Post-processing

C. Fernandes, L.L. Ferrás, J.M. Nóbrega

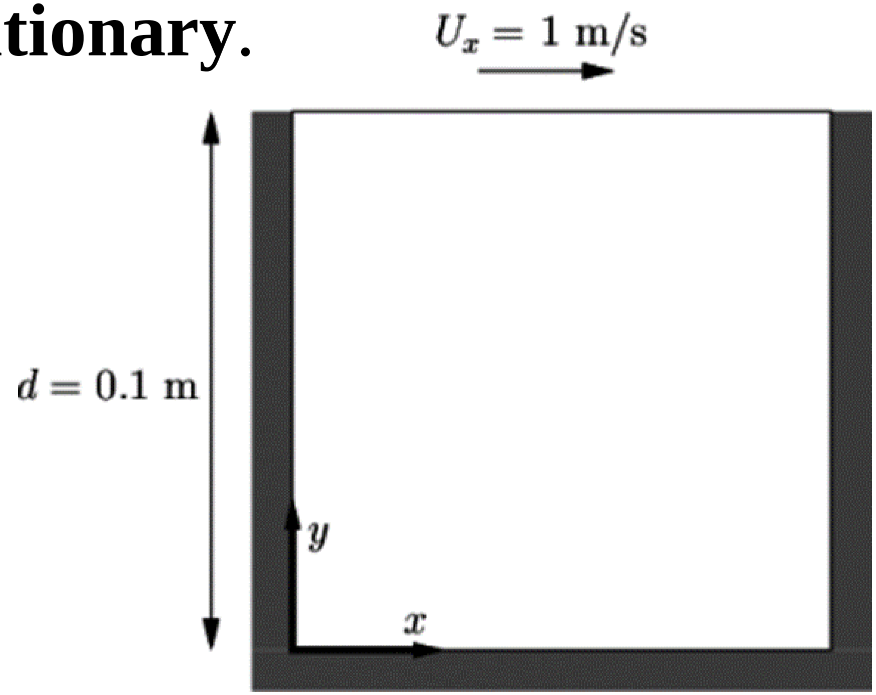
Tutorial

Lid Driven Cavity

Flow Involving Isothermal and Incompressible Fluid

The geometry is a **two-dimensional** square domain which all the boundaries of the square are walls.

The **top wall moves** on the x-direction at a speed of 1 m/s while the **other three are stationary**.



Inside the **$FOAM_RUN** directory you will find the **cavity** case.

On the terminal **enter inside the cavity case**

```
>> cd $FOAM_RUN/cavity
```

Create the mesh

```
>> blockMesh
```

Run the solver

```
>> icoFoam
```

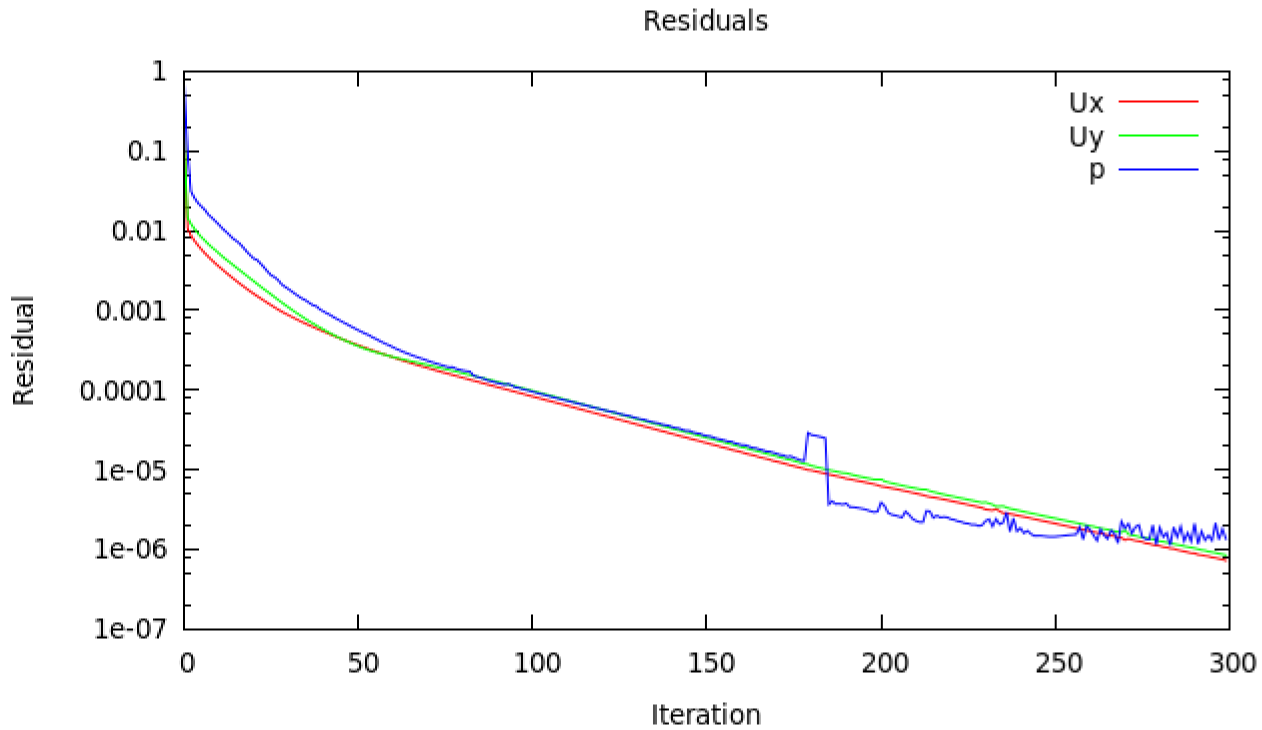
We can **check the convergence** of the simulation using the utility **gnuplot to draw the residuals** of pressure and velocity profiles.

```
set logscale y
set title "Residuals"
set ylabel 'Residual'
set xlabel 'Iteration'
plot "< cat log_simulation | grep 'Solving for Ux' | cut -d' ' -f9" title 'Ux' with lines,\
"< cat log_simulation | grep 'Solving for Uy' | cut -d' ' -f9" title 'Uy' with lines,\
"< cat log_simulation | grep 'Solving for p' | cut -d' ' -f9 | sed -n 'p;N' | tr -d ','" title 'p' with lines
pause 1
reread
```

Save file with name **Residuals**.

