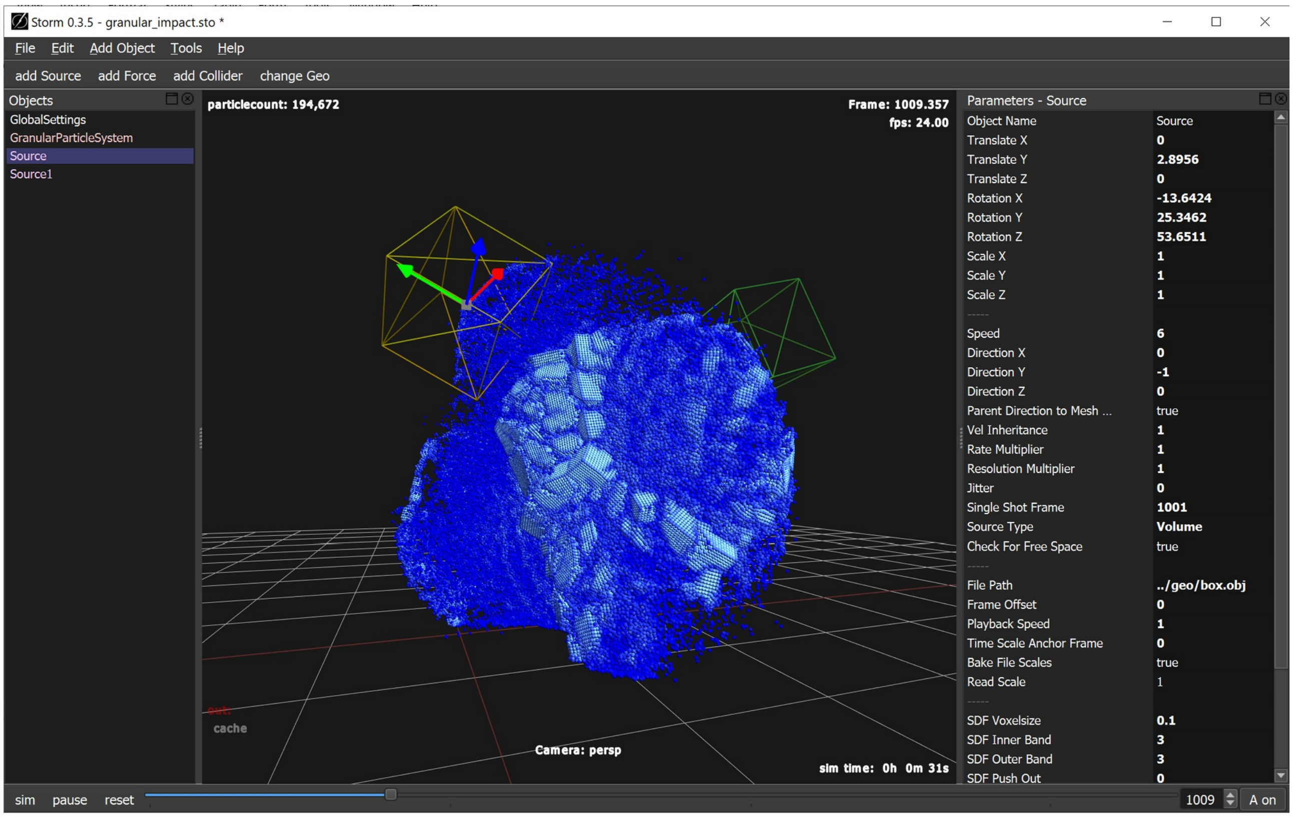
Storm Quick Start Guide

v0.6.0
info@storm-vfx.com



Object Name	Source
Translate X	0
Translate Y	2.8956
Translate Z	0
Rotation X	-13.6424
Rotation Y	25.3462
Rotation Z	53.6511
Scale X	1
Scale Y	1
Scale Z	1

Speed	6
Direction X	0
Direction Y	-1
Direction Z	0
Parent Direction to Mesh ...	true
Vel Inheritance	1
Rate Multiplier	1
Resolution Multiplier	1
Jitter	0
Single Shot Frame	1001
Source Type	Volume
Check For Free Space	true

File Path	../geo/box.obj
Frame Offset	0
Playback Speed	1
Time Scale Anchor Frame	0
Bake File Scales	true
Read Scale	1

SDF Voxelsize	0.1
SDF Inner Band	3
SDF Outer Band	3
SDF Push Out	0

Interacting with the sim while it is running.

Tips and Tricks

- You can **Shift+LMB** a File Path parameter to open up a file browser!
- You can **Shift+LMB** a False/True parameter to toggle it on or off!
- Hitting **“M”** in the viewport switches between global and local transform manipulator.
- Hitting **“C”** in the viewport cycles between the cameras (if there are any) and the regular perspective view.
- Many parameters accept expressions, you can recognize an entered expression by a **green background** of that parameter.
- **Red** parameter backgrounds indicate animated parameters (with keyframe expressions).
- Just regular left mouse button clicking (LMB) on a parameter’s name with one of these expression types switches between showing the **entered expression** and it’s **evaluated result**.

Hotkeys

Camera Controls:
Alt+LMB: Orbit
Alt+MMB: Pan
Alt+RMB: Dolly
F: Frame selection
A: back to origin
C: cycle the active cameras

Transformation Controls:
Q: Selection mode
W: translation tool
E: rotation tool (click on arrow and move mouse left/right)
R: scale tool
M: switch between local and global handle for translation/rotation
+/-: increase/decrease handle size

Display Controls:
X: wireframe mode on/off
Shift+X: Wireframe on shaded on/off
Ctrl+h: hide object (only in viewport)
Shift+h: show object (only in viewport)

Simulation Controls:
Space: Start/Stop sim
Enter: turn selected Source (Emitter) on/off
Shift+Del: reset sim (and restart sim if has been running before)
Del: remove selected object
G: show/hide grid
Shift+G: turn ground collision on/off

Keyframes:
Keyframes are controlled by expressions in the form of:
[frame]:[value] [frame]:[value] ...[frame]:[value]

A quick way to set a keyframe on any parameter is to:
ALT+LMB on the attribute name in the attribute editor: set keyframe

A quick way to remove a keyframe again is to:
CTRL+LMB on the attribute name in the attribute editor: remove keyframe
You can also just remove the whole keyframe expression to remove all keyframes.

To switch between expressions and a resulting values just click on it:
LMB on any attribute name: switch between expression and value

For translate, rotate and scale there are some extra keys:

Shift+W: create keyframes for translation
Shift+E: create keyframes for rotation
Shift+R: create keyframes for scale
Ctrl+W: remove keyframes for translation
Ctrl+E: remove keyframes for rotation
Ctrl+R: remove keyframes for scale

Jump between keyframes:
Ctrl+Left: go to previous keyframe
Ctrl+Right: go to next keyframe

Note that quaternion interpolation of rotation needs each keyframe to have all three axes keyed synchronized (are on the same frame).

Timeline:
Left/Right: go to previous/next frame

Storm being used for snow in the last episode of Game of Thrones (2018).

Basic workflow

First create a system (*Add Object… → Granular* or *SPH*).
Then click on *“add Source”* to create a simple plane emitter. You can press **W** and move the Source up a bit.
Now add a collider by clicking on “*add Collider*” in the toolbar. A standard collider with a box mesh is created.

To change the mesh click on “*Change geo*” to bring up a file browser. You can also directly edit the “*File path*” parameter on the collider.
Same goes for Sources, you can select the plane source and click “*Change geo*” to select a mesh to use as an area of emission.
When there are multiple systems in the scene first select the system and then add the Sources, Forces or Colliders. This way Storm knows which objects should work together.

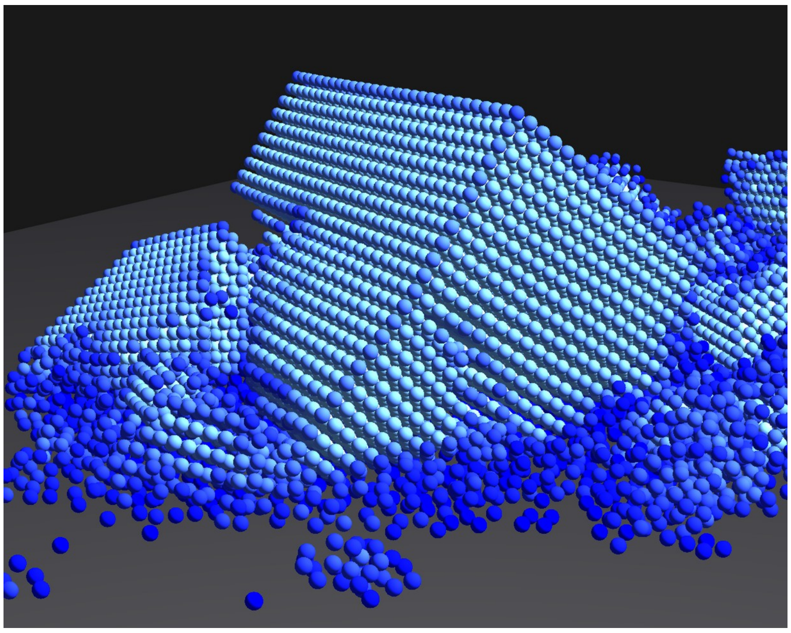
You can reposition the objects with the transform handles or by editing the transformation parameters on the right at the top of the list.
Scaling is now possible and works cleanly without issues anymore.
If the object has a different scene scale or you just need to uniformly scale the mesh, you can also use the “*Read Scale*” parameter (close to “*File path*”) to scale the object early at reading time (to bake-in the scaling). This way voxels remain in world scale thus making it easier to work with them and their voxel sizes.

Please note that having too many simulation threads can slow it down. Try having something between 4 to 10 threads as a start.

Storm being used to kick up ground gravel in X-Men: Dark Phoenix (2018).

Simulation notes

Following are some notes and tips on the granular particle system:



Particle stiffness acts as a particle repulsion force. Lowering this creates a more jelly-like matter. It's usually better to not go too low with it though, as at some point particles just pass-through each other.

Constraint stiffness acts as a force to keep the jointed structures intact.
It can be low on the other hand, which will make the particles crumble and detach more quickly.

In general higher stiffness values require more substeps (on the GlobalSettings object) to not make the simulation get unstable and explode.

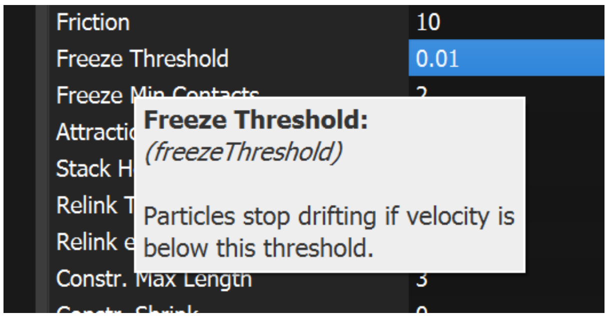
Constr. Max Length controls how long the constraints will be between particles at creation time. It's a multiplier of the current particle radius. 3.0 is default and higher values make the material more rigid and also create larger clumps when breaking. The simulation slows down with higher values though.
It's best to first try changing the *Constr. Max Stretch* parameter mentioned next to keep the sim speedy.

Constr. Max Stretch can also control how large chunks get or when the material breaks. It doesn't slow down the simulation or affect it's speed, so it's a good alternative to the above mentioned *Constr. Max Length* and should get tested first before altering *Constr. Max Length* to keep the sim speedy.
Interesting effects can get created when Constraint Max Stretch is high (e.g. 2.0) and the Constraint stiffness is low to get very wobbly results which don't directly break apart.

Substeps defines how many single steps one frame is divided in. For higher resolutions or fast moving particles usually more substeps are needed else there will be explosions and particle simulation artifacts happening.

Iterations defines how many iterations are done on each of these steps to resolve particle-particle collisions and particle constraints.
Imported geometry can be used for emission areas or collision geometry. Please note that imported geometry has to have a volume to be able to be used. Flat planes won't have an effect yet.

To get rid of unnatural sliding in some cases you can use the "Freezing" methods -
Find the "Freeze Threshold" parameter on the granular particle system:



Friction	10
Freeze Threshold	0.01
Freeze Min Contacts	2
Attracti	
Stack H	
Relink T	
Relink e	
Constr. Max Length	3
Constr. Shrink	0

Freeze Threshold:
(freezeThreshold)

Particles stop drifting if velocity is below this threshold.

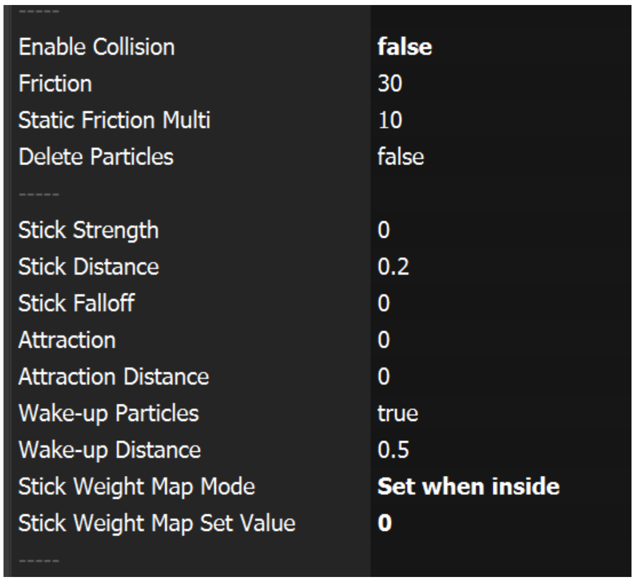
The threshold stands for the speed of particles that when they are below they will stop moving around. Collisions with other particles are still getting processed though (this is opposite to particle "Sleeping" where they fully stop and can only get woken up by a collider object in "Wake-up" mode).

