

Physics/Fluids Plugin Beta 0.5

READ ME

This is the most important thing to read. Just follow these instructions and most of the time things should work¹.

- Only vertices are used.
- All objects used by the plugin should be triangle meshes and should have the shading option set to “smooth”.
- Making the particle size smaller means that the space between vertices in obstructions will also need to be reduced.
- Smaller particle sizes means the time step also needs to be smaller.
- If the simulation “blows up” reduce the time step.
- Keep an eye on the Logs. Sometimes there are hints on what is wrong.
- Preview animation does not work. I have no idea why.
- future versions of the plugin will probably not load files from previous versions while in beta.

AoI Fluid Plugin Beta

This document is well, simple, and just meant to get everyone started with the physics plugin. It is not complete but should serve and be a starting point for a good tinker with the code as it is now. Note that this document will only be valid for the beta. In particular the beta 0.5. The GUI is likely to change dramatically over the course of time to version one.

Unfortunately to use fluids properly you need a little background on the method used to solve the mathematics that describes fluids. So please read this whole document. Its important to always know that it is a simulation, and an approximation to what a real fluid would do. It has and will always have limitations. You will need to work with them, because even though the final version will have a lot more features than the current beta, some of the limitations will still be present. The single biggest limitation is computational power. Fluids are hard and hence slow to simulate. Also there are 3d scaling laws that a lot of people don't appreciate. For example if I double the size

¹Yea right.

of my swimming pool I have 8 times more fluid to simulate, and so will take 8 times longer to solve. Setting everything up so that simulations won't take so long is important.

SPH numerical method

The approach I use is called Smoothed Particle Hydrodynamics (SPH). As the name implies the fluid is represented as a set of particles. However the particles are not like say a whole lot of ping pong balls, because the particles have no sharp boundary, they overlap each other. This is the smoothing part of the name. At each time step the force on each particle from every other overlapping particle is calculated and its new position calculated. At this point the time between steps is fixed. Large time steps means we need to do less calculations, but unfortunately if the time steps are too small things blow up. What this means is that particles can move too far in a single step, creating forces that are too large which leads to particles that move even further in the next step, and so on. Its called numerical instability and will happen whenever the time step is too large². The time step required depends on the particle size as well. So halving the particle size, should be followed by halving the step size. Further you end up with 8 times more particles for the same volume, so the whole thing will be 16 times slower. Always use the largest size particles you can get away with.

Because the fluid is a set of particles, the easiest way to get obstacles and the like into the simulation is to make them a set of particles as well. Therefore the only thing that this plugin uses are vertices. It does not work with triangles, closed surfaces or edges. An important consequence of this is that you need to convert everything you use into a triangle mesh, and then subdivide it finely enough so that the fluid won't leak. Another very important point is that it uses vertex normals. The only way to get all vertices in a triangle mesh is to have normals defined for all vertices, is to use the "smooth" shading method³.

Current Status, and development cycle

The current status is very much a beta. Not a GPL/GNU type beta either, more like the commercial worlds version of a beta⁴. So don't expect everything to just work. It won't. In particular the GUI is not even close to a

²Note that this will also cause the simulation to slow down dramatically. Sometimes it even appears "locked up", however its just unresponsive

³As far as I know. I might end up doing a few extra plugin tools to help manage normals. But we will see what others think.

⁴That's much worse than a GPL/GNU beta

final version. It really is just a cut on the basic idea on how to design the plugin. We are expecting plenty of feedback and ideas in this regard, but bear in mind the “whys” of the current design.

The design ideas are based around isolation. That is, the only thing that knows about fluids is the fluids plugin. This affects the basic GUI design at a fundamental level, and provides isolation from the rest of AoI as much as its possible. It also means that the way you interact with the fluid plugin is perhaps a little less conventional. Many of the basic ideas are taken, more or less, from blenders fluid implementation. So it bakes a simulation to a file.

