

**Fluidity** Documentation  
llt@dunnalex.com

# Getting Started

---

**NOTE :** In order to use Fluidity with Unity, you will need to have DirectX 11 enabled on your project. To do so, go to 'Player Settings' and under the 'Other Settings' tab check the tick box with the label 'Use Direct3D 11'.

## Creating the Fluid Controller

After importing Fluidity into your Unity project, you will find all the assets necessary to start making your fluid systems in the 'Fluidity' folder, in the root of your project.

To get up and running in an existing scene first you will need to create a fluid controller. Just create a new GameObject and add the 'Controller' component, this can be done by navigating to it in your project view ('Fluidity/Scripts/Simulation/Controller.cs'), or by using the component drop-down on the menu strip ('Component/Fluidity/Scene Controller Component...').

**NOTE :** It's worth noting at this point, that there should only ever be one fluid controller in your scene at a time, if you have more than one, the editor will show you an error message and your fluid systems will stop working.

Once you have the controller in your scene, it's time to create a fluid simulation.

## Creating Fluid Simulations

With the controller selected in the hierarchy, the inspector will show you the fluid simulation editor. From here you can press the '+' button (in the lower right of the editor) to create a new simulation. Your new simulation will appear above the '+' button.

To edit the parameters of the simulation you can press the green 'Edit' button next to your new simulation, and if you want to remove this simulation, just press the red 'X' next to the edit button. The options presented in this view will effect how the fluid system looks and perform, but you can read more about that in the 'Advanced' section of this document. For now, just give the simulation a name, and move on to the next step.

Once you have a simulation in your controller, you can now create a fluid volume in your scene and link it to the simulation.

## Creating Fluid Volumes

To create a fluid volume, just create a new GameObject in the scene, and add the 'FluidVolume' component to it. This can be done by navigating to the component in the project ('Fluidity/Scripts/Volumes/FluidVolume') or by using the component drop-down menu on unitys menu strip ('Component/Fluidity/Fluid Volume...').

Once you have created your fluid volume, you will notice in the scene view that a green box has appeared around the fluid volume, this box represents the size of the fluid volume, please resize and position it to your pleasing.

At this point it's worth noting that you should change the colour gradient of the fluid volume to something more appropriate than the default, to do this, select the fluid volume in the hierarchy, go to the inspector

and look at the section titled 'Colours'. In there you will notice a drop down titled 'Channel 1'. If you open this up, and click on the colour gradient bar, you can change the colours to your pleasing. The right hand side of the bar represents the colour of the fluid when it has reached the end of it's life (so you probably want to use a colour with 0 alpha) and the left hand side represents the fluid colour when it has just been injected into the simulation.

## **Rendering**

Finally, before you can see the fluid simulations being rendered in the scene, you will need to add a 'VolumetricRenderer' component to any camera that needs to render fluid.

Select the camera you wish to have render fluids and add the 'VolumetricRenderer.cs' script ('Fluidity/Scripts/Rendering/VolumetricRenderer') to it. This script acts like a post effect, so if you wish to, you can move it up and down in the chain, just right click on it to open the context menu and select move up or down. This will allow you to apply post effects to the fluid simulations after rendering.

## **Finally**

Hit play in the inspector and you should see your first fluid system emitting fluid and rendering to the scene.

Now that you have everything in place to start making basic fluid systems in your scene., read on to the 'Advanced' section to learn more about some of the things that have been glossed over in this section.



## Advanced

---

### Simulations

Creating more advanced simulations is easy, I will walk you through all of the options available to you with Fluidity.

Select your fluid controller in the hierarchy and in the inspector you should see the simulation editor. If you haven't already created a fluid simulation yet, then please refer to the 'Getting Started' section of this document to see how.

Once at the simulation editor, simply click the green 'Edit' button on the simulation of your choice and underneath you will be presented with a bunch of options relating to your simulation.

