Next Steps: Procedural Animation

M04 - Wire Solvers

Ari Danesh
ari@sidefx.com

SIDE EFFECTS SOFTWARE

Agenda

- **Intro to Wire Solvers**
  - Creating a Basic Wire Solver
- **Constraints**
- **Projects**
  - Wire Glue Constraint
  - Dangling Wires

SIDE EFFECTS SOFTWARE

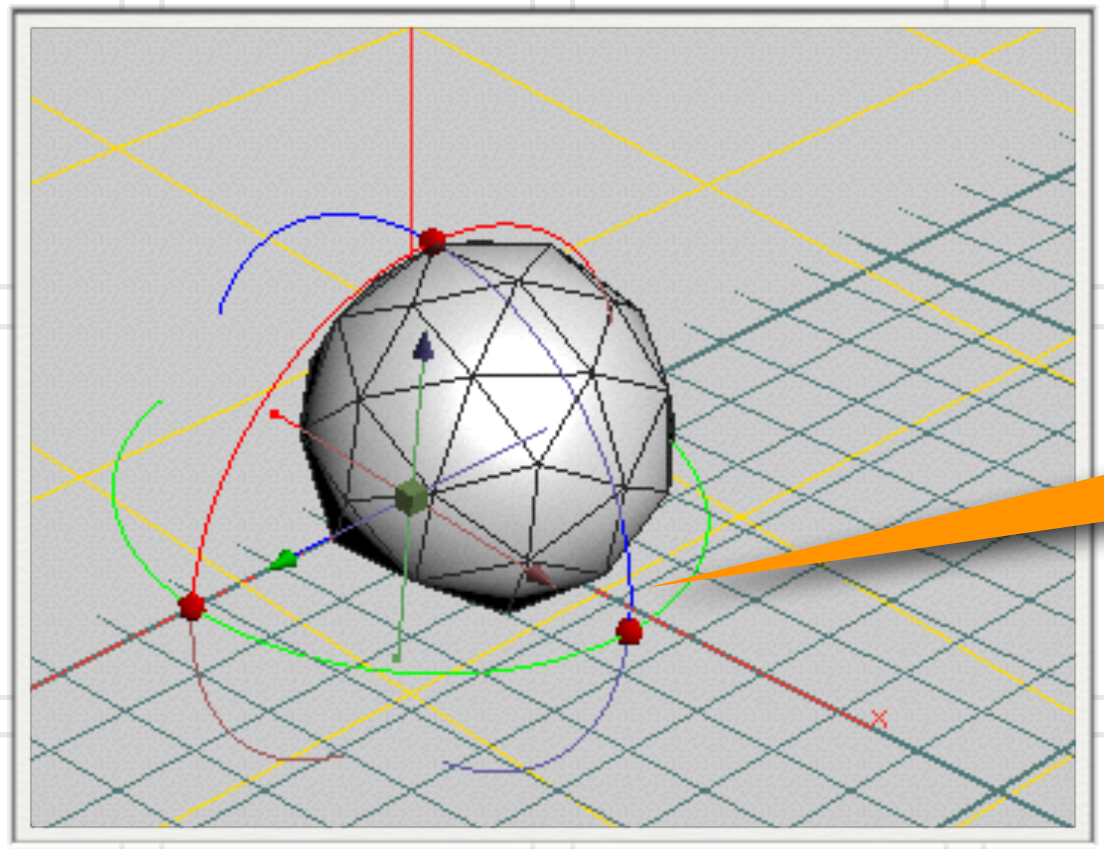
Wire Solver Intro

- **Wire Solvers are light weight**
  - They are just a point-edge solver. They use existing point-edges of the geometry unlike cloth solvers that have their own internal make up
- **Wire Solvers are good for doing soft body collisions and “hair like” simulations**
- **It only supports one object per sim. It can not be used for fracturing or tearing like the cloth solver**

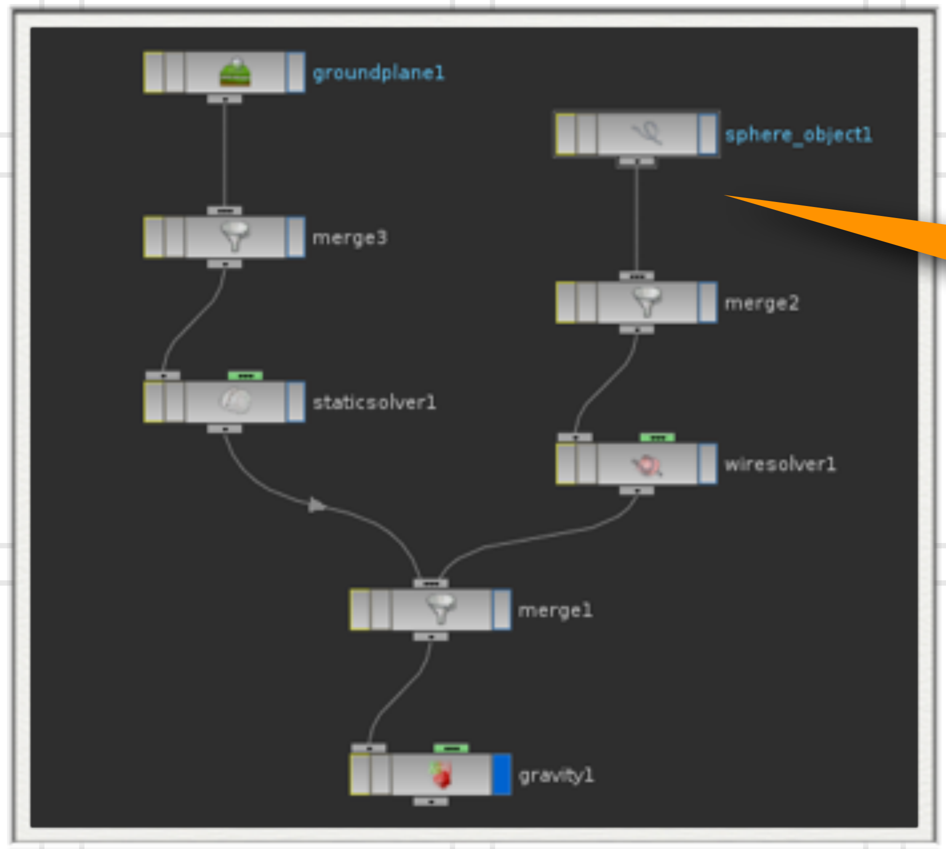
Test Bed

Wire Object

Wire Tab



Notice how the ball slightly collapses like a cloth solver but much lighter weight

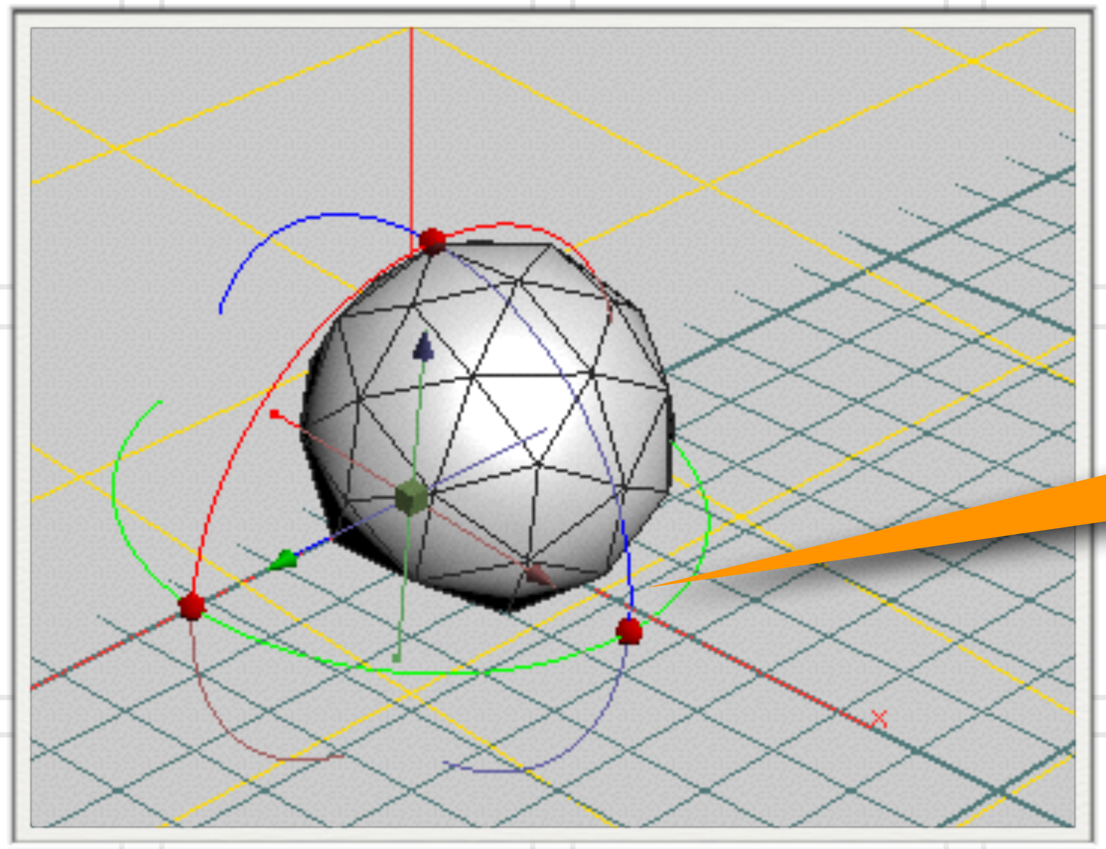


The AutoDOPNetwork looks almost the same as last week. The RBD object is now replaced with a Wire Object

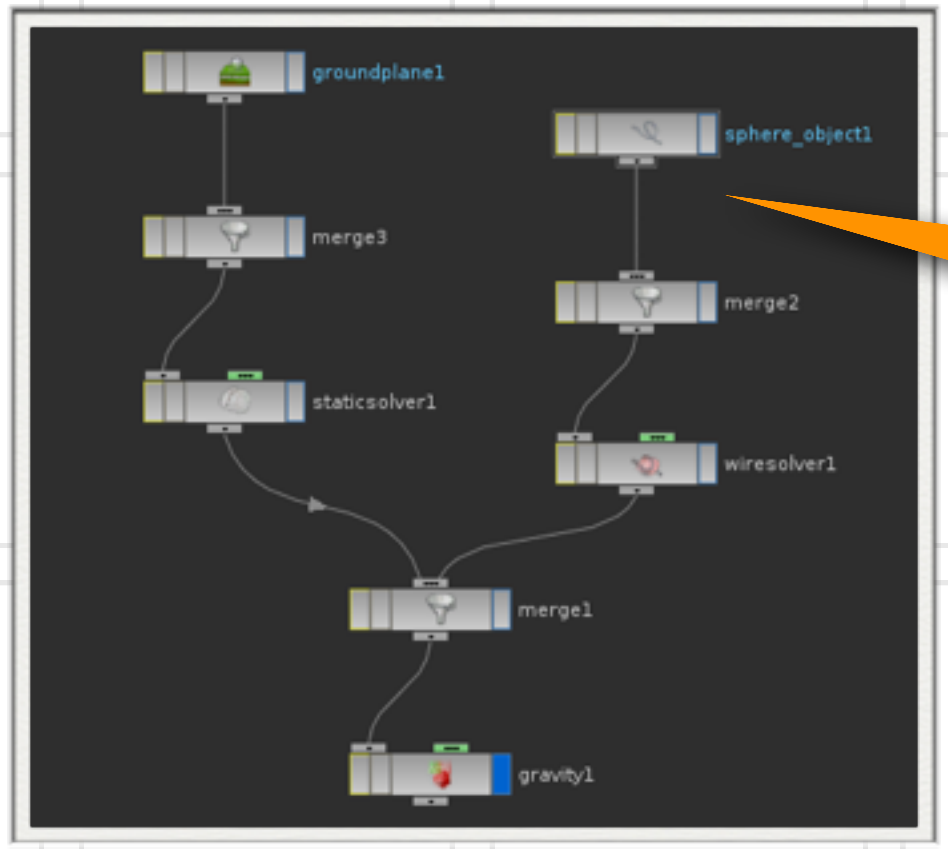
- **Very similar to last weeks**
  - Drop down a sphere object
- Make it polygonal and bring it up 5 units on the y-axis
- While the Sphere is selected make it a **Wire Object**
  - Drop down a Ground Plane
  - Run simulation - Remember to turn on the real time flag

SIDE EFFECTS SOFTWARE

Differences between Wire and RBD Setup



Notice how the ball slightly collapses like a cloth solver but much lighter weight

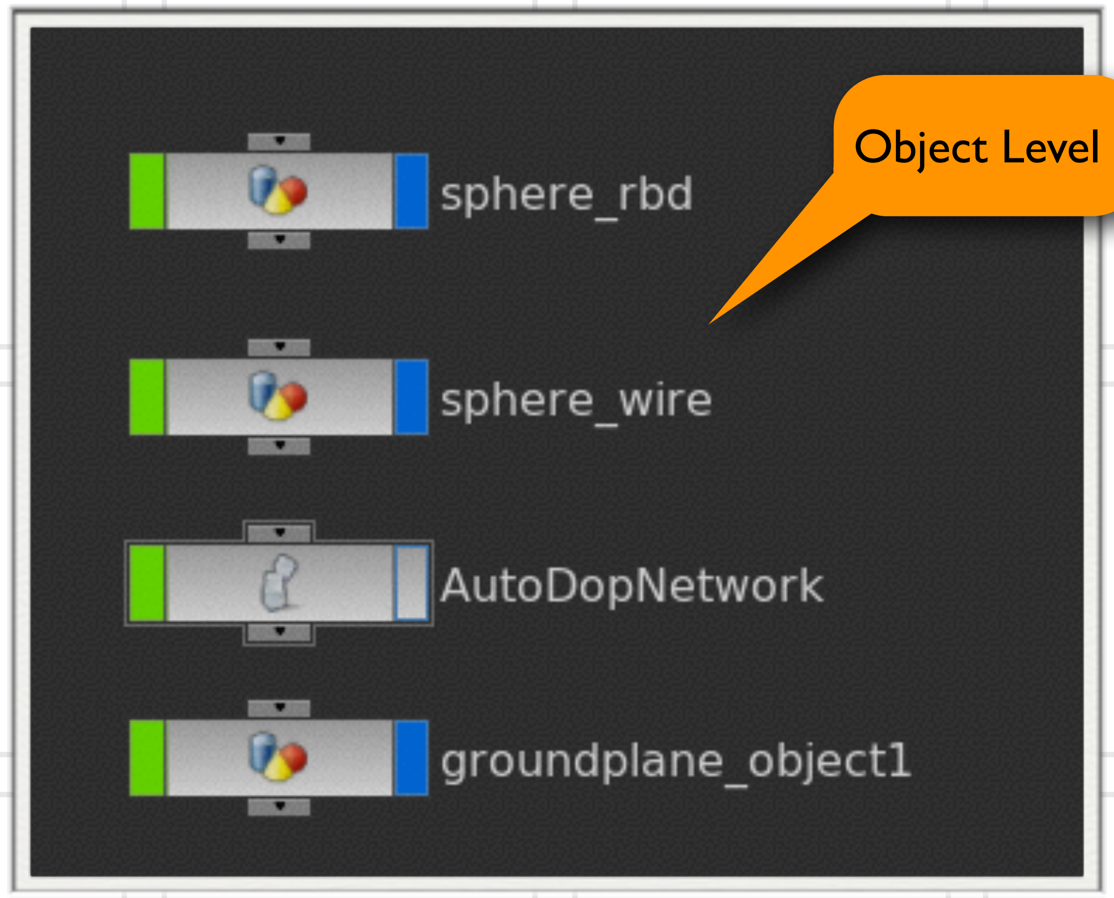


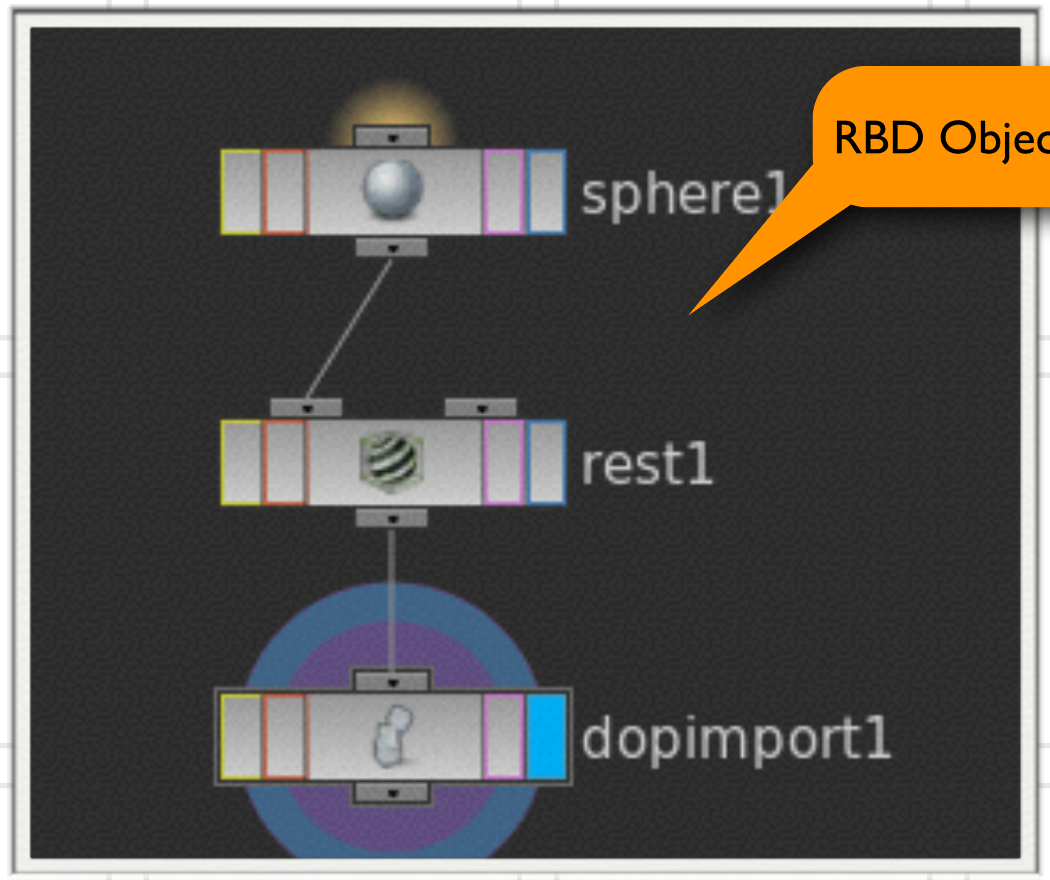
The AutoDOPNetwork looks almost the same as last week. The RBD object is now replaced with a Wire Object

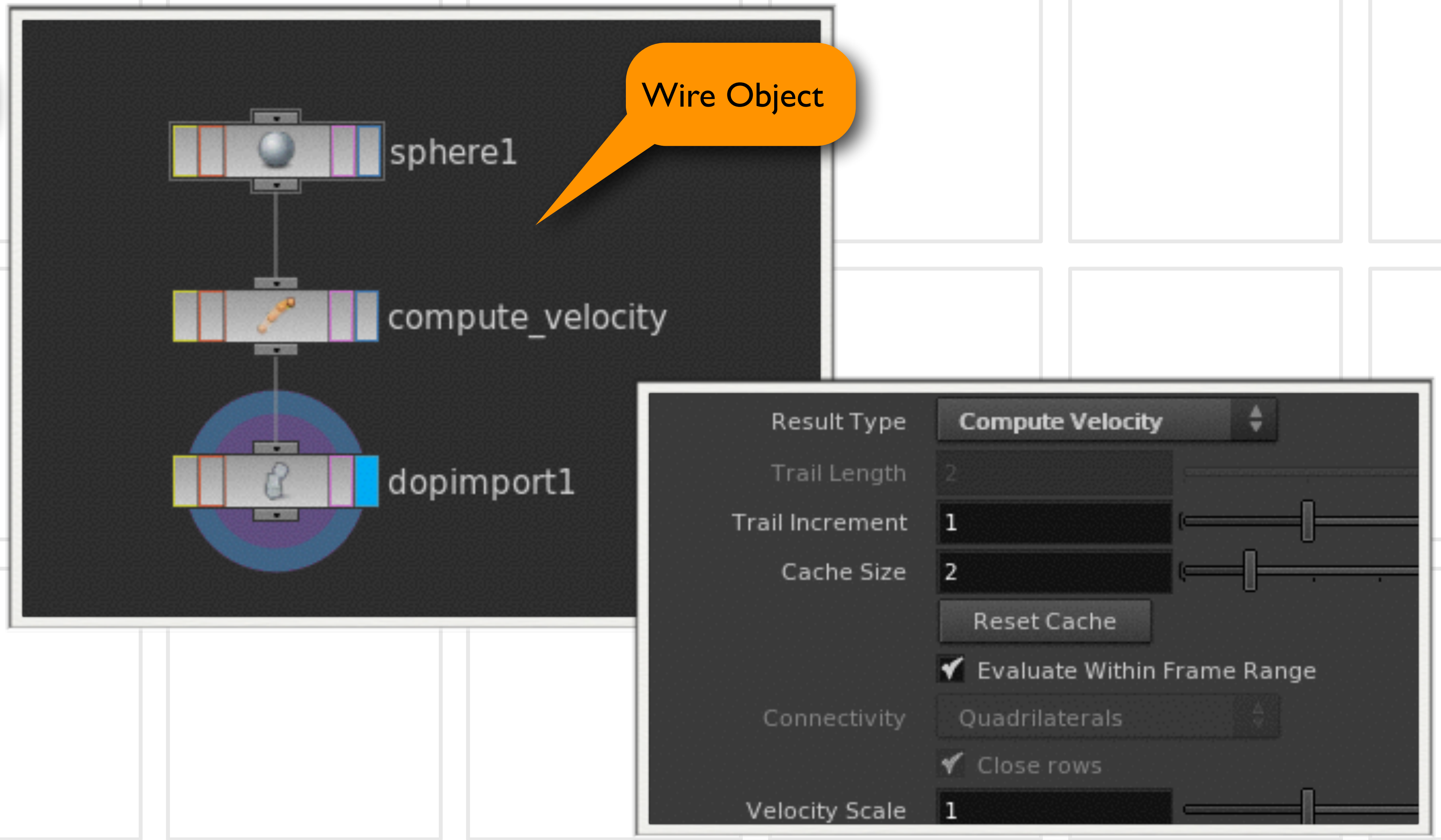
- **Let us create a RBD Object next to the Wire Object**
  - Drop down a sphere object
- Make it polygonal and bring it up 5 units on the y-axis
- Move it over on the x-axis away from the wire object
- While the Sphere is selected make it a **RBD Object**
- **Run the simulation**
  - **They drop at the same time**
  - **The RBD is rigid and the Wire is squishy**

SIDE EFFECTS SOFTWARE

RBD Object vs Wire Object

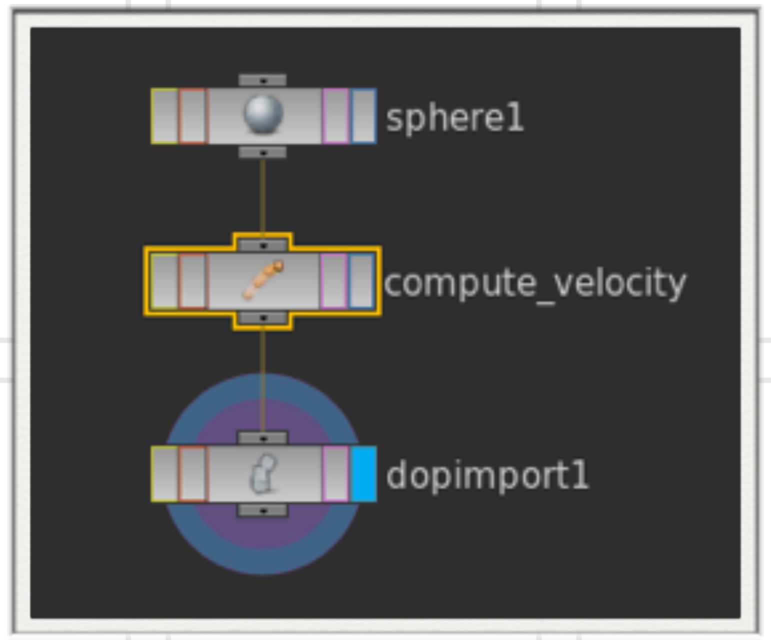


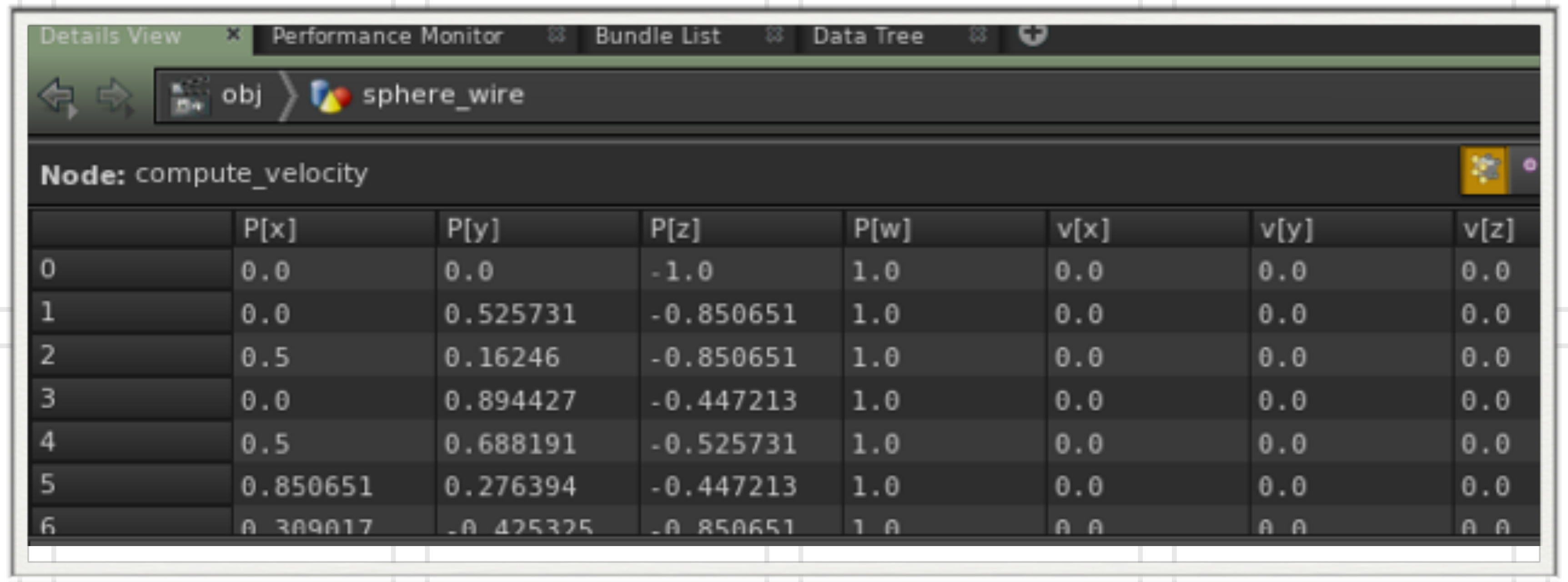




SIDE EFFECTS SOFTWARE

What is this Compute Velocity?