I will explain the GUI below, but must make one point here. The advanced panel has a spawn button. This uses java reflection to find any “beanish” properties that it can edit. Most, if not all of these parameters should *NOT* be touched. They are there for workarounds. So if its possible to sidestep a bug without an update we can. But these are not generally editable parameters.

Finally it is my intention that there will be a bit of time before a big update. So I want the bug reports to accumulate and then go over all of them and perhaps restructure things if needed.

Simple scene

Right so now onto the meat of this document. Lets start with an empty file so we get to go over all the details. For this exercise I want to produce an inflowing section of water into a box in such a way that is swirls.

1. Create a big box in one of the four screens.
2. Convert it to a triangle mesh.
3. Edit this object and delete a vertex in the middle of a face. You now have a box with one end open.
4. Select all remaining vertices and set edge mode of the editor.
5. Subdivide edges repeatedly until you have a fine mesh as shown in figure 1.
6. Set the shading method to “smooth”.
7. Use the top view to ensure that the open end of the box points up. If not rotate the box until the open end is pointing up.
8. Create a small flat type box inside previous box and convert to a triangle mesh.

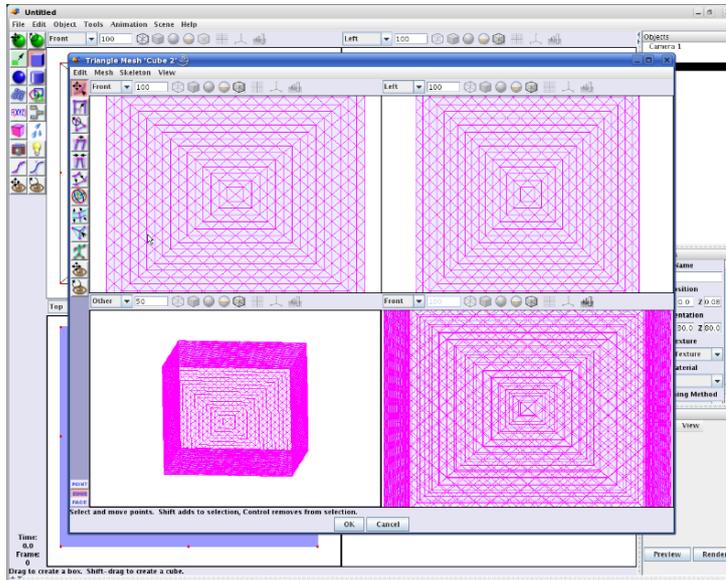


Figure 1: What the box should look like

9. Edit the small box and delete all the vertices except for one face.
10. Again set edge mode. Select all edges and subdivide until you have about the same spacing between vertices as the previous triangle mesh (see Figure 2).
11. Now select the fluid tool. It the one with the drops of water.
12. Drag a box completely around the main Box.
13. Edit the physics object. (right mouse click in the objects window on the right.)
14. Under the object management tab you will see the 2 objects you have created listed. Most of the options are for “inflow” objects only. Set the fluid property for the big Box to obstruction. This object is not animated so we leave the isMoving checkbox unchecked.
15. Set the fluid property for the small “square” to inflow. Now the normal factor is how much velocity in the normal direction the created fluid should have, set this to something between 2 and 5. The velocity factor is how much of the objects velocity the fluid should have. This source object is not animated so we can ignore this. Finally we have the density factor. This controls how low the relative density should