Freezing can help particles to stack like sand.

The default frame range starts on frame 0 and goes to frame 200. The range can get adjusted on the *GlobalSettings* node (accessible by using the *selection* button in the upper toolbar).
There's also a *config.ini* in your Storm install's bin folder to adjust these defaults.

Entering values in the Attribute Editor

When selecting an object in the scene or the selection box there will be parameters shown at the right side of the interface by default.



Enable Collision	**false**
Friction	30
Static Friction Multi	10
Delete Particles	false

Stick Strength	0
Stick Distance	0.2
Stick Falloff	0
Attraction	0
Attraction Distance	0
Wake-up Particles	true
Wake-up Distance	0.5
Stick Weight Map Mode	**Set when inside**
Stick Weight Map Set Value	**0**

There are:

- **numeric** parameters (e.g. *Gravity: -9.81*)

- **boolean** on-off switches (e.g. “Enable Collision"*: true* or *false*)

- **text** parameters (e.g. names and file paths)

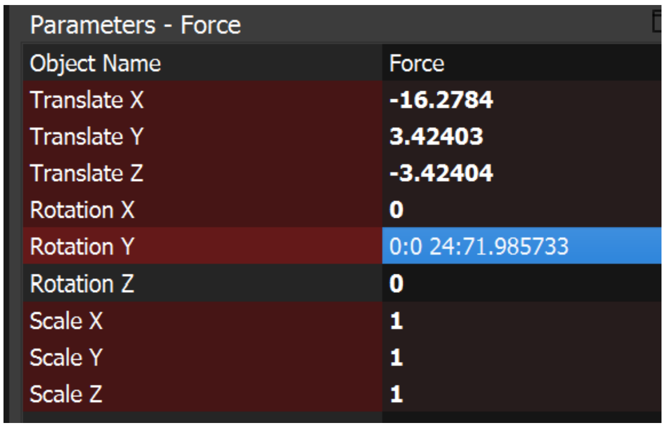
- and **multiple choice** option fields (e.g. *Source Type: Plane or Volume)*

To switch on-off parameters you can enter *0* for *false* and *1* for *true*.
To switch a multiple choice field you currently have to enter the number of the option
(this shouldn't be needed too often currently and future releases will have dropdown menus for this).

Some attributes already have tooltips and show them by hovering over the attribute name.

Keyframing expressions

Keyframing expressions have the form of frame and value pairs, e.g.
[frame]:[value] [frame]:[value] ...[frame]:[value]
with each pair separated by a whitespace.



Parameters - Force

Object Name	Force
Translate X	-16.2784
Translate Y	3.42403
Translate Z	-3.42404
Rotation X	0
Rotation Y	0:0 24:71.985733
Rotation Z	0
Scale X	1
Scale Y	1
Scale Z	1

An expression like *1001:0.0 1010:5.0 1030:1.0* means that value on frame 1001 is zero, on frame 1010 it is five and on frame 1030 it's down to one.

You can type in these expressions or use following shortcuts:
You can use the shortcut **Shift+W** to set a keyframe on each axis of the translation.
Ctrl+W removes these keyframes again.

The same goes with rotation [E] and scale [R].
Just use **Shift+E** to key all rotation axes or use **Shift+R** to key all scaling axes for the current frame.
To the keys on the current frame use **Ctrl+E** and **Ctrl+R** accordingly.
Use **Ctrl+Left** and **Ctrl+Right** to jump between keyframes.

General expressions
This version of Storm supports basic general parameter expressions.
You can use them for example to animate translations, rotate objects by time or frame or switch on/off by simple conditions.

Some of the most important expression functions:

sin(value)
Returns the sinus of value.
E.g. sin(framef)

cos(value)
Returns the cosinus of value.
E.g. cos(framef)

rand(value)
Returns a random number based on value.
E.g. rand(time)

noise(value)
Returns a noise value taking value as input.
*E.g. noise(framef * 0.1)*

remap(value, oldMin, oldMax, newMin, newMax)
Remaps a value between min and max values to new min and max values.

remaps(value, oldMin, oldMax, newMin, newMax)
Smoothly remaps value to the new min and max values.

Note that these expressions can get combined.

Examples:
Put the following for example into the *Translate X* parameter to linearly animate an object along x with 0.1 meters per frame (when the starting frame is 1001):
*4.1 + (framef – 1001) * 0.1*

The following example uses the time variable and the sin function to get values changing with a sinus wave:
*sin(time * 10) * 0.5*

The following expression is true when the current frame is *1010* and for example can get used on the "E*nabled"* parameter of a Source to set a single frame of emission:
frame == 1010

Similarly the following would emit on all the frames before 1020:
frame < 1020

built in environment variables:
$SCENENAME
$SCENEFOLDER
$STORMFOLDER
$F
$FF

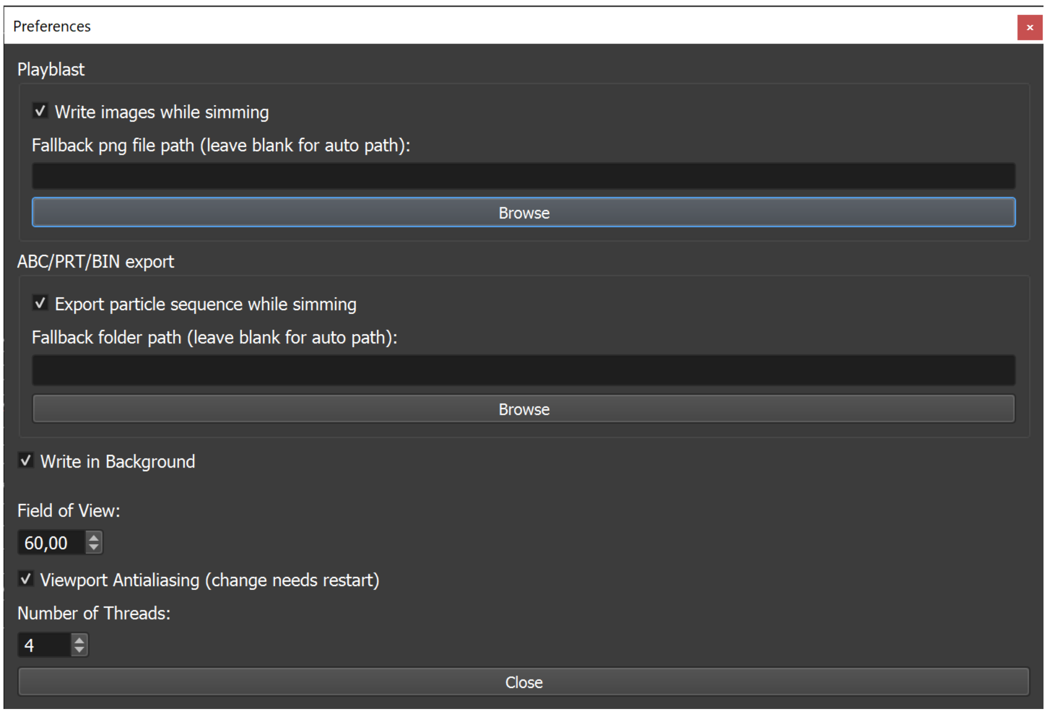
The upper environment variables can get used in any parameter, numbers, switches and file paths.

Built in numeric variables:
framef*:* floating point version of the current frame, which updates also on subframes.
frame*:* integer version of the current frame. It doesn’t change in the frame’s substeps.
time: current time in the simulation in seconds.

These numeric variables can only get used in numeric and boolean parameters, but not in text inputs.

Playblast and Cache Export

You can export caches and playblast while the sim is running. Go the the preferences and switch on the check marks:



Preferences

Playblast

✓ Write images while simming

Fallback png file path (leave blank for auto path):

Browse

ABC/PRT/BIN export

✓ Export particle sequence while simming

Fallback folder path (leave blank for auto path):

Browse

✓ Write in Background

Field of View:

60,00

✓ Viewport Antialiasing (change needs restart)

Number of Threads:

4

Close

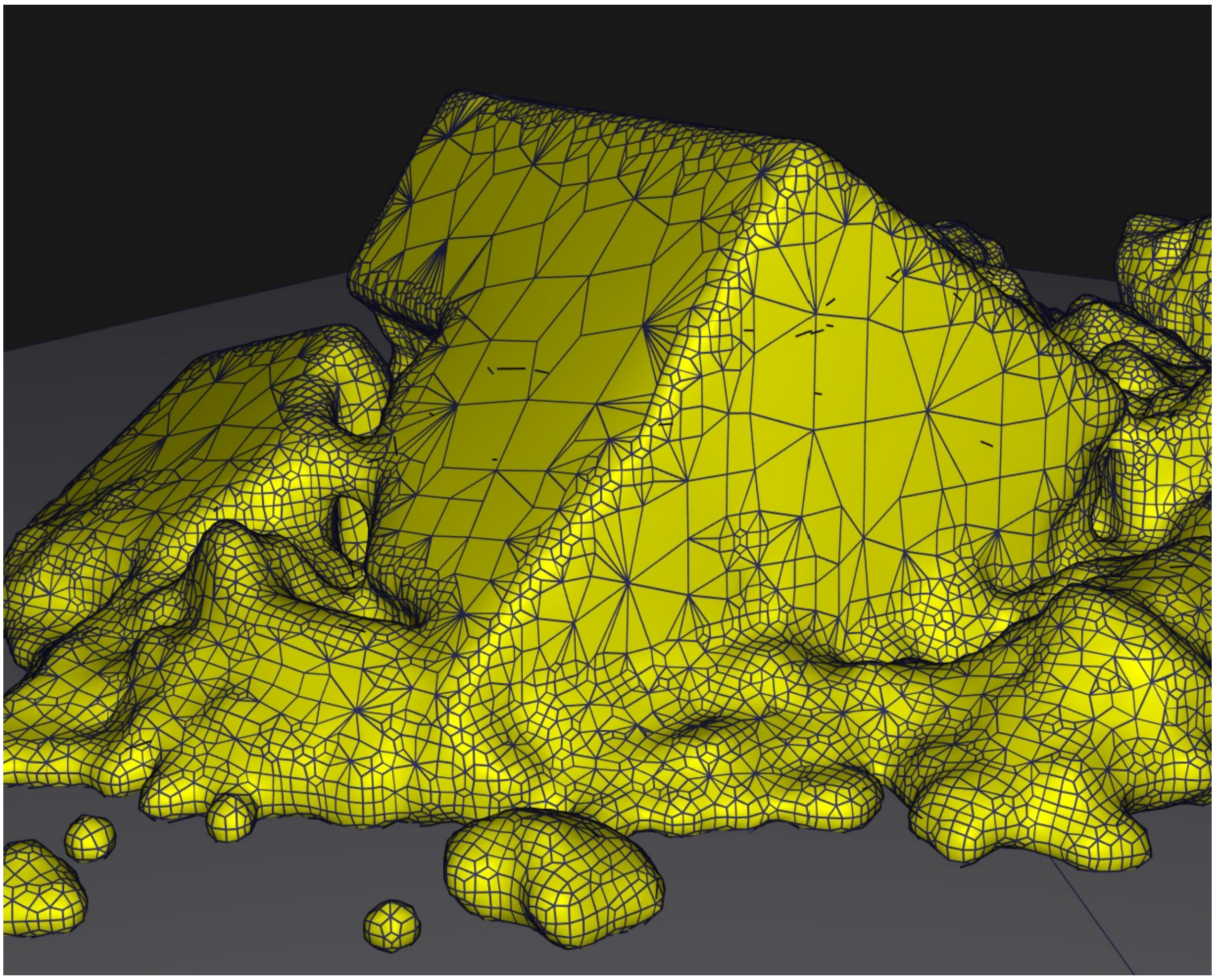
To export a playblast png sequence or particle caches the easiest way is to first save the scene and then enabling "*Write images*" and/or "*Export particle sequence*" in the preferences.
Now when simulating the export happens every frame. The data is written into subfolders next to the saved scene file.
The default particle cache export format is *prt*. To change this find the "*Export Format*" parameter in the lower section of the particle system parameters. Enter here *bin* for the uncompressed bin format or *abc* for alembic cache export.

In the demo version there's currently no option to save the scene file. You can still create a playblast sequence though by manually specifying the playblast location through the "*Fallback Path*" field in the preferences by entering a path for example like *C:\img\sim1.%04d.png*

Alembic and PRT export formats are currently the best supported ones.

Mesher Usage

This version of Storm comes with a built-in particle mesher. Point the mesher to an already existing particle cache on disk and run the frames from start to end to write out your combined single alembic mesh file or a sequence of mesh files. The mesher can load abc or prt particle files and it outputs alembic or obj files for your meshes.



To use it go to “*Tools*” and select “*Mesher*”.
Shift+Click on “*Input File Sequence Path”* to open a file browser. Choose a file of the particle cache you want to mesh and replace the frame number with sharps like in this example: “*particles.####.abc*” - This will get marked green and indicates an expression. You can left click the parameter name to toggle between evaluated display and input display showing the sharps).
Now you can also enter the name of a particle system in the “*Input File Sequence Path”* to directly reference an active particle system here.

