generated to materials used in your scene. Non-square dimensions are supported. Often you will want to change the resolution of the simulation to reflect this.

Setting	Description
Draw falloff textures	Controls visibility of falloff textures for simulation fields.
GPU acceleration	When using Unity Pro run the simulation on the GPU. Much faster than CPU simulation.
Max Thread Count	When not using GPU acceleration this lets you set the maximum number of threads to be used. The value is clamped between 1 and the number of processors available to Unity.
Add fluid	Creates an Add fluid field.
Remove fluid	Creates a Remove fluid field.
Force	Creates a Force field (Vortex, Directional, Calm, etc.) The type of field can be changed by selecting the field GameObject.
Heightmap	Creates a Heightmap field.
Simulation Fields	A list of all fields that affect this generator. You can add fields by hand, or parent them under the generator GameObject and use Add child fields to add all fields parented under the generator.
Auto Add Child Fields	Automatically add all fields that are parented under the generator's transform.

Add child fields	Add all fields parented under the generator GameObject.
Clear fields	Remove all fields from this generator, doesn't delete the fields from the scene.
Dimensions	The bounds of the simulation, in world units.

Flowmap Simulator Component



☑ Shallow Water Simulator (Script)

Simulation Steps: 2000/2000

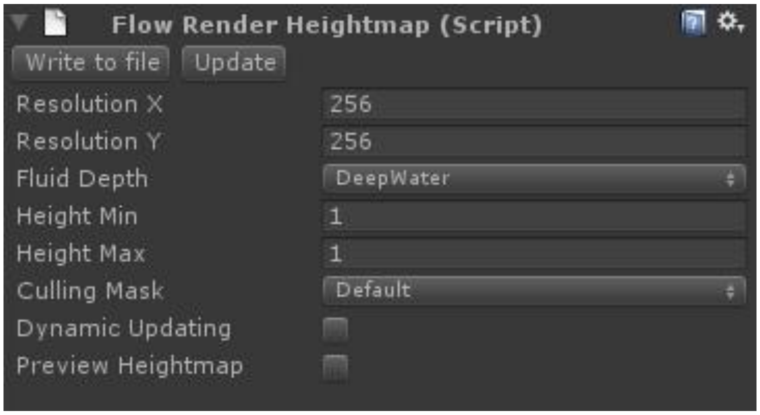
Write single texture	Flowmap
Simulate on play	□
Max Simulation Steps	2000
Write to file on max steps	☑
Output folder	C:/Projects/Flowma
Output Prefix	
Output Height/Fluid	□
Simulate Foam	□
Foam Velocity Scale	1
Output Foam Separately	□
Assign To Materials	
Materials	
Size	0
Assign Flowmap	□
Flowmap Name	_FlowmapTex
Assign Height/Fluid	□
Height/Fluid Name	_HeightFluidTex
Assign UV	□
Assign UV Name	_FlowmapUV
Timestep	0.4
Gravity	1
Velocity Scale	1
Resolution X	256
Resolution Y	256
Border Collision	Collide
Fluid Depth	DeepWater
Evaporation Rate	0.001
Initial Fluid Amount	0
Fluid Add Multiplier	0.05
Fluid Remove Multiplier	0.01
Force Multiplier	0.02
Accumulation Rate	0.02
Blur Filter Strength	1

Each Flowmap Generator can have one simulation component attached. Right now the only type of simulation is a Shallow Water simulation.