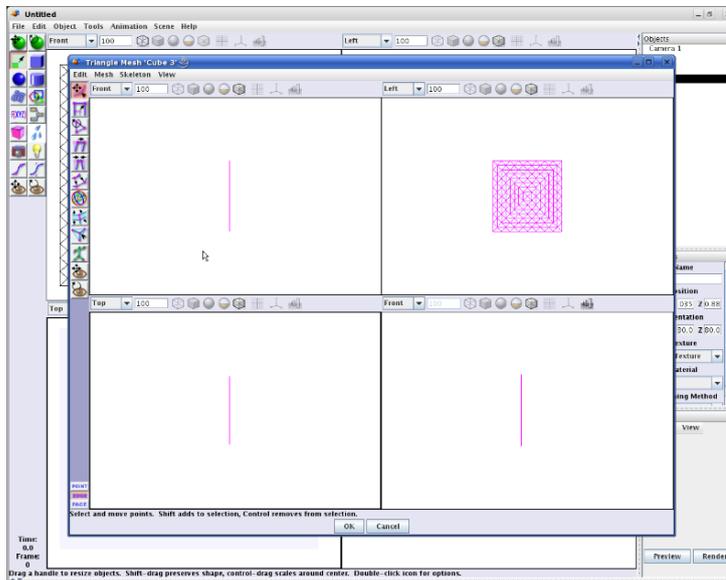


Figure 2: The source object. Note that we only have vertices from one face left.

be before new particles are spawned. This should always be below 1 IIRC. I usually use about 0.5. Finally this object is not moving so we don't need to check the isMoving box.

16. Note that under the Object management tab, objects can be set to fluid. This option has no effect at this time. However outflow does. Any fluid that gets close to an outflow vertex is removed from the simulation.
17. Select the Fluid Params tab. Note that if you have a slow computer you should probably leave this alone. But if you have a lot of ram and a fast CPU you can change the particle size to 0.1. Other options are not needed as the time step is small enough.
18. Select the Baking tab. All simulations are done though this tab. You back them to the file name listed at the top. If this is not pointing to a drive with enough free space (100+MB) change it. Baking can be cancelled at anytime. But the latest simulation frame must finish before the code checks to see if cancel was pushed. So if you have big simulations this can be unresponsive.
19. Click on bake. The bake progress window is always too small, just resize it so you can see current progress. Once about 1 second has been

simulated, cancel the simulation and click on OK in the main editor window. Should now see something like figure 3.

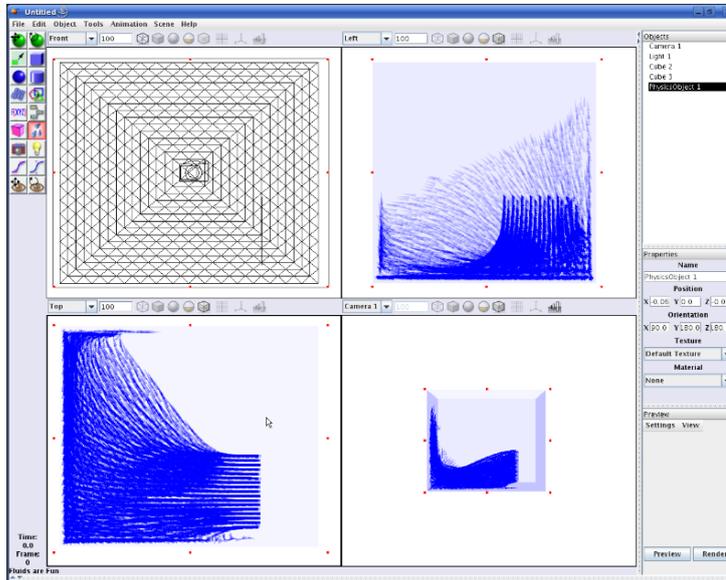


Figure 3: The result after one second of simulation.

20. Edit the physics object again.
21. Under the object management tab, make the source object (the one that's inflow at this point in time) fluid property "none".
22. Go to the bake tab. For the change to take effect, we need to click on the "clear object cache" button. Then bake again for as long as you like. The fluid that was already there continues to get simulated, but no extra particles are added. You can move to anytime you want and rebake from that time. But don't change the particle size. you need to "restart" by first setting the time to zero and then clearing the bake file.

So that's a run through. I wanted to have a better set of examples and a video tut. But its already taken too long. I will try to provide more info on the other parts of the GUI during the week.