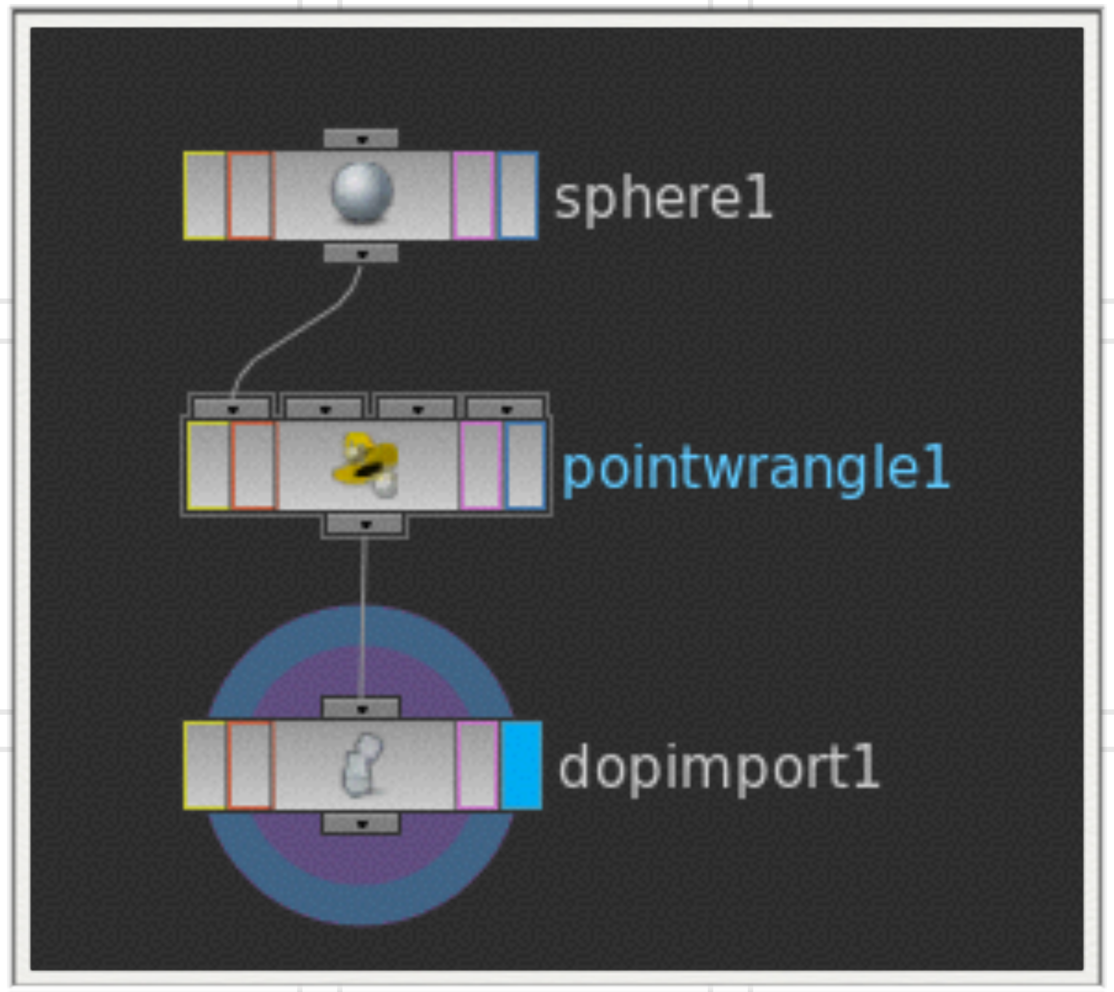




	P[x]	P[y]	P[z]	P[w]	v[x]	v[y]	v[z]
0	0.0	0.0	-1.0	1.0	0.0	0.0	0.0
1	0.0	0.525731	-0.850651	1.0	0.0	0.0	0.0
2	0.5	0.16246	-0.850651	1.0	0.0	0.0	0.0
3	0.0	0.894427	-0.447213	1.0	0.0	0.0	0.0
4	0.5	0.688191	-0.525731	1.0	0.0	0.0	0.0
5	0.850651	0.276394	-0.447213	1.0	0.0	0.0	0.0
6	0.309017	-0.425325	-0.850651	1.0	0.0	0.0	0.0

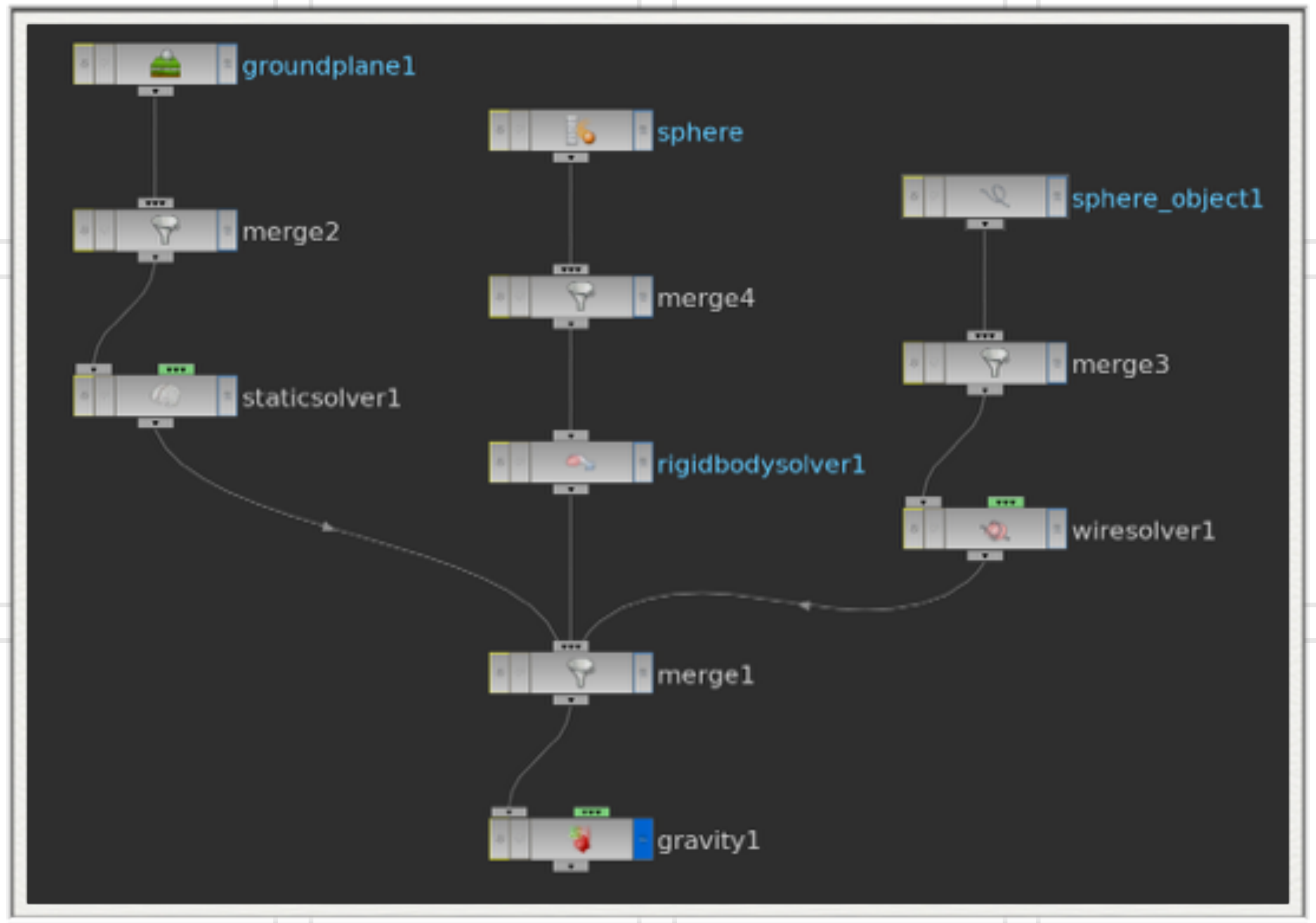
For the Wire Solver to Work you have to add a Velocity attribute. But as we know from the introduction to dynamics the sphere does not actually move therefore the velocity is zero.

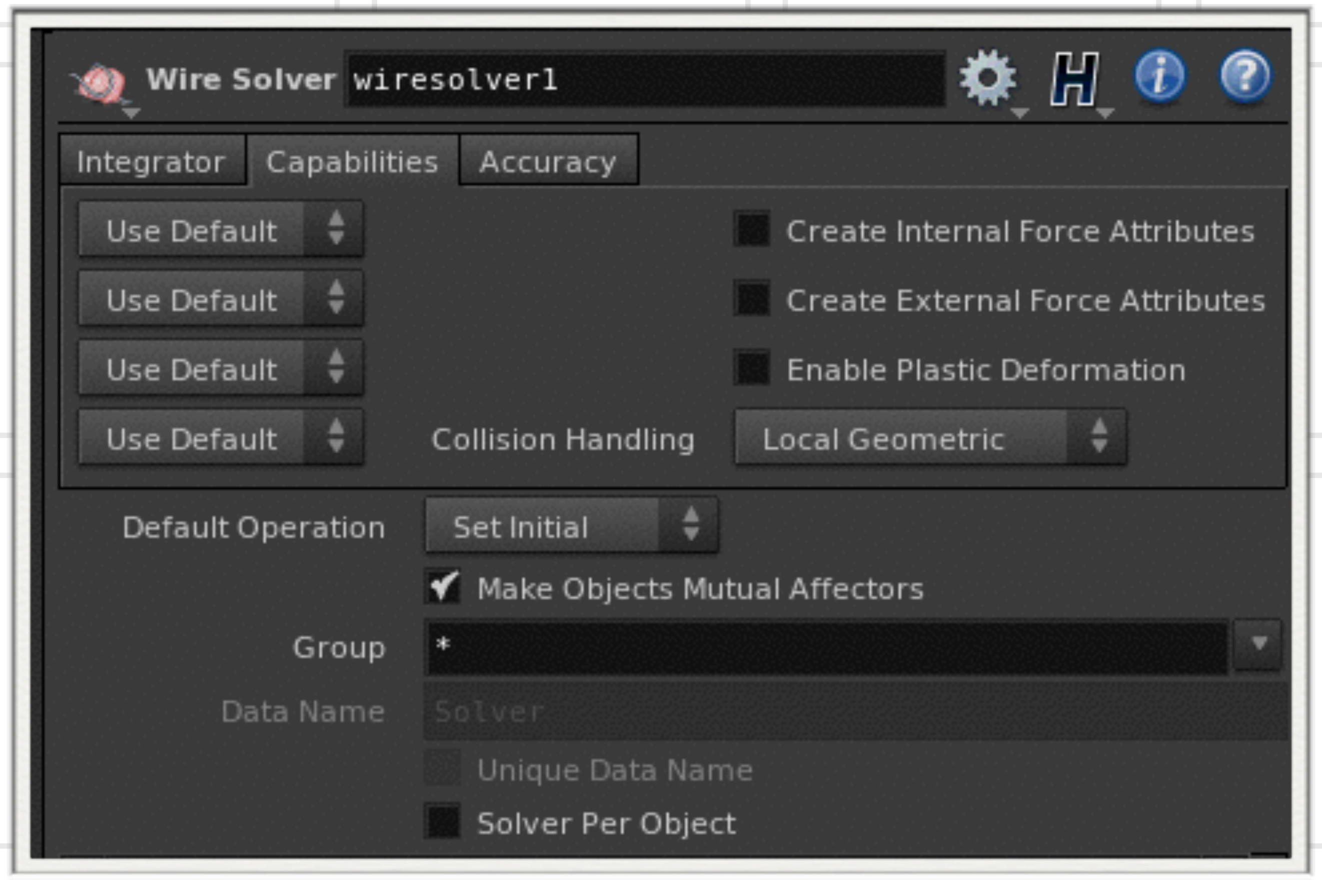
Therefore if you were to wire this on your own you could also just add a point wrangle

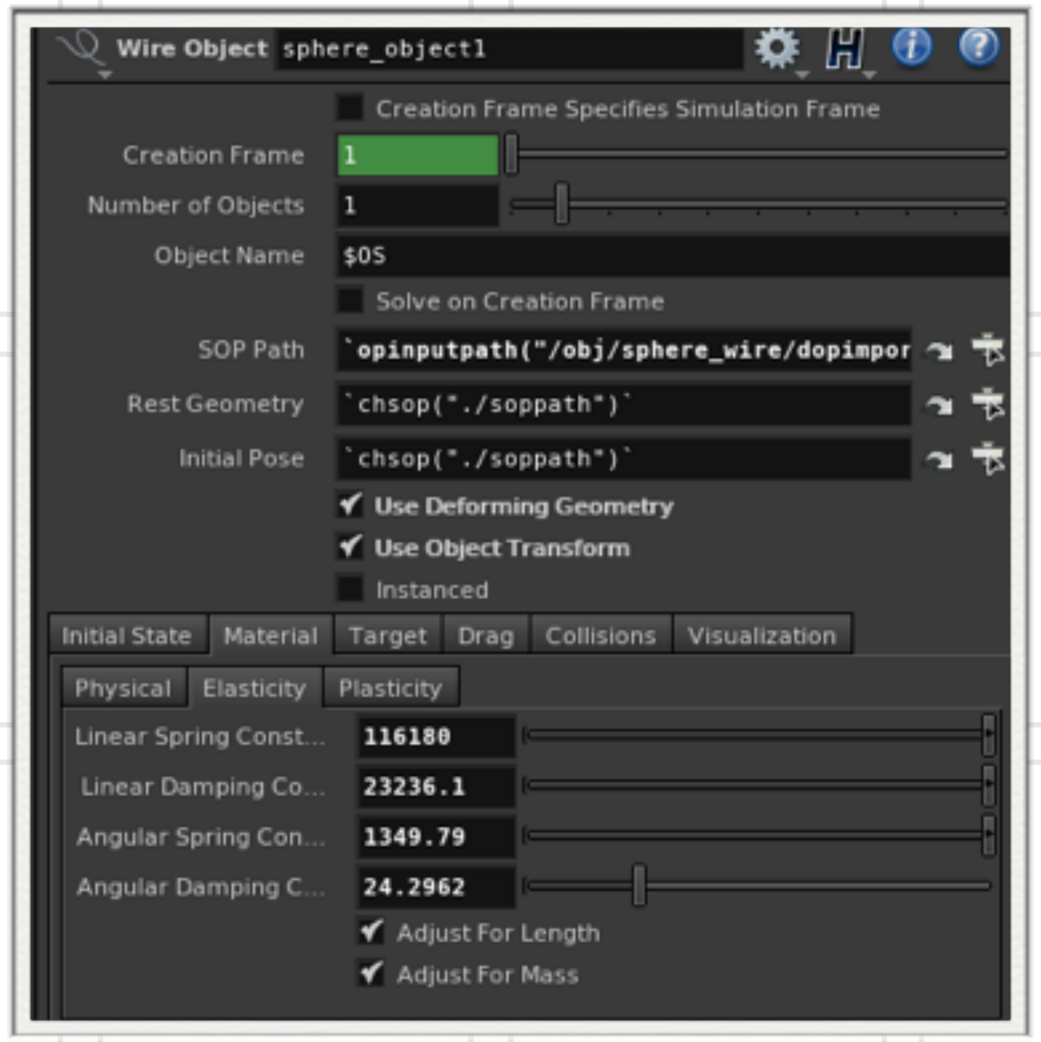


SIDE EFFECTS SOFTWARE

Diving into the AutoDOP Network







The Wire Object looks quite similar to the RBD Object

Look at the “Capabilities” Tab of the Wire Solver

SIDE EFFECTS SOFTWARE

Capabilities - Wire Solver

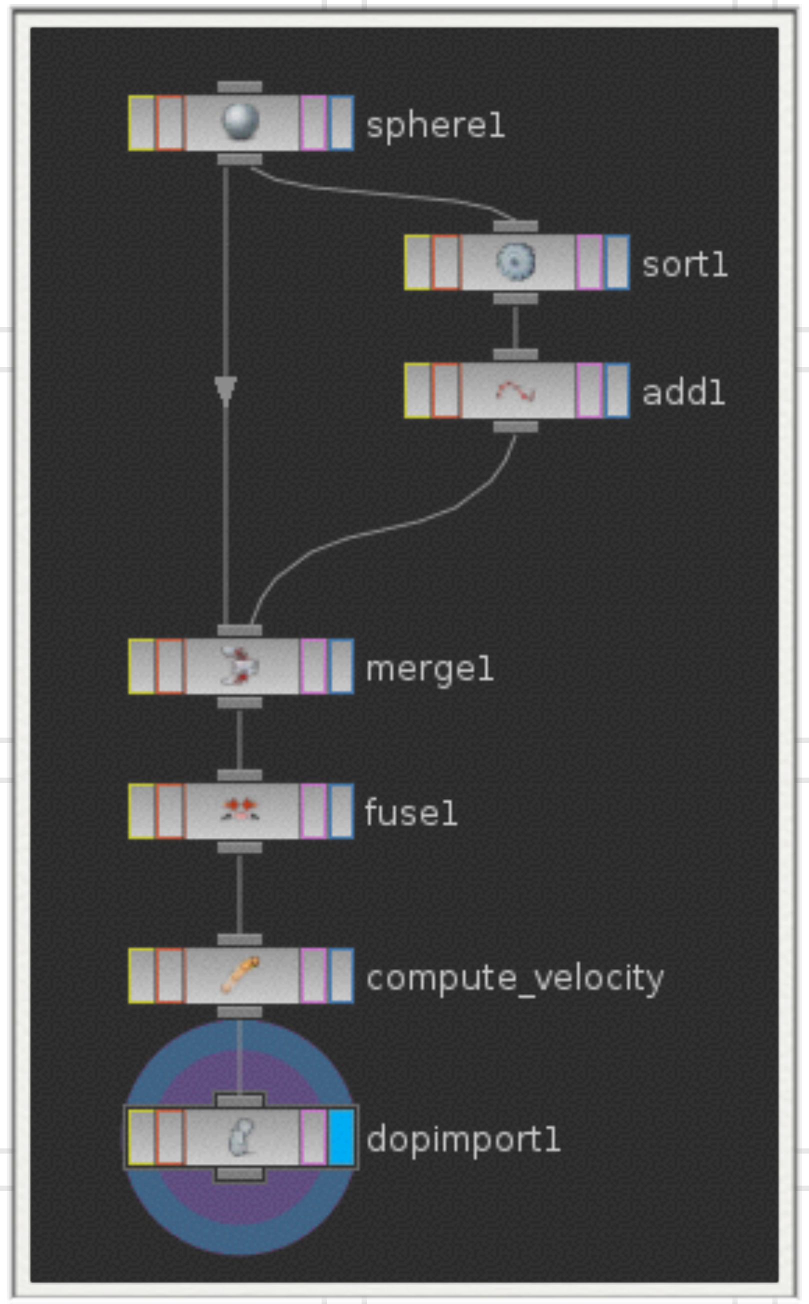
Capabilities

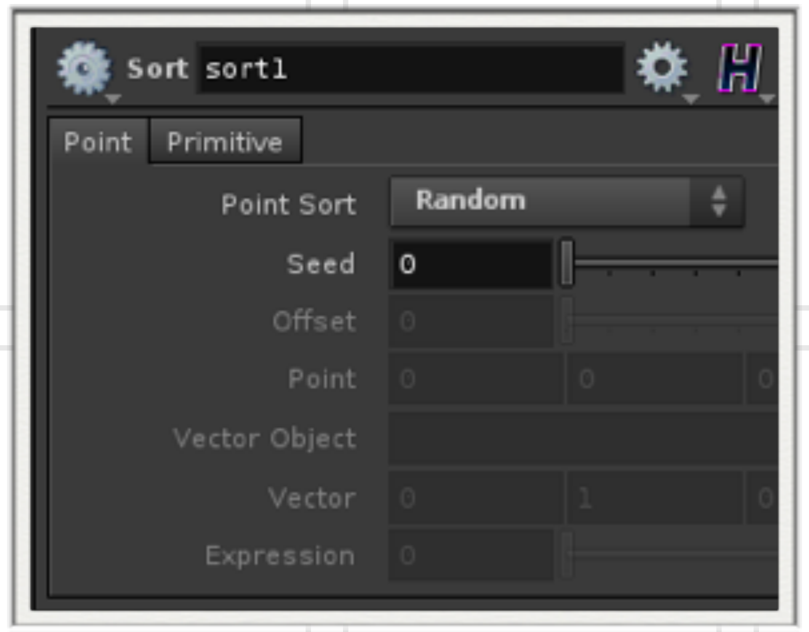
These parameters control attribute creation on the wire object.

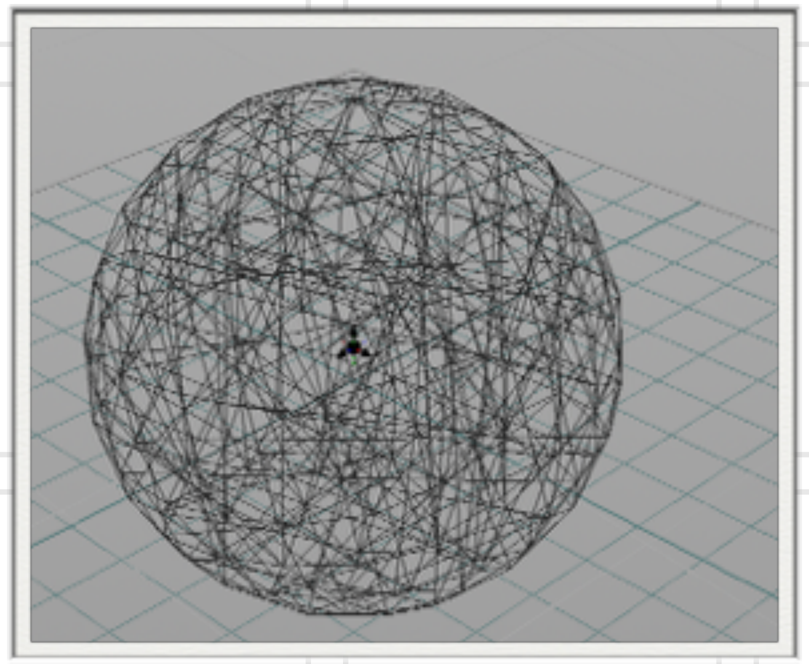
Create Internal Force Attributes	If set, the wire solver will create attributes on each point showing the force applied to resist stretching and bending. This is required for wire visualization to work, but can be disabled to save memory during simulation.
Create External Force Attributes	If set, the wire solver will create attributes on each point showing the force applied by Force DOPs, and applied to meet constraints, prevent collisions and create friction. This is required for wire visualization to work, but can be disabled to save memory during simulation.
Enable Plastic Deformation	If set, the wire solver will create and modify attributes on Wire Object geometry for describing plastic deformation information.
Collision Handling	Determines the collision detection and resolution strategy used by the wire solver.

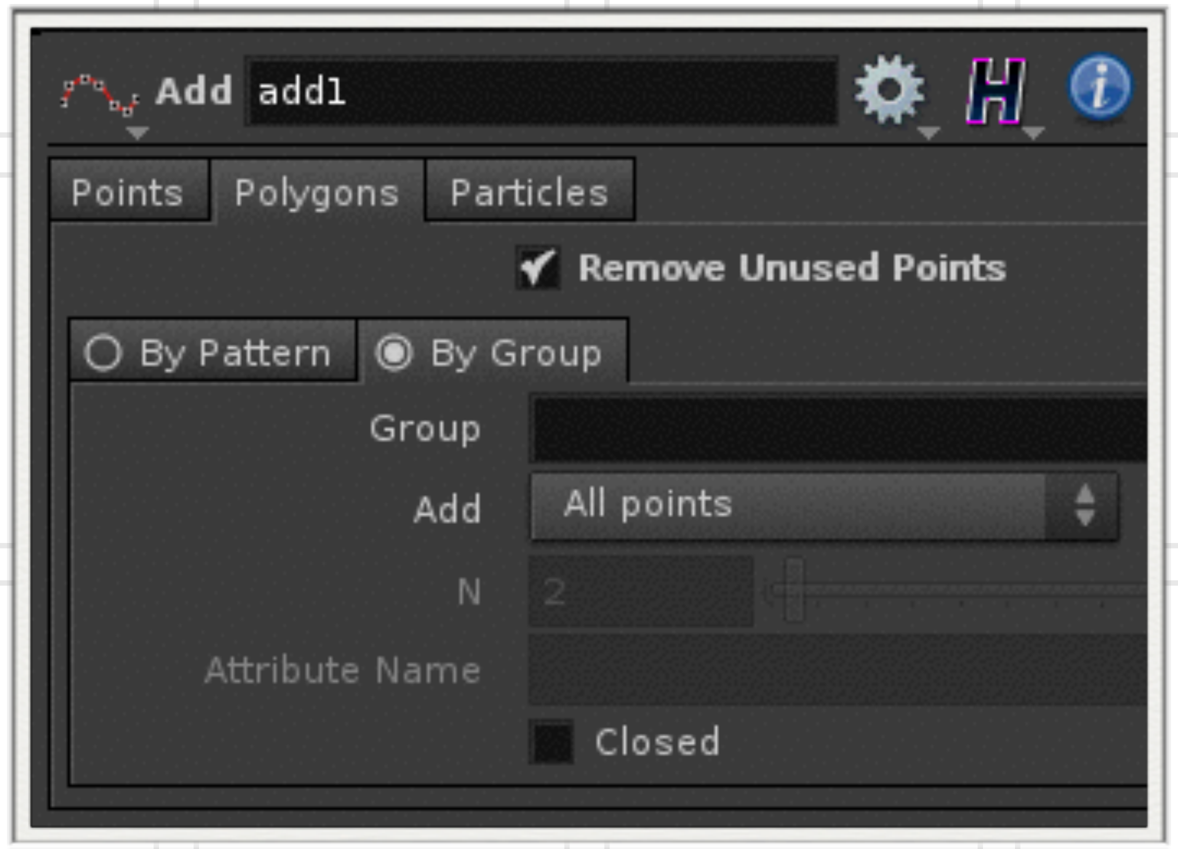
SDF	A strategy that supports SDF representations of rigid bodies.
Local Geometric	A collision response strategy that ignores the influence of elements distant from the collision. This strategy is faster that “Global Geometric” but may produce visual artifacts.
Global Geometric	A collision response strategy that considers the influence of all elements.