The command to see the graphic is

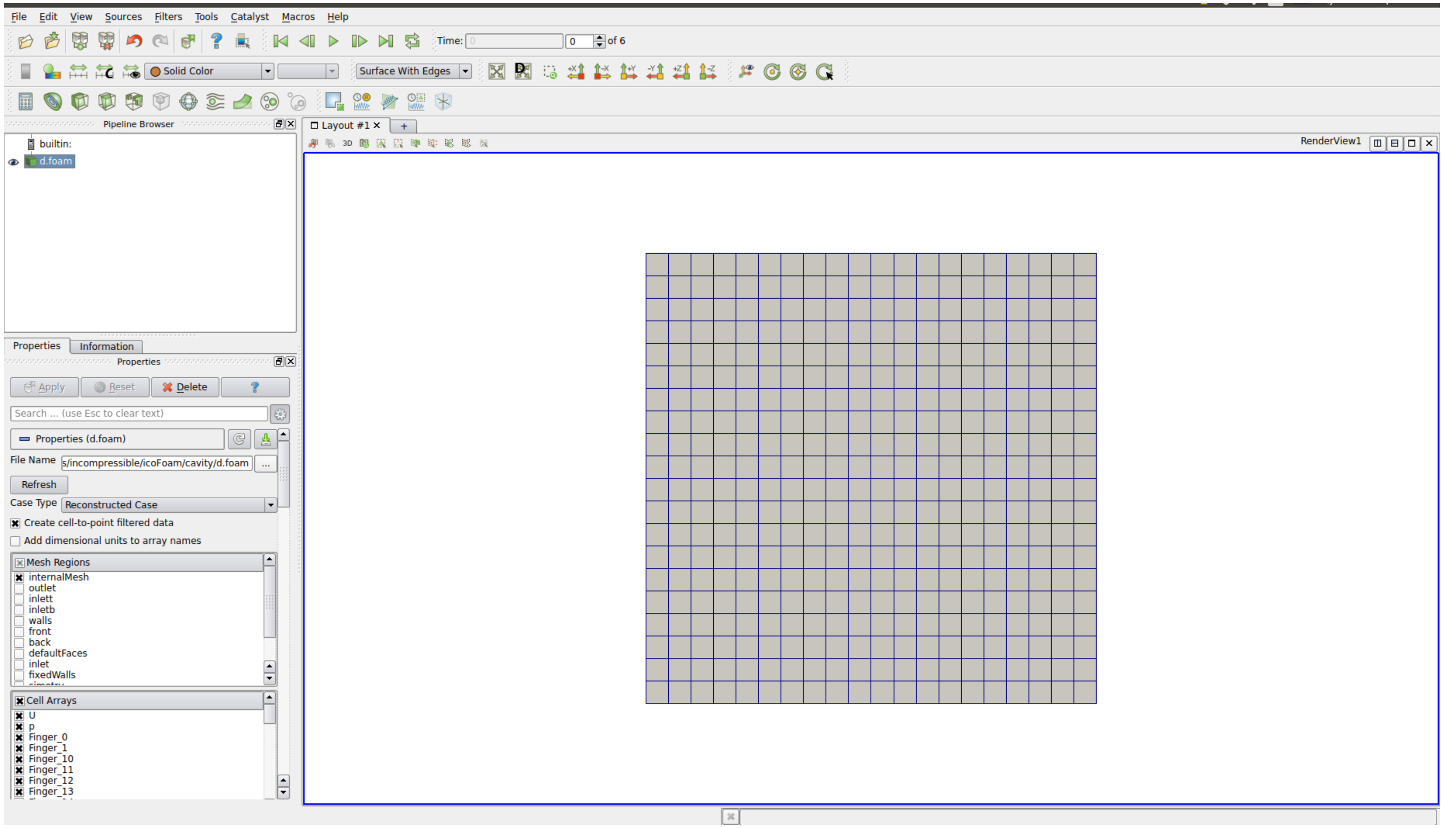
>> gnuplot Residuals

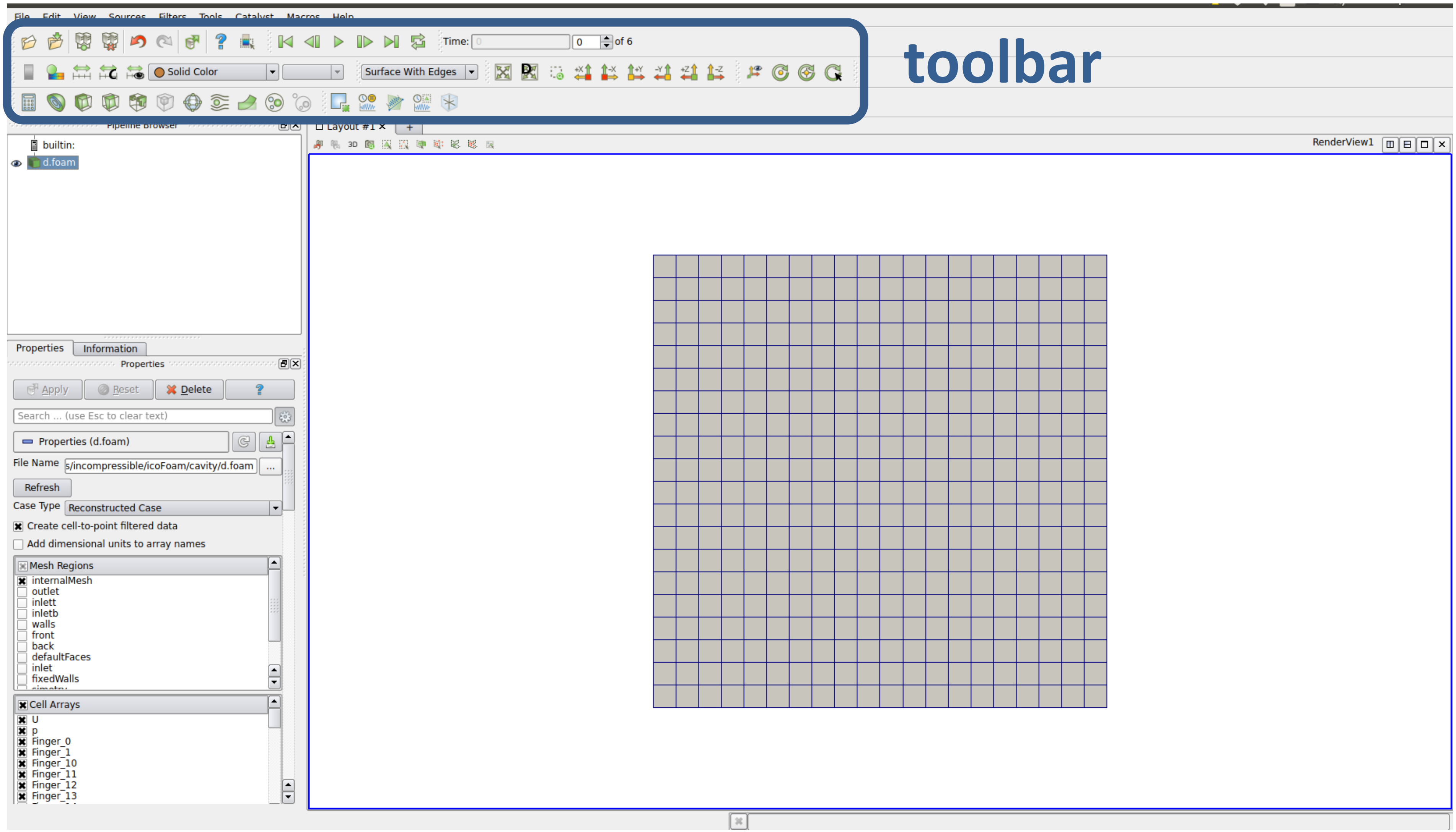


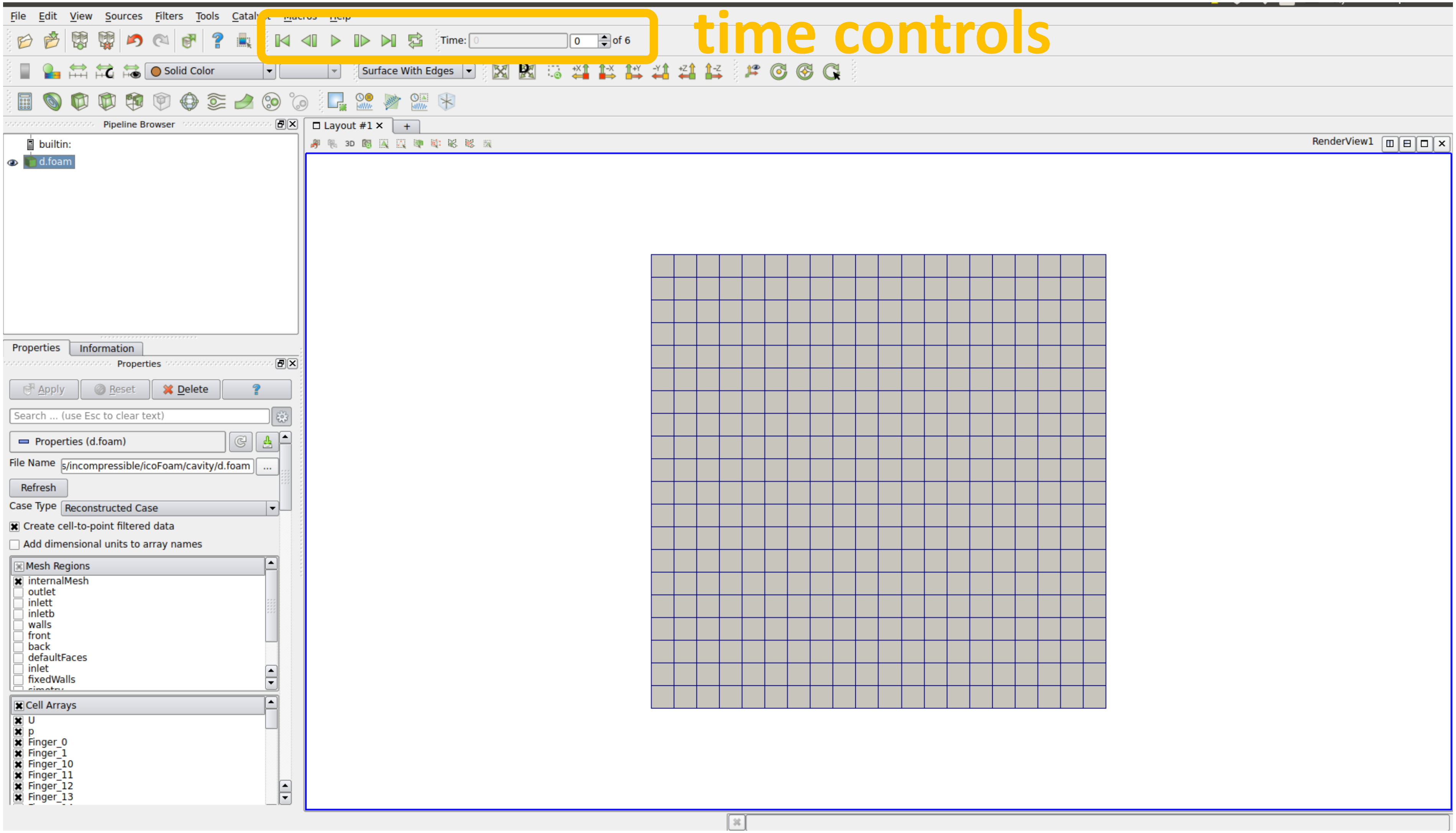
View results

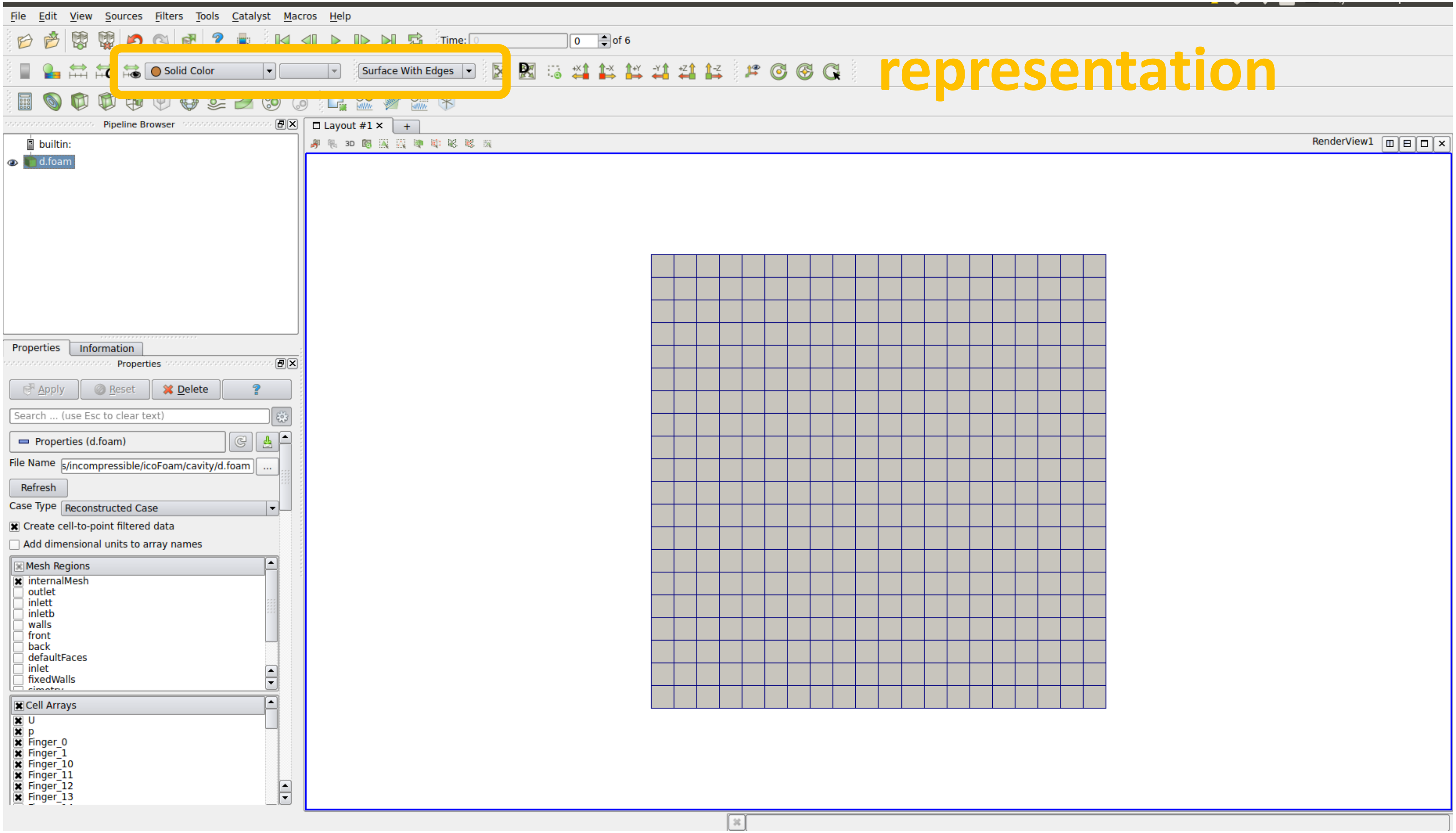
>> touch cavity.foam

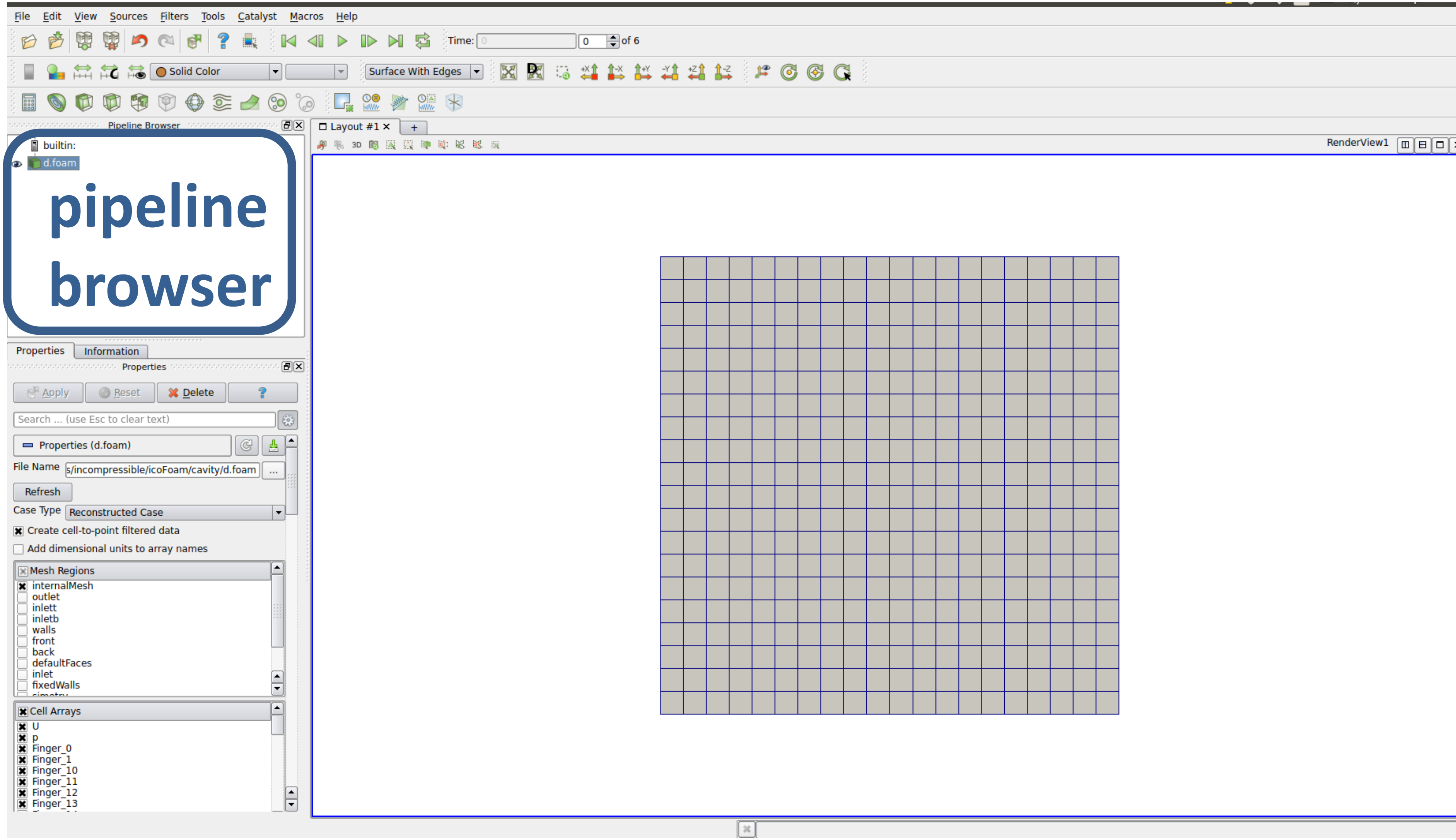
>> paraview cavity.foam

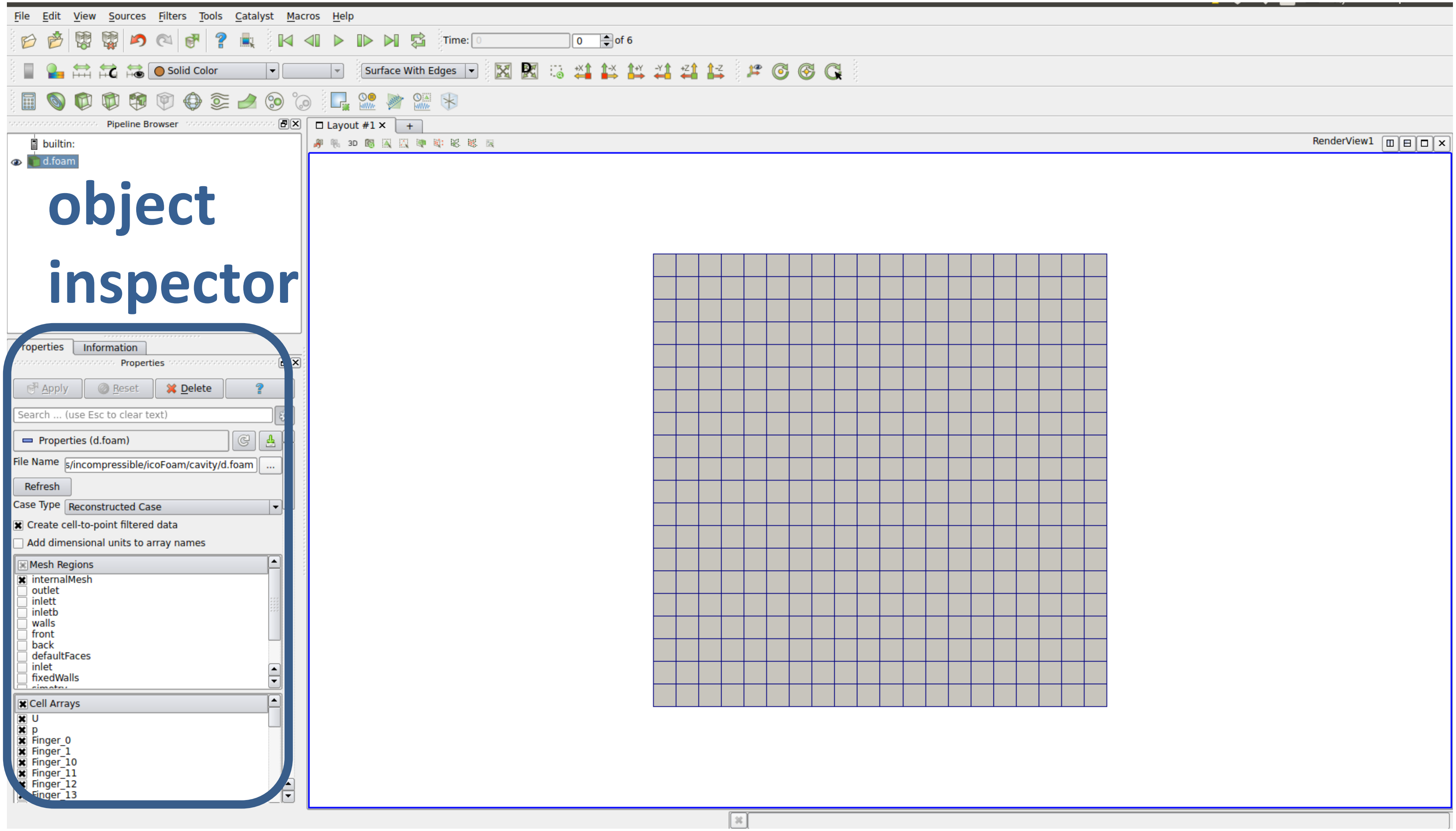


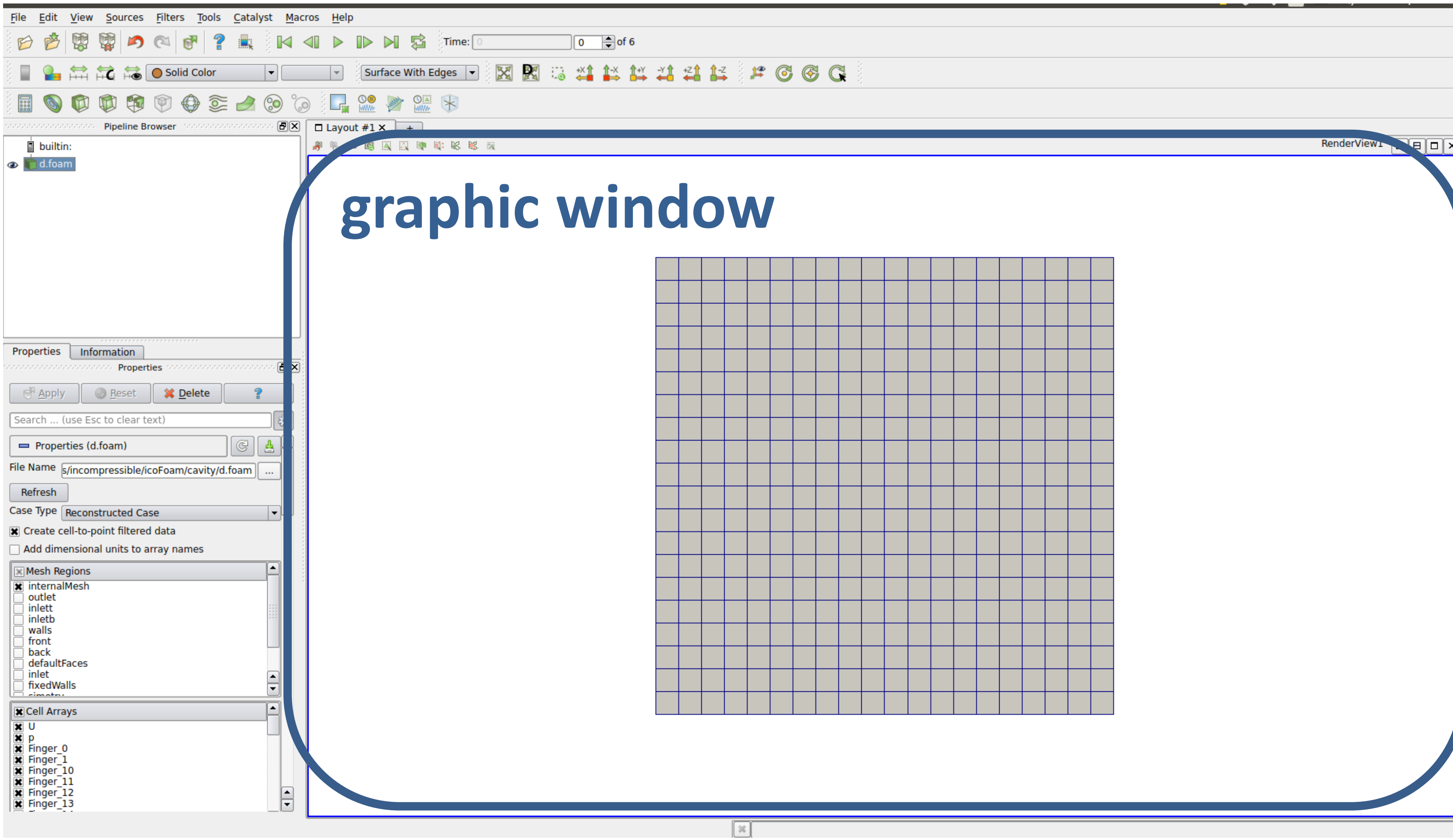

toolbar
Time:
of 6
Solid Color
Surface With Edges
Layout #1
RenderView1
builtin:
d.foam
Properties
Information
Apply
Reset
Delete
Search ... (use Esc to clear text)
Properties (d.foam)
File Name
s/incompressible/icoFoam/cavity/d.foam
Refresh
Case Type
Reconstructed Case
Create cell-to-point filtered data
Add dimensional units to array names
Mesh Regions
internalMesh
outlet
inlett
inletb
walls
front
back
defaultFaces
inlet
fixedWalls
Cell Arrays
U
p
Finger_0
Finger_1
Finger_10
Finger_11
Finger_12
Finger_13


time controls
File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0 of 6
Solid Color
Surface With Edges
Pipeline Browser
builtin:
d.foam
Layout #1