Move the time slider to update the mesh and to show it in the viewport. Adjust the “*Voxel Size*” to capture the detail you need but try to also keep it as high as possible to have the meshing run fast and to keep the polygon count manageable.
Lower “*Blending*” to speed up the meshing a bit. Blending merges together individual particles.
Raise “*Final Smoothing*” to smooth out remaining bumps in the mesh.

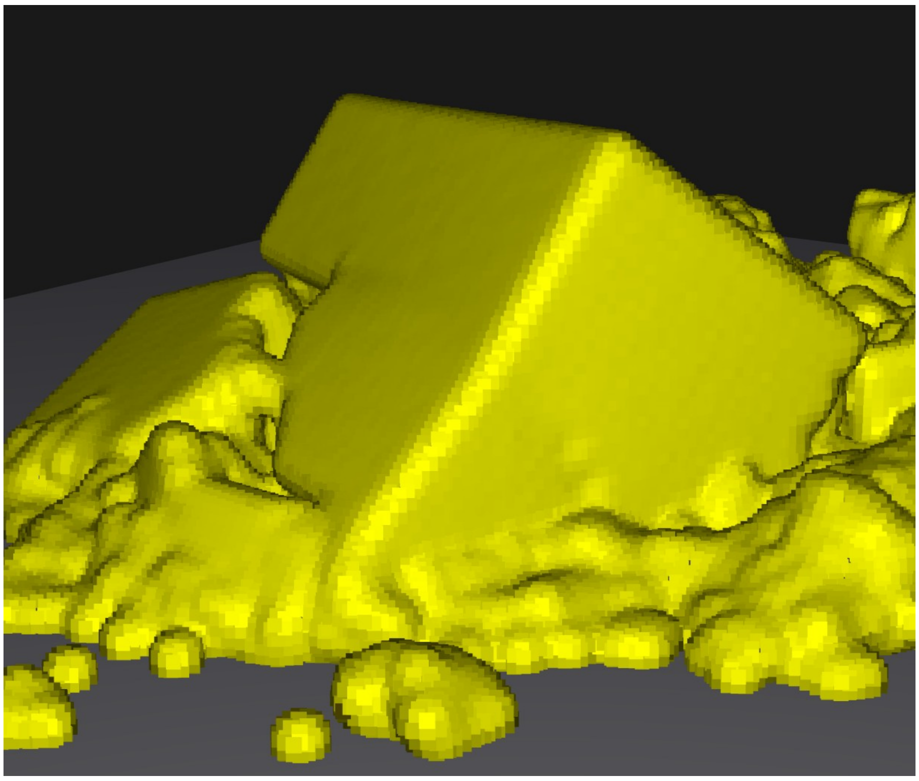
When you are happy with the result you can write out the full sequence animated mesh, either in a single alembic file containing the whole animation or into a file sequence.
To do that first adjust the output frame range on the "*Global Settings*" object and reset the sim by hitting the reset button in the lower left corner. Now enter an output path for "*Output Alembic File Path*", either by manually entering a path or by Shift+clicking on the parameter name to get a file browser.
To output a file sequence use hashtags as seen with loading in the particle caches above. For example "mesh.####.abc" or "mesh.####.obj" for sequences. For a single alembic file containing the whole animation enter a file without sharps, for example just "mesh.abc".
To toggle the parameter from evaluated display to input display and back left click the parameter name.

When you're ready hit the *sim button* in the lower left corner or use the spacebar.
For a combined, single alembic file please note that the sim has to run from the first frame to the last frame of the range, then the file will be properly created and finalized. The sim also has to start on the first frame, so to be safe it's easiest to always also once hit the *reset button* just before starting the writing process.

The mesher now automatically updates when one of it's values gets changed.

You can press "*v*" to show the mesh's velocity trails.

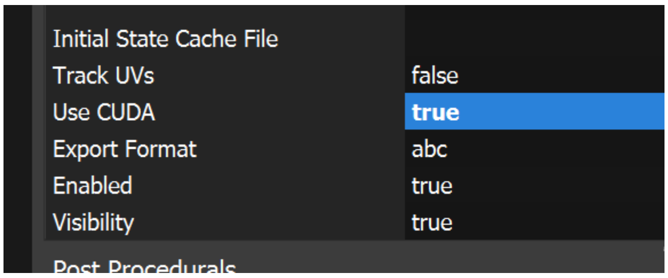
It's possible now to export VDB volumes from the mesher.
Enable "Export VDB" and enter an output path to a vbd file sequence, e.g.:
D:\Storm\output\surface.####.vdb



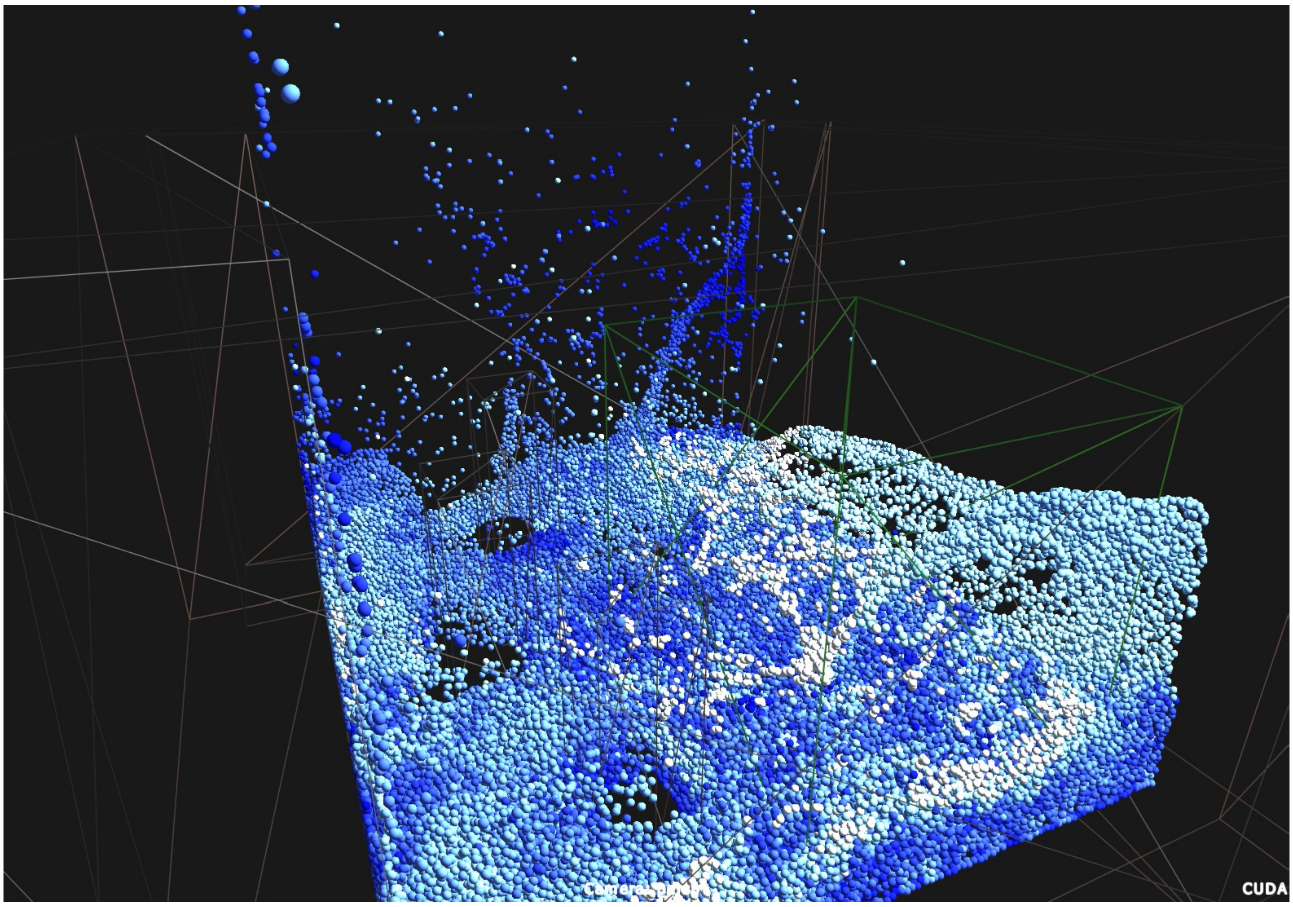
Direct VDB output.

CUDA support

This version of Storm brings CUDA support to accelerate your sims (Windows only).
Find the parameter “Use CUDA” on the granular and the sph system.
It’s experimental and not all parameters are supported yet.
The granular system usually might need a couple of iterations more, compared to the CPU version.



Initial State Cache File	
Track UVs	false
Use CUDA	true
Export Format	abc
Enabled	true
Visibility	true
Post Procedurals	



The CUDA SPH system demo scene “sph_damBreak_cuda.sto” being run.

STØRM

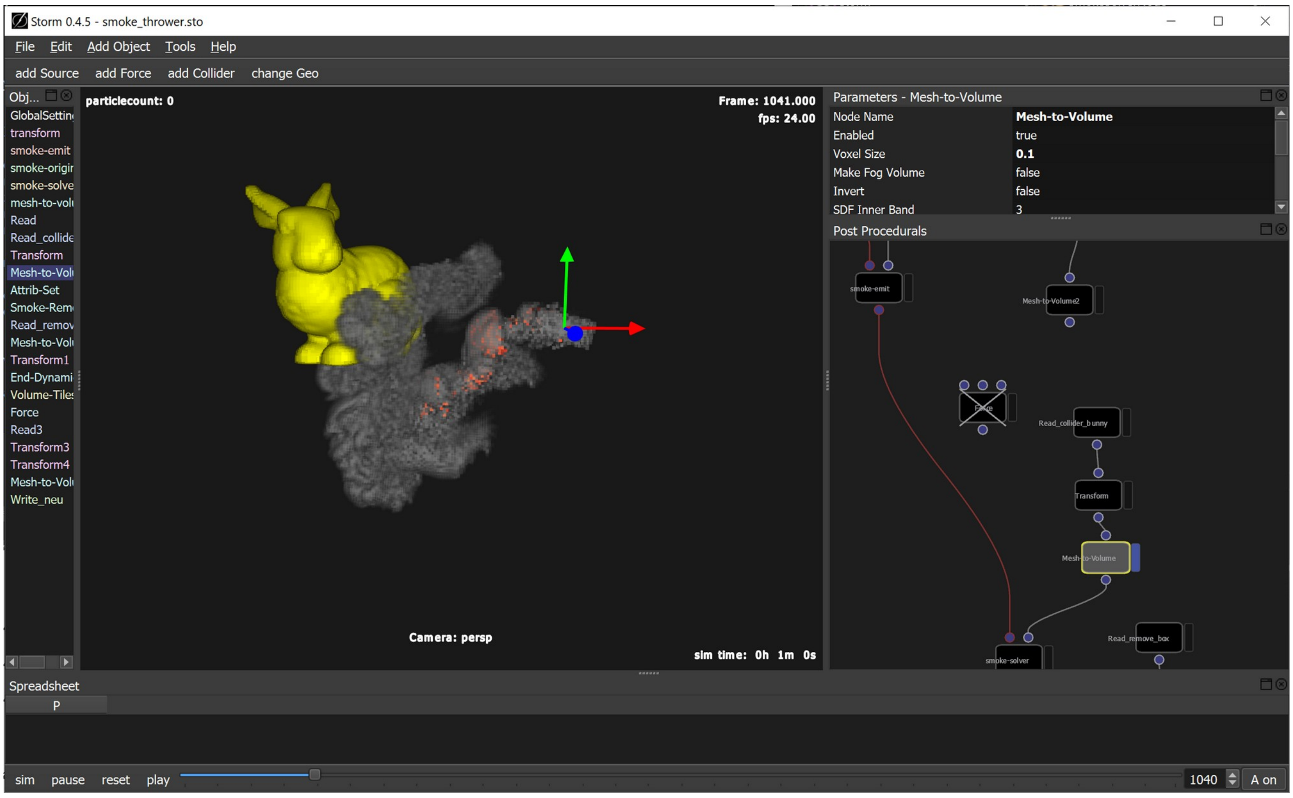
THE SIMULATION TOOL

Storm Nodes Overview

v0.6.0

info@storm-vfx.com

Storm just got an experimental node interface!



Node Name	Mesh-to-Volume
Enabled	true
Voxel Size	0.1
Make Fog Volume	false
Invert	false
SDF Inner Band	3

Tips:

- Press **“R”** to get a **Read** node
- Press **“T”** to get a **Transform** Node
- Press **“M”** to get a **Mesh-To-Volume** Node
- Doing so will automatically connect the last selected node with the new one.
- Hit **“Tab”** or use the right mouse button to get a menu for available nodes.
- The most used ones are **“Read”**, **“Transform”, “Mesh-To-Volume”** for mesh operations.
- For smoke simulations you would need **“Smoke-Origin”** (the start of each simulation step), **“Smoke-Emit”** to emit density and/or velocity into the sim (with a connected volume) and the **“Smoke-Solver”** node.
- You can shake nodes to disconnect them
- You can select multiple nodes and hit “S” to wire them up together

- Hit **“1”** on the keyboard to toggle the selected nodes visibility on or off

- On default the graph is only evaluated **once per frame** – to change this find the parameter **“Run Graph on Subframes”** on the GlobalSettings object. When turning this on you can tweak the amount of substeps needed as usual.