Adding Internal Structure to the Wire Simulation





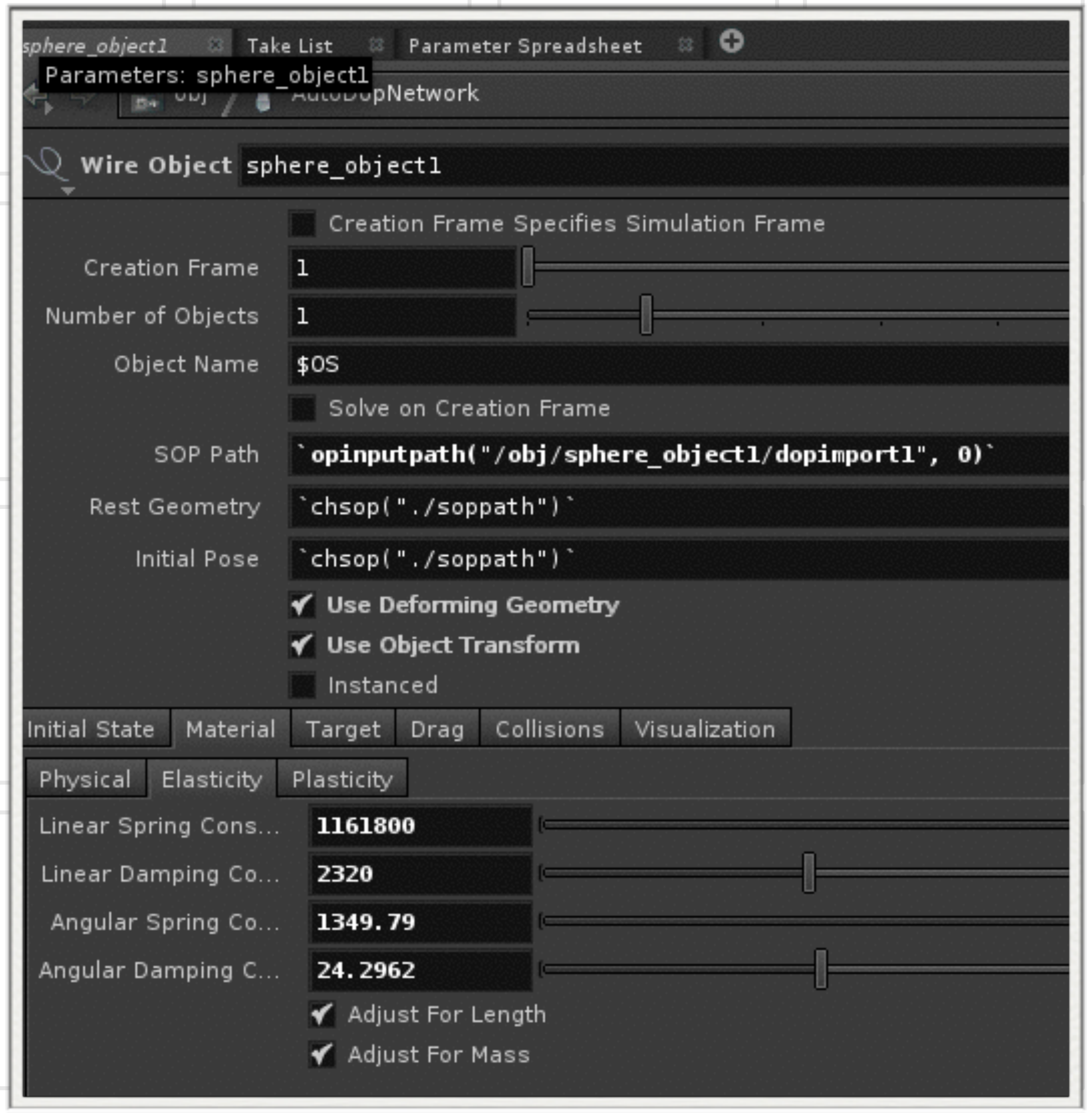




- Wire simulations can be considered as a simulation that looks at each edge of the geometry as a spring. If the
- Split off a Sort SOP from the Sphere and set Point Sort to Random
- Append an Add SOP
  - Polygons Tab
  - By Group - All Points
- Merge and then set the viewport and see the results
- Run the simulation to see the bounce and roll are slightly different due to the internal springs

SIDE EFFECTS SOFTWARE

Increasing Stiffness



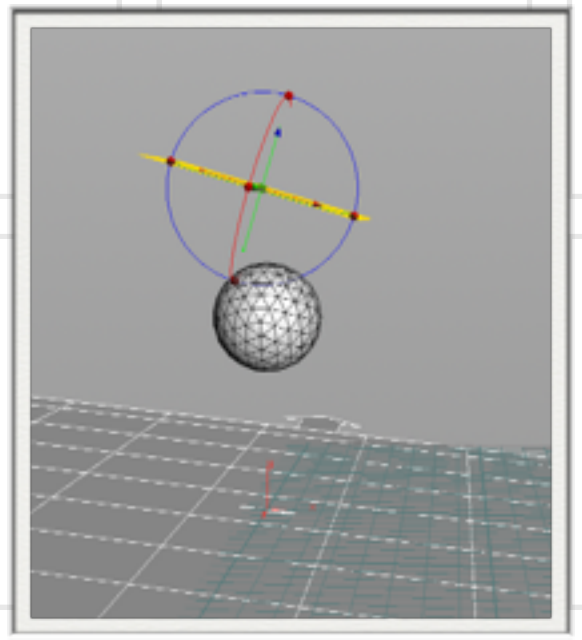
- **Dive inside the AutoDOPNetwork and Select the Wire Object**
- **Select the Material Tab**
- **In the Material Tab Select the Elasticity Tab**
  - Increase the Linear Spring Coefficient by a factor of 10
  - Decrease the Linear Damping Coefficient by a factor of 10
- **Test the Simulation**

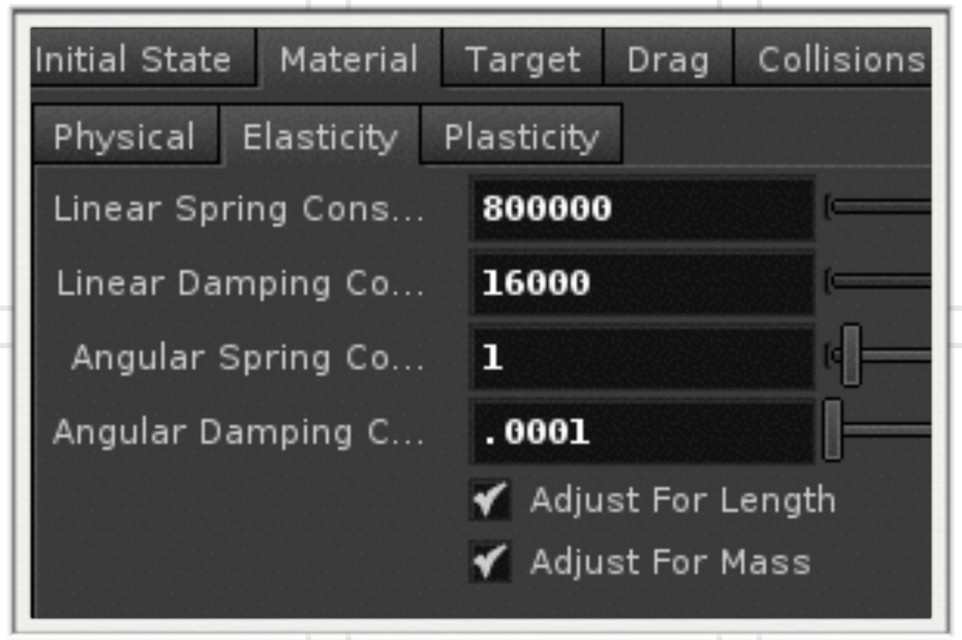
SIDE EFFECTS SOFTWARE

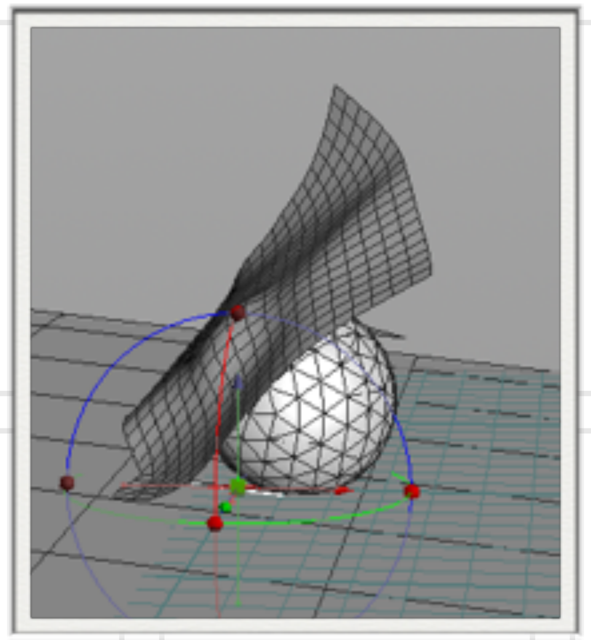
Elasticity

Elasticity	
Linear Spring Constant	This parameter defines how strongly the wire resists stretching.
Linear Damping Constant	This parameter defines how strongly the wire resists oscillation due to stretch forces.
Angular Spring Constant	This parameter defines how strongly the wire resists bending.
Angular Damping Constant	This parameter defines how strongly the wire resists oscillation due to bending forces.
Adjust For Length	Enabling this parameter will adjust spring and damper strengths according to segment lengths. This allows wire flexibility behavior to be independent of segment resolution.
Adjust For Mass	Enabling this parameter will adjust spring and damper strengths according to segment masses. This allows wire flexibility behavior to be independent of mass.

Add a Grid to the Simulation



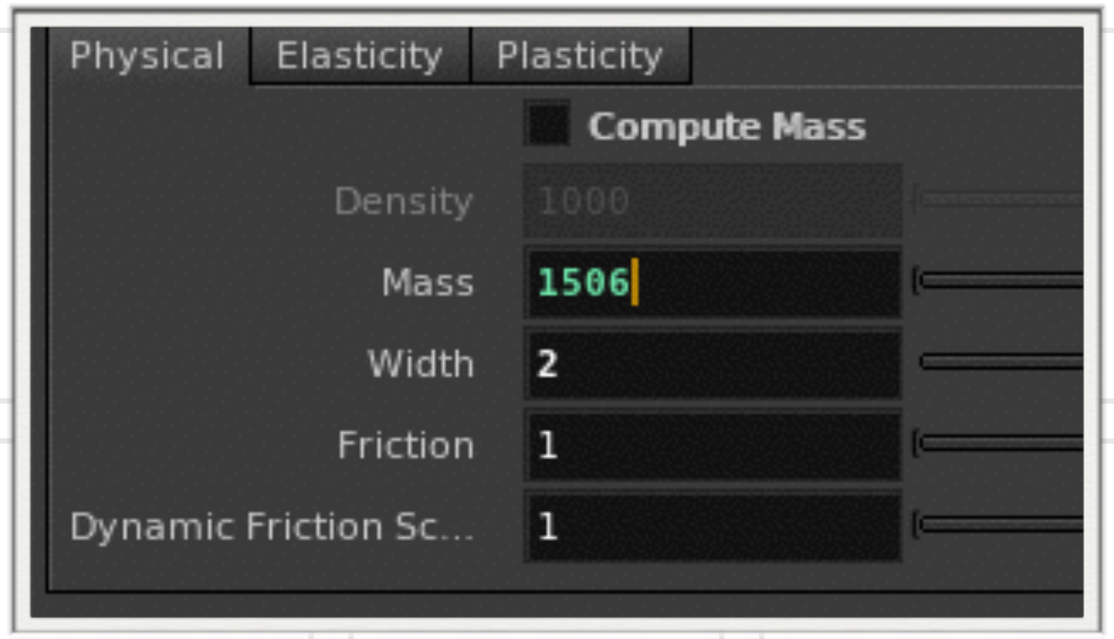




- **At the Object Level Drop Down a Grid**
  - Dive inside and set the size to 4x4
  - Divisions - 20x20
- **At the Object Level Position the Grid so it is over the Ball**
- **Make the Grid a Wire Object**
- **Run the simulation and observe the Results**
- **In the AutoDOP Network dive inside and select the Wire Object for the Grid Object**
- **Set Elasticity to values shown on left**
  - Re-run Simulation

SIDE EFFECTS SOFTWARE

Compute Mass?



By default simulations are calculating a mass for you based on the size of the object and the velocity.

If you want more accuracy try turning off Use Mass and enter your own mass.

Try using this website to get your masses

http://www.simetric.co.uk/si_materials.htm

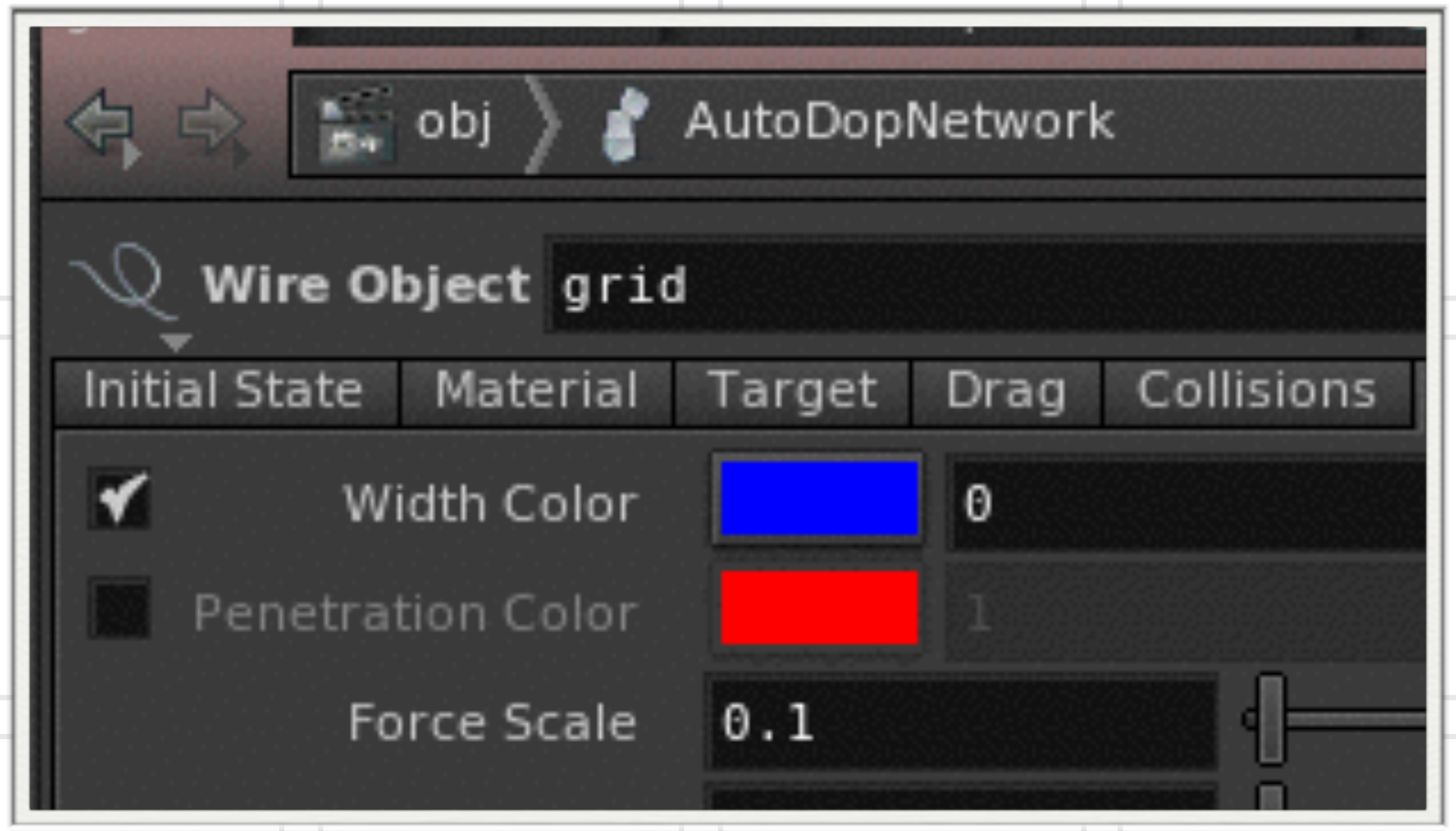
Make the Sphere a Mass of Portland Cement - 1506

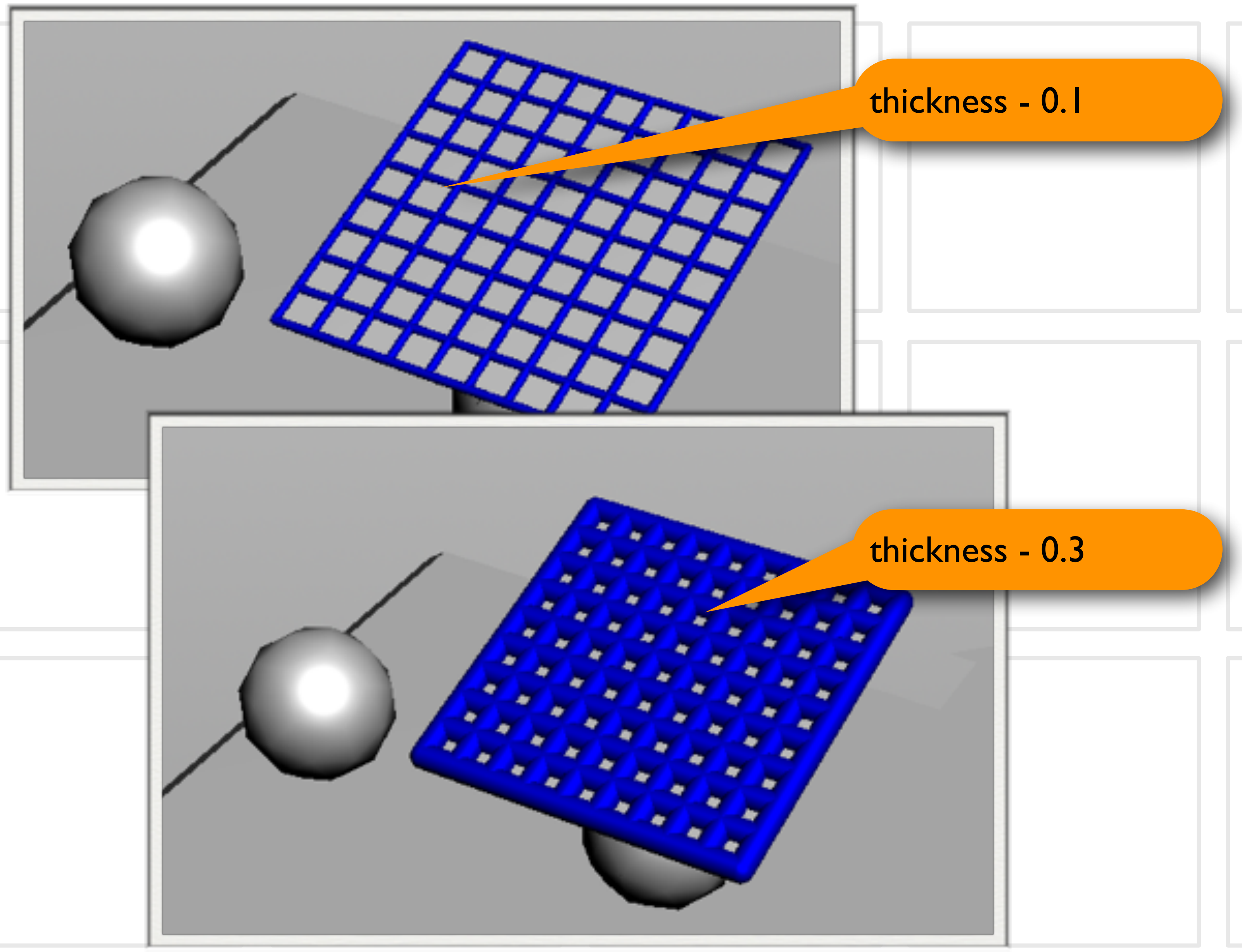
Make the Grid a Mass of Tobacco - 320

Run Sim and then reverse the numbers and rerun the sim

SIDE EFFECTS SOFTWARE

Understanding Width





Width is not the Width of the Object but rather the width of the springs going from one vertex to the other. Think of it as how thick is the coils of your spring. To see them go to the Visualization tab and turn on “Width Color”

Now go back to Physical tab and change widths. You will see in blue the thickness of the coils.

What does thickness do?

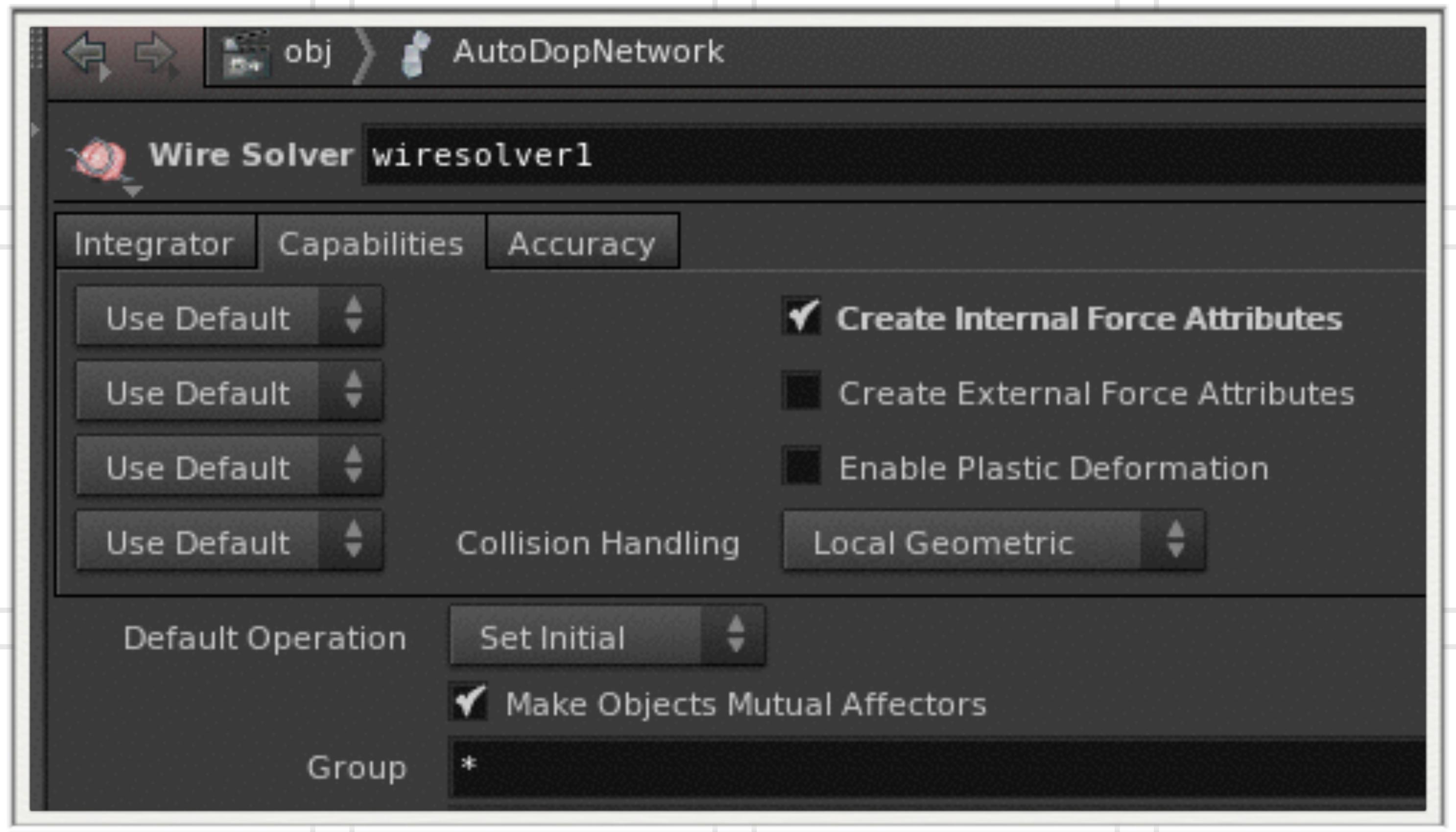
It acts as a multiplier of the forces

Try Using the Visualization Tab

The Visualization Tab helps in understanding what your simulation is doing?