RenderView1
Properties Information
Properties
Apply Reset Delete
Search ... (use Esc to clear text)
Properties (d.foam)
File Name s/incompressible/icoFoam/cavity/d.foam
Refresh
Case Type Reconstructed Case
Create cell-to-point filtered data
Add dimensional units to array names
Mesh Regions
internalMesh
outlet
inlett
inletb
walls
front
back
defaultFaces
inlet
fixedWalls
Cell Arrays
U
p
Finger_0
Finger_1
Finger_10
Finger_11
Finger_12
Finger_13

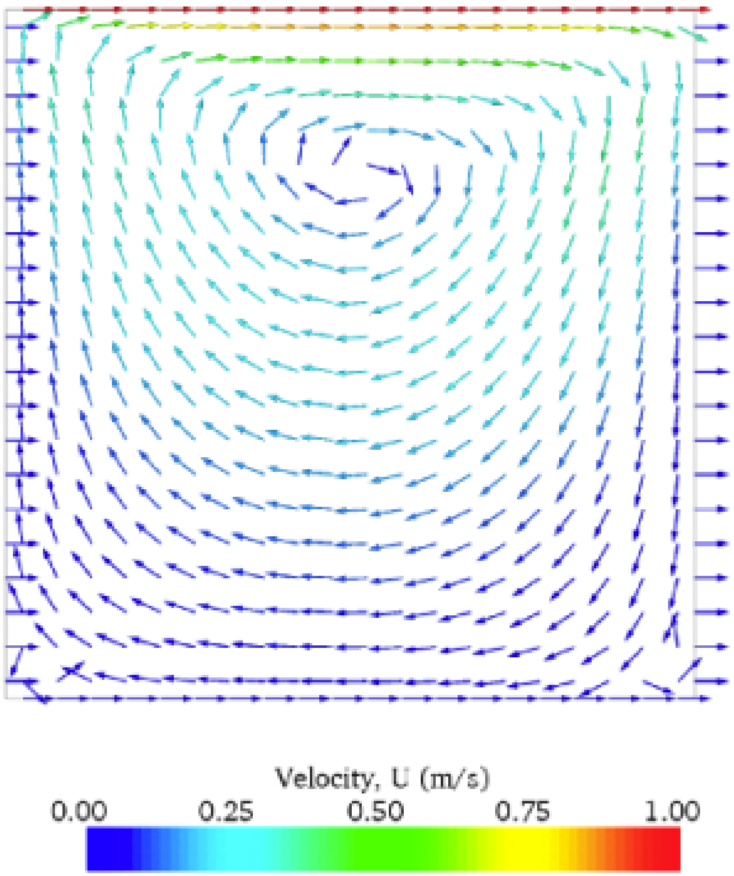

File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0 of 6
Solid Color
Surface With Edges
representation
Pipeline Browser
builtin:
d.foam
Layout #1
RenderView1
Properties Information
Apply Reset Delete
Search ... (use Esc to clear text)
Properties (d.foam)
File Name s/incompressible/icoFoam/cavity/d.foam
Refresh
Case Type Reconstructed Case
Create cell-to-point filtered data
Add dimensional units to array names
Mesh Regions
internalMesh
outlet
inlett
inletb
walls
front
back
defaultFaces
inlet
fixedWalls
Cell Arrays
U
p
Finger_0
Finger_1
Finger_10
Finger_11
Finger_12
Finger_13