Setting	Description
Reset	Reset the simulation, keeping all settings.
Bake	Run the simulation from the first frame until it reaches the max number of simulation steps, then write all specified textures to the Output folder.
Simulate	Run the simulation from the current simulation step.
Pause	Pause the simulation at the current simulation step.
Write All	Write all output textures at the current simulation step.
Simulation Steps	Shows the current simulation step.
Write single texture	Save a texture from the current simulation step to file.
Simulate on play	If enabled, start running the simulation when the game starts. This can be used for continuously updating flowmaps.
Max Simulation Steps	If set to a non-zero number the simulation will pause at this simulation step and optionally write output textures to file.
Output folder	When baking all textures will be saved to this folder.
Output Prefix	Adds a prefix to the output texture filenames.
Output Height and Fluid	Write a texture that contains the heightmap in the red channel and fluid depth in the green channel.
Simulate Foam	Enable foam accumulation calculation.
Output Foam Separately	Write foam to a separate texture, otherwise the foam will be saved to the blue channel of the flowmap.
Assign To Materials	
Materials	Apply to these materials.
Assign Flowmap	Assign the flowmap currently being generated to the materials.
Flowmap Name	Assign the flowmap to materials using this texture name.
Assign Height/Fluid	Assign the heightmap and fluid depth currently being generated to materials.
Height/Fluid Name	Assign the heightmap and fluid texture to materials using this texture name.
Assign UV	Assigns a float4 to materials containing the generator's scale in xy and the generator's position in zw.
Assign UV Name	Sets the name of the float4 that contains the generator's dimensions and position.
Timestep	Controls how large of a time difference will be used for each simulation step. Too large of a timestep will result in the simulation "exploding" into pixel artifacts. The maximum good timestep is dependent on Gravity.
Gravity	Controls how fast the fluid flows. Lower gravity settings will result in force fields having a stronger effect.
Resolution X	The flowmap texture's width.
Resolution Y	The flowmap texture's height.
Border Collision	If set to Collide, the fluid will bounce off the borders of the simulation. If set to Pass Through, the fluid will flow out of the

	simulation.
Fluid Depth	Set the fluid depth style, several other settings are affected, such as initial fluid amount.
Evaporation Rate	Removes a bit of fluid every simulation step.
Initial Fluid Amount	Starts the simulation with fluid already existing. If Fluid Depth is set to DeepWater, no fluid will be added where the heightmap has a value of 1.
Fluid Add Multiplier	A global multiplier for all fluid add fields
Fluid Remove Multiplier	A global multiplier for all fluid remove fields
Force Multiplier	A global multiplier for all force fields
Accumulation Rate	Controls how quickly the flowmap accumulates velocity. A lower rate will make sure that short lived velocity changes are ignored.
Blur Filter Strength	Sets the strength of the blur filter.

Flow Render Heightmap Component



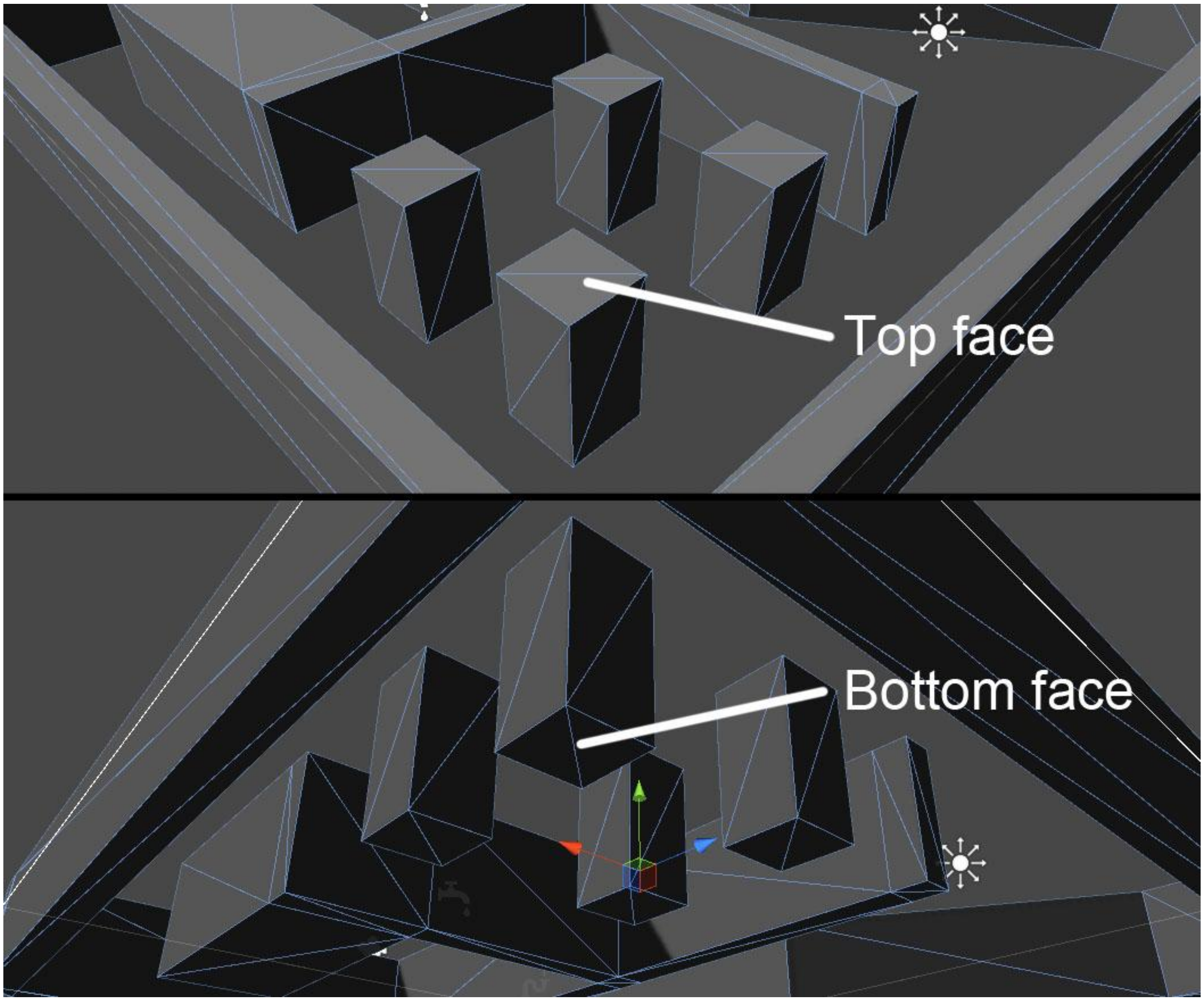
Resolution X 256
Resolution Y 256
Fluid Depth DeepWater
Height Min 1
Height Max 1
Culling Mask Default
Dynamic Updating □
Preview Heightmap □