- When visualizing smoke in the viewport performance may drop. Find the parameter **“Volume Downsample”** on the GlobalSettings object to speed up the viewport rendering.

- Running a simulation without UI is the **fastest option**. An easy way to do this is going to **“Tools”** and selecting **“Launch Background Sim”** (find all commandline parameters by launching Storm like this *“Storm –help”).*

This version of Storm has an additional node interface called **“Post Procedurals”**.
It currently is a new set of nodes and a fully separate from Storm’s previous interface and features.
It allows you to read back caches, check about channels and edit or delete particles. You can convert particles to volumes or meshes and see this in a simple procedural flow.
It also comes with a mesh deformer and a granular system upressing method, as well as an experimental new smoke solver.
Storm’s classic interface of adding a granular system, and sph system isn’t changed and not replaced by the node interface; they both live under the same hood.

In this guide are three tutorials, about how to deform poly geometry, upres a grain sim and smoke & fire.

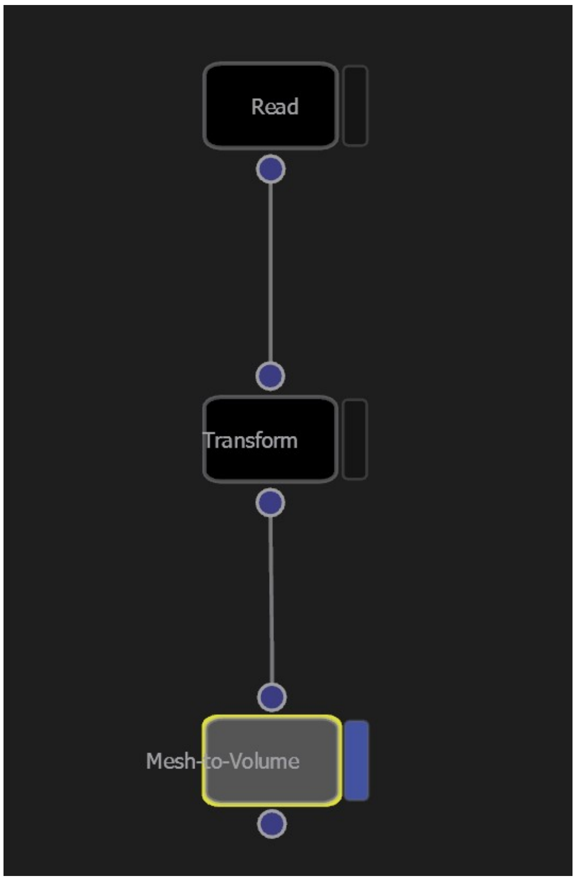
Let’s start with the smoke!

1. Basic Smoke & Fire Tutorial

With the mouse cursor over the node graph (aka “Post Procedurals”) press the **“R”** key to quickly create a **“Read”** node. Alternatively you can also hit “**Tab”** or the right mouse button and search for the read node in the context menu.
You will see a box appear in your viewport. On default the read node will load an obj file containing a box mesh.

Keep the node selected (indicated by the yellow outline) and hit **“T”** to create and connect a **“Transform”** node. The new transform node will be the only visible node (indicated by the fully blue button on the right of the node).

Now hit **“M”** to create a **mesh-to-volume** node which will convert the box mesh into a volume – needed for the next steps.



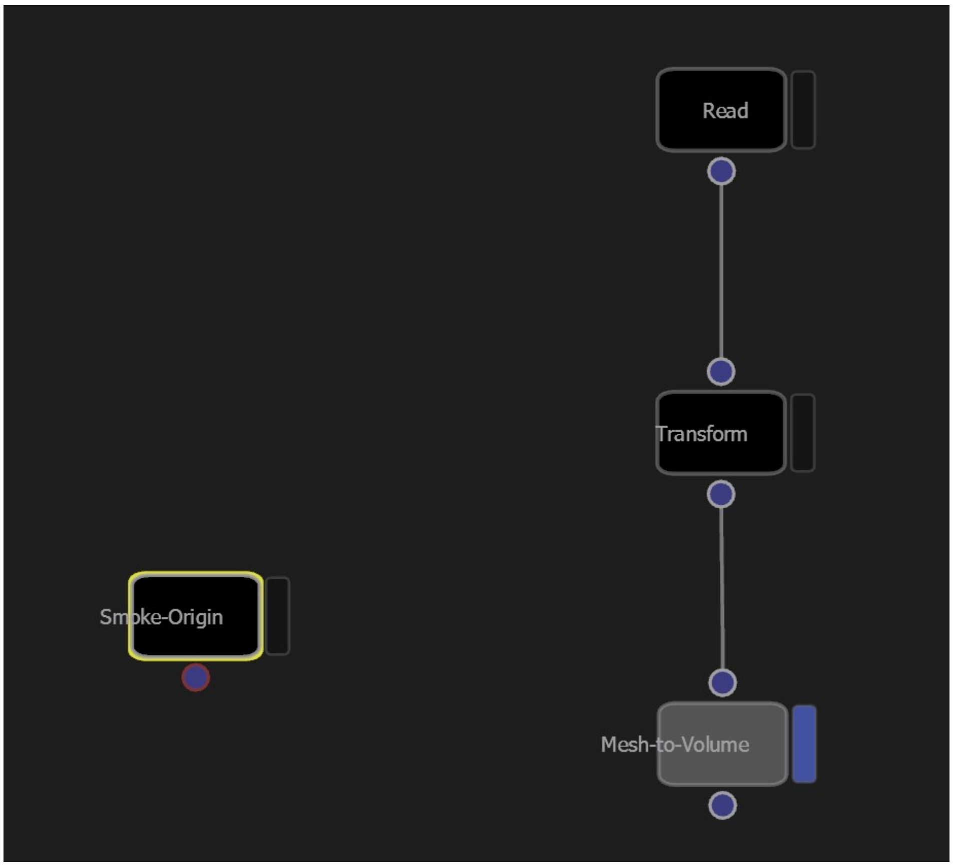
Creating an emission volume.

This will serve as an emission volume of smoke.

Following is the creation of the remaining three nodes for the smoke simulation, an **origin**, **emission** and **solver node**...

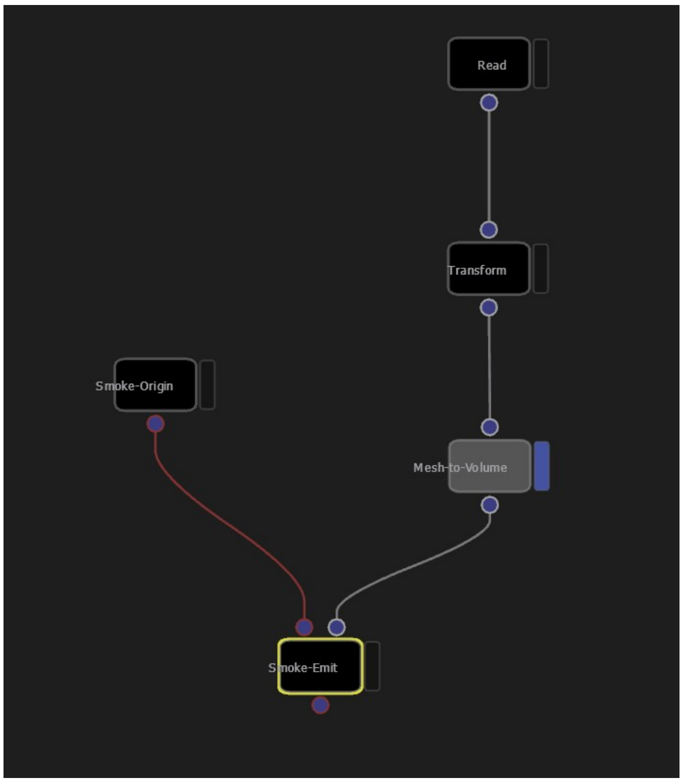
Hit **"Tab"** and enter **"Smoke-Origin"** and place it into the scene, a little to the left of your currently existing node chain.

This will be the starting point for each simulation step of your smoke sim. The advection will happen here.

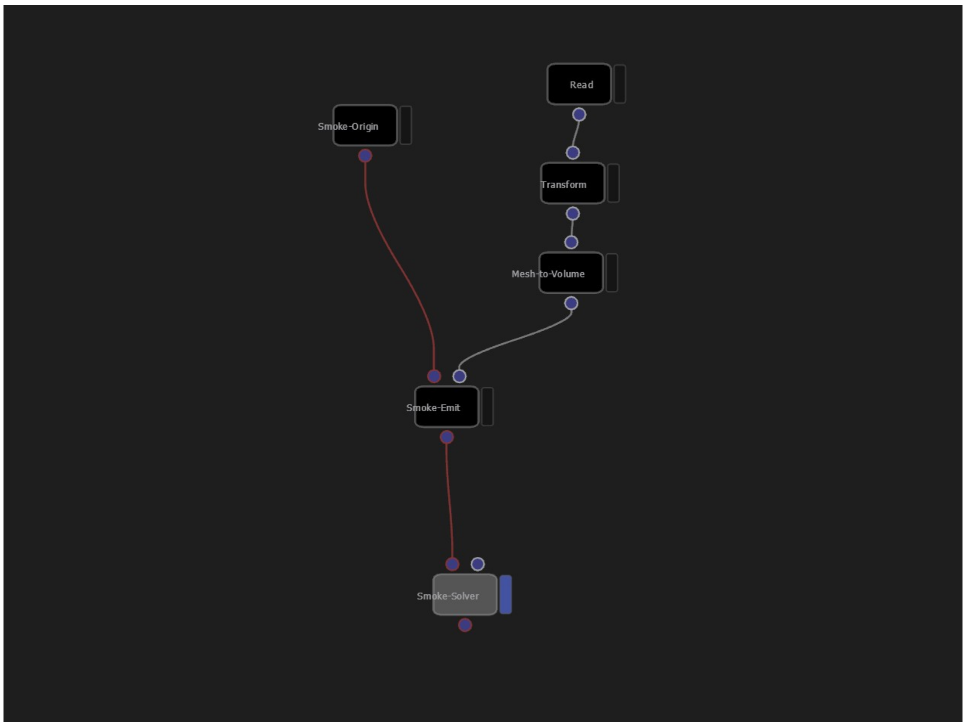


Now hit **“Tab”** once more and enter **“Smoke-Emit”** and place it below the “Smoke-Origin” node and make sure they are connected together. You can also connect them by either dragging down an output connector to an input connector or by keeping the first node selected before creating the second and Storm will wire them up automatically.

Click and drag the right upper input plug of the **“Smoke-Emit”** node onto the “Mesh-To-Volume” on the right node to connect them together too and define your emission area:

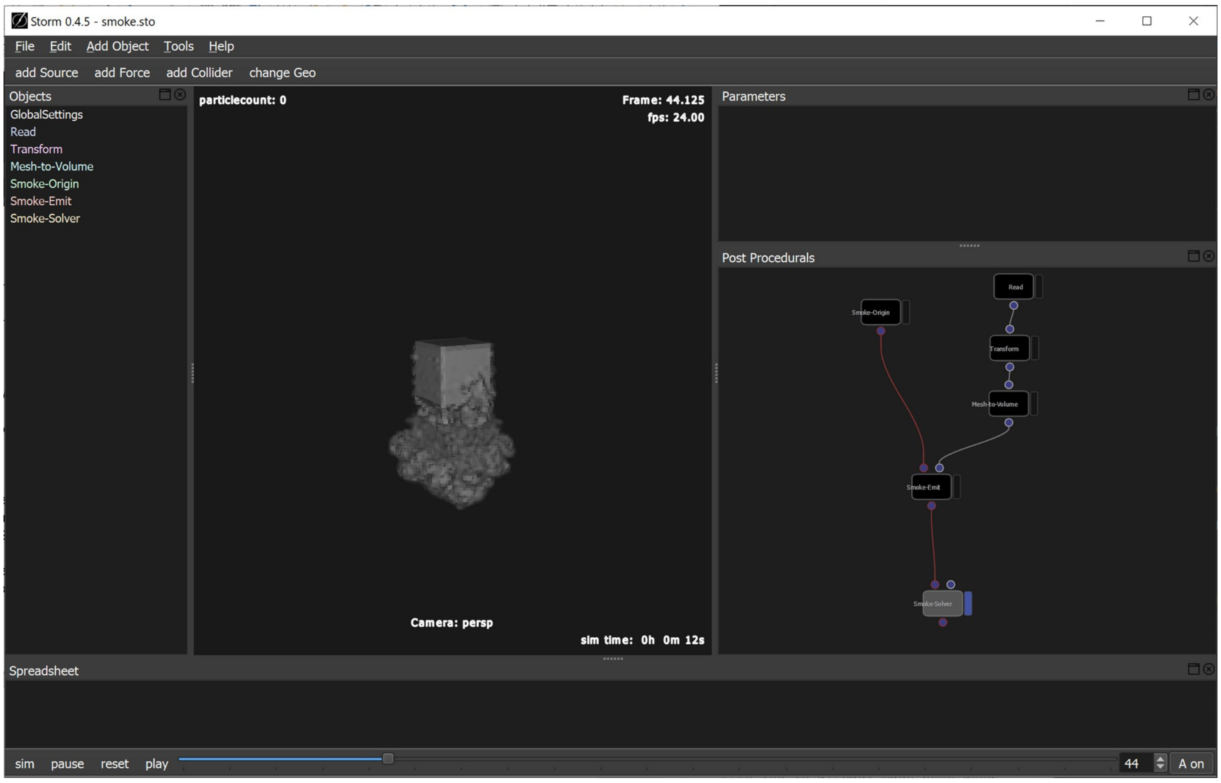


Lastly make sure the “Smoke-Emit” node is selected and hit the “Tab” key - look for the “**Smoke-Solver”** node and place it in the graph. Doing this will lead to the following arrangement:



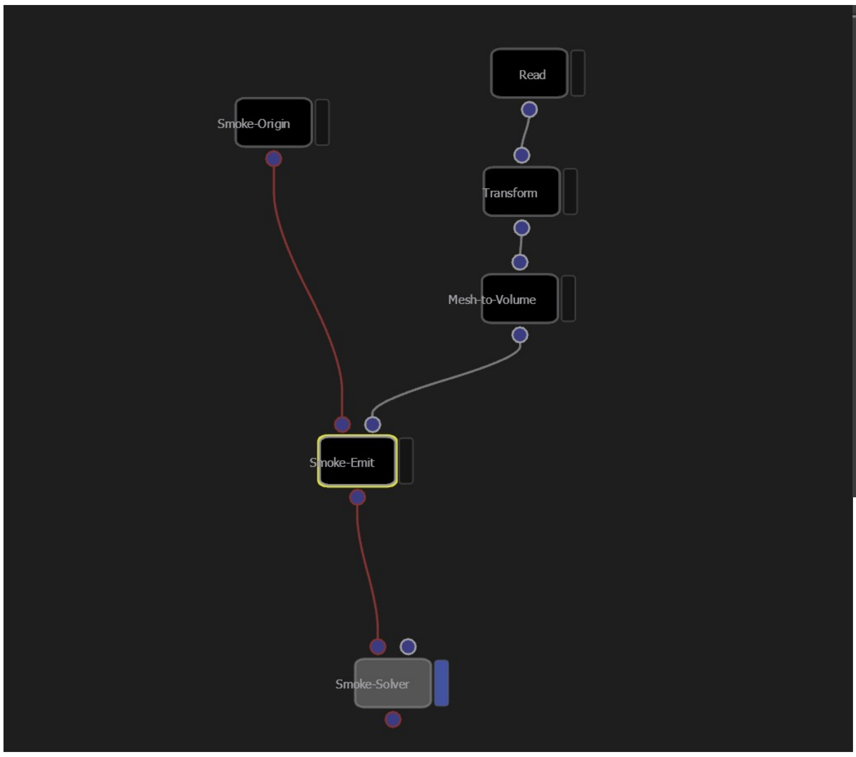
The **red** line on the left indicates a dynamic simulation. It will be happening in the **left three smoke nodes** and evolving there whereas the right nodes will be purely procedural.

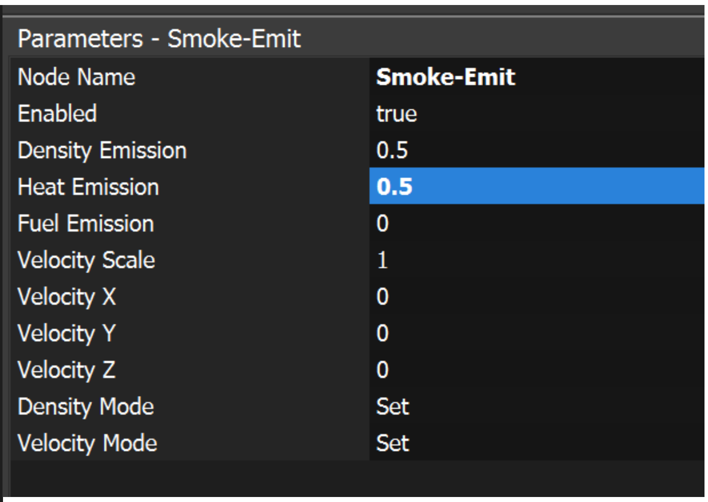
Once you’re done hit the **sim** button in the lower left corner and you should see smoke being emitted!



Smoke being emitted and falling down due to it's weight

The smoke is falling down due to it's own gravity. To change this find the **"Smoke-Emit"** node and change the emitted **heat**, for example to *0.5*:

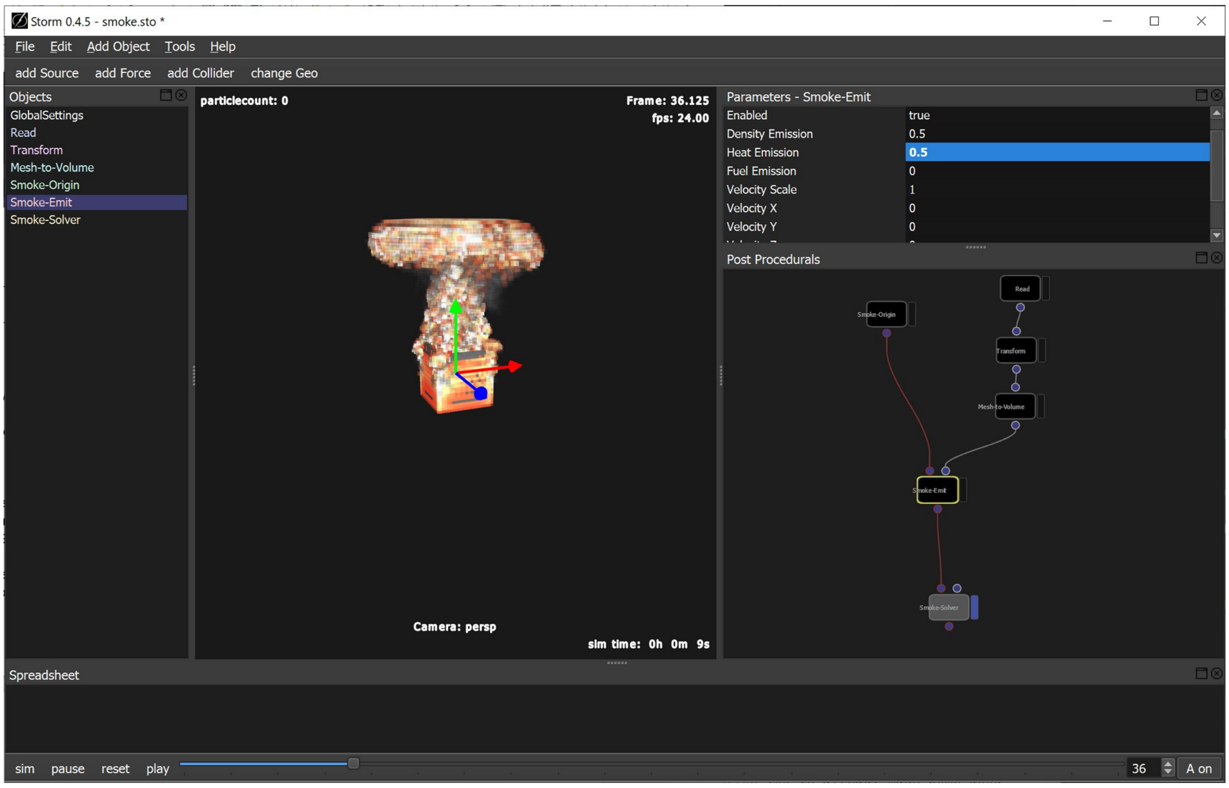




Parameters - Smoke-Emit

Node Name	**Smoke-Emit**
Enabled	true
Density Emission	0.5
Heat Emission	**0.5**
Fuel Emission	0
Velocity Scale	1
Velocity X	0
Velocity Y	0
Velocity Z	0
Density Mode	Set
Velocity Mode	Set

Reset and sim again and there will be hotter smoke – and **rising** due to more temperature:

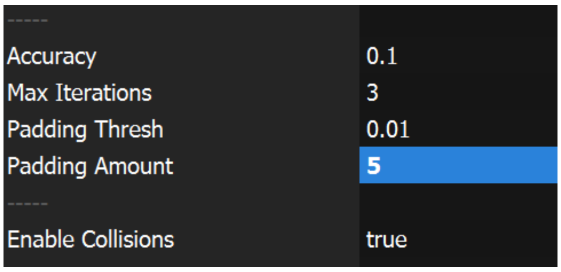


Parameters - Smoke-Emit

Enabled	true
Density Emission	0.5
Heat Emission	0.5
Fuel Emission	0
Velocity Scale	1
Velocity X	0
Velocity Y	0

You might notice unnatural edges at the top of the plume - this is due to the expansion of the underlying voxelgrid which is happening too slow in this case.

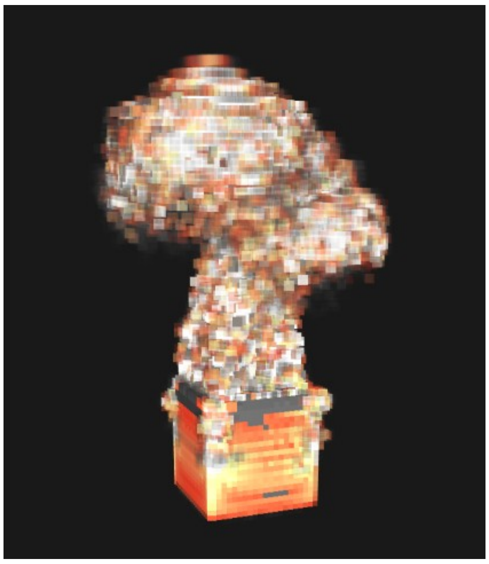
Select the **"Smoke-Solver"** node again and find the parameter **"Padding Amount"** and change it to *5*.



Accuracy	0.1
Max Iterations	3
Padding Thresh	0.01
Padding Amount	5

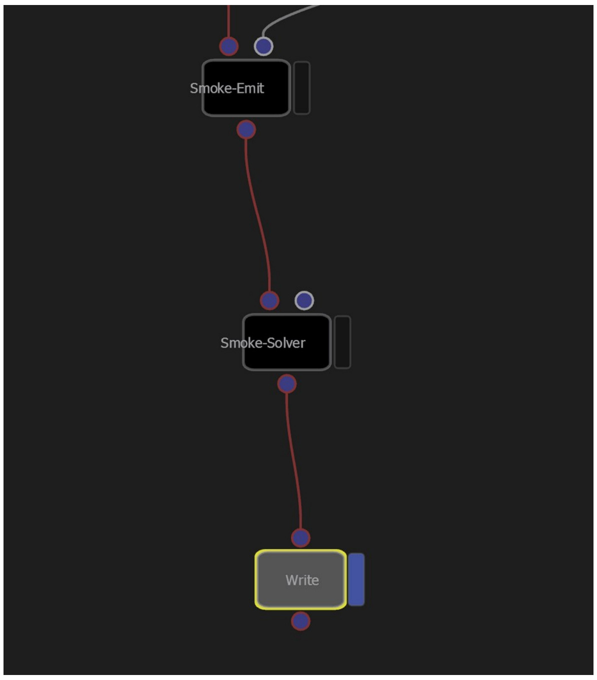
Enable Collisions	true

...The smoke should appear more natural:

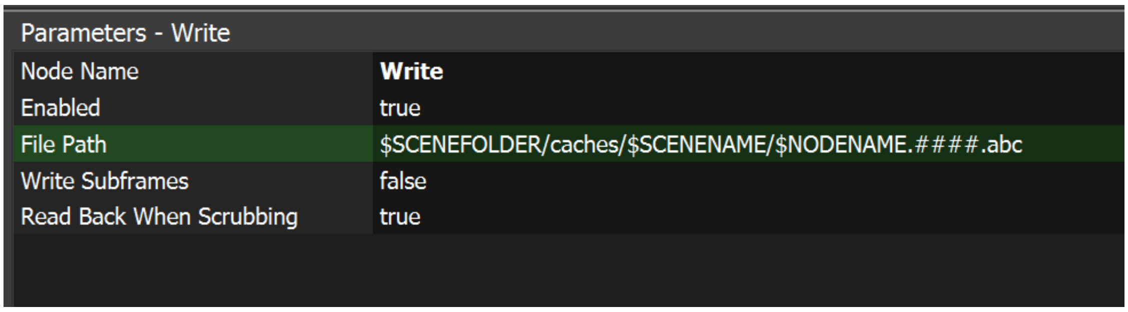


Writing out the sim

To write out the sim select the smoke solver node and hit **“W”** to create a write node.