File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0 of 6
Solid Color
Surface With Edges
Pipeline Browser
builtin:
d.foam
pipeline browser
Layout #1
RenderView1
Properties Information
Properties
Apply Reset Delete
Search ... (use Esc to clear text)
Properties (d.foam)
File Name s/incompressible/icoFoam/cavity/d.foam
Refresh
Case Type Reconstructed Case
Create cell-to-point filtered data
Add dimensional units to array names
Mesh Regions
internalMesh
outlet
inlett
inletb
walls
front
back
defaultFaces
inlet
fixedWalls
Cell Arrays
U
p
Finger_0
Finger_1
Finger_10
Finger_11
Finger_12
Finger_13

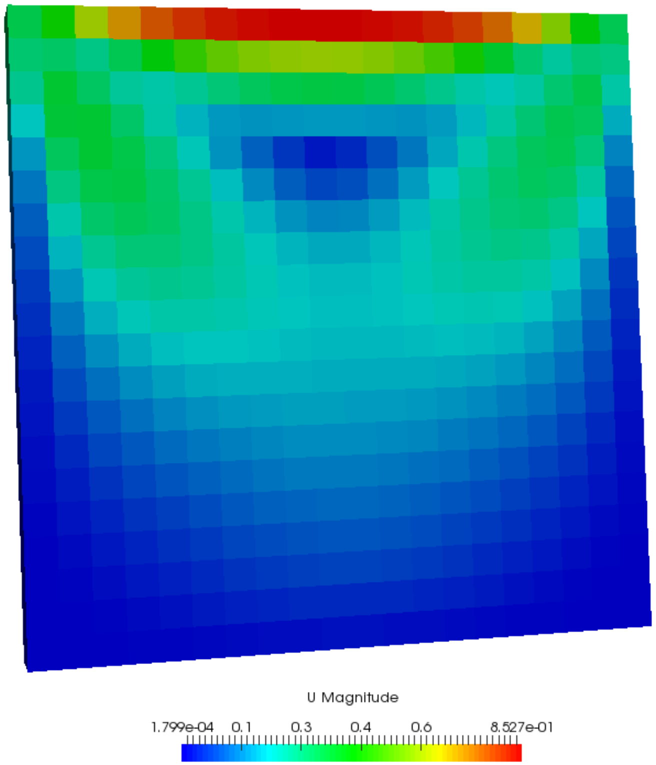

File
Edit
View
Sources
Filters
Tools
Catalyst
Macros
Help
Time:
0
of 6
Solid Color
Surface With Edges
Pipeline Browser
builtin:
d.foam
object
inspector
Layout #1
RenderView1
roperties
Information
Properties
Apply
Reset
Delete
Search ... (use Esc to clear text)
Properties (d.foam)
File Name
s/incompressible/icoFoam/cavity/d.foam
Refresh
Case Type
Reconstructed Case
Create cell-to-point filtered data
Add dimensional units to array names
Mesh Regions
internalMesh
outlet
inlett
inletb
walls
front
back
defaultFaces
inlet
fixedWalls
Cell Arrays
U
p
Finger_0
Finger_1
Finger_10
Finger_11
Finger_12
Finger_13

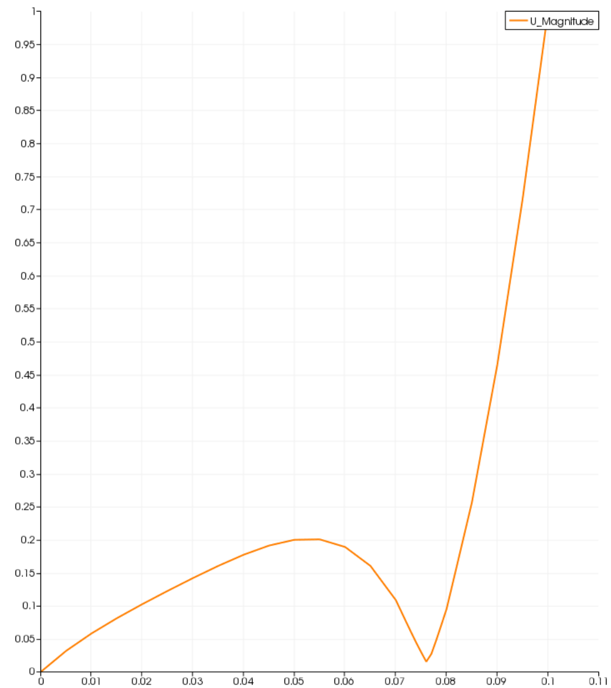

File Edit View Sources Filters Tools Catalyst Macros Help
Time: 0 of 6
Solid Color
Surface With Edges
Pipeline Browser
builtin:
d.foam
Layout #1
RenderView1
graphic window
Properties Information
Properties
Apply Reset Delete
Search ... (use Esc to clear text)
Properties (d.foam)
File Name s/incompressible/icoFoam/cavity/d.foam
Refresh
Case Type Reconstructed Case
Create cell-to-point filtered data
Add dimensional units to array names
Mesh Regions
internalMesh
outlet
inlett
inletb
walls
front
back
defaultFaces
inlet
fixedWalls
Cell Arrays
U
p
Finger_0
Finger_1
Finger_10
Finger_11
Finger_12
Finger_13

Use **Glyph** option to plot **velocity field**

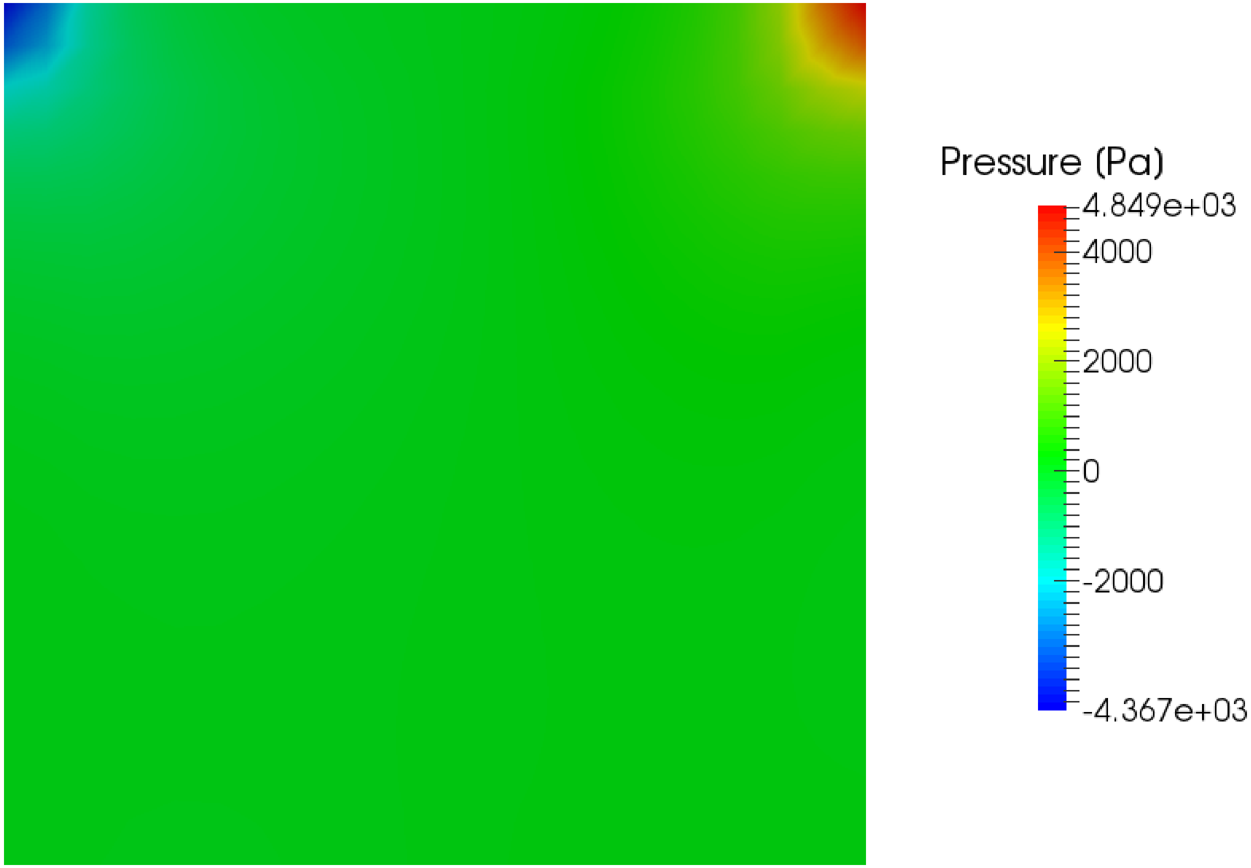


Plot Over Line the **velocity field** in the middle of the channel along ***y*** direction, P1(0,05;0;0,005) and P2(0,05;0,1;0,005).

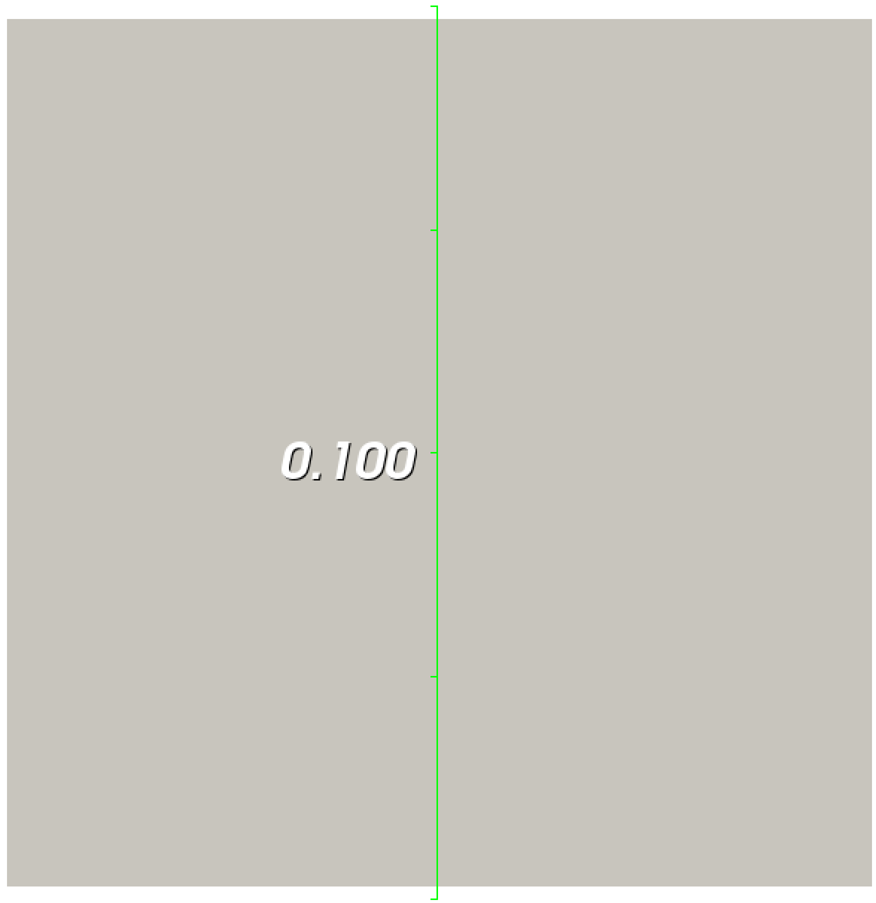




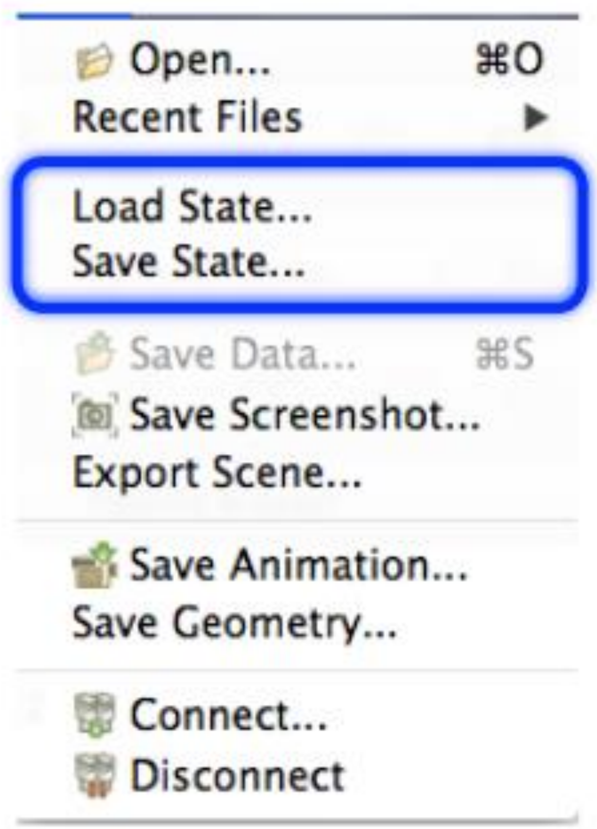
Use **Calculator** to plot **real pressure** (assume density equal 1000 kg/m^3)