This component will render a heightmap from your scene geometry. It is only supported for Unity Pro. A wireframe cube shows the Height Min and Max in relation to the generator's plane.

Setting	Description
Write to file	Save the current heightmap to a texture.
Update	Force the heightmap to update.
Resolution X	The heightmap texture's width.
Resolution Y	The heightmap texture's height.
Fluid Depth	If set to Deep Water, the resulting heightmap will be a cross section of geometry that intersects with the generator plane. Use this when you are simulating a deep pool of water. If set to Surface, the resulting heightmap will look more like a traditional heightmap. Use this when you are simulating a fluid flowing on a surface.
Height Max	The maximum height, the result depends on the fluid depth chosen.

Height Min	The minimum height, the result depends on the fluid depth chosen.
Culling Mask	Only geometry on these layers are rendered to the heightmap. Often it's best to use geometry specifically made for rendering to a heightmap.
Dynamic Updating	If enabled the heightmap is rendered before every simulation tick. This is most useful when using continuously updating flowmaps.
Preview Heightmap	Displays the heightmap in the scene.

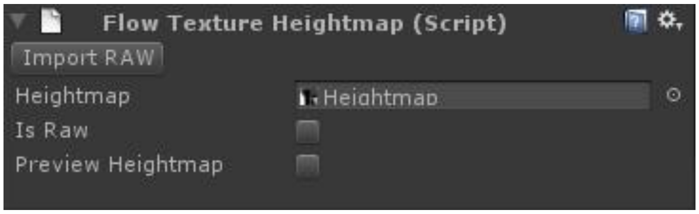
There are certain requirements for the geometry to be used. If the Fluid Depth is set to Deep Water the geometry should be watertight; there should be both front and back faces when looking at the geometry from a top-down view. Height Max when Fluid Depth is set to Deep Water should be set to a value larger than the highest intersecting geometry; the bounding box should enclose the highest intersecting geometry. The Height Min will control whether objects below the water surface will contribute to the heightmap. This can be used for objects that are only covered by a small amount of water.



Overhanging geometry such as bridges will be ignored, as long as they don't intersect with other geometry. It's often better to make sure they are not on the layer being used as the Culling Mask.