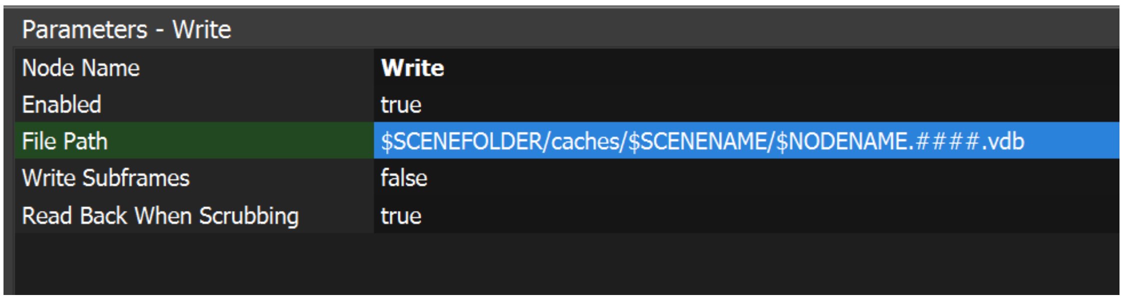
Click on the parameter name “File Path” to switch to and see the entered expression of the path:



Parameters - Write

Node Name	**Write**
Enabled	true
File Path	$SCENEFOLDER/caches/$SCENENAME/$NODENAME.####.abc
Write Subframes	false
Read Back When Scrubbing	true

The expression indicates writing will happen next to the scene files into a subfolder called “caches” and another subfolder of the name of the scene file plus it shows the filename itself will be the node’s name and a file sequence because of the four hashtags.
Change the file extension from **“abc”** to **“vdb”** and save the scene.



Parameters - Write

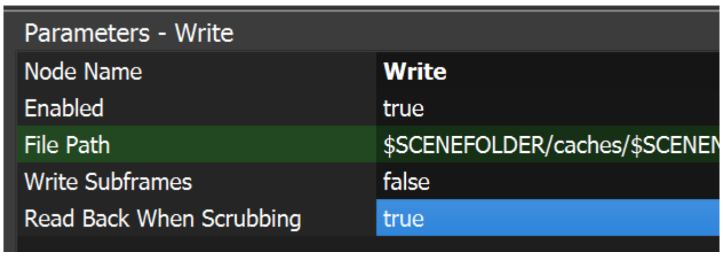
Node Name	**Write**
Enabled	true
File Path	$SCENEFOLDER/caches/$SCENENAME/$NODENAME.####.vdb
Write Subframes	false
Read Back When Scrubbing	true

Reset and simulate for a couple of frames.

When done you can drag the timeslider to go through your sequence.

*Notes on the write node and **reviewing** functionality:*
On default the node will only write out when you simulate by hitting the sim button.
When reviewing the sim by dragging the timeslider or playing back the node will function as a read node that reads back the previously written out results.

To change this behaviour find **"Read Back When Scrubbing"** in the write node's parameter list. Turning this off will make the node evaluate the dynamic nodes above itself and write out the sim – also by just playing back or dragging the timeslider.

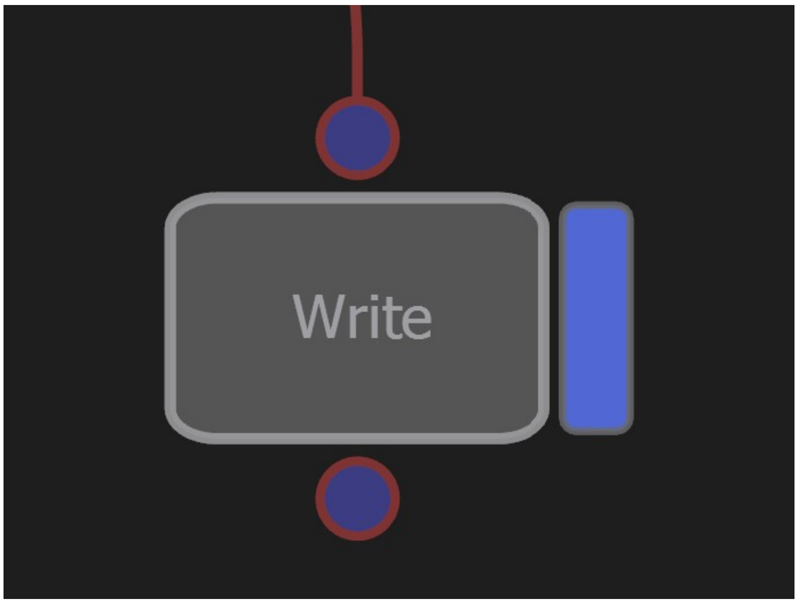


Parameters - Write

Node Name	**Write**
Enabled	true
File Path	$SCENEFOLDER/caches/$SCENEN
Write Subframes	false
Read Back When Scrubbing	true

Visualizing nodes and geometry

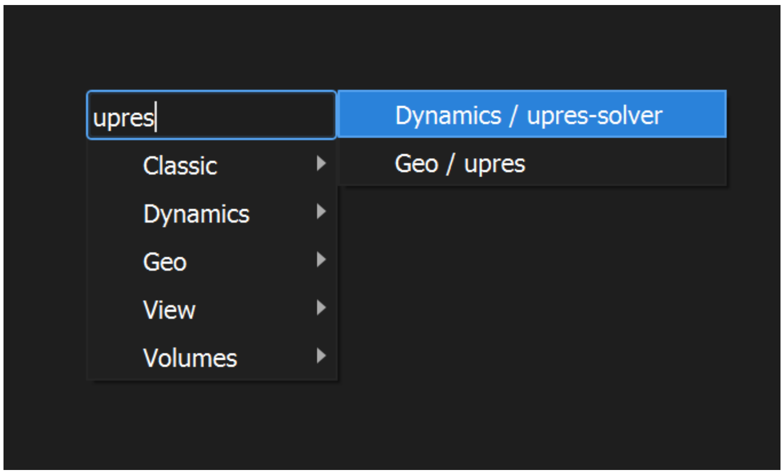
Select the node you want to visualize in the viewport and hit **"1"** on the keyboard (or use the **Visibility Button** on the right side of the node).



2. Basic Upressing Tutorial

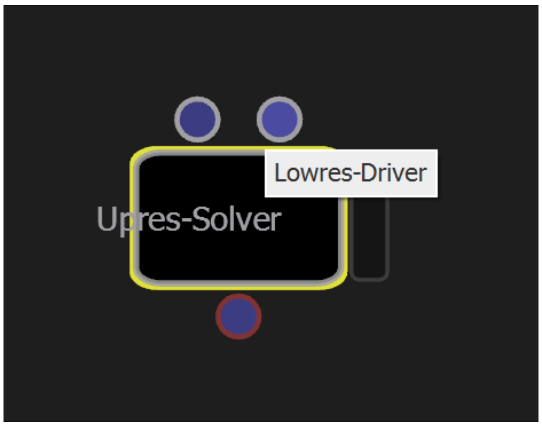
This version of Storm comes with a way to **upres** an existing grain particle simulation using it's new node interface.

Hit **"Tab"** in the node interface or right click, enter **"upres"** and you will see two offered nodes:



Upressing: two nodes are availabe, a dynamic upressing solver node and a procedural one.

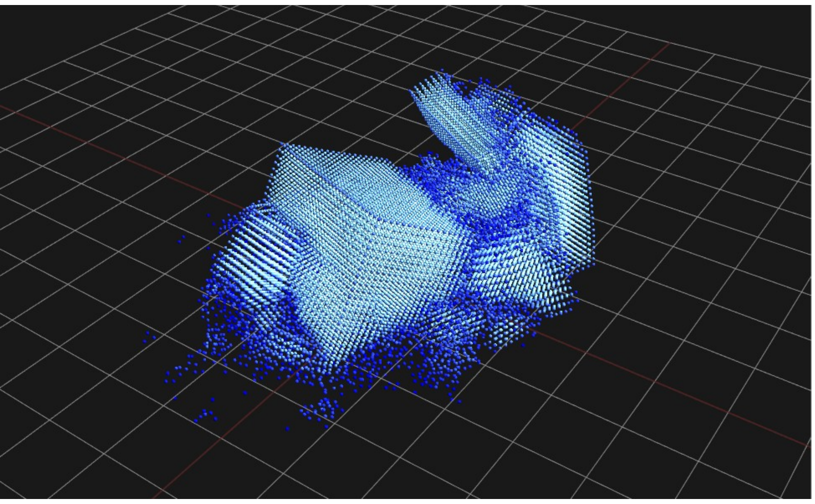
Pick the **"upres-solver"** node and place it in the scene. You will see that it has two inputs, the one on the right is labeled **"Lowres-Driver"** – it is where you can plug-in your existing particle simulation that you want to upres.



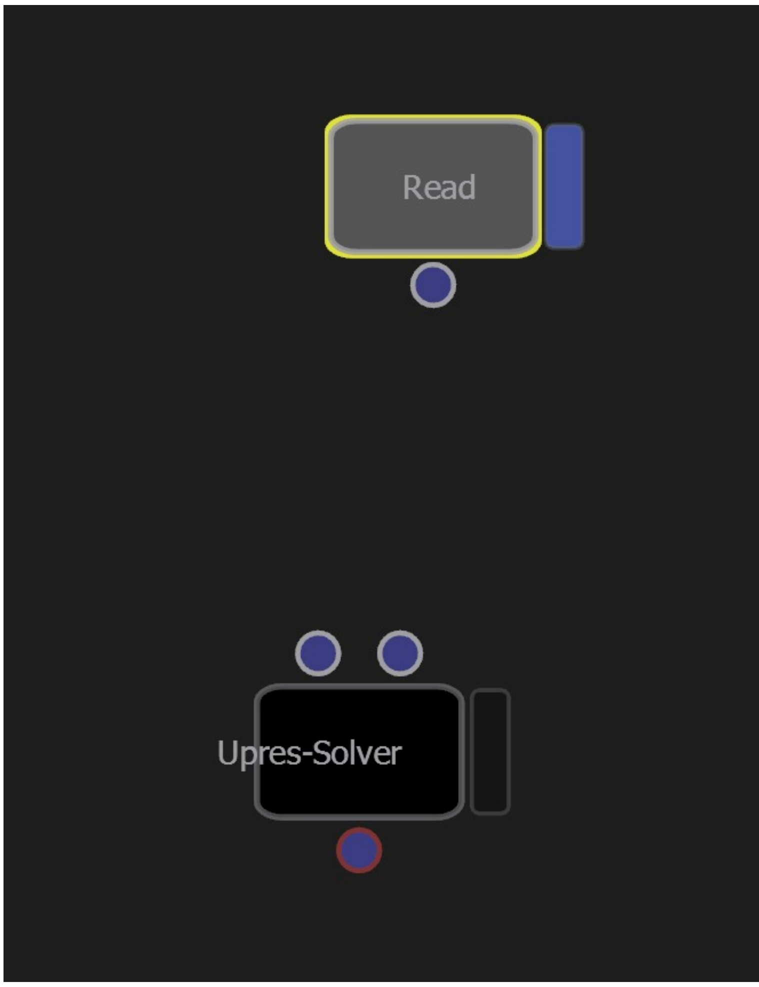
The left one accepts one highres frame of the animation as an initial setup of points for the new dynamic simulation this node performs. Lets look at this later.

For now add a **"Read"** node, point it to an existing particle simulation cache by **shift-clicking** the

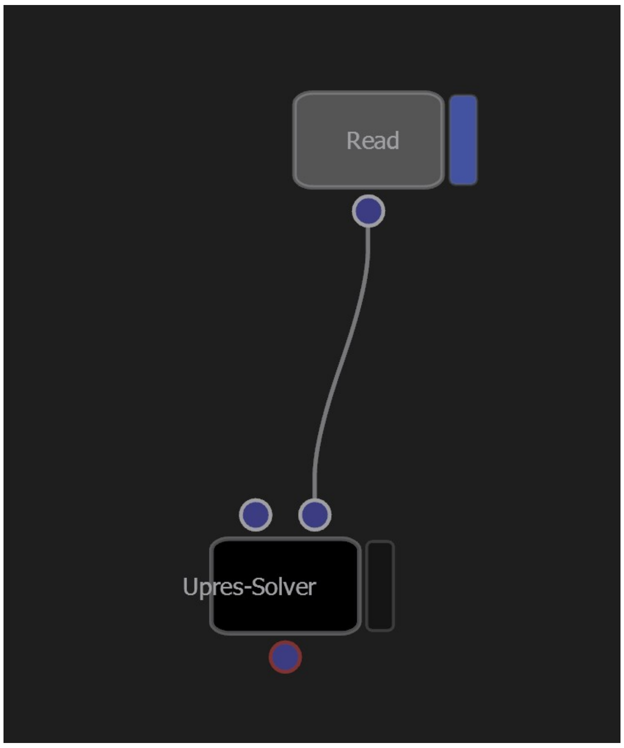
“File Path” parameter and make the node **visible** by hitting **“1”** on the keyboard.



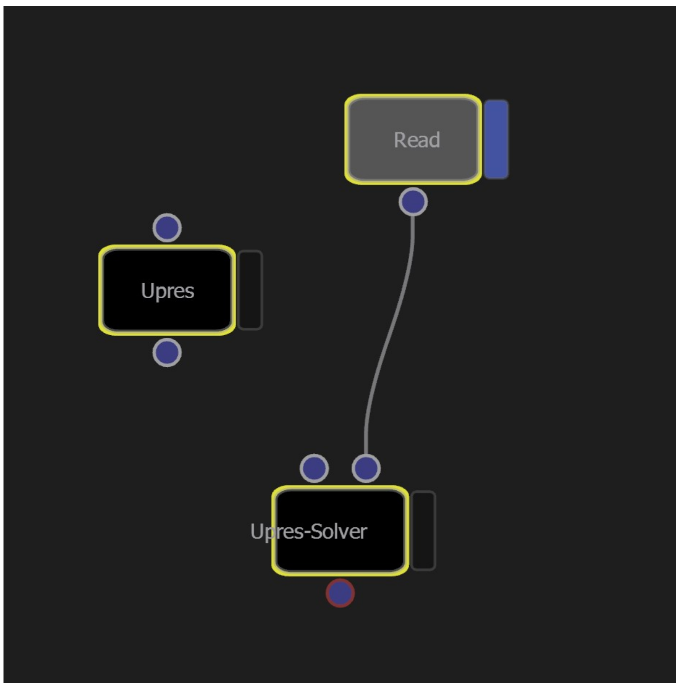
Loading in your existing particle simulation.