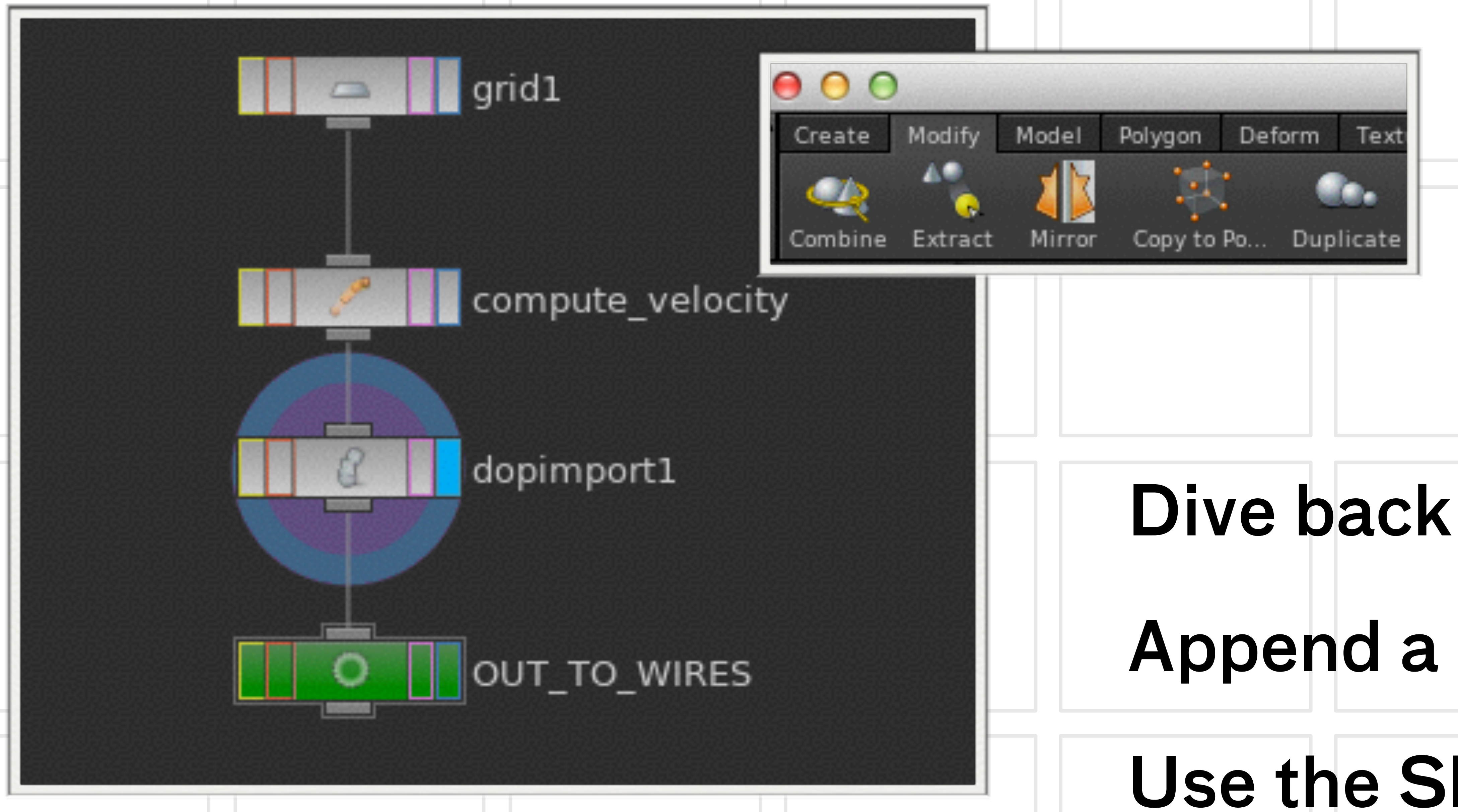
Try turning on internal force.

If you do not see the lines remember you have to turn them on in the Wire Solver Node



SIDE EFFECTS SOFTWARE

Adding “Hairs” to the Grid



Dive back into the Grid Object

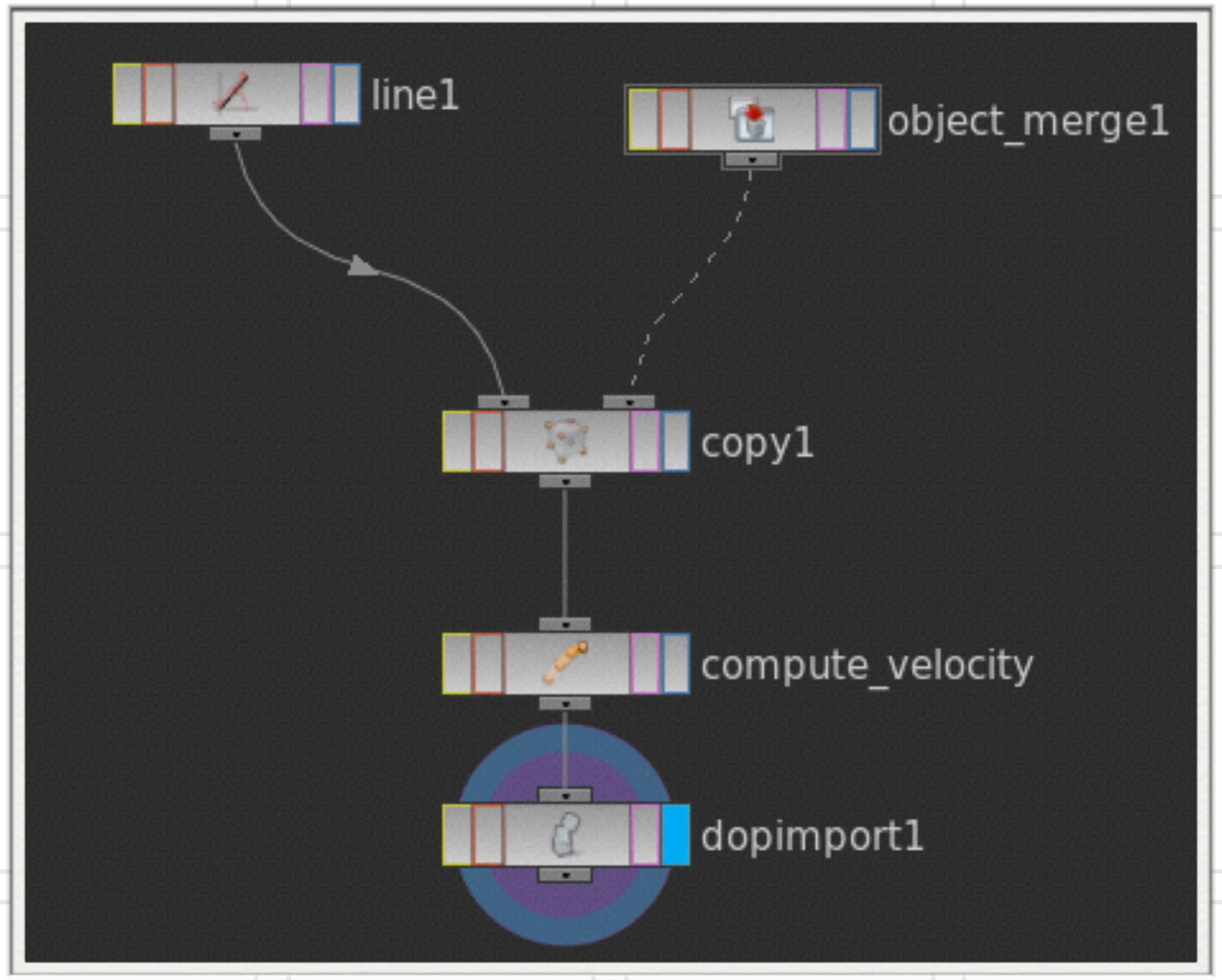
Append a Null to the dopimport1 - name it OUT_TO_WIRES

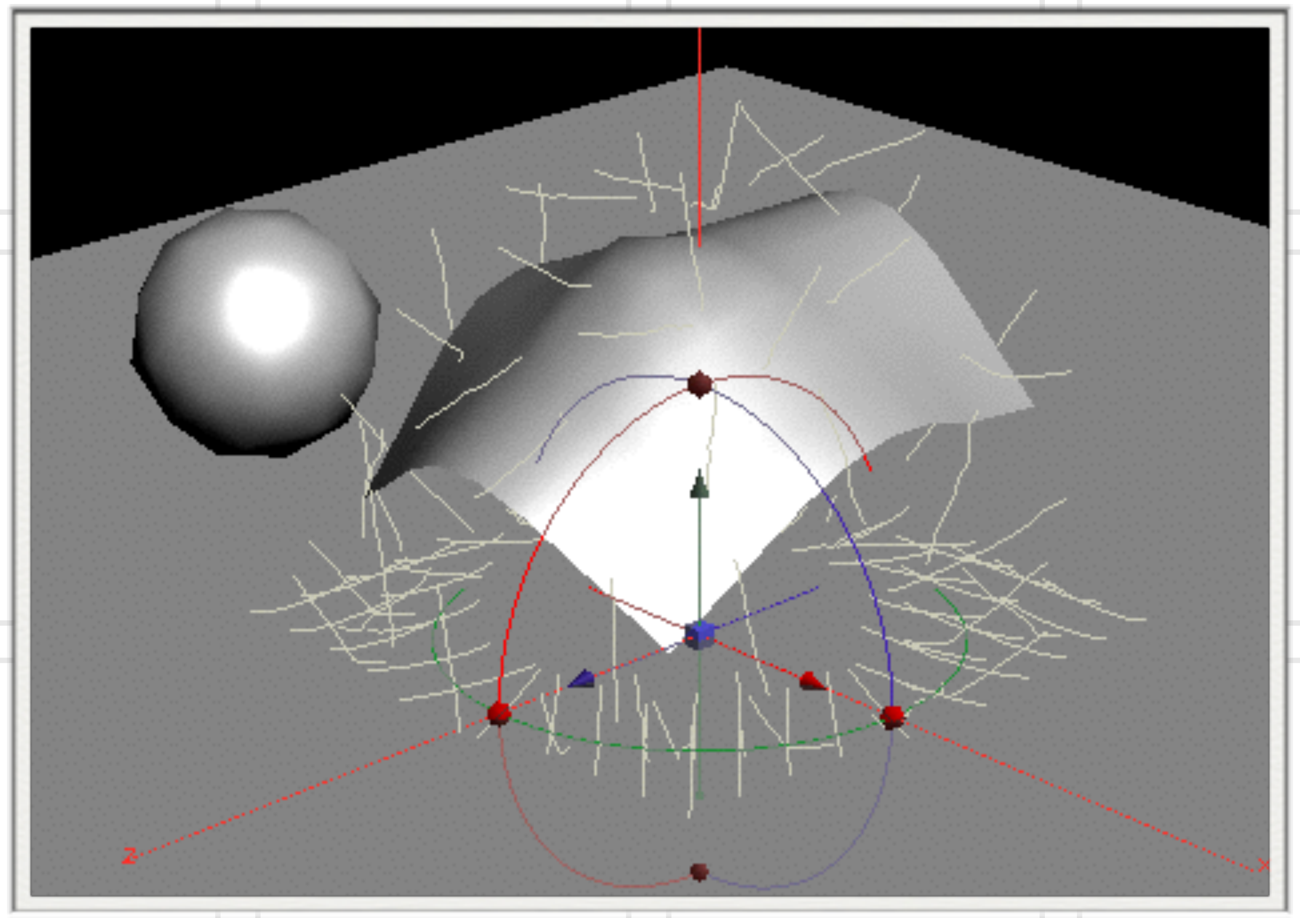
Use the Shelf Tool to Extract the node into a new Geometry node

Rename the new Geometry “Wires”

Dive Inside

Adding “Hairs” to the Grid (cont.)





Go up to the Object level and name the new extracted object “wires”

Dive inside - drop down a line. change the direction to (0,0,1)

Copy the line onto the imported object

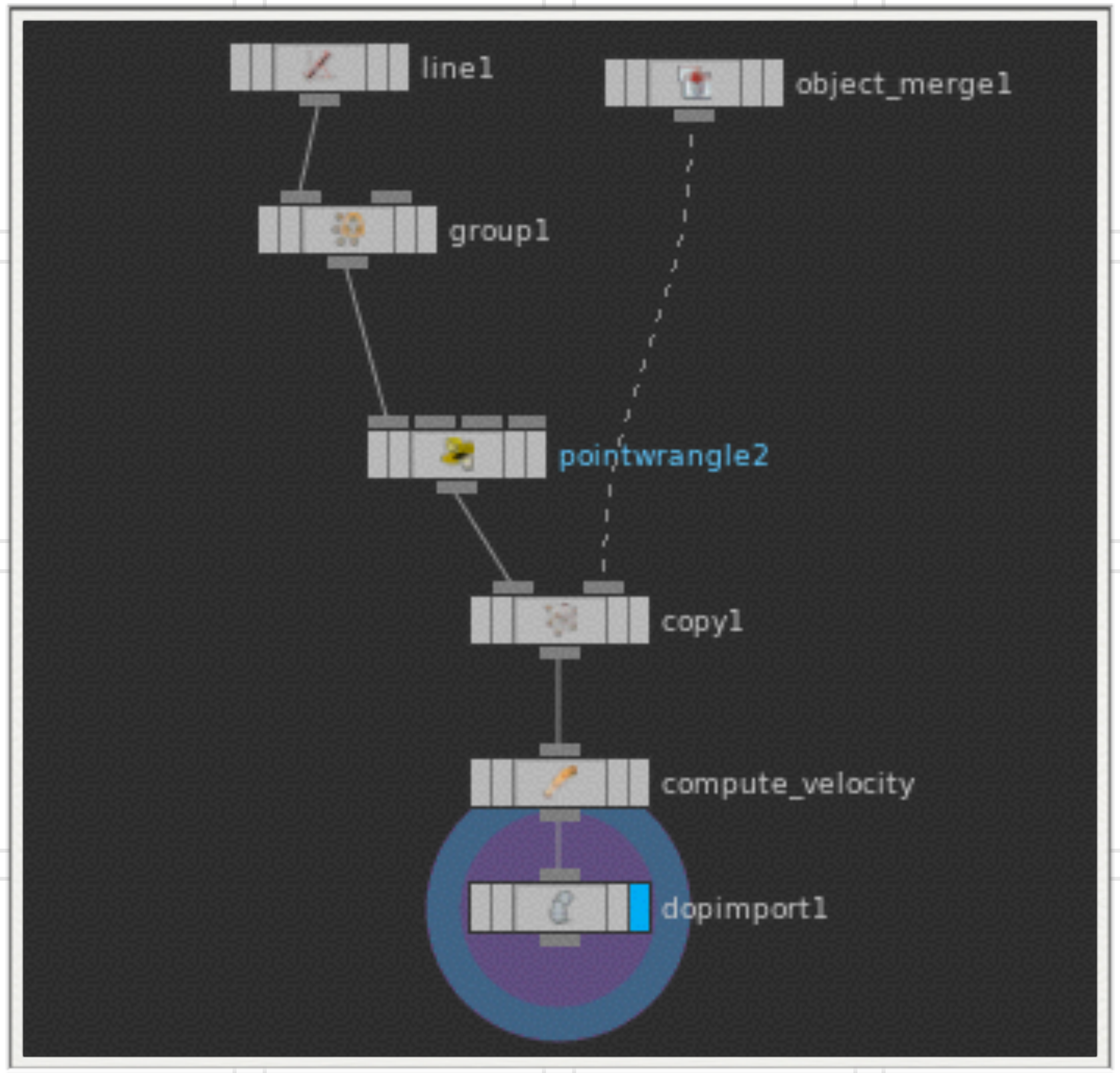
Go back to the object and make it also a wire object

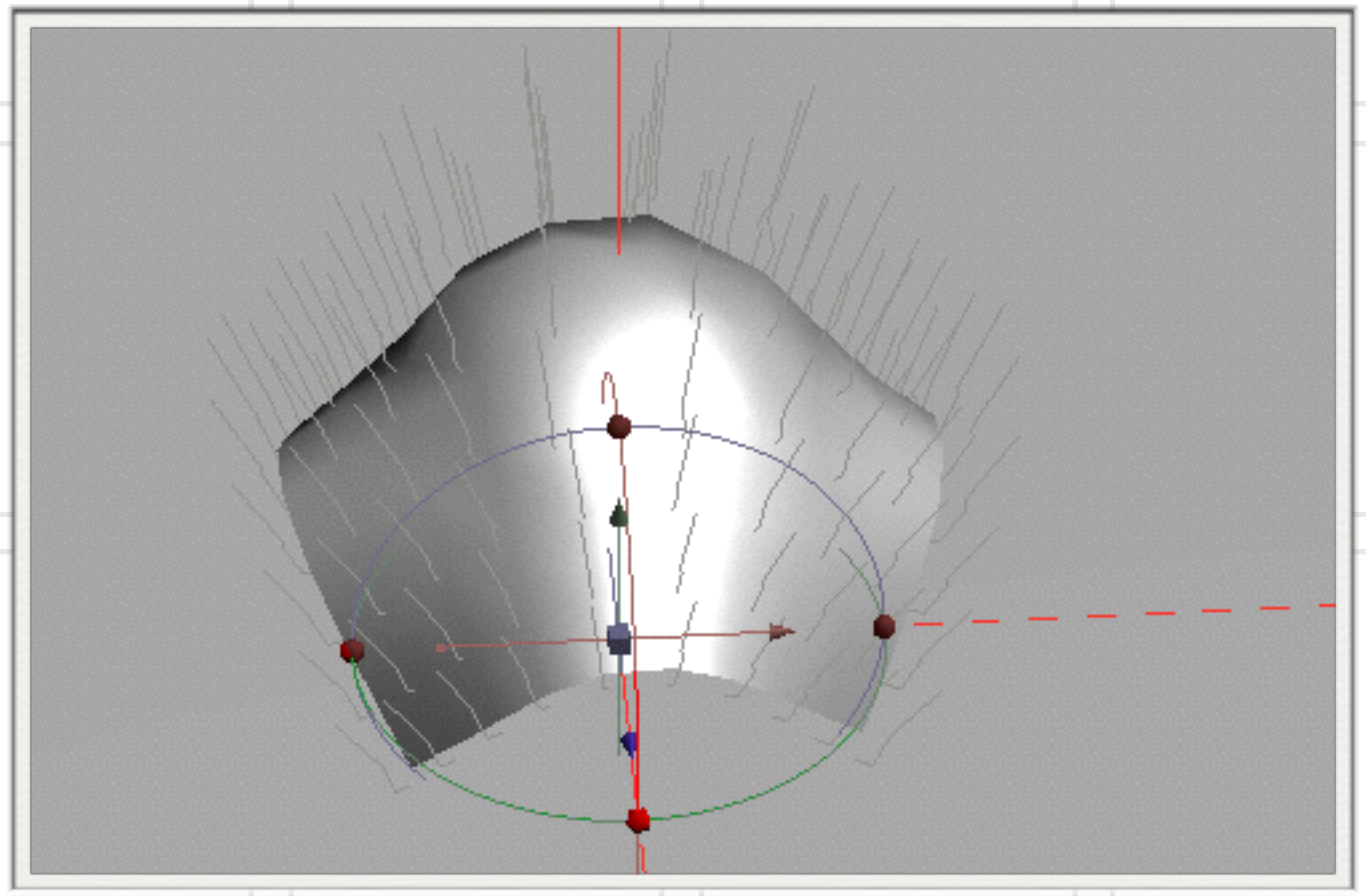
Run the simulation

The wires break off

SIDE EFFECTS SOFTWARE

Adding “Hairs” to the Grid (cont.)





Go back into the Wire Object and Append a Group SOP to the line

Name the group “wire” set to “points” and use point 0 1 as the pattern

Append a Point Wrangle Node - Set the Group to Wire

```
f@gluetoanimation = 1.0;
```

Rerun the simulation

The Wires no longer break off. What happened?

SIDE EFFECTS SOFTWARE

Wire Attributes

gluetoanimation - Values greater than 0.5 cause a point's position and orientation to be constrained to the input geometry.

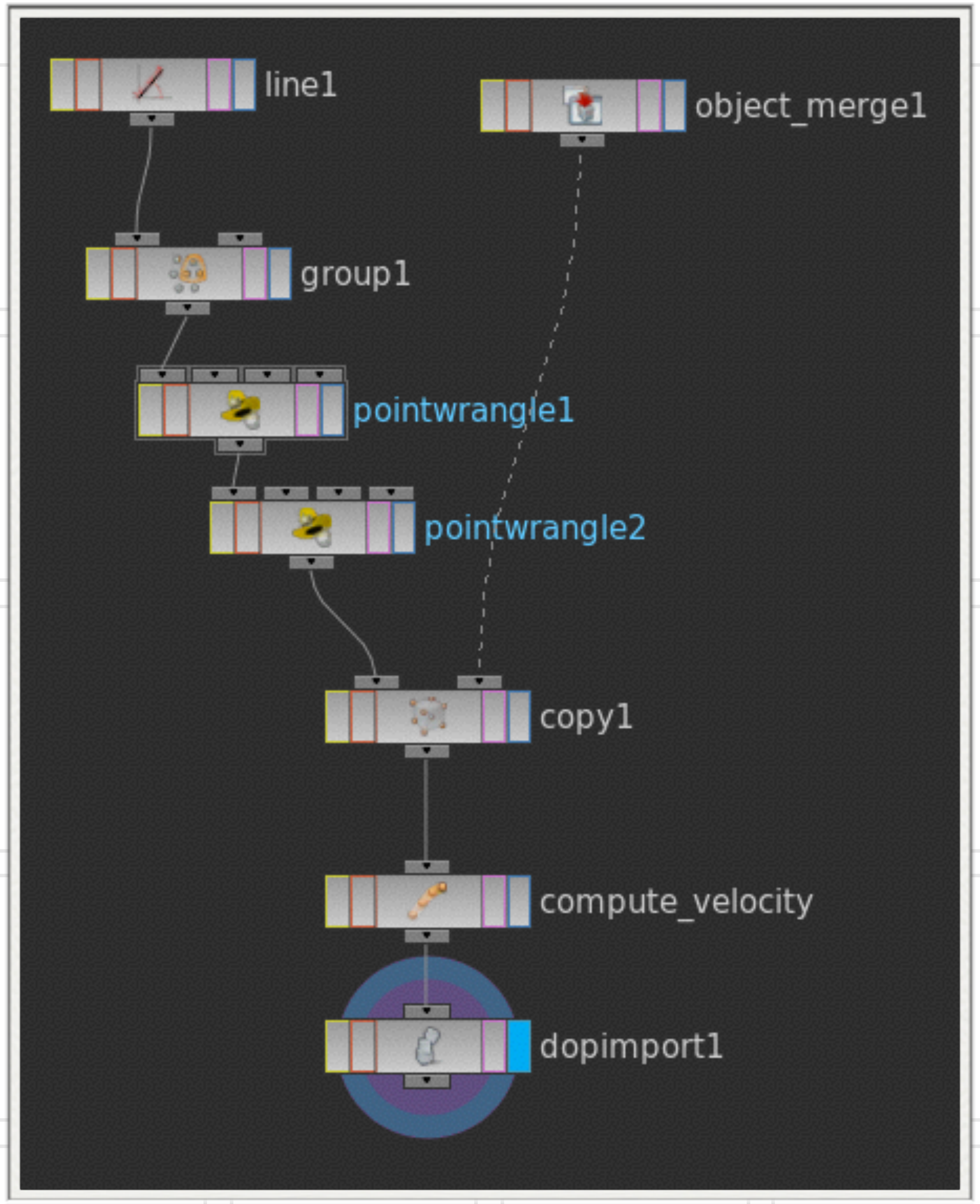
pintoanimation - Values greater than 0.5 cause a point's position to be constrained to the input geometry.

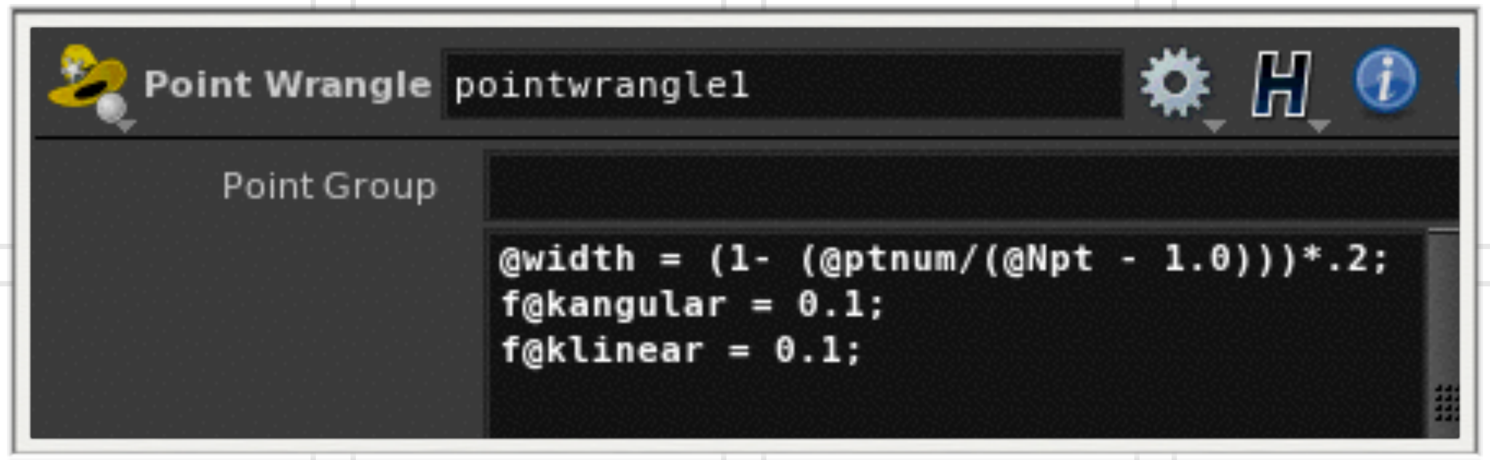
kangular - Defines how strongly the wire resists bending.

klinear - Defines how strongly the wire resists stretching.