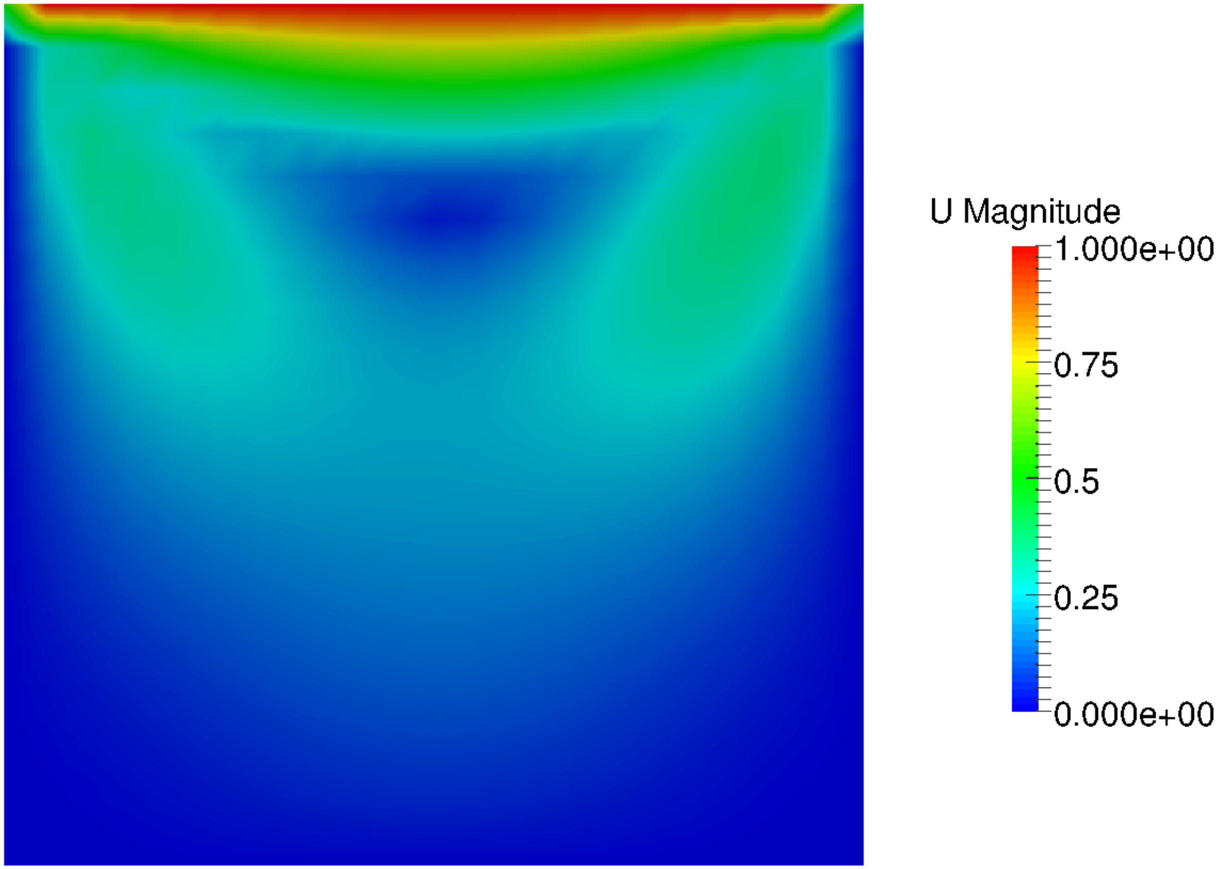
Sources



Load/Save States



Export Scenes



Tutorial

Goldschmidt

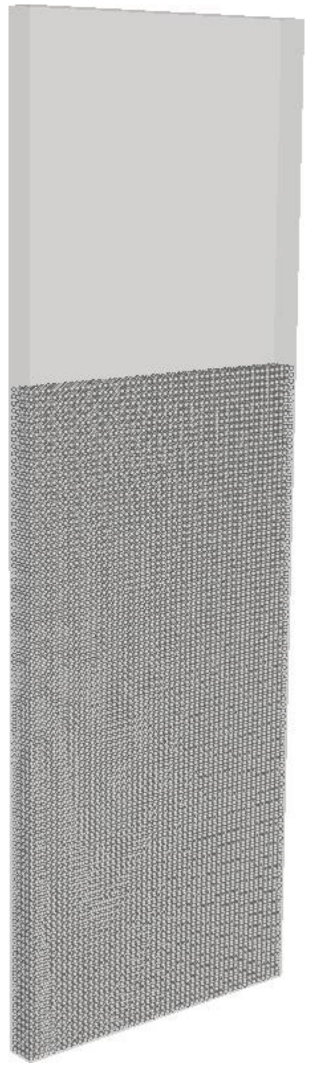
Fluid Flow Through a Packed Bed of Particles

OpenFOAM includes a transient solver for the coupled transport of a single kinematic particle cloud including the effect of the particulate volume fraction on the continuous phase, suitable for **dense particle flow** simulation.

The solver name is **DPMFoam**.

A **bed of particles** (24750) is initially setup in a rectangular geometry.

For the **gas phase** a prescribed **influx condition** is applied at the bottom, **no-slip boundary conditions** are applied at the **side walls** and a **prescribed pressure** condition is applied at the **top of the bed**.



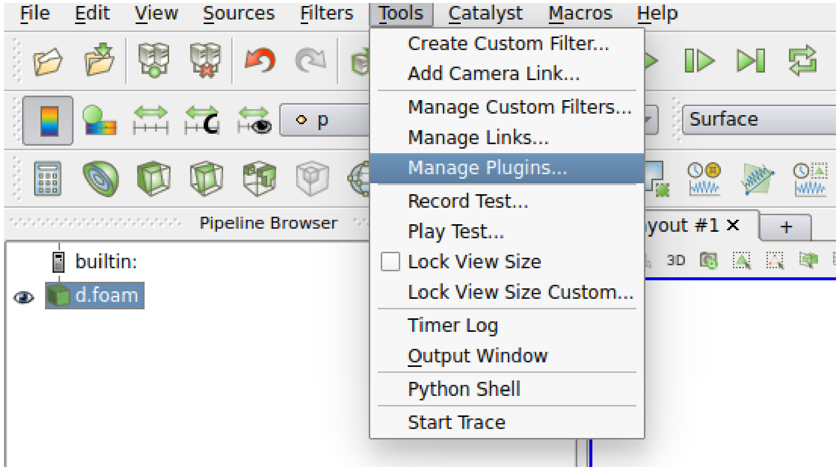
Inside the **$FOAM_RUN** directory you will find the **Goldschmidt** case with the results already computed due to the large time necessary to run this tutorial.

On the terminal enter inside the Goldschmidt case
>> cd $FOAM_RUN/Goldschmidt

And open paraview to visualize the results
>> touch goldschmidt.foam
>> paraview goldschmidt.foam

First, to obtain better visualization of the particulate field we can load the **Point Sprite** plugin.

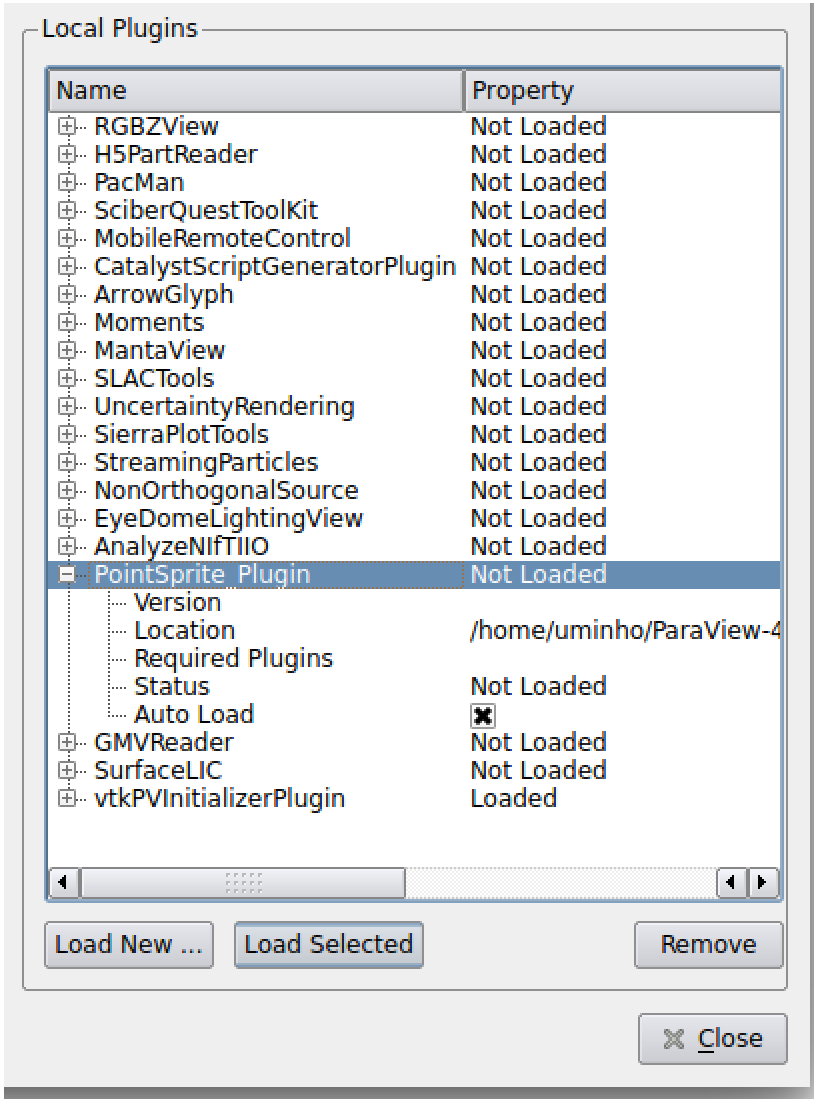
In the **Tools** menu choose the **Manage Plugins…** submenu and there we can find the **PointSprite Plugin**.