When Fluid Depth is set to Surface the Height Max and Height Min act as clipping planes, anything at Height Min or lower on the Y axis will be set to 0 in the heightmap. Anything above Height Max is ignored, so make sure your scene fits within the bounding box. Setting Height Min and Height Max larger than the bounds of your scene will cause the heightmap to have smaller values, and therefore affect the simulation less strongly.

Flow Texture Heightmap Component



Flow Texture Heightmap (Script)

Heightmap: Heightmap

Is Raw ☐

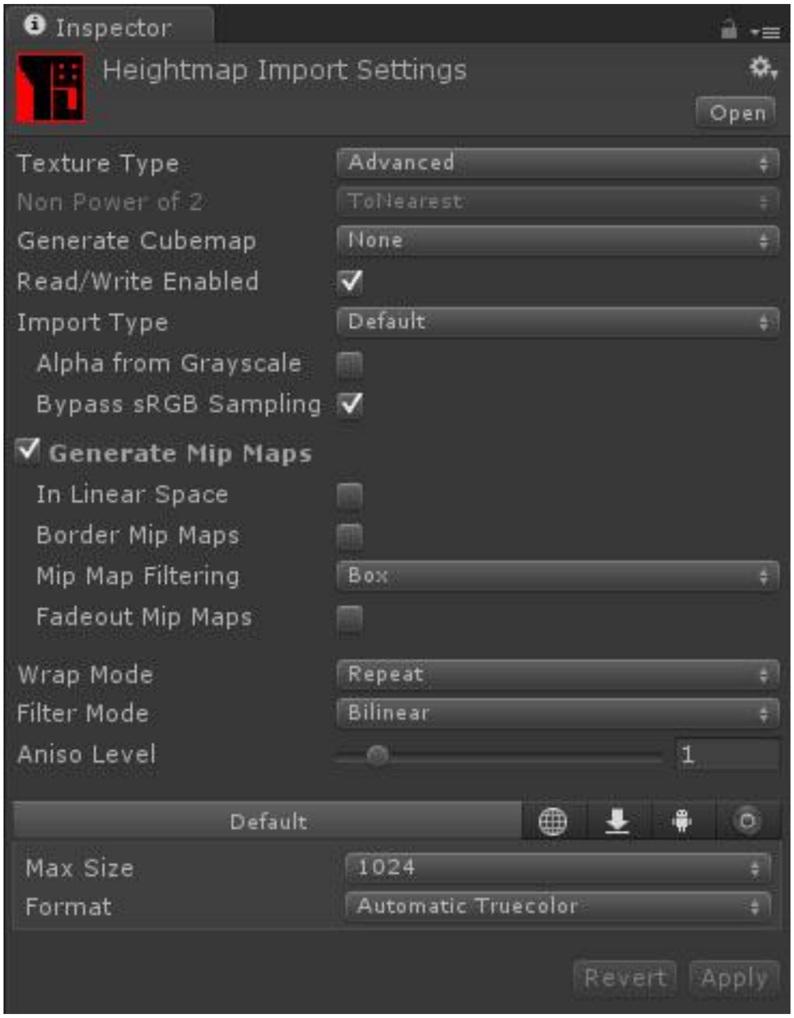
Preview Heightmap ☐

This component loads in a texture to be used as a heightmap in the simulation. For Unity Free this is the only way to load heightmaps. The heightmap can be created in a 3D application, or created by hand in an image editing application. The red channel will be used as a heightmap. Greyscale images work as well, as the red channel will contain the same information as all other channels.

Setting	Description
Import RAW	Import a 16bit .raw heightmap from file. The texture will be saved in the Unity project as a texture that uses all four channels to store a float value.
Is Raw	Should be enabled when using an imported Raw texture.
Preview Heightmap	Displays the heightmap in the scene.

Regular Heightmap Texture

The texture to be used needs to have some specific settings set. Here's how a non-Raw texture should be set up.