SIDE EFFECTS SOFTWARE

Adding klinear, kangular, & width





Append another PointWrangle right after the group sop

This time there is no need for a group

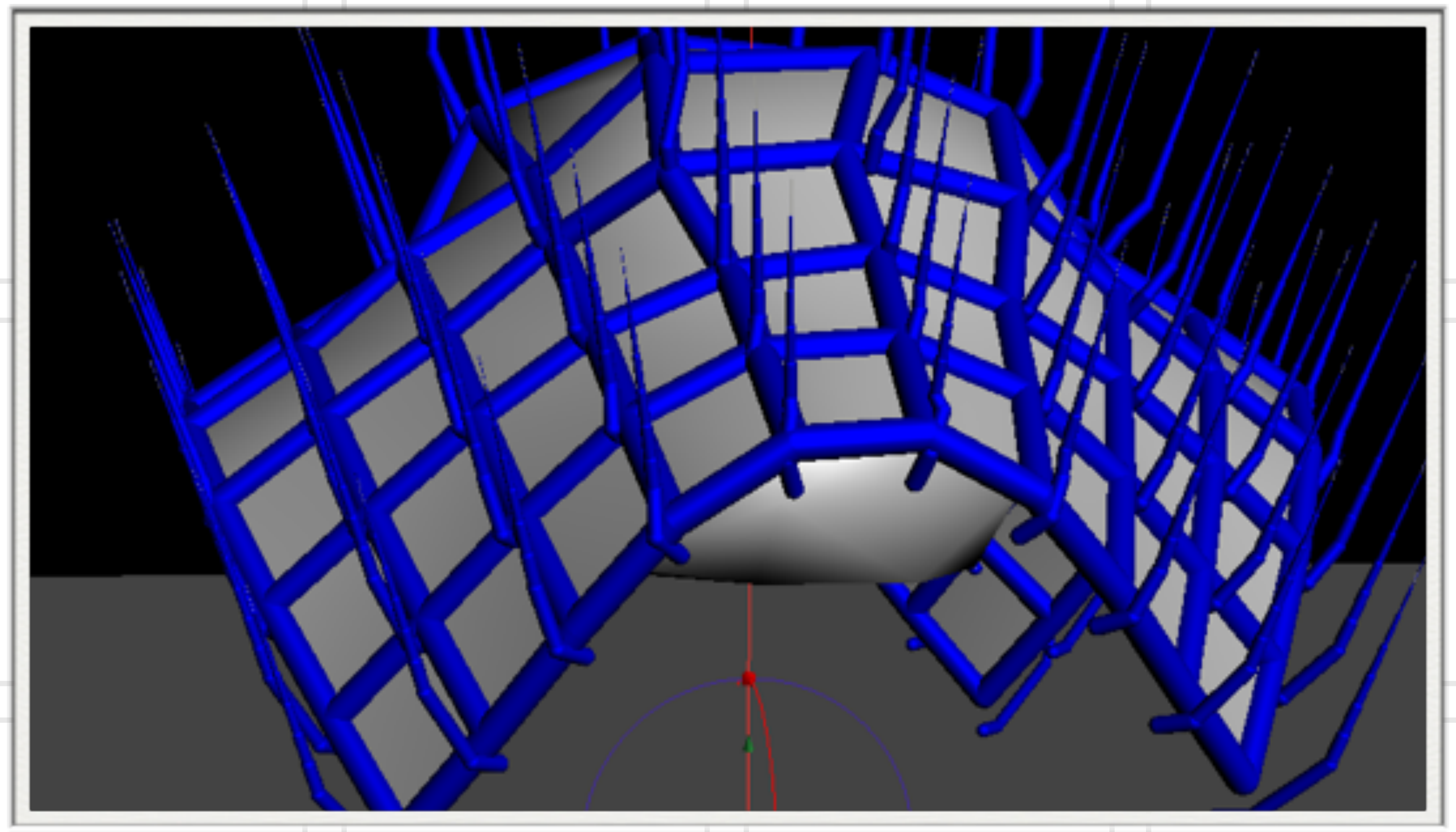
Inside the point wrangle

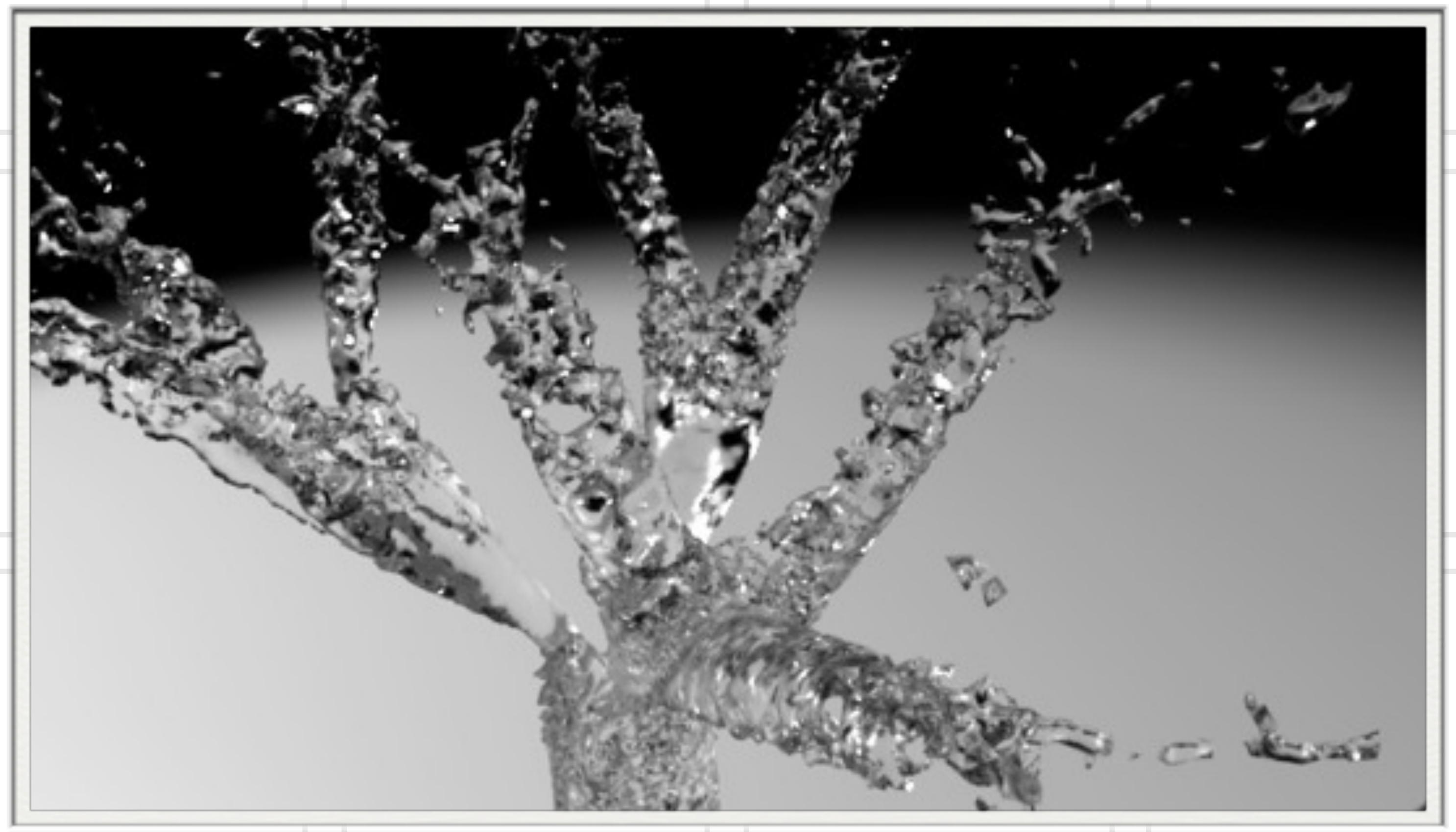
```
@width = (1- (@ptnum/(@Npt - 1.0)))*.2;
f@kangular = 0.1;
f@klinear = 0.1;
```

Rerun the simulation

Notice the wires bend and stretch

Visualizing the Point Wrangle





Dive into the Autodop Network and select the Wire Object

Go to the Visualization tab and turn on “width”

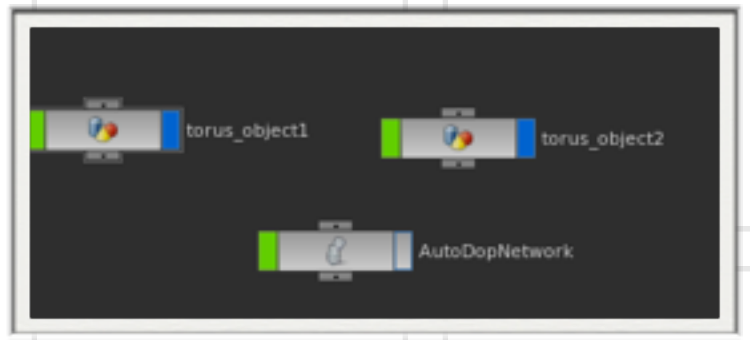
Notice the width and therefore the forces taper off along the wire

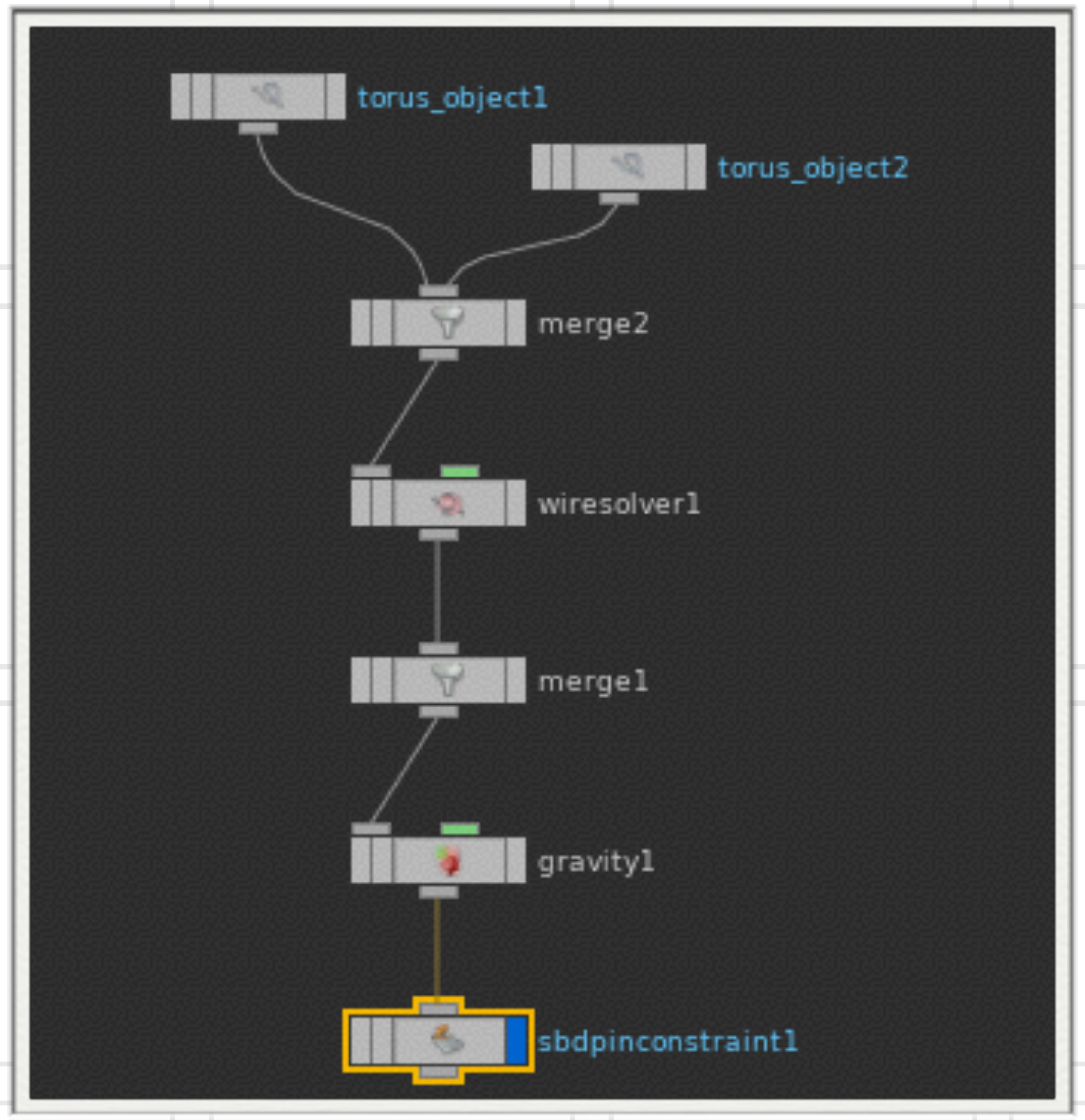
A Little on Constraints

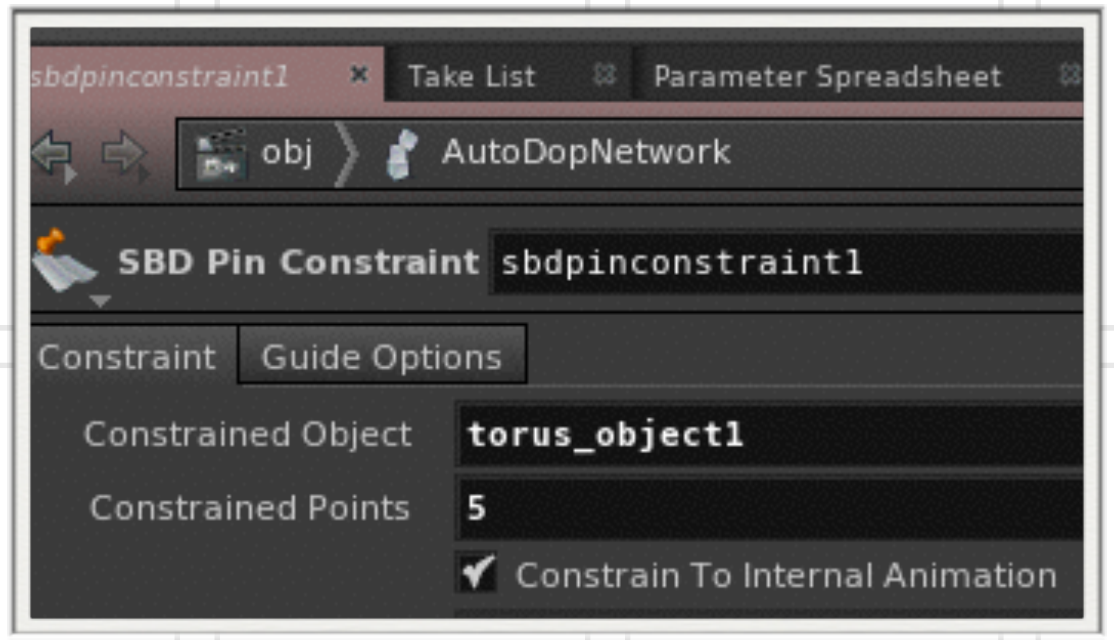
SpringNetworkConstraints

SIDE EFFECTS SOFTWARE

Setup







At the object level drop down two torii separated in distance

Make both of them a wire object

On the first torus add a pin constraint attached to a point on the outside

Run the simulation - One tire swings while the other drops

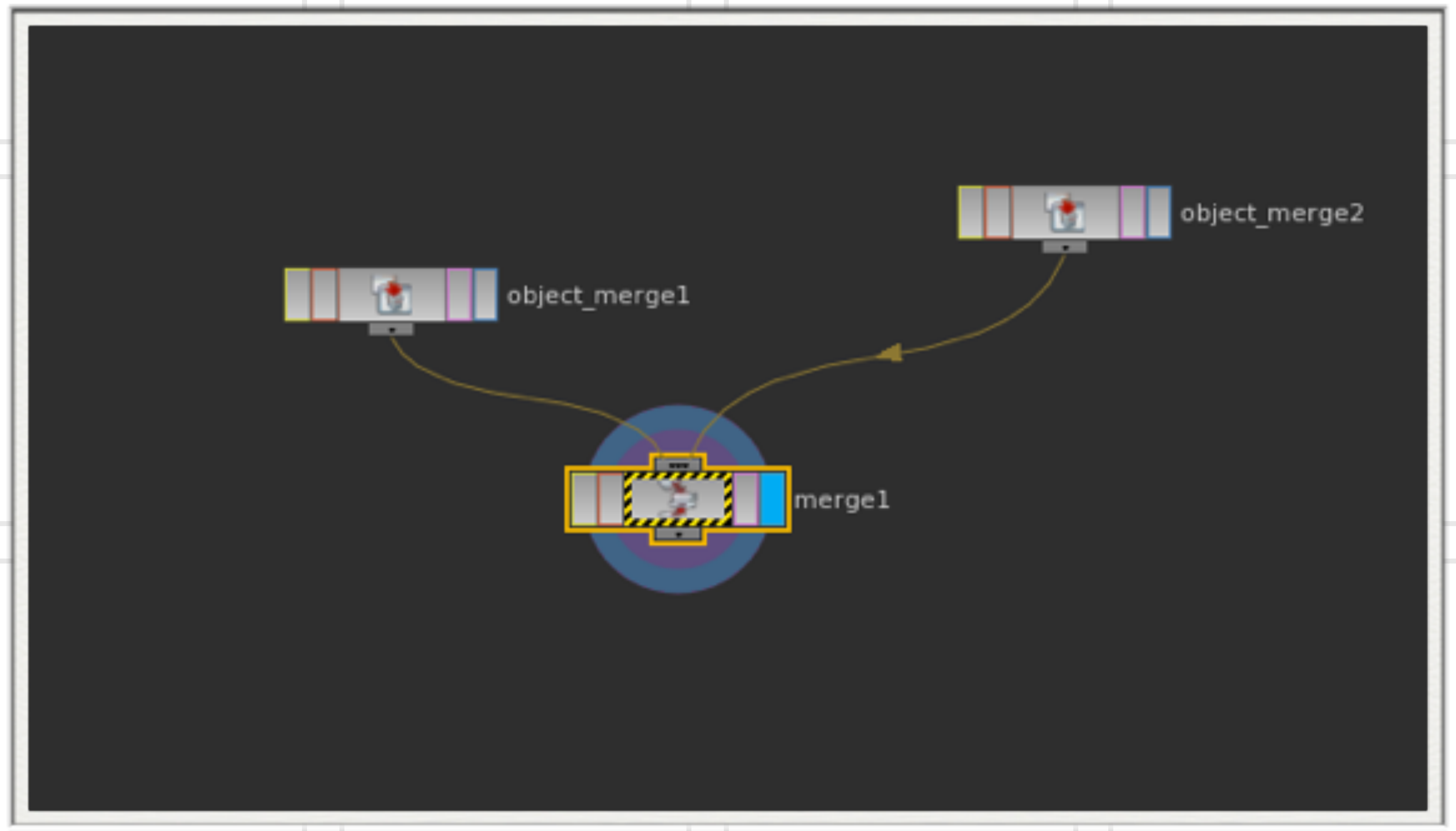
Dive into the AutoDop Network

Notice the sbdpinconstraint appended to the end of the network

The sbdpinconstraint points to the point number you selected

SIDE EFFECTS SOFTWARE

Hand Making a springnetworkconstraint



Append a springnetworkconstraint node to the pin constraint

We will set the parameters of this node in a bit

Go up to the object level and drop dow a geometry node

Dive inside, delete the File node, and add two object merge nodes

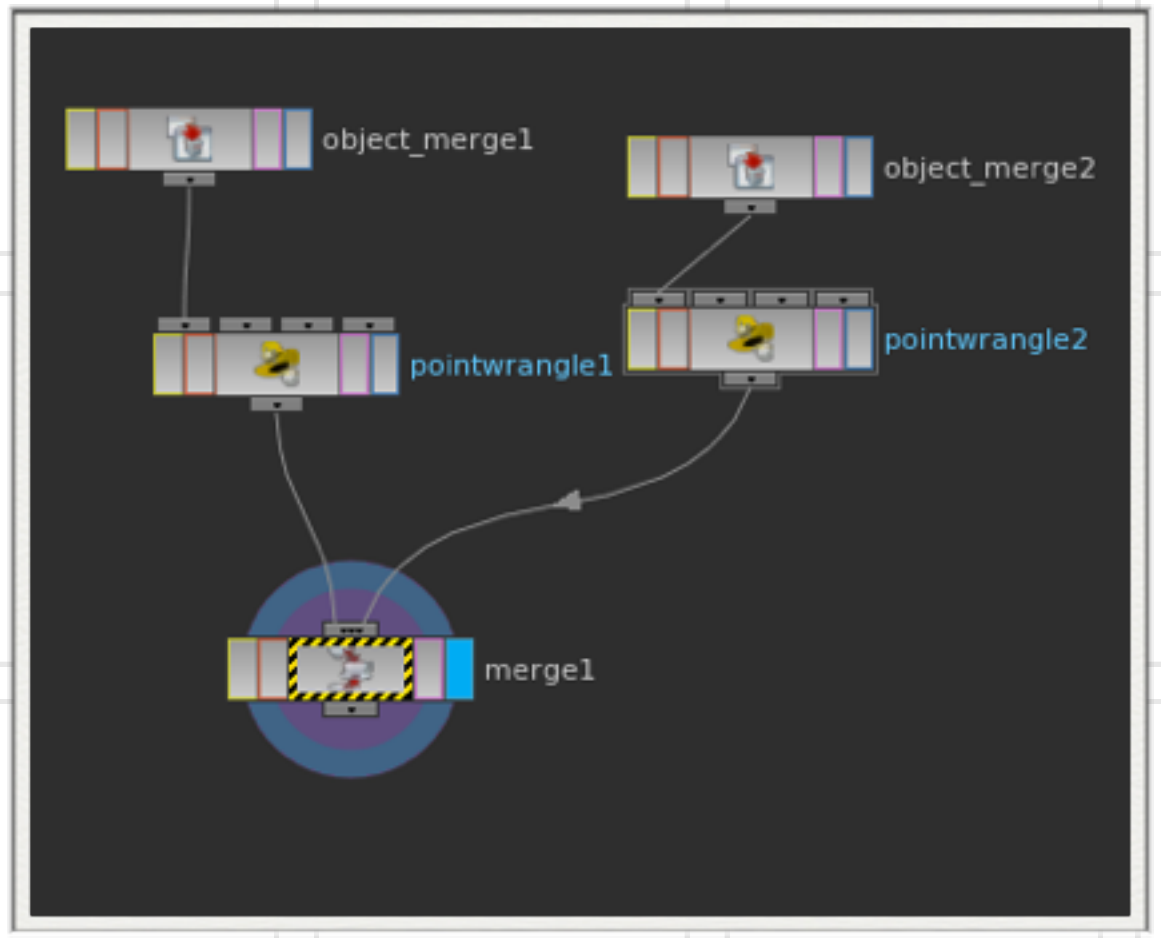
Point the two object merge nodes to the to torii's dopimport nodes

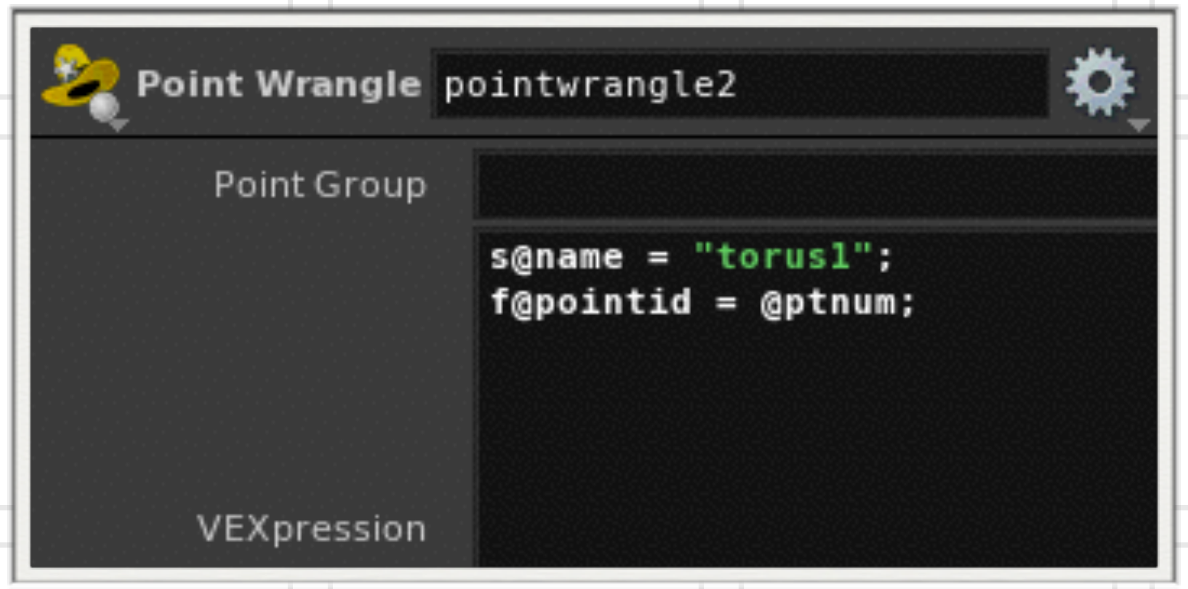
Set transform - "into this object"

Append a Merge and wire the two object merges

SIDE EFFECTS SOFTWARE

Hand Making a springnetworkconstraint (cont.)





Append a point wrangle to both object merge nodes.