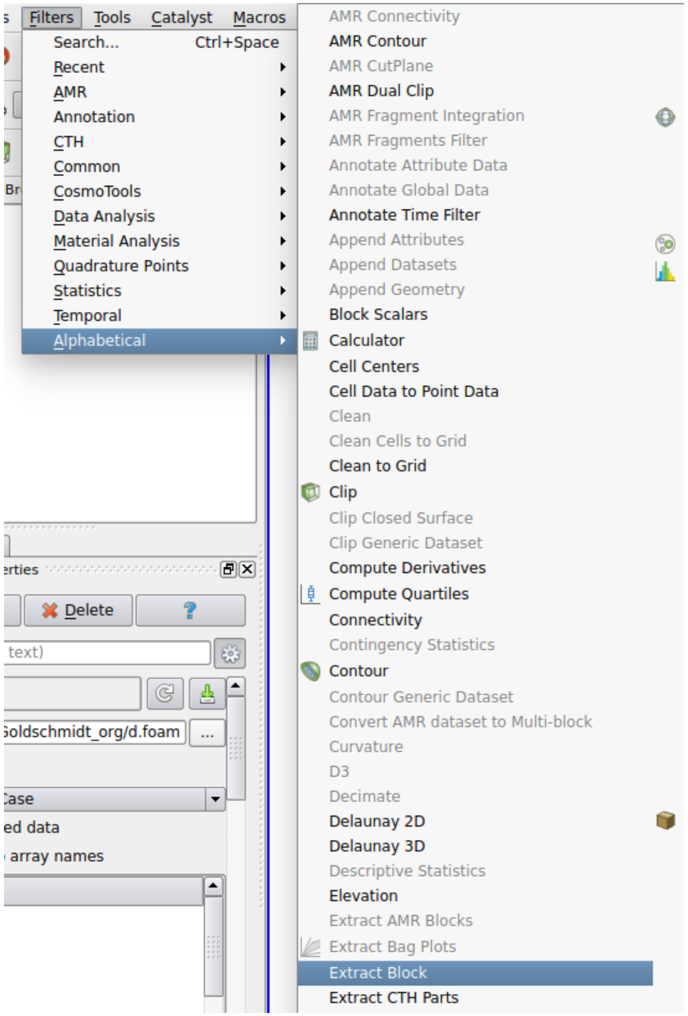
Click on the **plus symbol** and activate the **Auto Load** option.

Finally, click on **PointSprite_Plugin** and then on **Load Selected**.

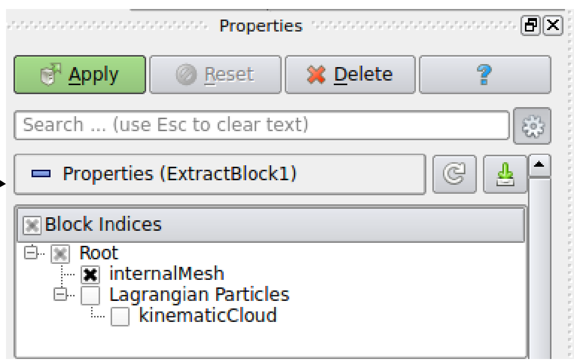
Close paraview and open again.



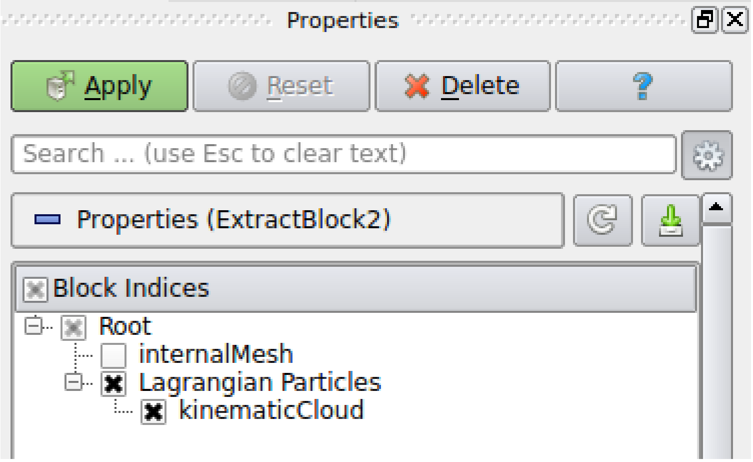
Next, in the **Filter** menu choose the **Alphabetical** submenu and click on the **Extract Block** option.



Choose the **internalMesh** field in the **left side Properties** panel and press **Apply**

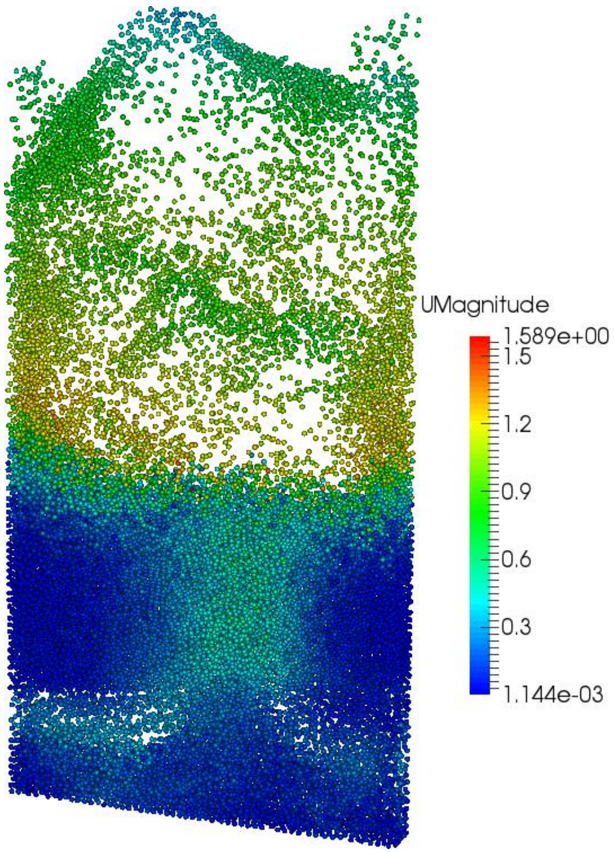


Repeat the same procedure but now choose the **Lagrangian Particles** field.



Now we can see separately the behavior of the continuous (gas) and discrete (particles) fields.

With the **PointSprite** option in the **representation bar** and the **Max Pixel size** of the **PointSprite menu** try to reproduce the image on the right.



Tutorial

Propeller

Analysis of Flow around a Ship Propeller

OpenFOAM includes the **Arbitrary Mesh Interface** technique (AMI) for non-conformal patches.

AMI is a technique that allows simulation across **disconnected, but adjacent, mesh domains**. The domains can be stationary or move relative to one another.

The sliding interface capability has been tested on engineering geometries, including a **propeller**.

Flow was simulated using the **pimpleDyMFoam** solver.

Transient solver for incompressible flow of **Newtonian fluids** on a **moving mesh** using the **PIMPLE** (merged PISO-SIMPLE) algorithm.

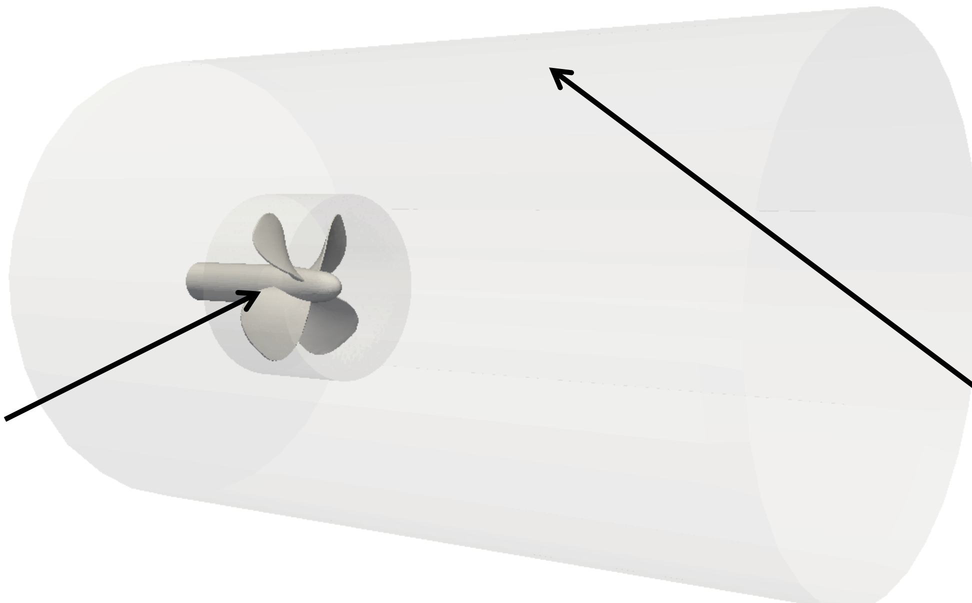
The **propeller geometry** is represented in the figure.

Inlet fixedValue
Velocity – 5 m/s

Inlet zeroGradient
Pressure

Outlet inletOutlet
Velocity

Outlet fixedValue
Pressure – 0 Pa



propeller
movingWallVelocity

outerCylinder
fixedValue
Velocity – 0 m/s

outerCylinder
zeroGradient
Pressure

Inside the **$FOAM_RUN** directory you will find the **Propeller** case with the results already computed due to the large time necessary to run this tutorial.

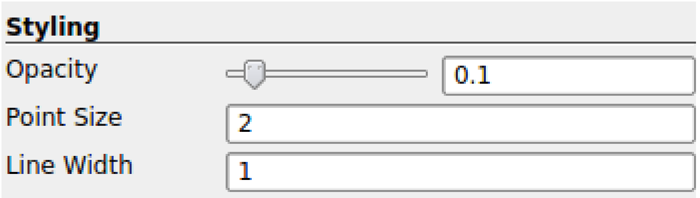
On the terminal enter inside the Propeller case
>> cd $FOAM_RUN/Propeller

And open paraview to visualize the results
>> touch propeller.foam
>> paraview propeller.foam

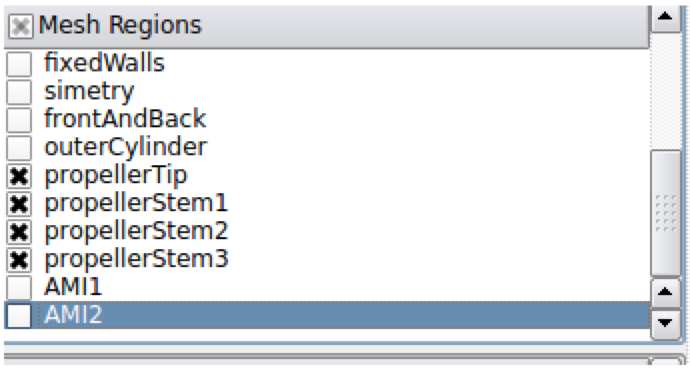
In Paraview try to **animate** the **propeller velocity** field and **export the video with streamlines**.

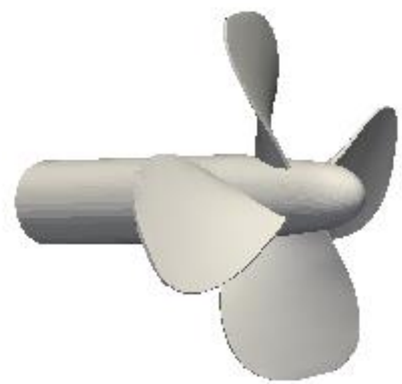
First, open **only the internalMesh** and press **apply**.

Change the **Opacity value** to 0.1.

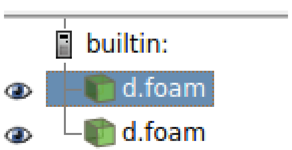


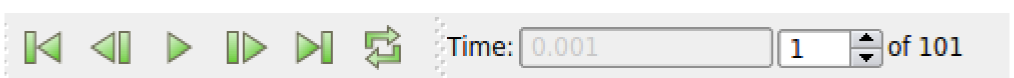
Then, **open again the same case** and choose only the **propeller patches**.



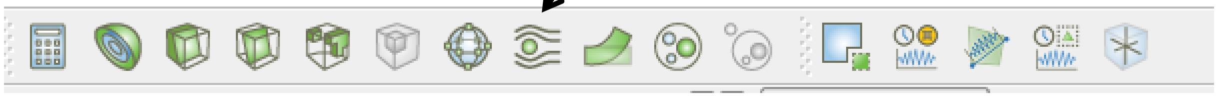


Now, have sure that the **two cases are selected** and **advance one time step**.