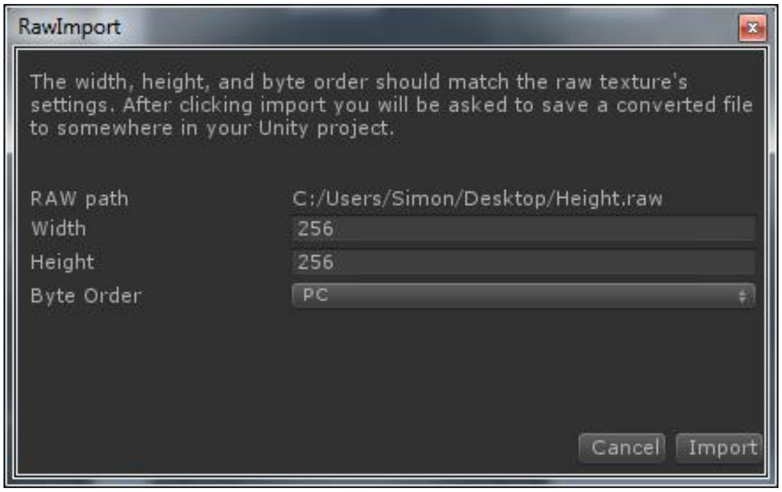
Heightmap Import Settings

Texture Type: Advanced
Non Power of 2: ToNearest
Generate Cubemap: None
Read/Write Enabled: ☑
Import Type: Default
Alpha from Grayscale: □
Bypass sRGB Sampling: ☑
☑ Generate Mip Maps
In Linear Space: □
Border Mip Maps: □
Mip Map Filtering: Box
Fadeout Mip Maps: □
Wrap Mode: Repeat
Filter Mode: Bilinear
Aniso Level: 1
Max Size: 1024
Format: Automatic Truecolor

The texture type should be set to Advanced so the extra settings are visible. Read/Write needs to be enabled. Bypass sRGB Sampling should be enabled. It's also a good idea to set the Format to Automatic Truecolor.

Raw Heightmap Texture

In some situations you may need more information in your heightmap. 8bit heightmaps may cause artifacts when there is a lot of smooth transitions between heights. This is often the case when generating surface flowmaps. A 16bit .raw heightmap can be imported and saved in your Unity project as a compressed texture, with the height information spread out over all four colour channels. Clicking the Import Raw button will open a file browser where you can select your .raw heightmap. Once selected you will need to set the import settings to match how the .raw file was saved.

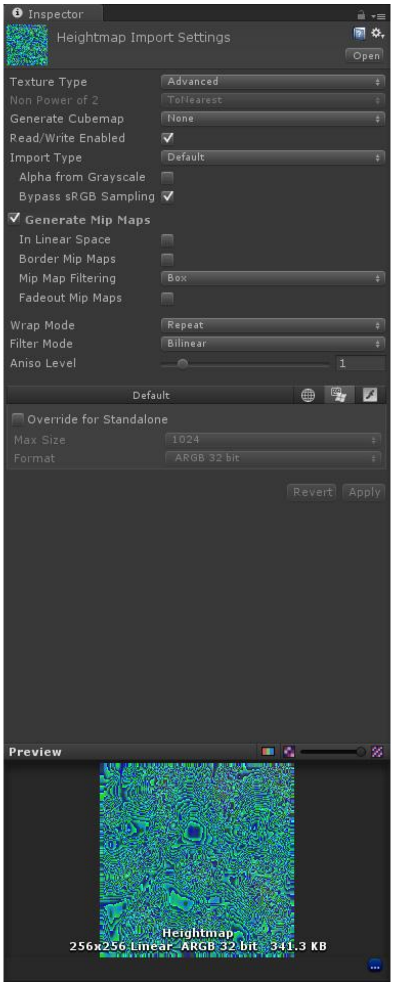


RawImport

The width, height, and byte order should match the raw texture's settings. After clicking import you will be asked to save a converted file to somewhere in your Unity project.

RAW path	C:/Users/Simon/Desktop/Height.raw
Width	256
Height	256
Byte Order	PC

After clicking Import, you will be prompted to save the compressed texture in your Unity project. The imported texture will look odd, but as long as the heightmap preview on the Flowmap Generator looks correct it will work.