The spring network constrain node needs three attributes

name, pointid, and restpos

For the point wrangle of the first object merge

s@name = “torus";

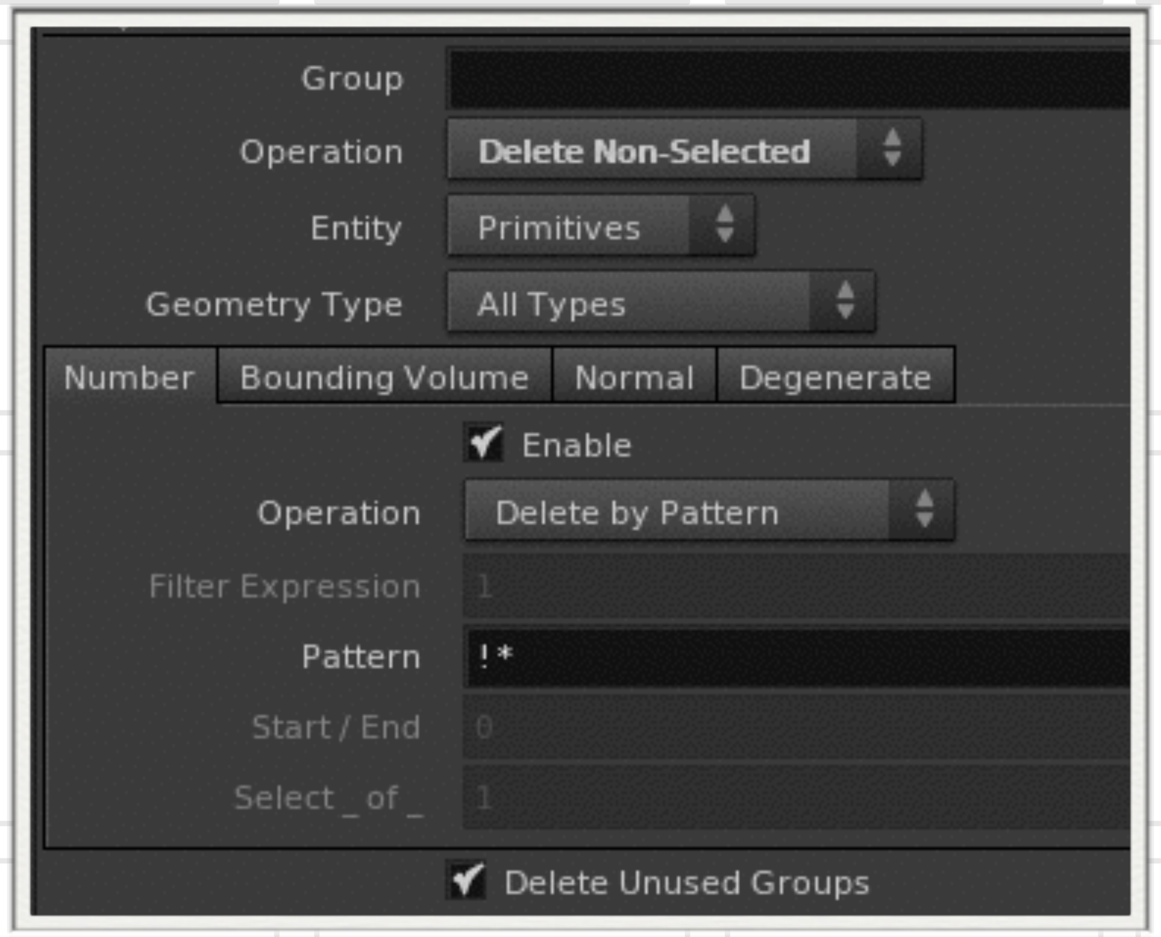
f@pointid = @ptnum;

For the point wrangle of the second object merge

s@name = "torus1";

f@pointid = @ptnum;

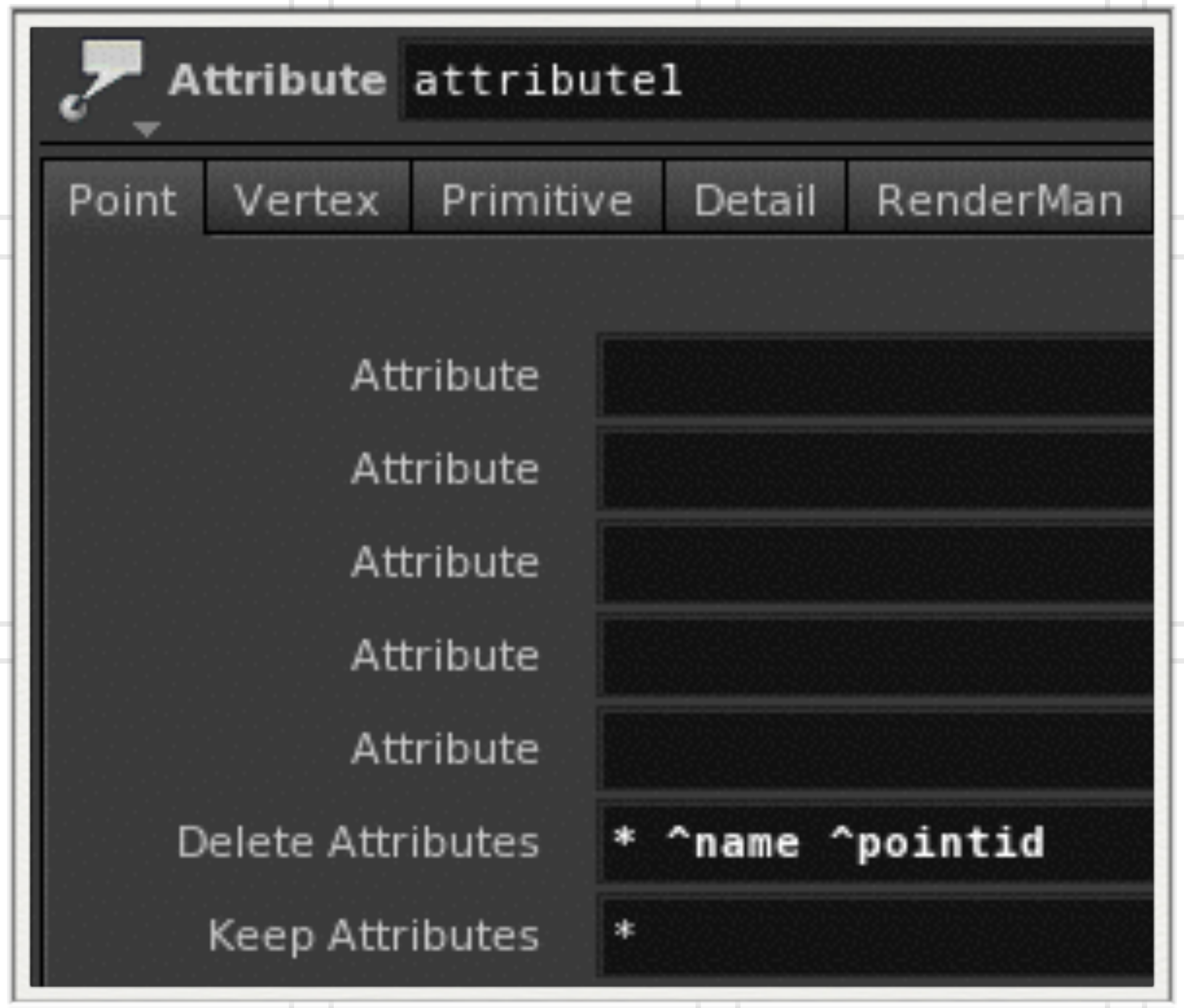
Hand Making a springnetworkconstraint (cont.)

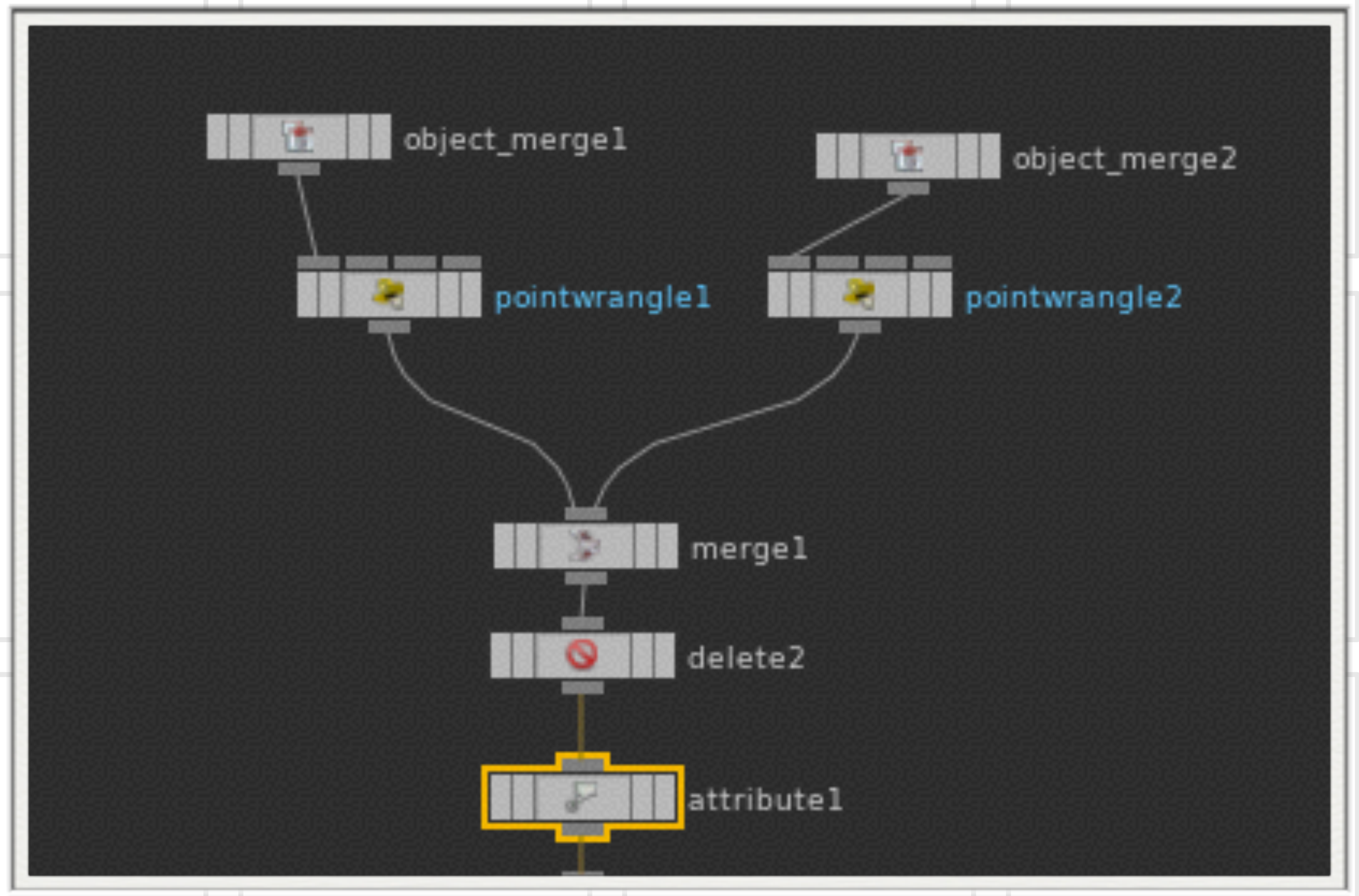


After the Merge append a Delete SOP

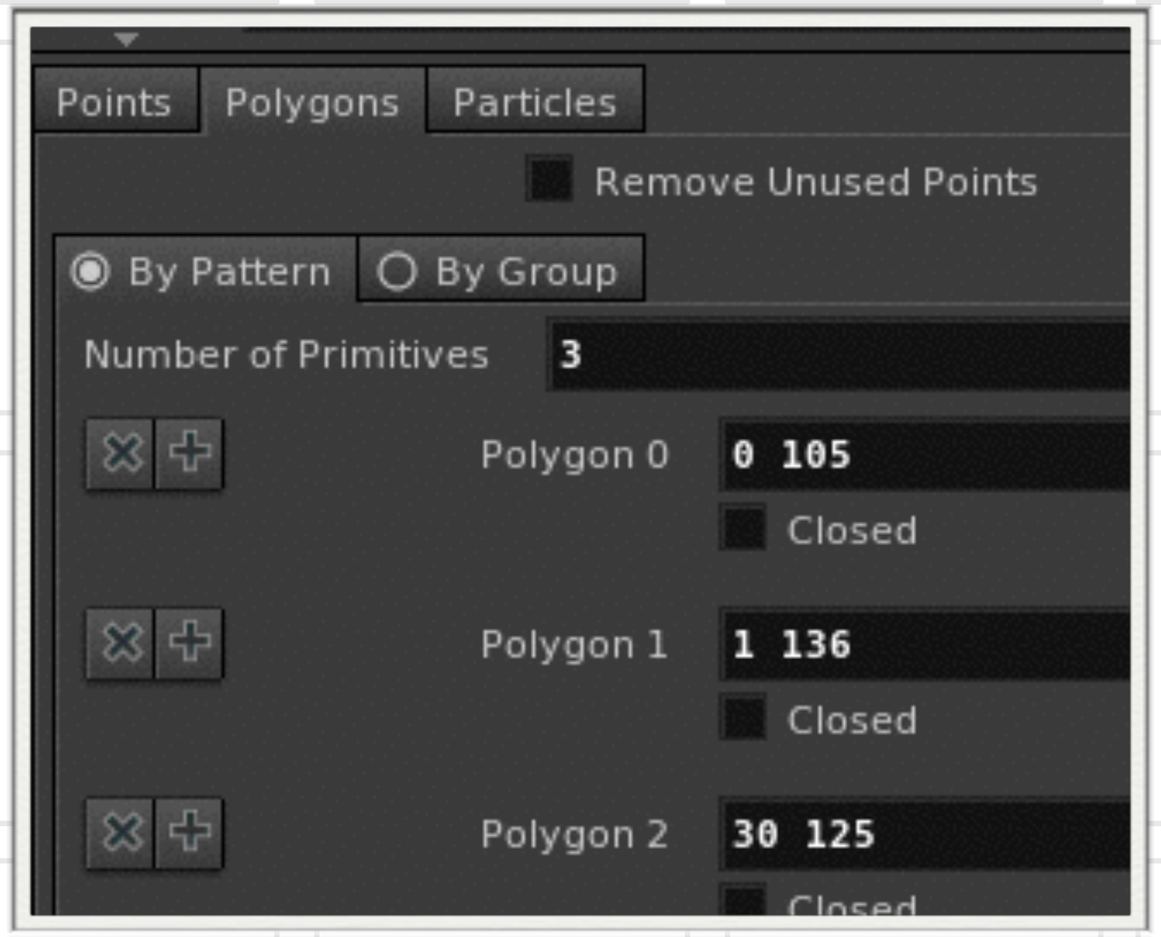
We want to delete all the primitives and keep the points. See image on the left.

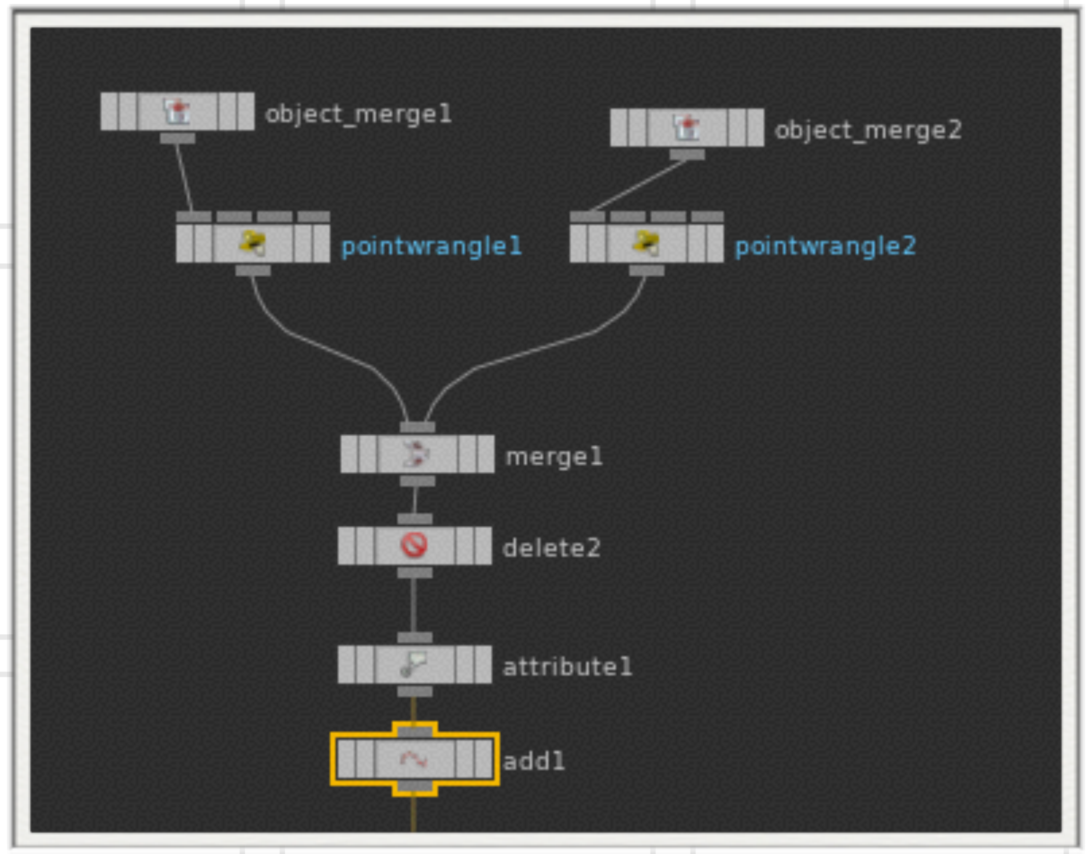
We then want to delete all the attributes except for name and pointed. See image on the left.





Hand Making a springnetworkconstraint (cont.)

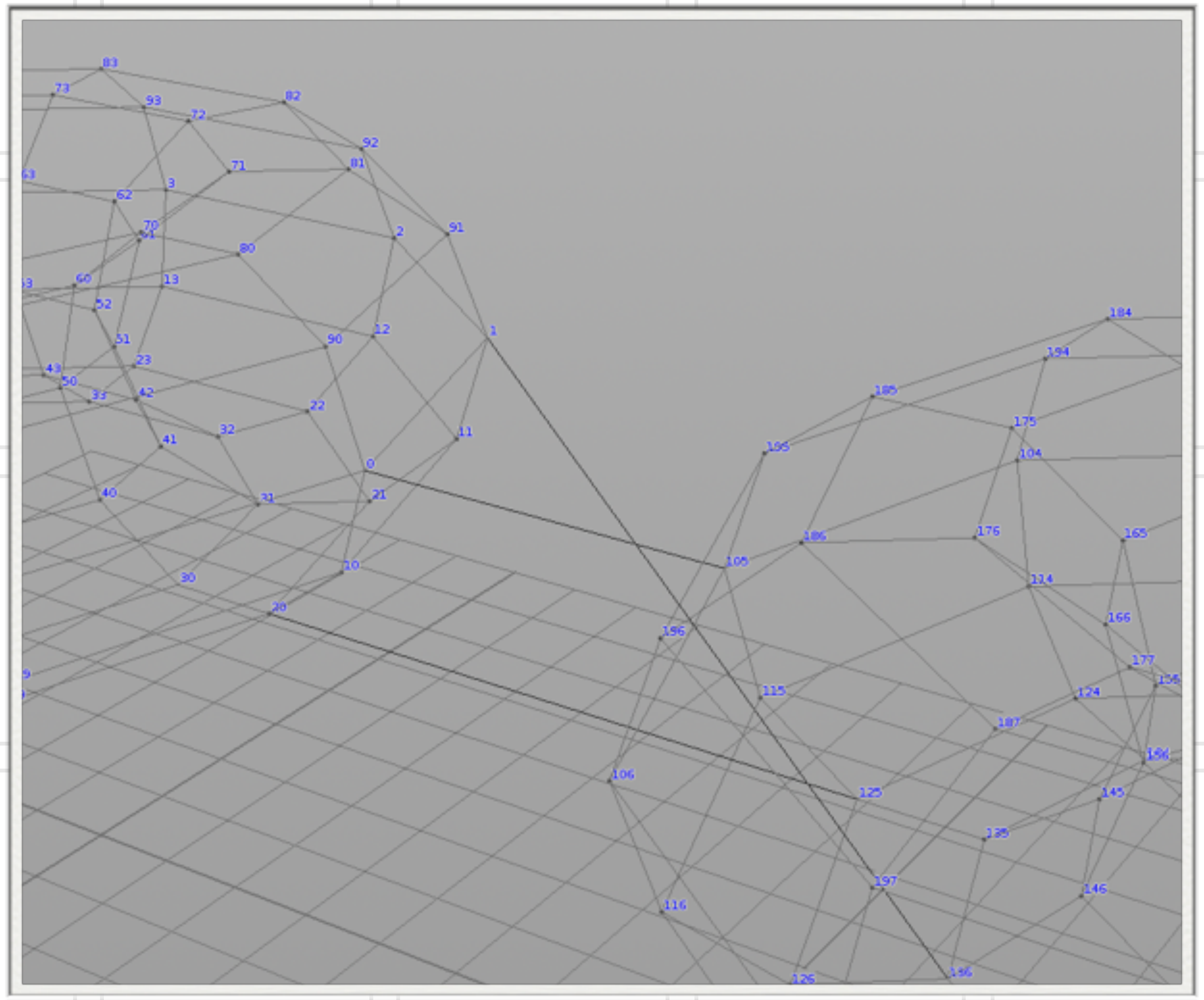




Now we go to create lines between the torii

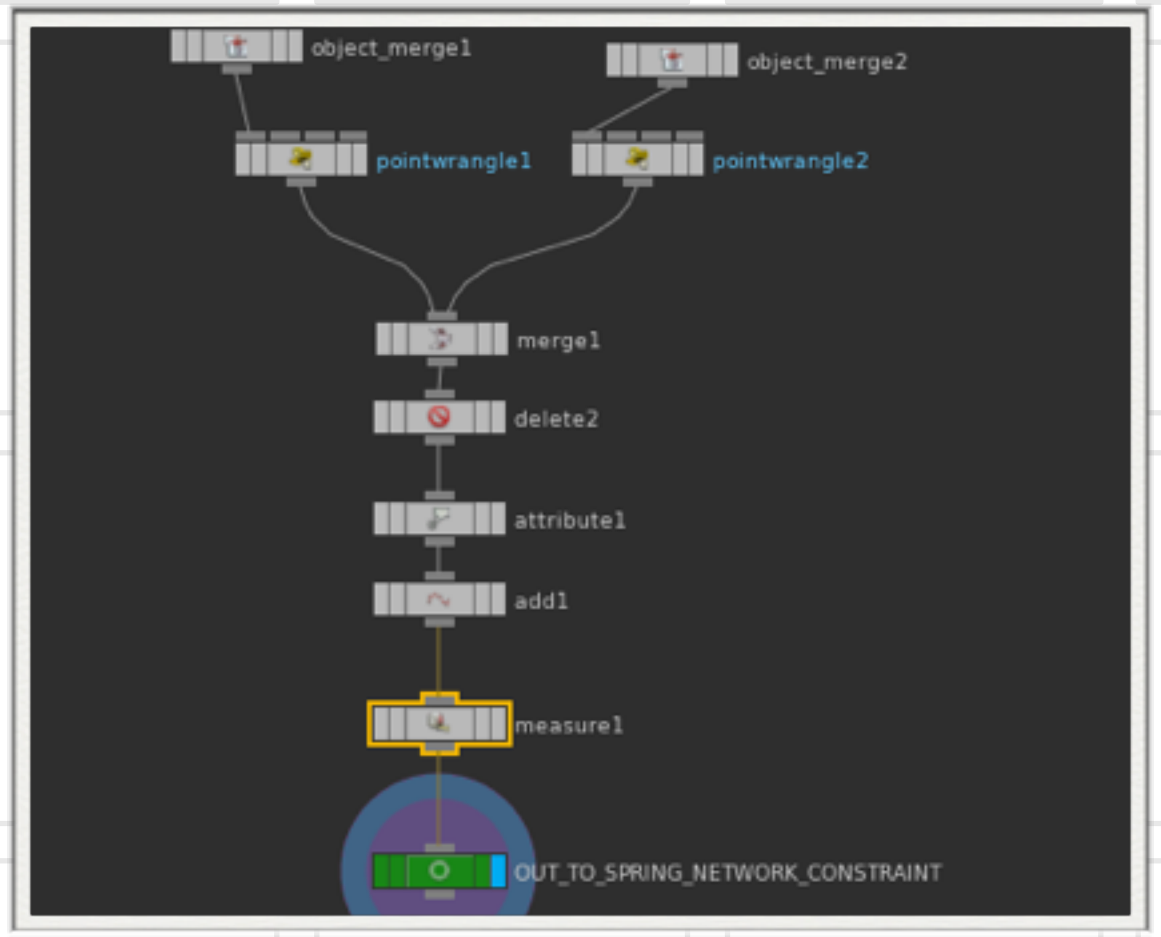
Append an add SOP and create primitives where the point pairings have the start on one torus and the end on the other.

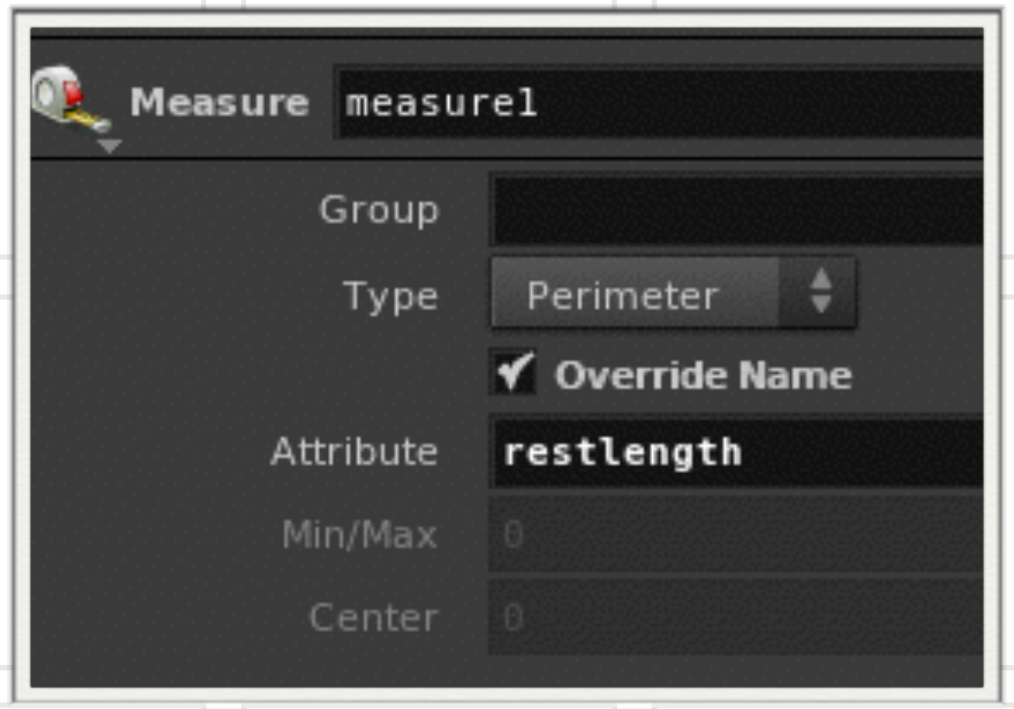
See attached image



SIDE EFFECTS SOFTWARE

Hand Making a springnetworkconstraint (cont.)