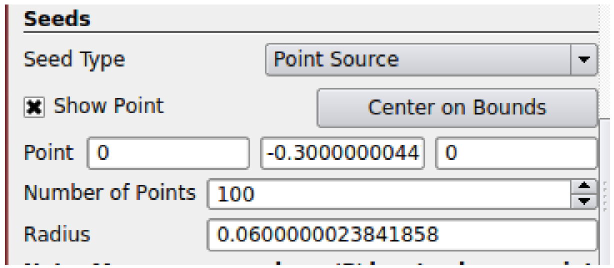




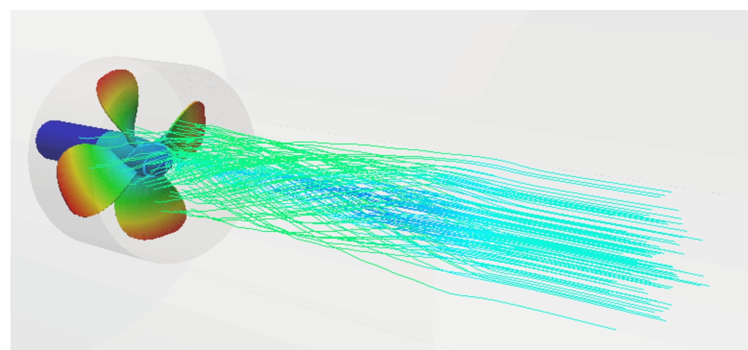
Choose the **streamTracer** option.



Press the **Center on Bounds** button**.** Finally, press **apply**.

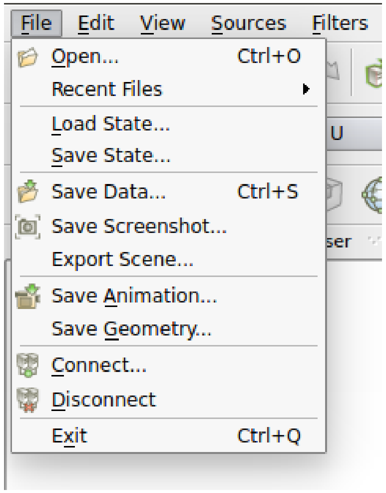


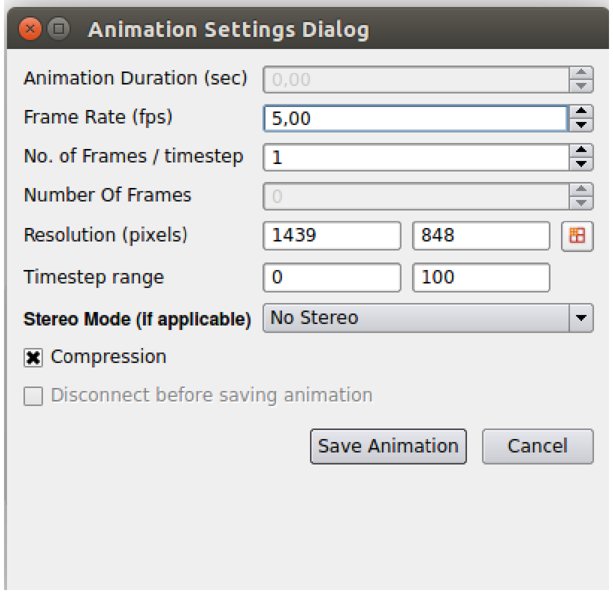
We should obtain an image similar to the one below, **coloring by the velocity field**.



To finalize we will **export a video** with the movement of the propeller.

For that in the **File** menu click on **Save Animation** and setup **Frame Rate** to 5 and finally click on **Save Animation**.





<u>Tutorial</u>

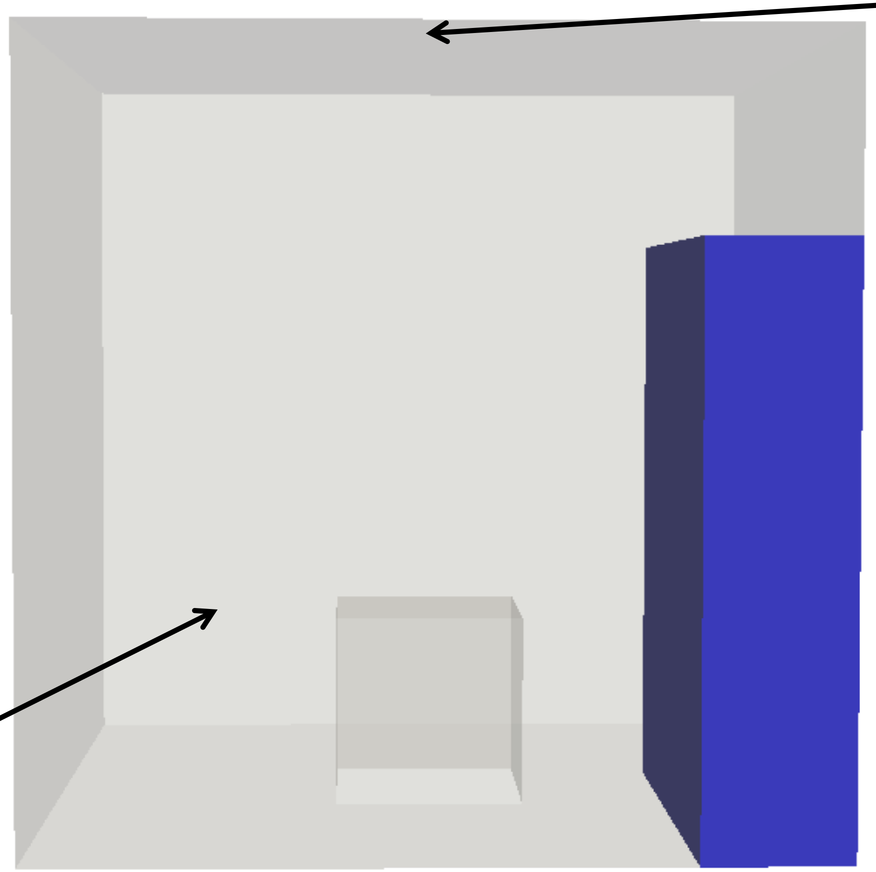
DamBreak3D

Analysis of River Flow around a Obstacle

Flow was simulated using the **interDyMFoam** solver.

Solver for **2 incompressible, isothermal immiscible fluids** using a **VOF (volume of fluid) phase-fraction** based interface capturing approach, with optional mesh motion and mesh topology changes including adaptive re-meshing.

The **damBreak geometry** is represented in the figure.



Top boundary is free to the atmosphere so needs to permit both outflow and inflow according to the internal flow.
Total Pressure
PressureInletOutletVelocity

Walls velocity - 0 m/s
Pressure – fixed flux pressure

Inside the **$FOAM_RUN** directory you will find the **DamBreak 3D** case with the results already computed due to the large time necessary to run this tutorial.

On the terminal enter inside the DamBreak3D case
>> cd $FOAM_RUN/DamBreak3D

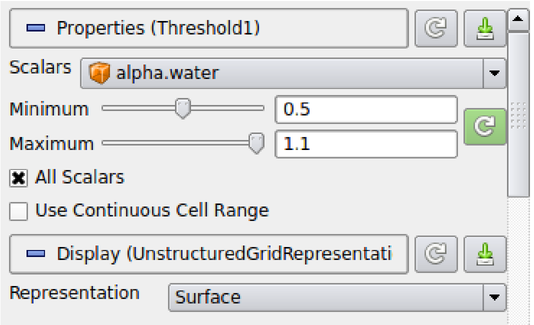
And open paraview to visualize the results
>> touch dambreak3D.foam
>> paraview dambreak3D.foam

In Paraview try to **animate** the water fall and **export the respective video**.

First, select the internal mesh and atmosphere patch and choose **Opacity to 0.3**.

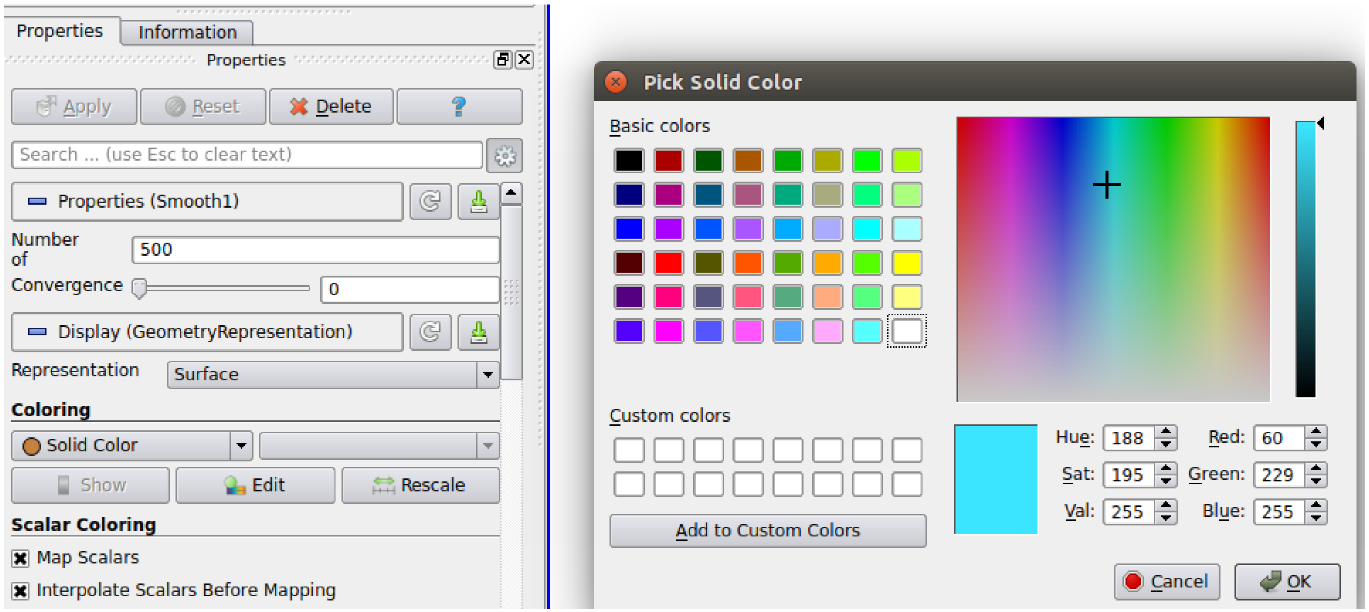
Choose the **Threshold** option.

And apply the following values.

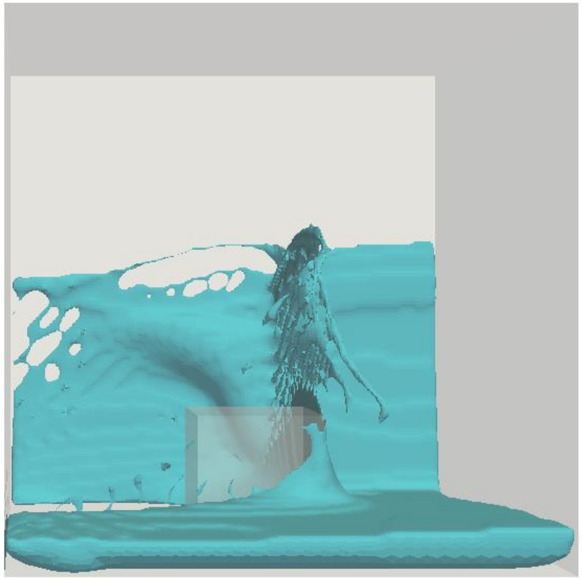


Next, apply the **Extract Surface** filter and the **Smooth** filter with **Number of Convergence** of 500.

Finally, color the surface with **Solid Color** and edit the color choosing a light blue color.



We should obtain



To finalize we will **export a video** with the movement of the water.

Tutorial

motorBike

Analysis of Flow around a Motor Bike

For this tutorial we will look at the simulation of the flow around a **motorbike** model.

Flow was simulated using the **pisoFoam** solver.

Transient solver for **incompressible** flow. **Turbulence** modeling is **generic**, i.e. laminar, RAS or LES.