Inspector

Heightmap Import Settings

Texture Type	Advanced
Non Power of 2	ToNearest
Generate Cubemap	None
Read/Write Enabled	☑
Import Type	Default
Alpha from Grayscale	□
Bypass sRGB Sampling	☑
☑ Generate Mip Maps	
In Linear Space	□
Border Mip Maps	□
Mip Map Filtering	Box
Fadeout Mip Maps	□
Wrap Mode	Repeat
Filter Mode	Bilinear
Aniso Level	1

Default

□ Override for Standalone

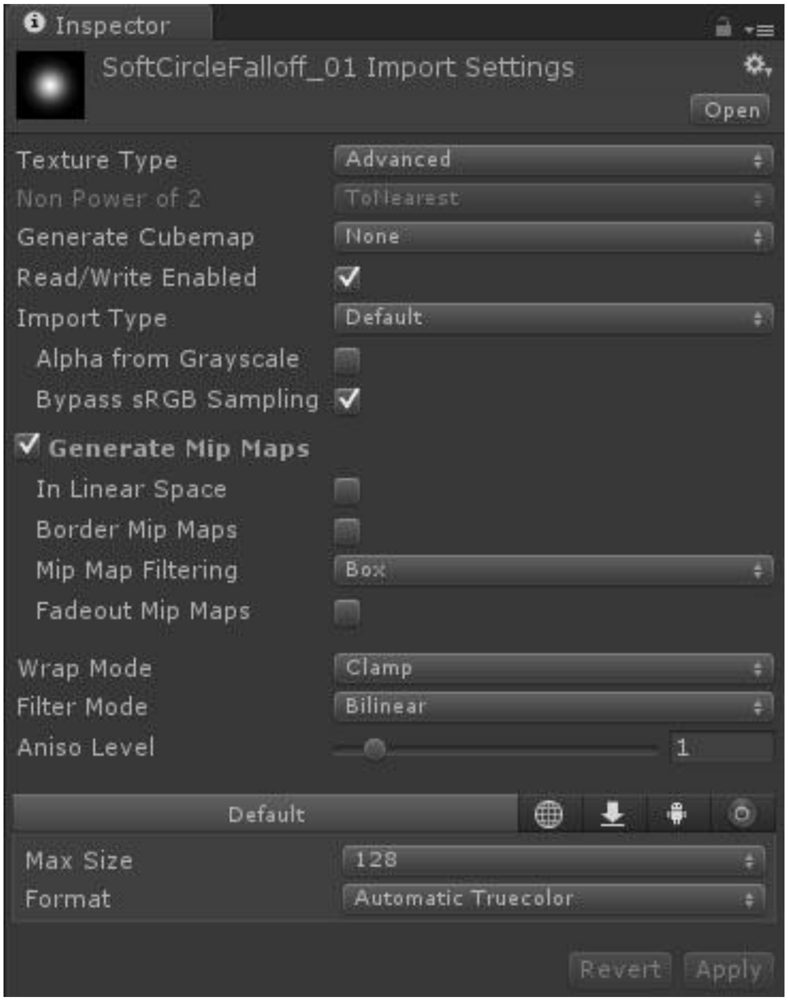
Max Size	1024
Format	ARGB 32 bit

Heightmap
256x256 Linear ARGB 32 bit 341.3 KB

Simulation Fields

Simulation fields allow you to control the fluid simulation. To have an effect on a generator they must be added to that generator's Simulation Fields list. If the field component is disabled it won't have an effect on the simulation.

The strength of a field is multiplied by the falloff texture. When using falloff textures and simulating on the CPU certain settings need to be set on the texture.

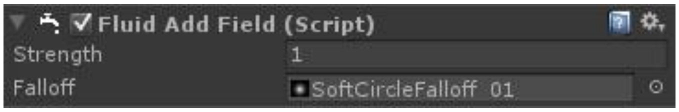


SoftCircleFalloff_01 Import Settings

Texture Type: Advanced
Non Power of 2: ToNearest
Generate Cubemap: None
Read/Write Enabled ☑
Import Type: Default
Alpha from Grayscale ☐
Bypass sRGB Sampling ☑
☑ Generate Mip Maps
In Linear Space ☐
Border Mip Maps ☐
Mip Map Filtering: Box
Fadeout Mip Maps ☐
Wrap Mode: Clamp
Filter Mode: Bilinear
Aniso Level: 1
Max Size: 128
Format: Automatic Truecolor

Setting Texture Type to Advanced will display the extra settings. Read/Write should be enabled as should and Bypass sRGB Sample. Setting the Format to Truecolor is a good idea as well.