- **'Grid Resolution'** - This is essentially the resolution of the simulation, higher numbers will give you more detailed looking systems at the price of performance.
- **'Emit Fluid'** - With this enabled the simulation will continuously inject fluid into the volume, disabled the fluid will stop.
- **'Ignore Culling'** - As an optimization, fluid simulations will automatically stop updating when they aren't in view of any cameras in the scene, with this option enabled, this fluid system will always update.
- **'High Quality Simulation'** - With this option enabled, the simulation will use a more expensive compute shader to calculate the fluid dynamics, this allows you to control the pressure solver iterations exactly and yields better looking fluid.
- **'MacCormack Advection'** - Provides more realistic fluid dynamics, velocity in the simulation is resolved more accurately.
- **'Vorticity Confinement'** - Again, this provides more realistic fluid dynamics, velocity in the simulation is resolved more accurately by taking into account vortices.
- **'Reaction Speed'** - This is the rate of decay of the simulation, ie. how quickly the fluid will dissipate. This can be used to fade the fluid off as it's lifetime expires.
- **'Emitter Settings'** - In this section, you can control how the fluid will be injected into the volume. Currently, there are two options for doing this, the default is to use Sphere emitter, but you can also emit fluid from a texture. In sphere mode you can specify the radius, position and direction of the emitter. You can see the position and direction of the emitter in the scene represented by the fluidity icon and white arrow in your fluid volume. In texture mode you just need to apply a texture with some data in the RED channel and the simulation will emit the fluid from the bottom (local to the volume) of the volume.

### Optional

Some of the options that you can see in the simulation editor will only appear when you are using certain modes.

- **'Iterations'** - Only available with 'High Quality Simulation' enabled. This allows you to control how many iterations used to solve the pressure in the fluid system. Greater numbers yield more realistic results at the price of performance.
- **'Vorticity'** - Only available with 'Vorticity Confinement' enabled. This is the scale at which to apply vortices when solving for velocity in the system, changing this number will affect the look of your fluid, but should not degrade performance.

## Fluid Volumes

Fluid volumes embed a fluid system into the scene and control how they will look when rendered to the screen.

With a fluid volume selected in the hierarchy, you can see in the inspector there are some options available:

- **'Simulation'** – Under this tab, you can select the drop down menu to choose which simulation you would like to render in this volume. Multiple volumes can reference the same fluid simulation, this is called simulation instancing.
- **'Colour Channels'** – The amount of gradient channels to use when rendering this volume. This can be used in conjunction with *Temporary Emitters* to render fluid emitters with more than one colour gradient. When not using temp emitters, the main emitter for this volume will just use the first channel.
- **'Channel 1'** – This is the main colour channel for this volume, the main emitter will use this channel by default. The gradient represents fluid when dissipated to freshly injected fluid from left to right.
- **'Render Mask'** – Rendering inside the volume can be masked using a mask texture, this allows you to create fluid volumes that fit perfectly into parts of your scene which aren't necessarily a box.
- **'Intensity'** – This is simply a modifier for the gradient curve, it controls the range of the texture to use when rendering.
- **'Link Light Source'** – Sometimes you will need to link a light source to a fluid system when conveying an effect like a fire. You can create a light source and link it to it using this field. The linked light source will modulate its colour based on the values from the gradient in channel 1.
- **'Use Active LOD'** – Active LOD is a setting which allows you to vary the amount of render steps depending on how far away the volume is from the rendering camera.
- **'Render Steps'** – When not using Active LOD, the volume will be rendered using a fixed amount of steps, defined here.

## Fluid Effectors

Fluid effectors currently can be used for two things; 1) Introduce spherical collisions into the fluid simulation, and 2) Introduce temporary emitters in to the fluid simulation.

To create an effector, first you will need a GameObject you want to use as an effector. Then add the 'Effector' component. This can be done by navigating to the script in the project view ('Fluidity/Scripts/Tools/Effector.cs') or by using the component drop-down menu ('Component/Fluidity/Effector...').

Once you have an effector in your scene, you can control its radius from the inspector, a visual representation of this can be seen in the scene view (blue sphere).

## Colliders

When using the collider functionality of effectors, the effector will now block fluid when inside a fluid volume. The radius (indicated by the blue sphere around the effectors transform in the scene view) effects the distance around the sphere to block fluid.

## Temporary Emitters

When using the temporary emitter functionality of effectors, the effector will now become another sphere emitter when inside of a volume. The optional parameters of the effector editor are now available to you.

- **'Power'** – Controls the force at which fluid will be injected into the system from this effector.
- **'Channel'** – Temporary emitters can write fluid data into the optional extra channels on a fluid volume, the value of this parameter represents the channel gradient index on the volume.

## Further Help?

---

If you ever need any help settings things up, or if you spot a bug, or even if you just want to share your fluid creations, let us know!

Email : [llt@dunnalex.com](mailto:llt@dunnalex.com)

Thanks for buying Fluidity!  
From : Low Level Tech.