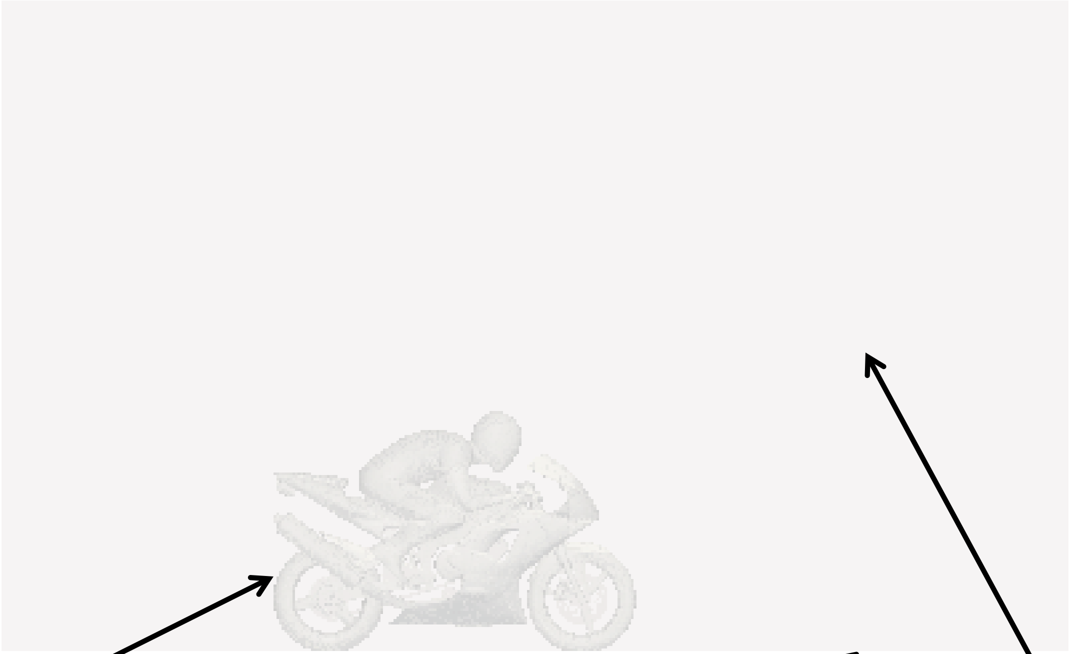
The **motorBike geometry** is represented in the figure.

Inlet fixedValue Velocity – 20 m/s

Inlet zeroGradient Pressure

Outlet inletOutlet Velocity

Outlet fixedValue Pressure – 0 Pa



motorBike fixedValue Velocity – 0 m/s

motorBike zeroGradient Pressure

Front, Back, Upper and Lower walls are of type symmetryPlane

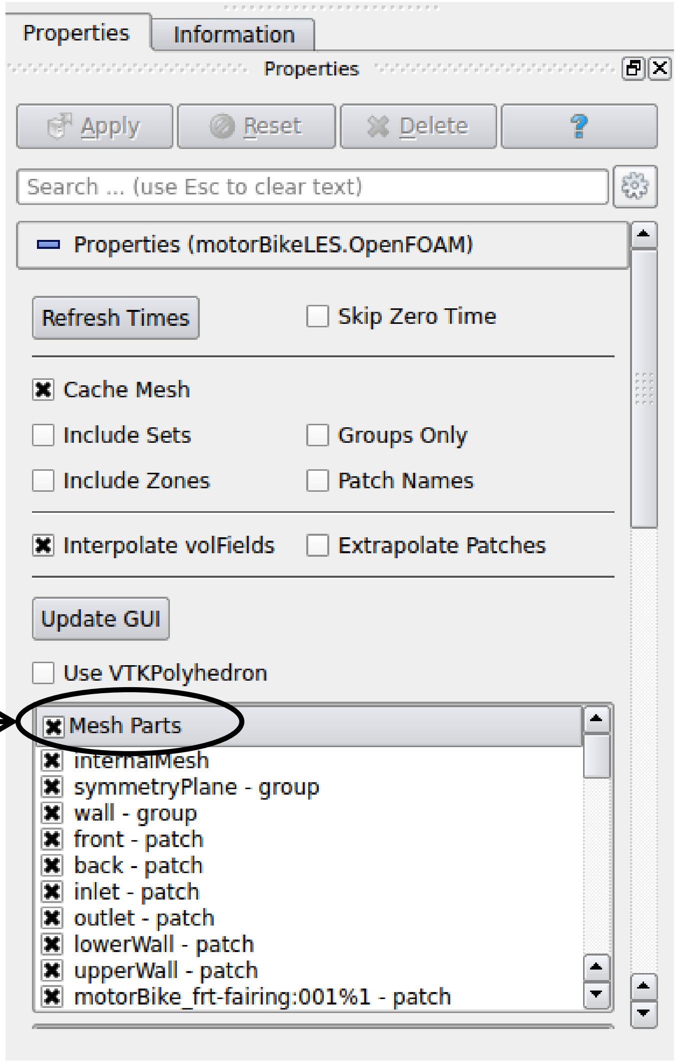
Inside the **$FOAM_RUN** directory you will find the **MotorBike** case with the results already computed due to the large time necessary to run this tutorial.

On the terminal enter inside the MotorBike case
>> cd motorBike

And open paraview to visualize the results
>> touch motorbike.foam
>> paraview motorbike.foam

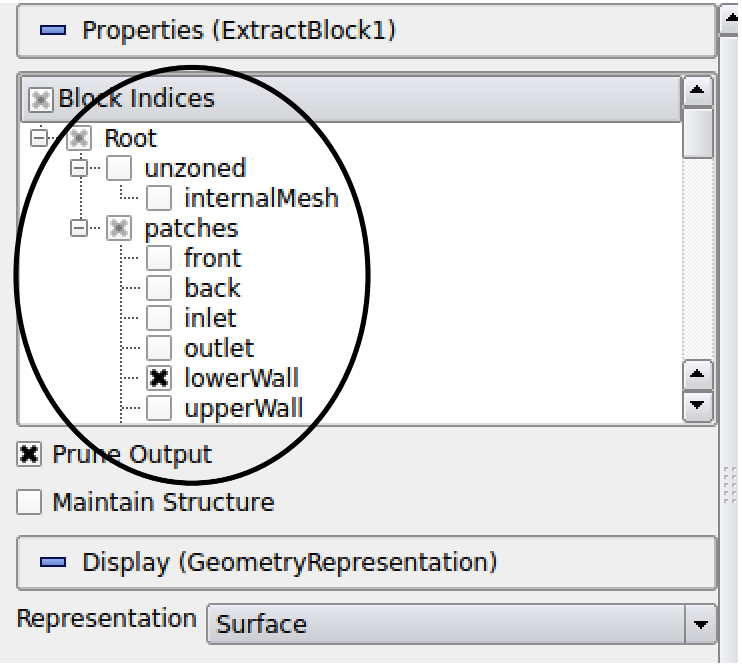
In Paraview try to plot the **streamlines** that were exported from the computations and are saved on the **postProcessing** folder.

First in the Paraview **Properties painel** select all **Mesh parts**.



Press **Apply** to obtain all the mesh and patches.

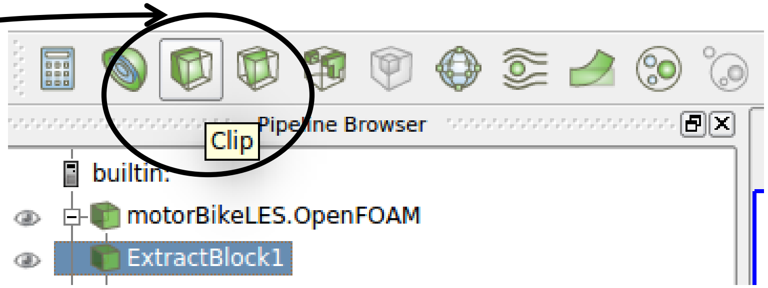
Next, as in the Goldschmidt tutorial select the **Extract block** filter.



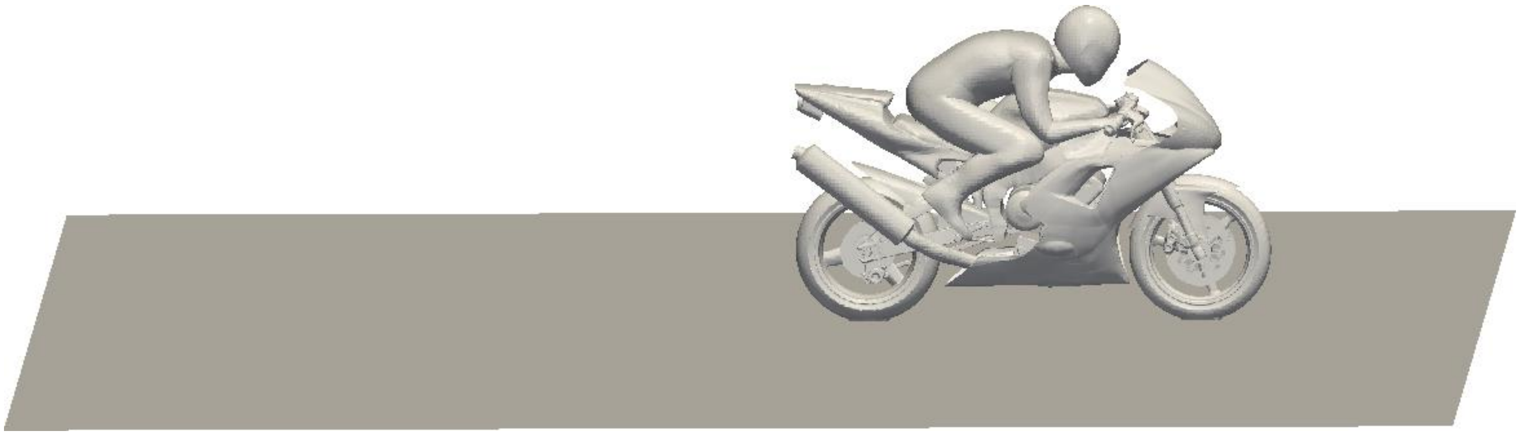
Unselect the internalMesh and the front, back, inlet, outlet and upperWall patches.

Press **Apply.**

Then, apply several **Clips** to reduce the geometry.

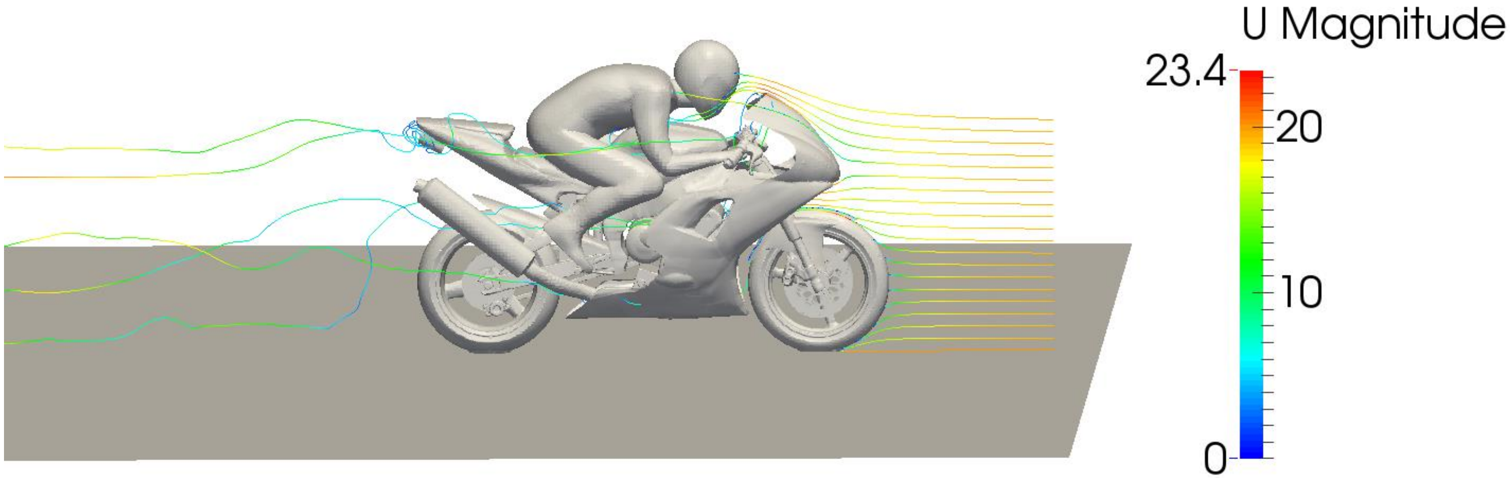


We should obtain one image similar to the one below.



Finally, open the streamline file for the last simulation time: postProcessing/sets/streamLines/0.7

And select **velocity field** to be seen. The result should be



Thank you for your attention!