Fluid Add Field



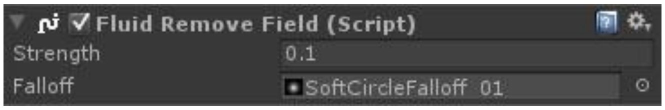
☑ Fluid Add Field (Script)

Strength: 1

Falloff: SoftCircleFalloff 01

A fluid add field adds fluid to the simulation. Every simulation needs some fluid, so there should be at least one fluid add field.

Fluid Remove Field



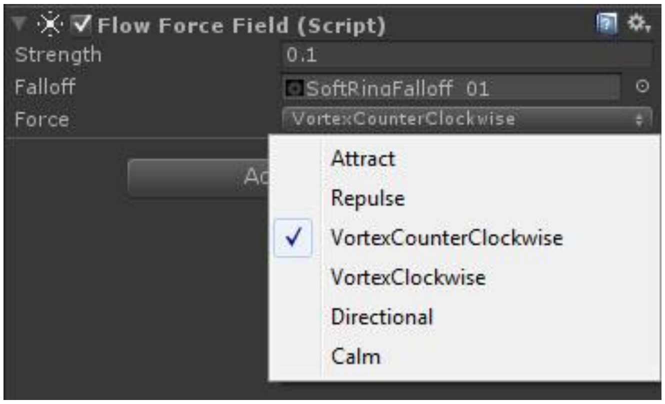
☑ Fluid Remove Field (Script)

Strength: 0.1

Falloff: SoftCircleFalloff 01

A fluid remove field acts as a drain, removing fluid from the simulation.

Force Field



☑ Flow Force Field (Script)

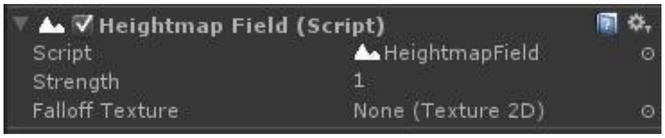
Strength: 0.1

Falloff: SoftRingFalloff 01

Force: VortexCounterClockwise

A force field applies a force to the fluid. Attract pulls fluid to the center of the field, repulse pushes away from the center, a vortex causes the fluid to swirl around the center of the field, directional pushes the fluid in the direction of the field's local Z axis, and calm slows down the fluid.

Heightmap Field



☑ Heightmap Field (Script)

Script: HeightmapField

Strength: 1

Falloff Texture: None (Texture 2D)

A heightmap field is blended on top the heightmap. This field works in Unity Free, and can be used to create a heightmap based on the scene by placing multiple fields.

Troubleshooting

Nothing happens when simulating

Some fluid must be present in the simulation. Try adding an Add Fluid field or setting the Initial Fluid Amount to a value larger than 0.

The Bake button is greyed out

To run a bake you need to set Max Simulation Steps to a value higher than 0 and have a valid Output Folder selected.

The simulation takes a long time to simulate

Make sure that the Timestep is set to a large enough value. It will be clamped to 40% of the Gravity value to prevent the simulation from giving incorrect values but try setting it as large as possible. The timestep only really needs to be changed when the flowmap dynamically updates, if a slower moving fluid is desired.

If you can't use GPU acceleration Flowmap Generator will use a multithreaded approach on the CPU. This usually is much too slow for dynamically updating. Setting a smaller resolution will help and flowmaps often don't require a large resolution for good results. In fact, sharper details in the flowmap often cause stretching artifacts.

A new field doesn't have an effect on the simulation

When using the CPU simulation path, all fields are baked at the start of the simulation. Try reseting the simulation and start simulating again.

Known Issues

GPU vs CPU simulations

The flowmap simulation when using GPU acceleration will give a slightly different result than the CPU simulation. It's recommended that you use the same simulation path for all flowmaps in a project.

GPU acceleration causes screen flickering and incorrect flowmaps

I've noticed that on my 2009 Macbook Pro that GPU acceleration causes some problems. It's recommended that you disable GPU acceleration if you encounter these problems.

Support

For more information and updates, visit http://www.superpositiongames.com/products/flowmap-generator/

To report a bug, send an email to support@superpositiongames.com