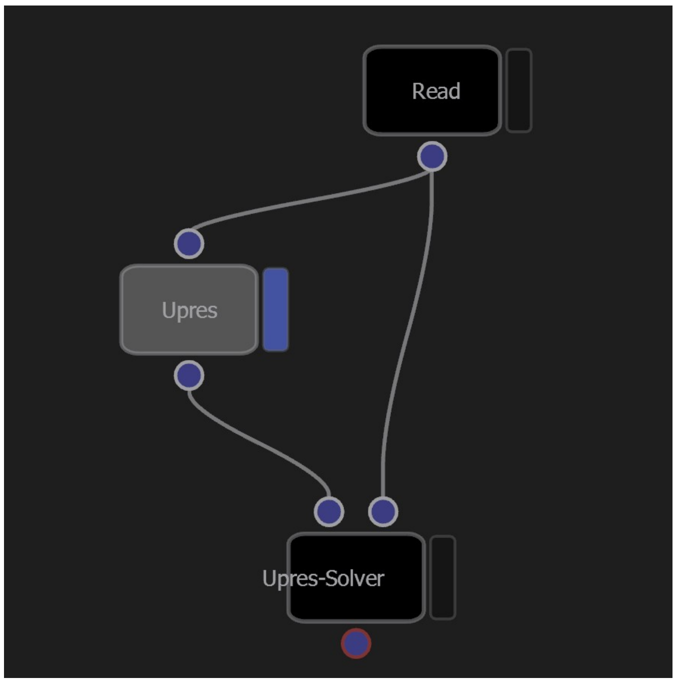


Then connect the read node with the right input of the upres-solver:



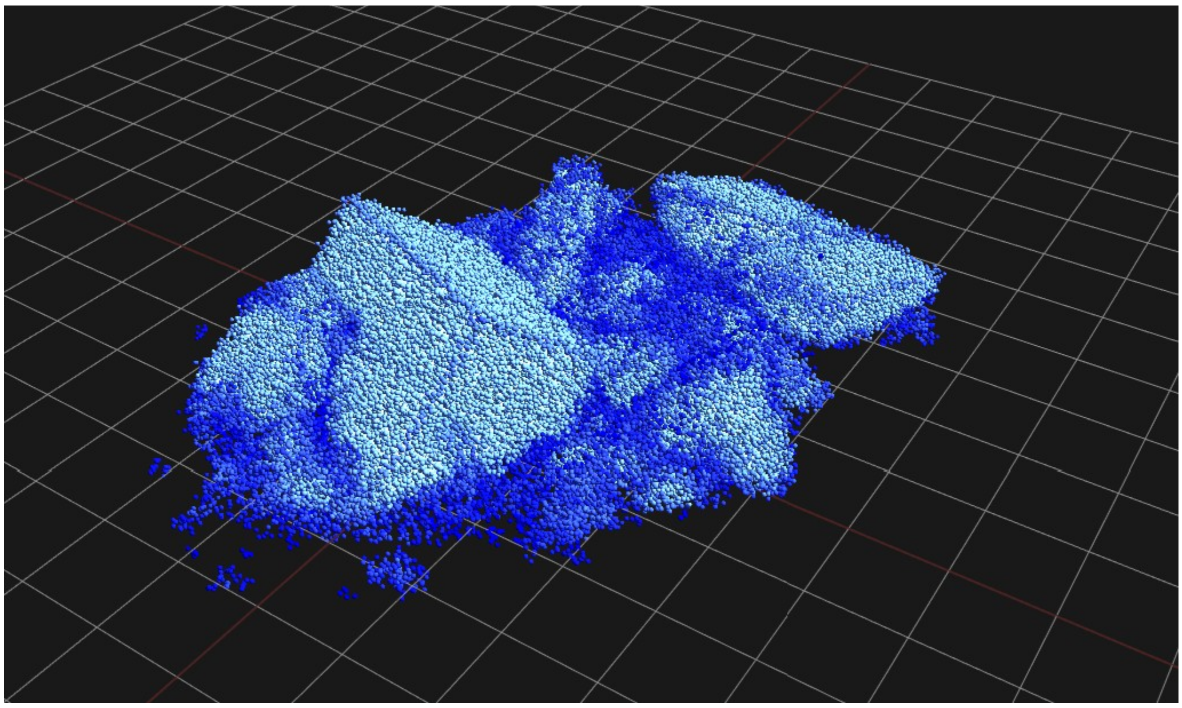
Lets make a new starting point of the new high res simulation.
Place an **"upres"** node in the scene, select all three nodes and hit **"S"** – to connect all three nodes automatically.





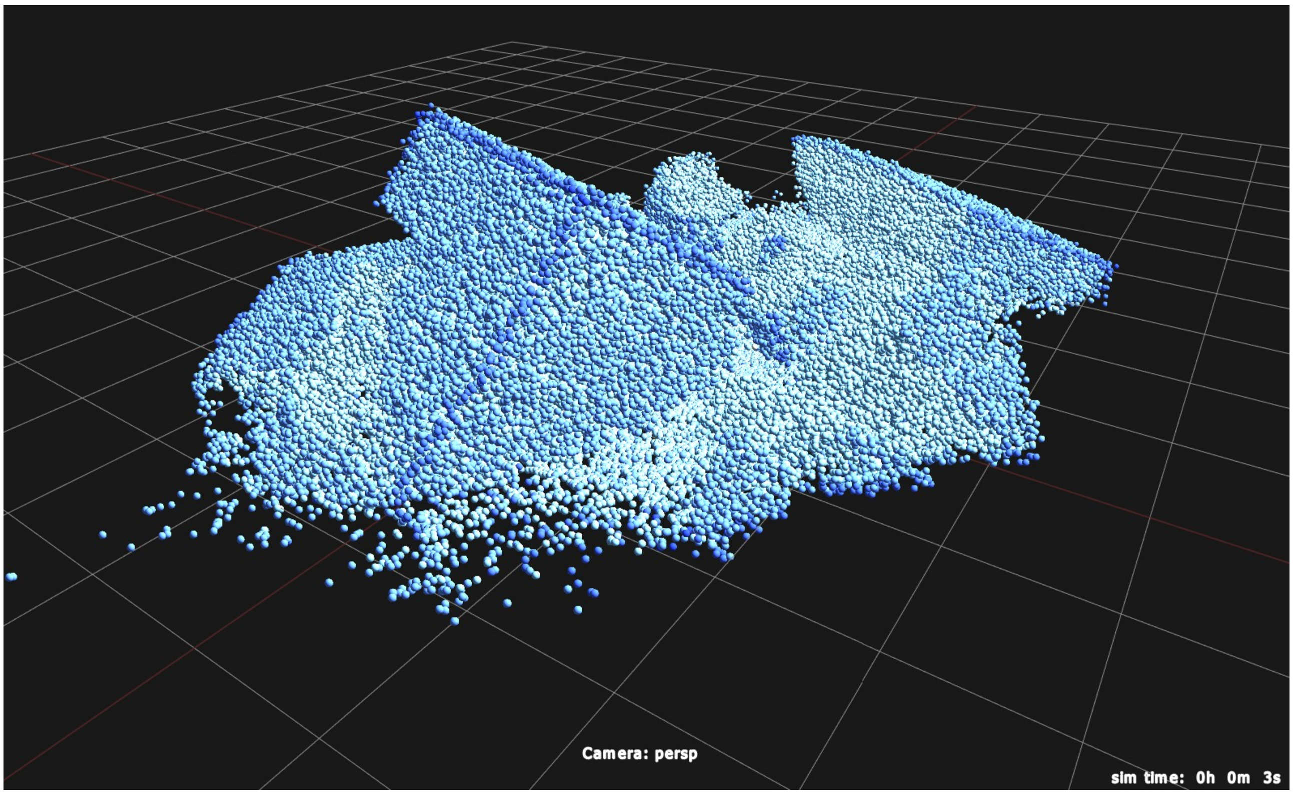
Connecting multiple nodes by hitting "S"

Make the **"Upres"** node visible and you should non-dynamic, procedurally multiplied point counts of your original sim:



Procedural upressing with the "upres" node – fully following the input anim.

Now make last node in the chain, the Upres-Solver node visible, reset the sim and hit the "sim button", both in the lower left corner of the screen – you will see a new dynamic simulation of particles resembling the movements of the original sim.



Dynamic upressing with the "upres-solver" node – following and able to detach from the input.

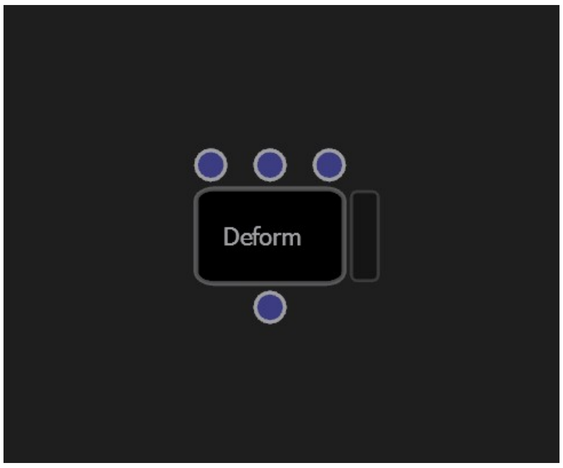
Notes for collisions:
For collisions please place "Collide" nodes below the "upres-solver" and feed them with your collision geometry. Make sure the last node in the chain is visible and run the sim.

3. Basic Mesh Deformer Tutorial

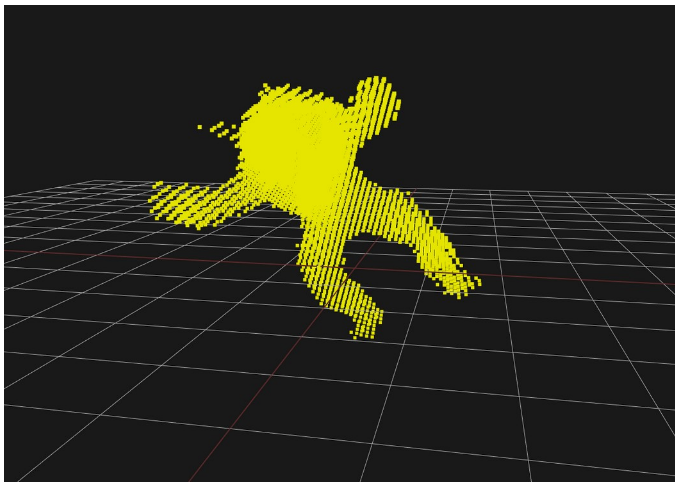
Now it's possible to deform your geometry with a particle simulation using nodes.

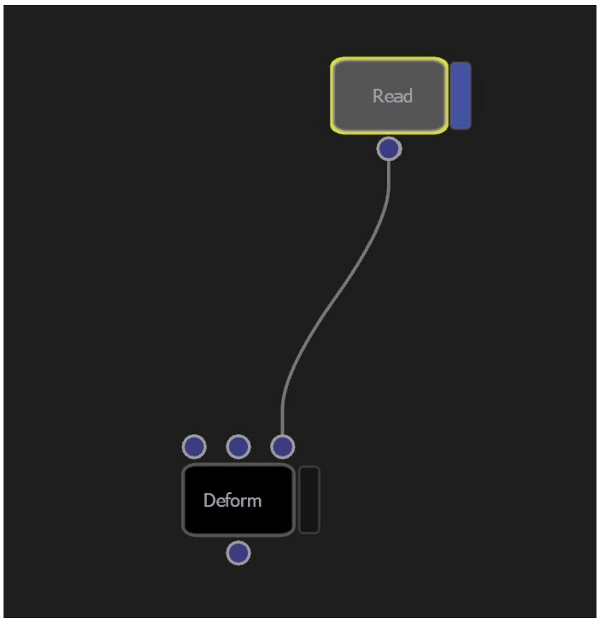
The workflow would be to first simulate your grain simulation the "classic" way – using Storm's granular particle system and emitting from sources and caching out your sim by setting the check mark in the preferences for cache export.

Then, disable your GranularParticleSystem (turn "Enabled" to "false by Shift+cicking the parameter name) or create a new empty scene. In the "Post Procedurals" window hit "Tab" and look for the **"Deform**" node.