We already have created name and pointed. The only attribute left to create is - rest length

Append a Measure SOP

Type - Perimeter

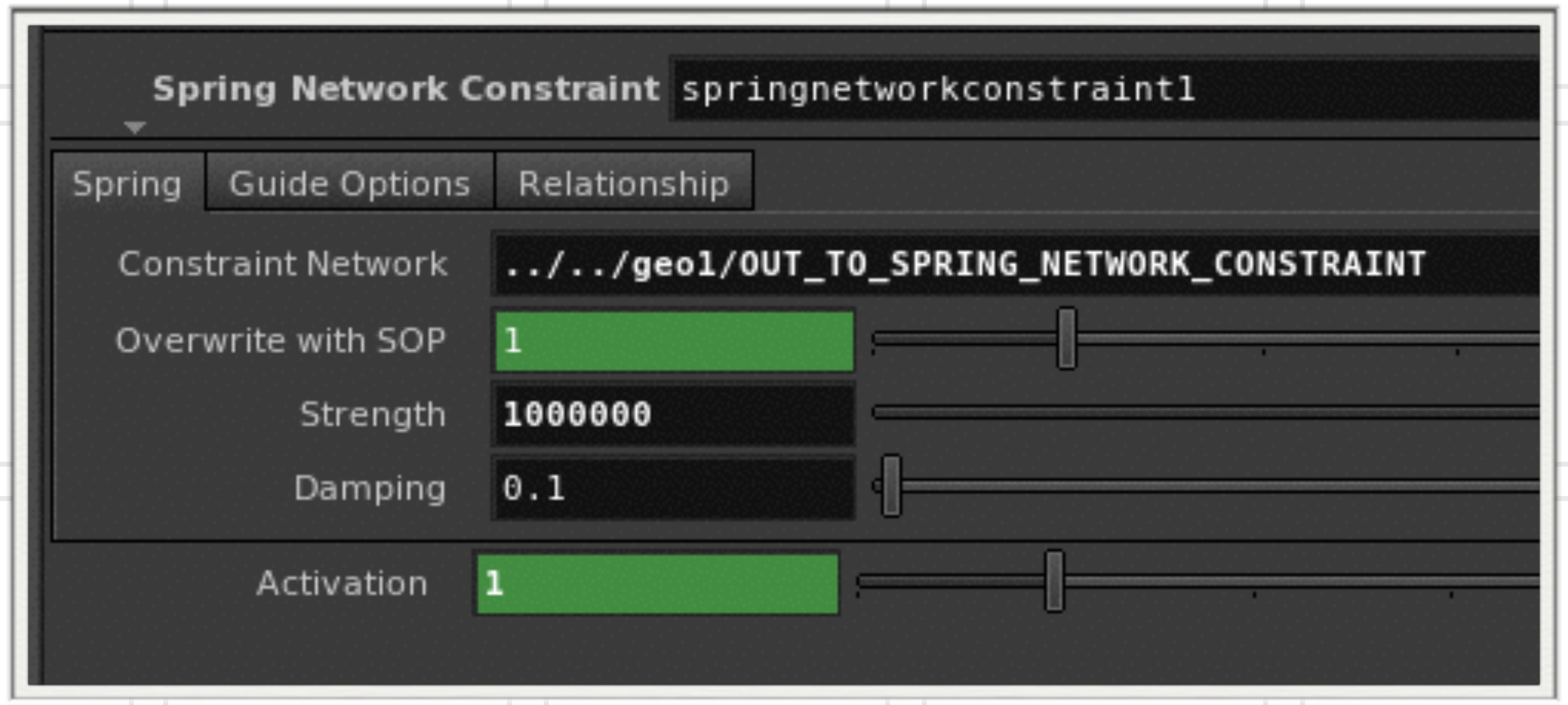
Attribute - restlength

Append a Null - name it OUT_TO_SPRING_NETWORK_CONSTRAINT

This is the node we will use to with the Spring Network Constraint in the Autodop Network

Dive inside the Autodop Network and select the Springnetworkconstraint

Set as shown and run simulation

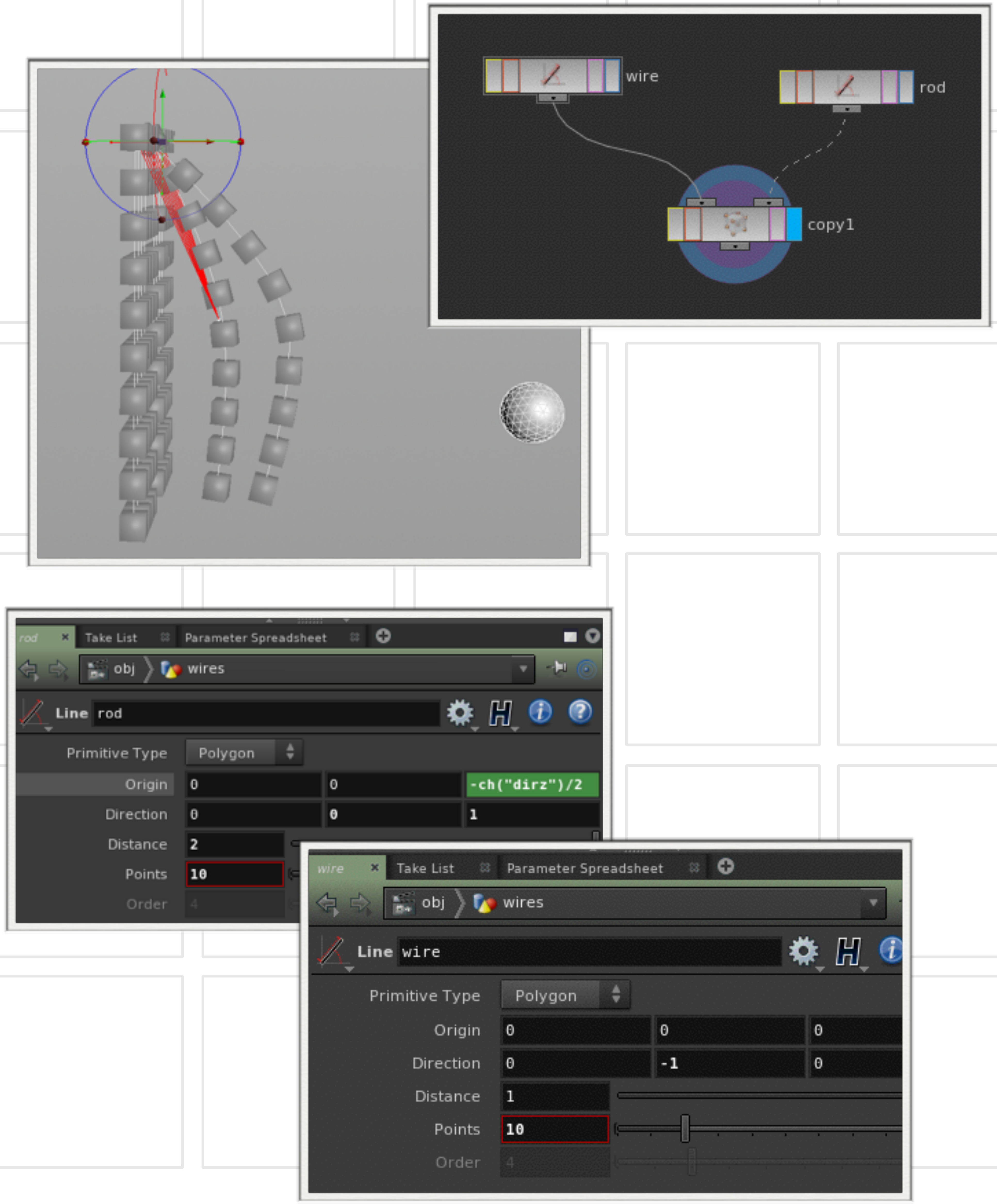


SIDE EFFECTS SOFTWARE

Project 1- Beaded Curtain

SIDE EFFECTS
SOFTWARE

Setup



Drop down a geometry object and dive inside

Delete the File SOP

Drop down a line in the Z Direction (0,0,1)

- Make its length 2, 10 points, name “rod”
- Origin (0,0, -ch(“dirz”)/2

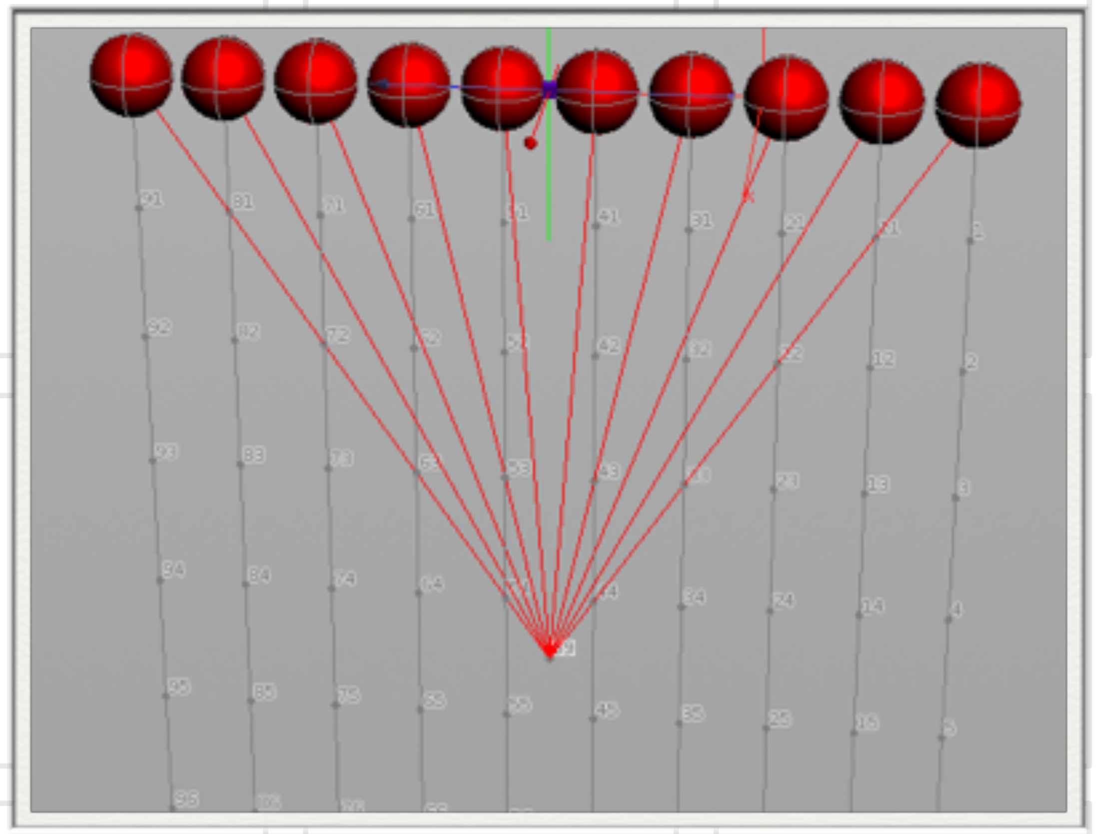
Drop down another line in the -y direction

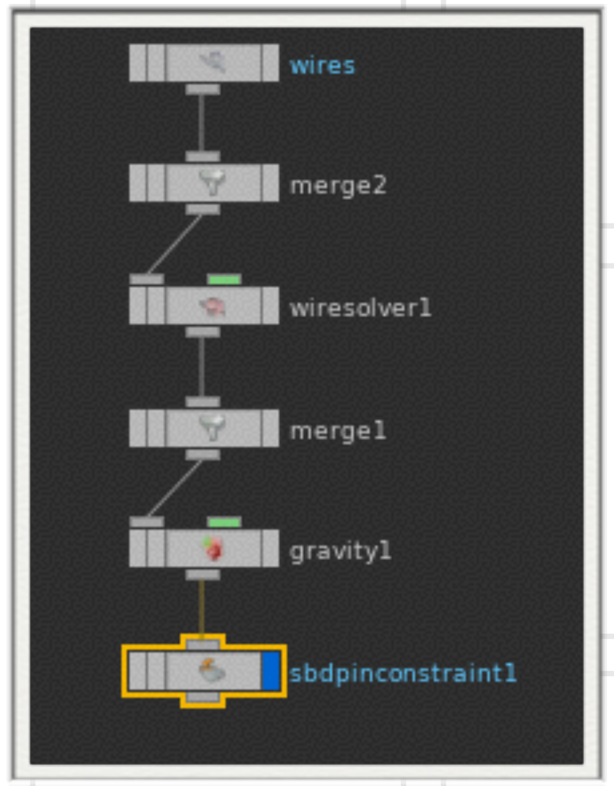
- Make its length 3, 10 points, name “wires”

Copy the wires to the rod

SIDE EFFECTS SOFTWARE

Setup (cont.)





Go up to the Object Level

Make the Wire Object, well a wire object

Run the sim. The wires should all fall down

In the Wire Shelf Tool select “Wire Pin Constraint

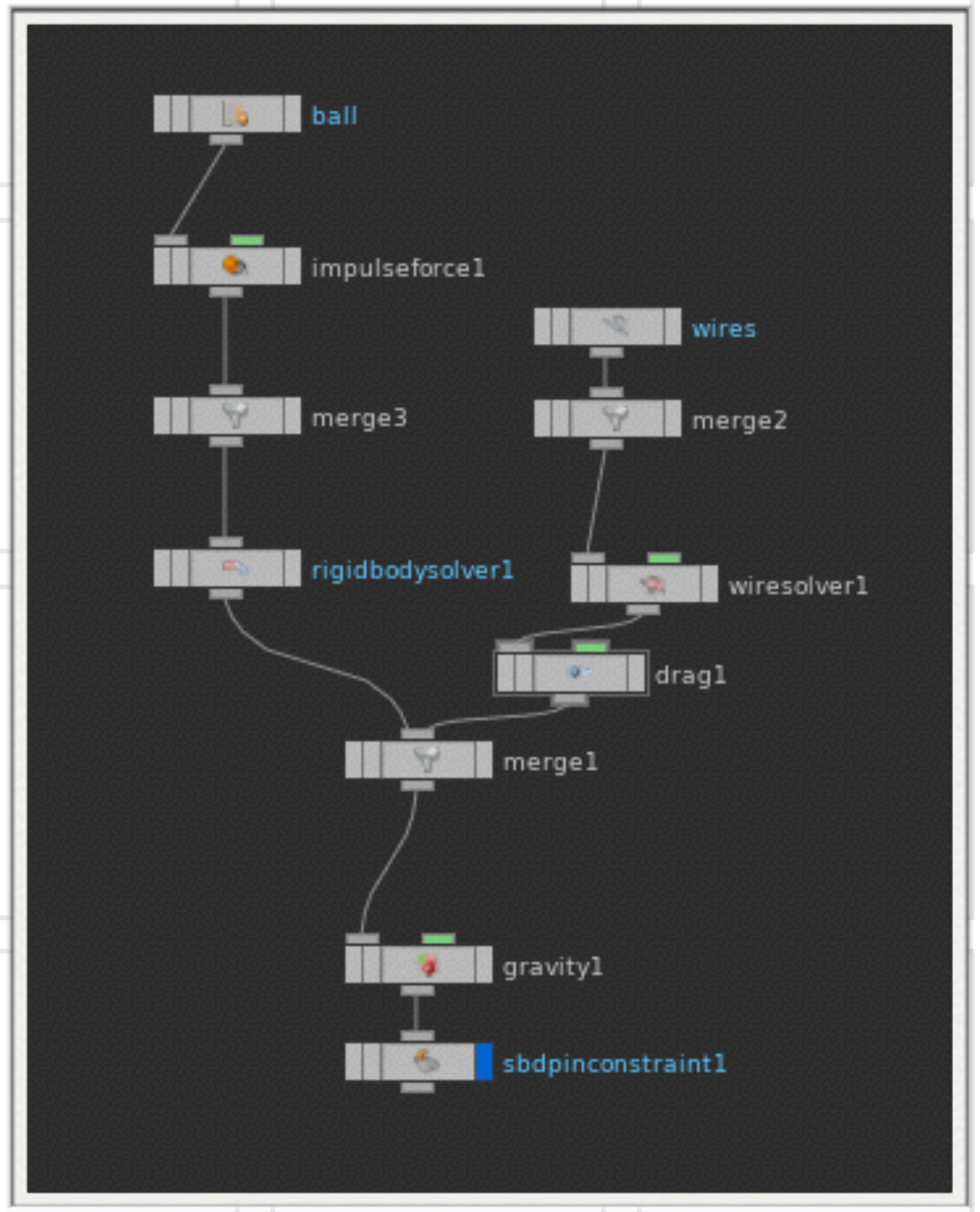
Select top row of points and hit enter

You will see pin constraint in the viewport

In the AutoDopNetwork you will see a SBDPinConstaint appended to gravity

Run the Sim = The wires no longer fall

Simulation



At the Object level drop down a Sphere, name it Ball

- Radius 0.3, and make it polygonal
- Translate (-5,-1,0)

At the Object Level make the Ball and RBD Object

- Run sim and see it fall straight down

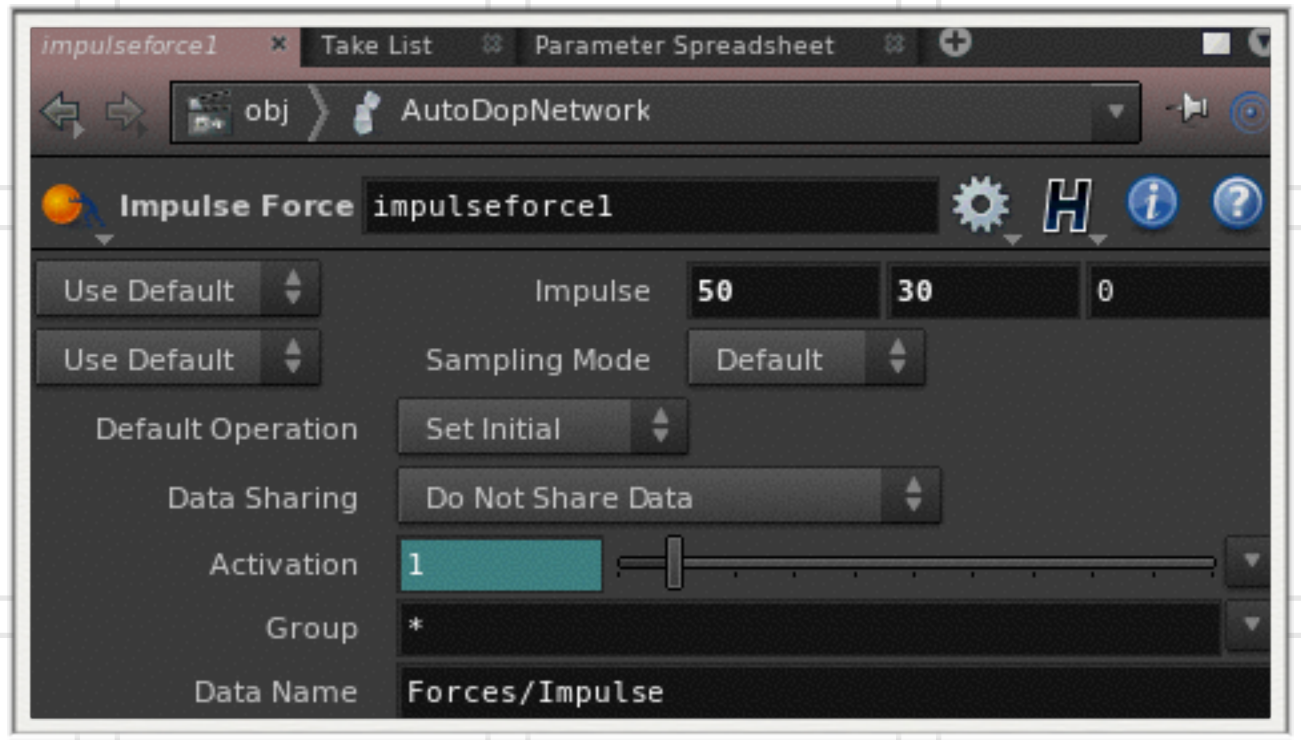
Dive into the AutoDOPNetwork

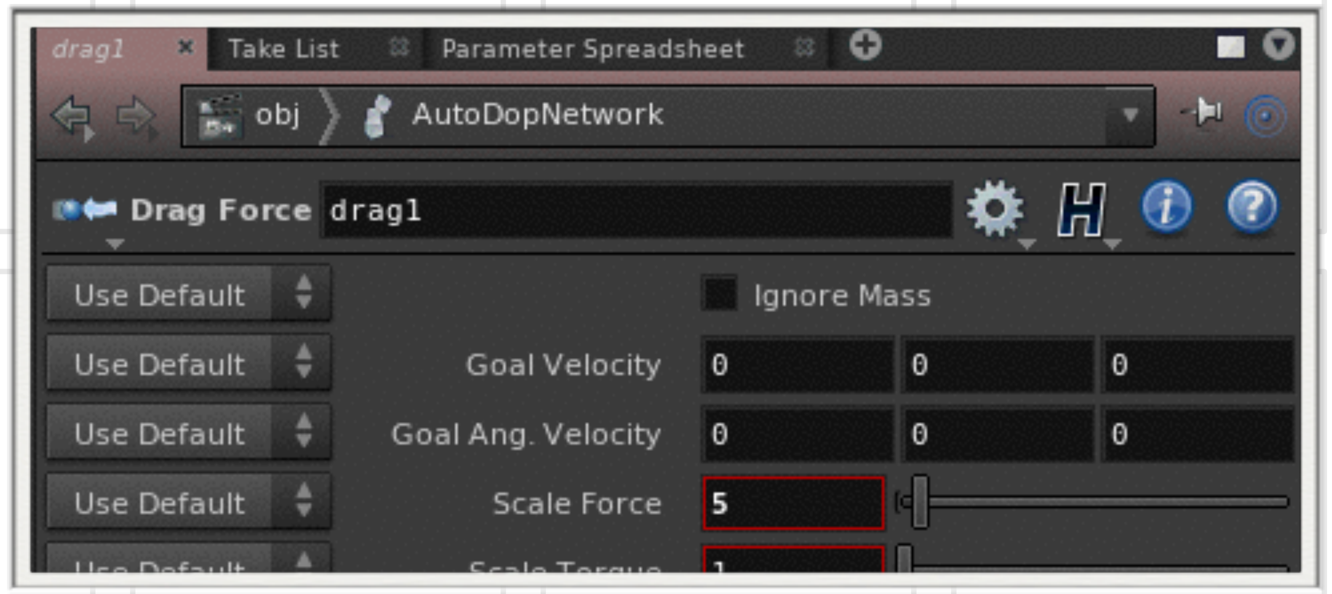
- Append a Impulse Force to the Sphere Object
- Set the Impulse Force to (50,30,0)

Run the sim and see the ball collide with the wire

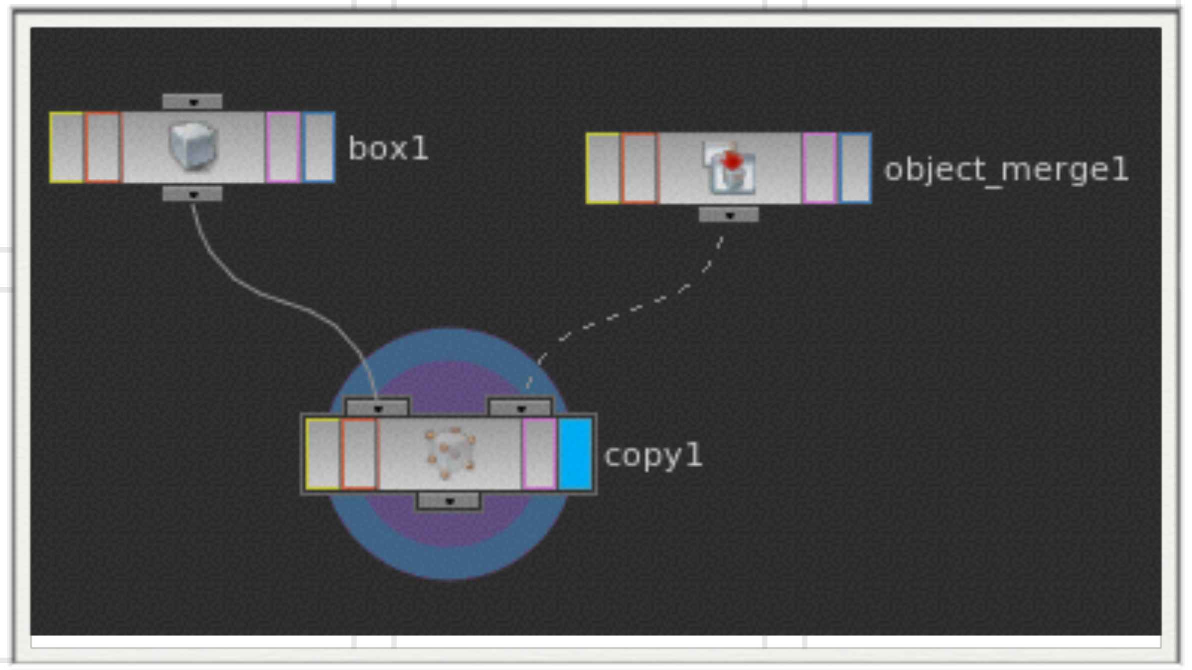
Append a Drag Force to the Wire Solver. I made the scale force = 5

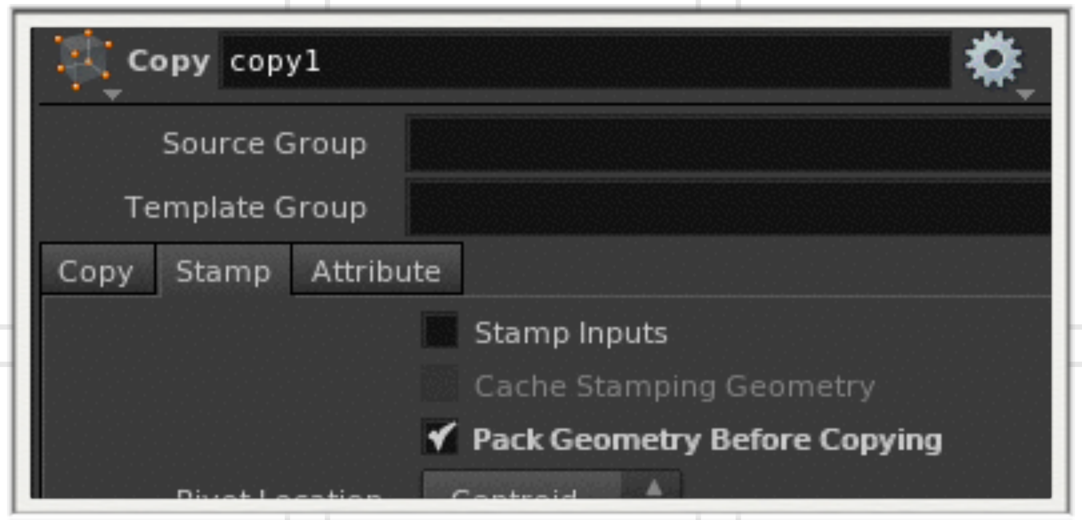
Run the Sim





Adding Beads





Dive into the Wire Object and append a NULL to the dopimport

Name it OUT_TO_BEADS

Go back up to the Object Level and create a Geometry Object

Name it Beads

Dive inside. Delete the File Node. Add a Object merge Node

Point the Object Merge node to “OUT_TO_BEADS” null you just created

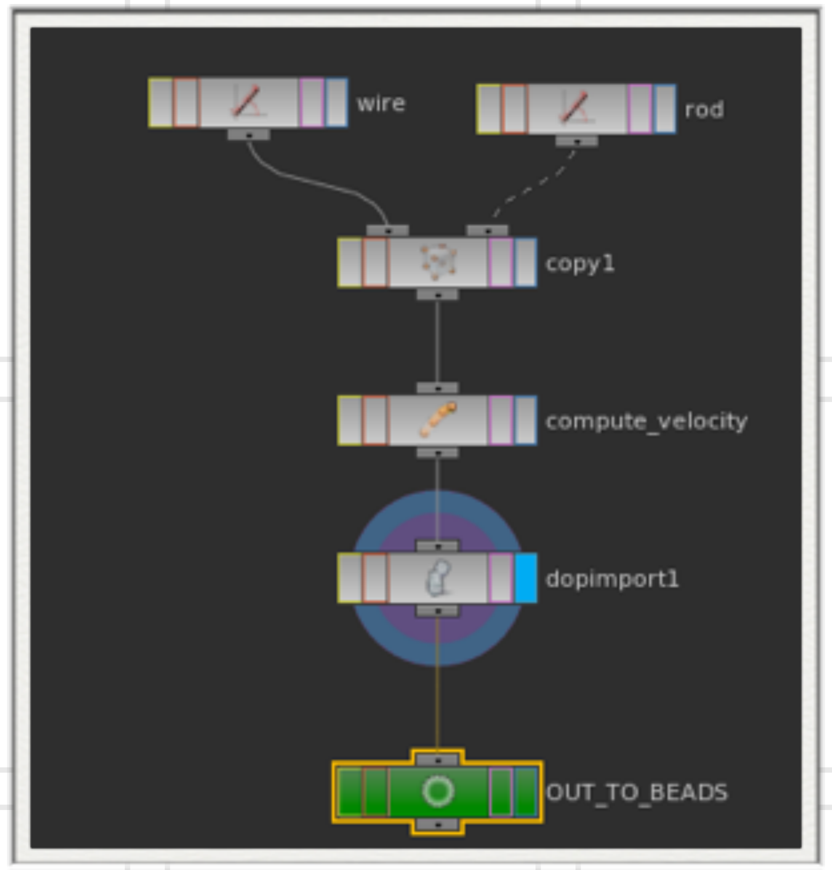
Drop down a box Node

Drop down a copy Node. Copy the Box to the Object Merge

Scale the box to make it looks like beads

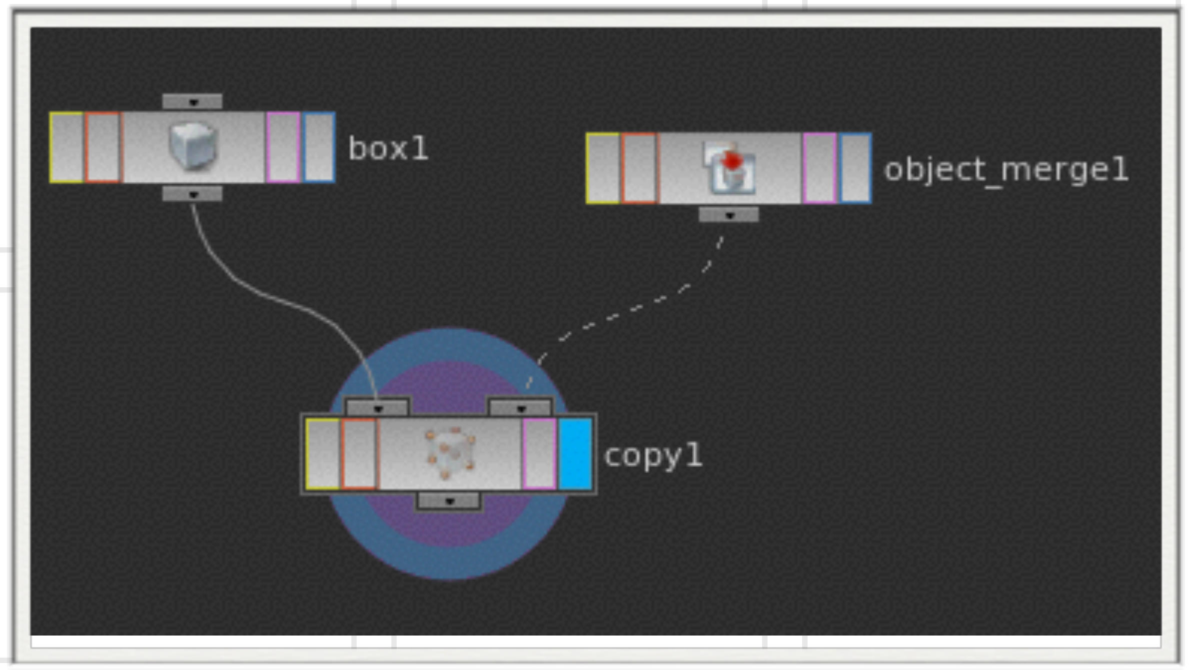
In the COPY SOP go to the Stamp Tab and select “Pack Geometry before copying”

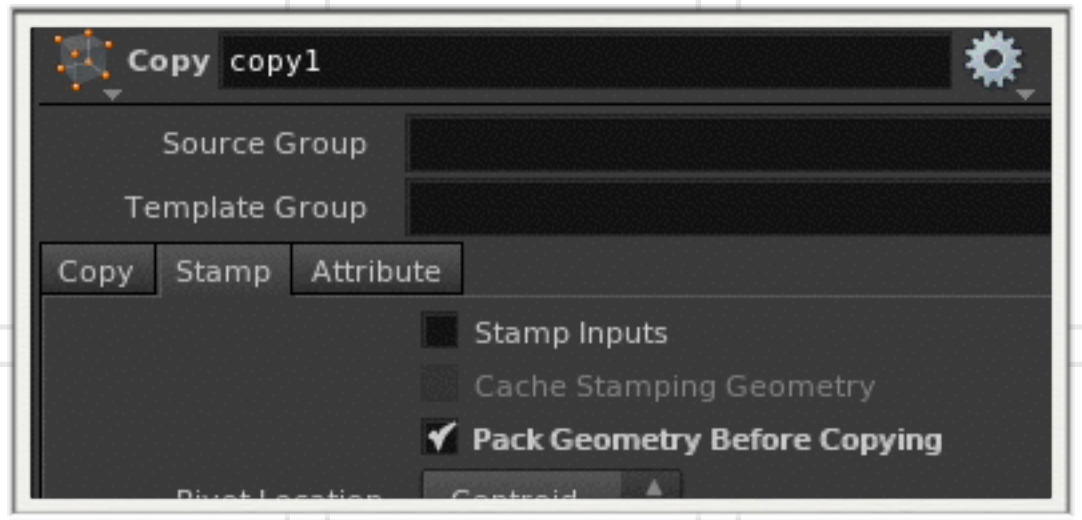
Run the Sim



SIDE EFFECTS SOFTWARE

Finishing Up





I am not happy with the stretchiness of the wire

Dive back into the AutoDOPNetwork

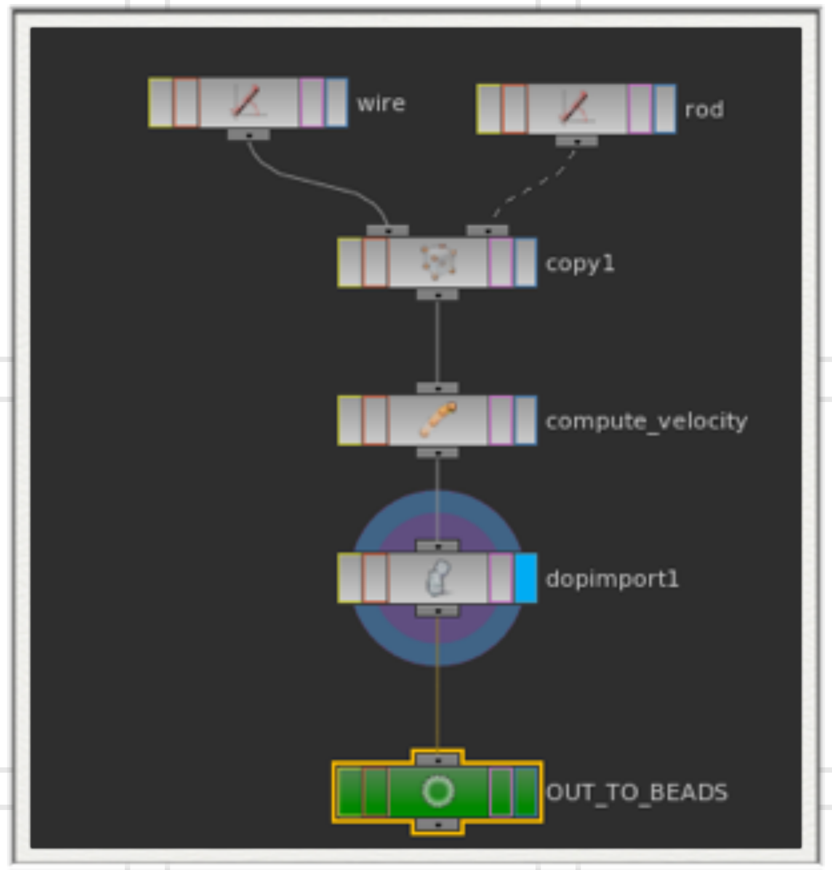
Select the Wire Object

In the Material Tab select the Elasticity Tab

Revert all the parameters to default

Increase the Linear Spring to 1000

Run sim - it looks better

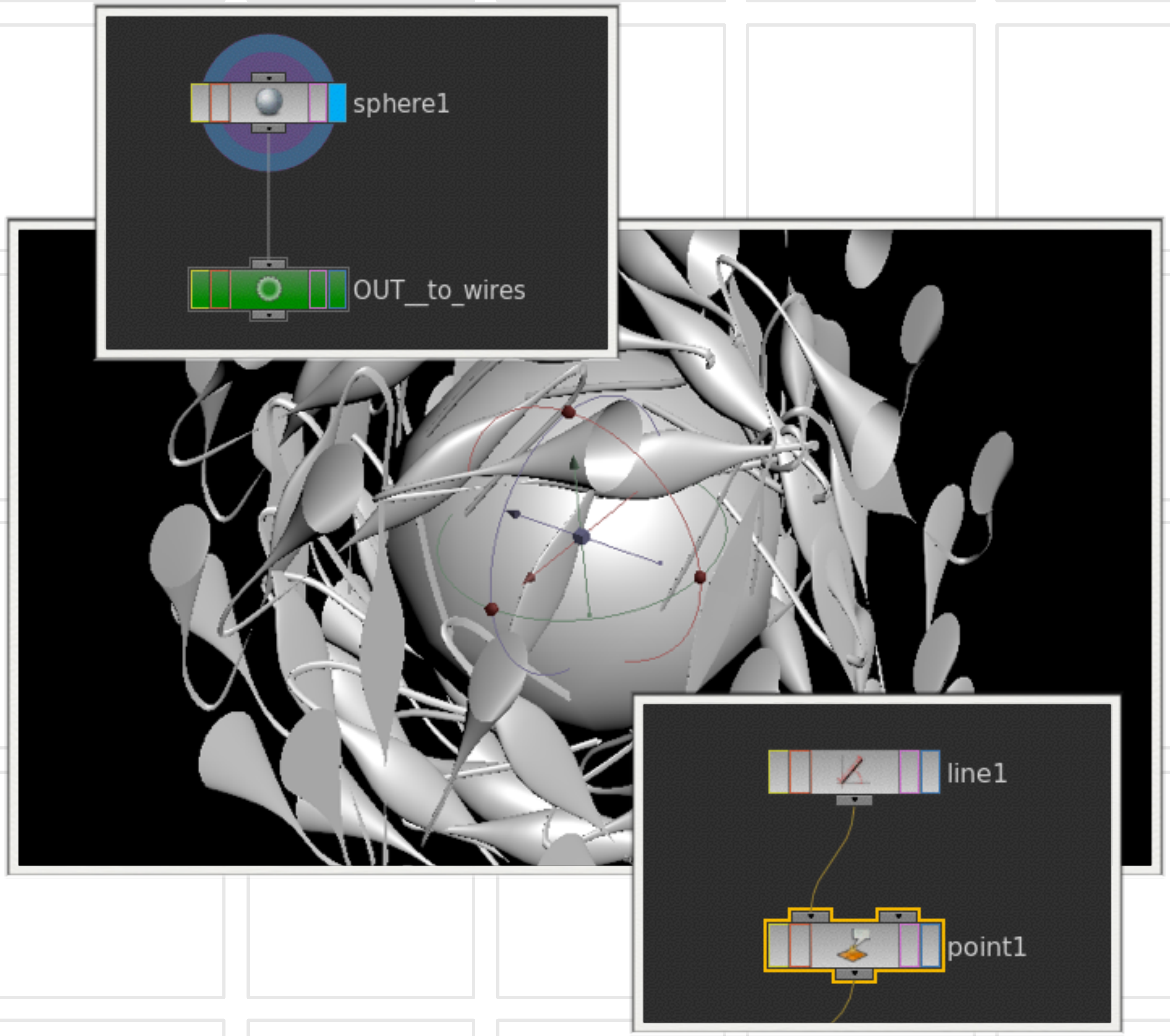


SIDE EFFECTS SOFTWARE

Project 2 - Tentacles

SIDE EFFECTS SOFTWARE

Setup



Drop down a sphere. Make it polygonal

Append a NULL. Name it OUT_TO_WIRES

Go up to the Object Level

Drop down a Geometry Object. Name it “Wires”

Dive Inside. Delete the File SOP

Drop down a Line SOP. Direction (0,0,1)

Append a Point SOP

Translate Y - $TY + sin($BBZ*360+$FF)*1

This will make it snake through time

Setup (cont.)

Drop down a Object Merge

- Point it to the OUT_TO_WIRES Null you just created

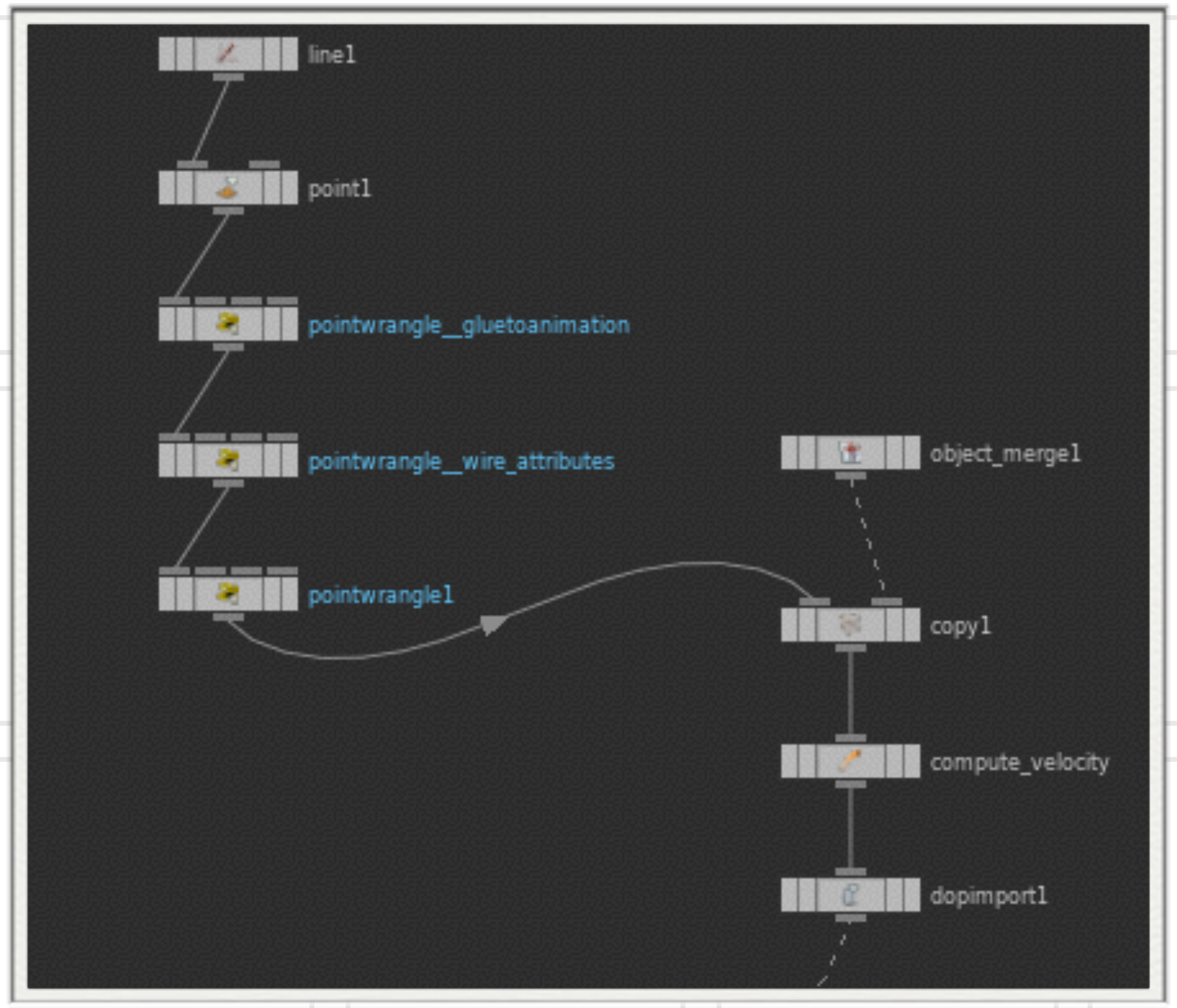
Drop down a COPY SOP

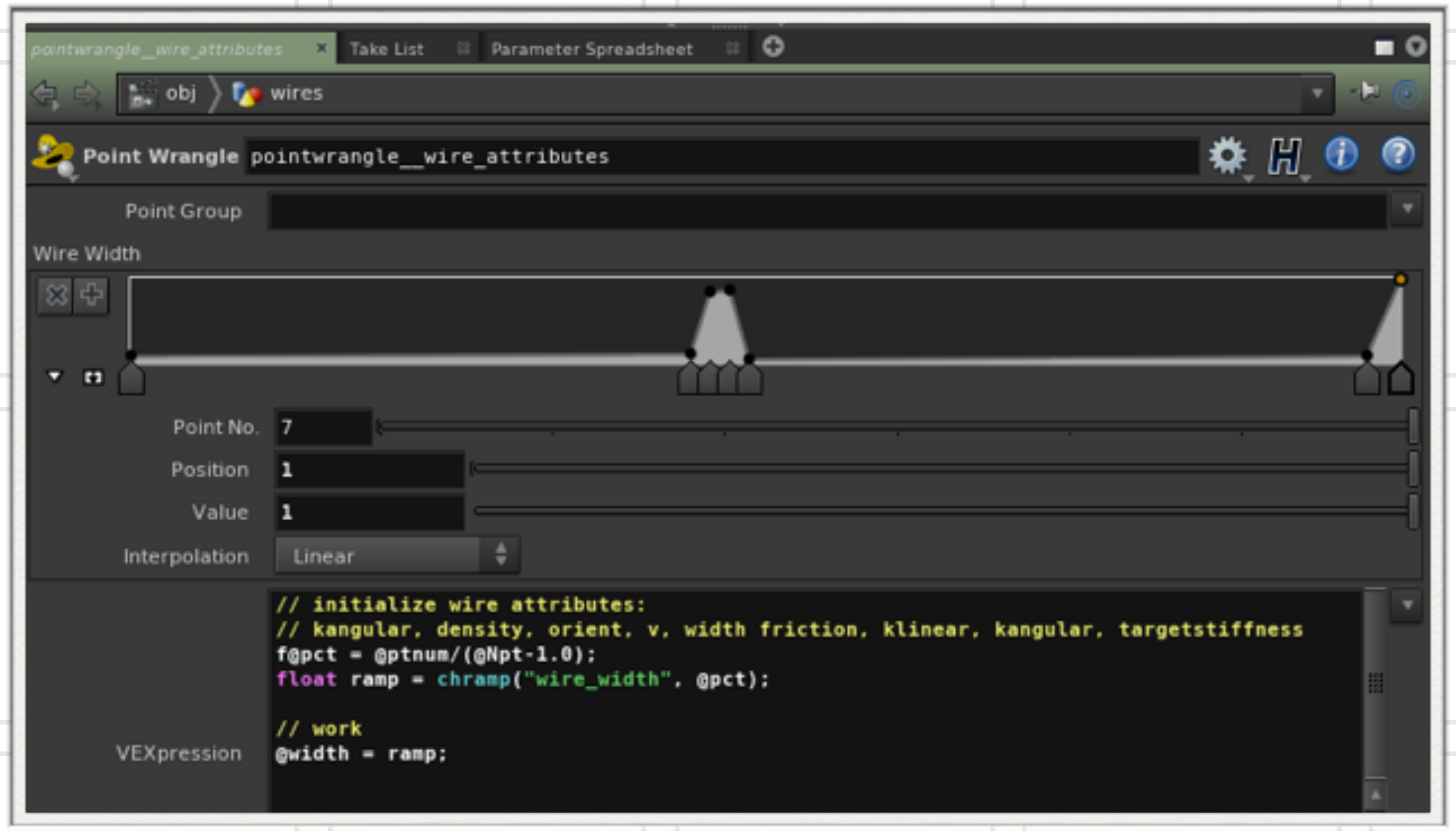
Copy the Point SOP to the Object Merge

Go up to the Object Level and make the “Wires Object” a wire object

Run the sim and watch the wires fall

Creating the Sim





Dive back into the Wires Object and append a PointWrangle to the Point SOP

We will use this to glue the wires to the sphere

```
f@pintoanimation = 1;
```

Append another point wrangle. We will use this to shape the rendering of the wire

Go into the parameters and create a ramp. Name it ramp

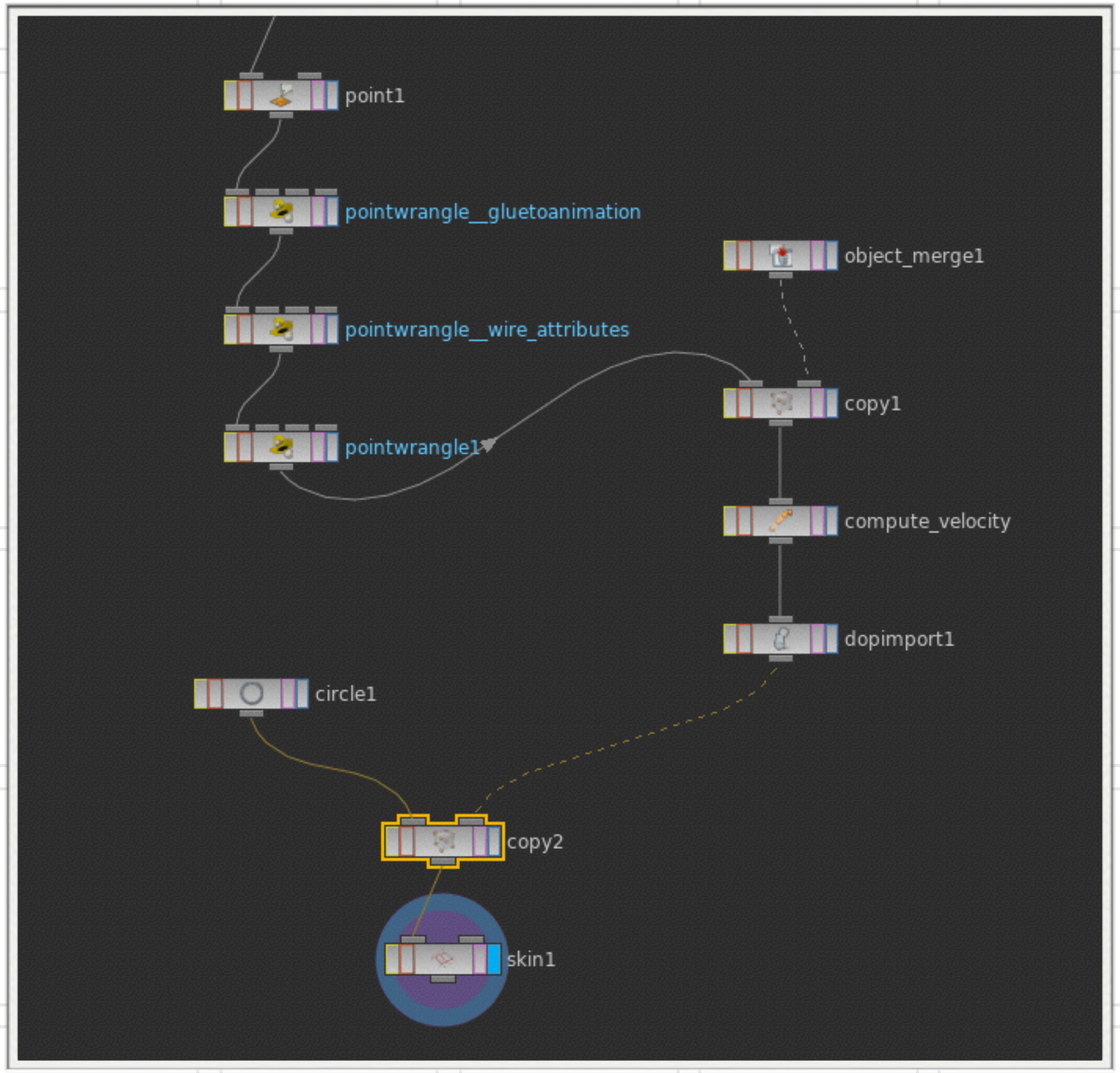
```
// initialize wire attributes:
f@pct = @ptnum/(@Npt-1.0); // percentage of wire length
float ramp = chramp("wire_width", @pct);
@width = ramp; // if rendering wire, shape of wire
```

Append another Point Wrangle

```
@pscale = @width; // scale forces
```

SIDE EFFECTS SOFTWARE

Creating the Shape of the wires



The rest is straight forward

- Drop Down a circle
- Drop down a Copy SOP. Copy the circle to the dopimport

Append a skin

End of MO4

SIDE EFFECTS
SOFTWARE