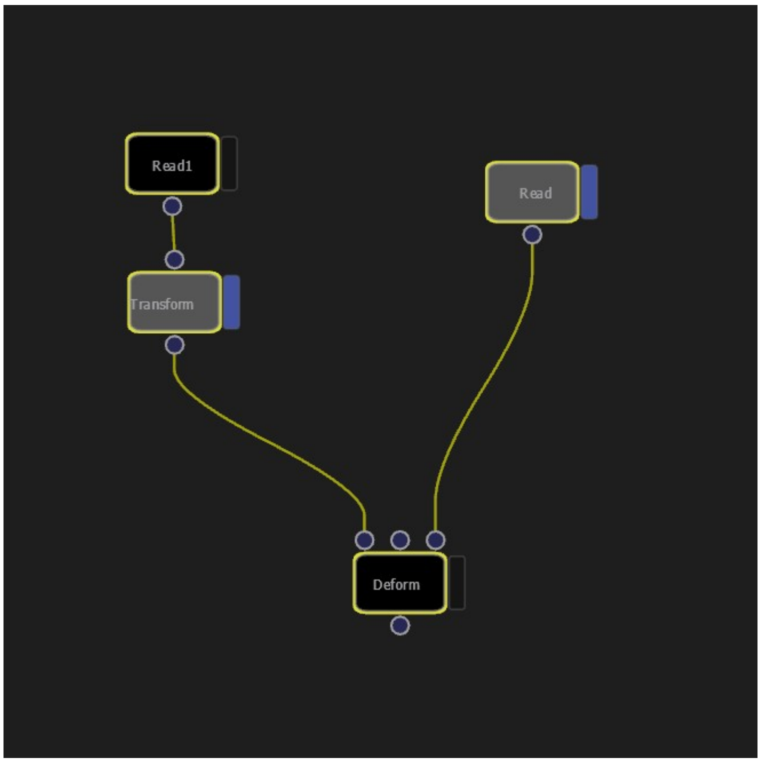


Place a read node in there and connect it to the defomer's right input – here we will load in your existing particle simulation cache that will then guide the deformer.
You should see your particle sim in the viewport.



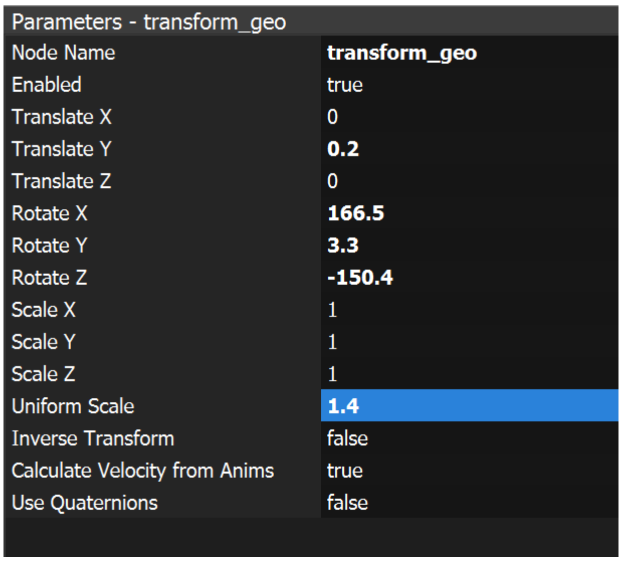


Now add another "Read" node and place it to the left and create and connect it to a "Transform" node. Wire these up to the deformer's left input. Point this read node to your geometry you want to deform.



Left in is the mesh to deform, on the right comes in your grain simulation.

The most tricky part is now to enter the same transform values on the "Transform" node so the geometry lines up with your original particle source geometry (usually solved by just using the identical transform values).

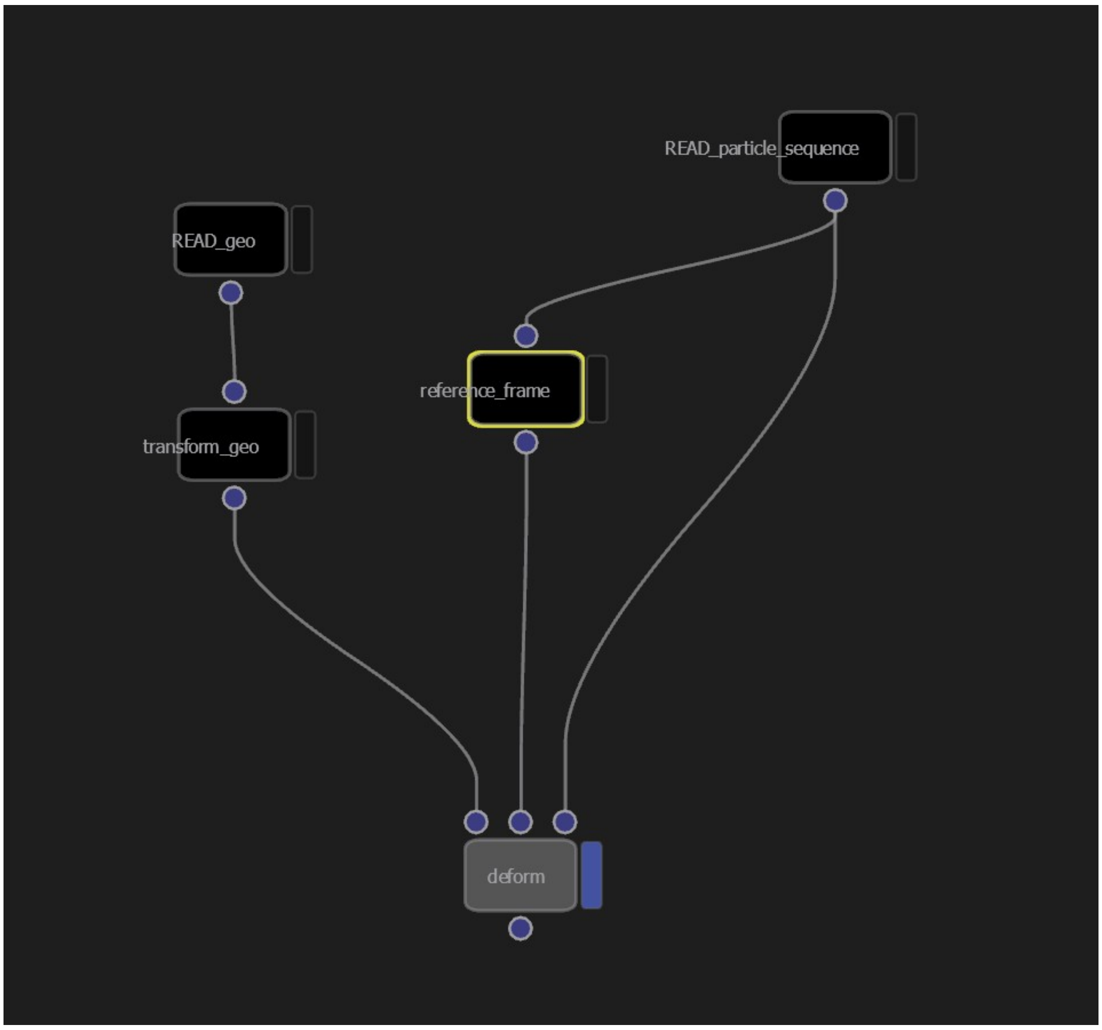


Parameters - transform_geo

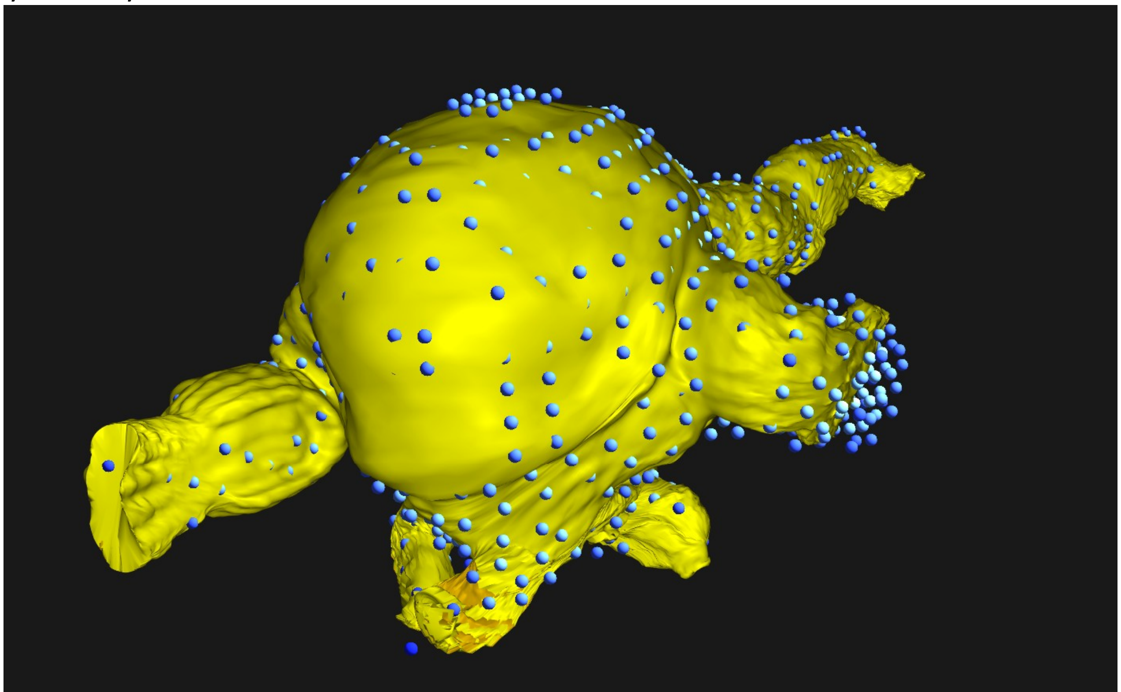
Node Name	**transform_geo**
Enabled	true
Translate X	0
Translate Y	**0.2**
Translate Z	0
Rotate X	**166.5**
Rotate Y	**3.3**
Rotate Z	**-150.4**
Scale X	1
Scale Y	1
Scale Z	1
Uniform Scale	**1.4**
Inverse Transform	false
Calculate Velocity from Anims	true
Use Quaternions	false

Transforming your geo to line up with the original source geometry.

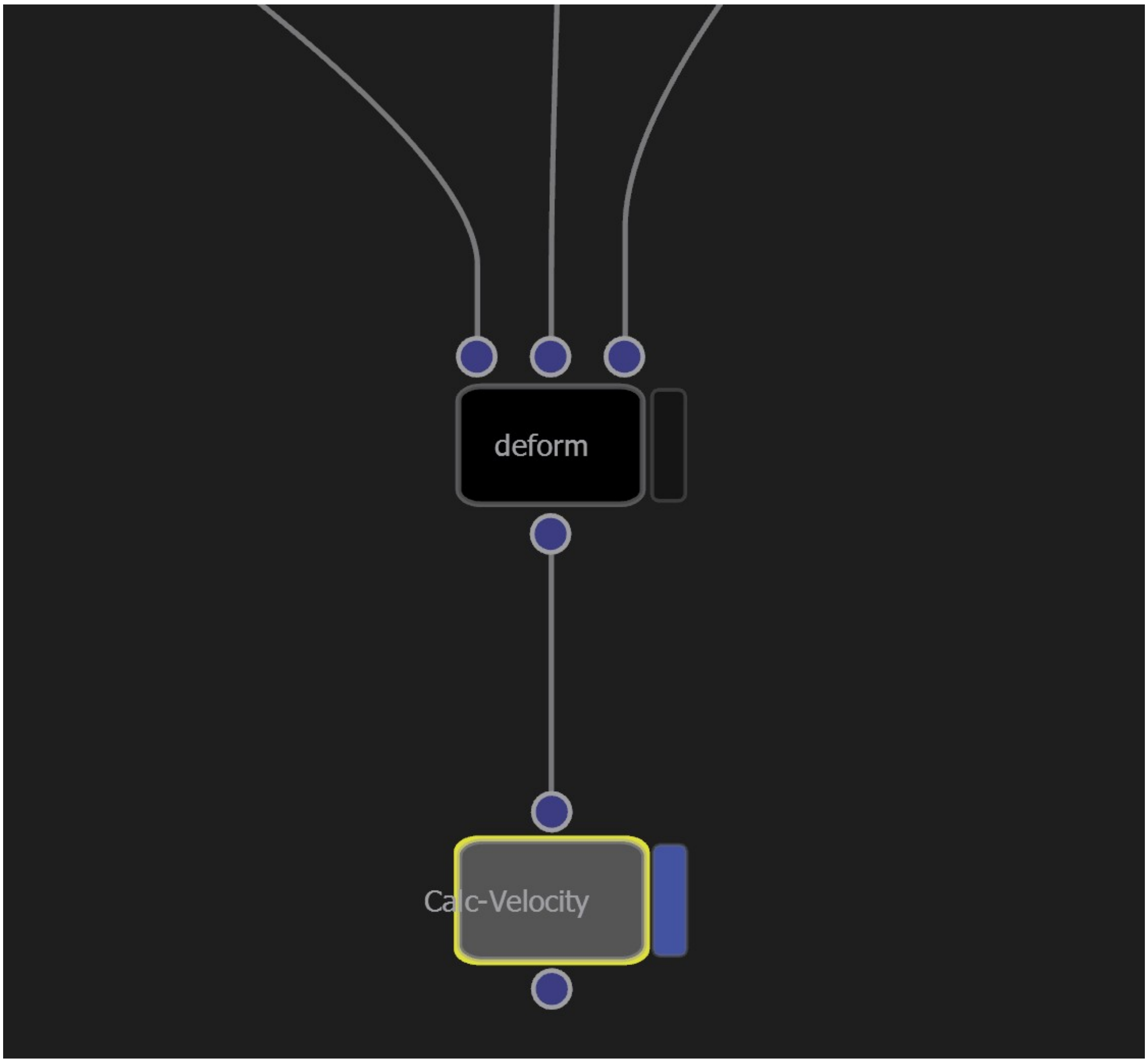
The last step is to define a frame on which the deformer should relate the mesh and the particles to each other. The easiest is by placing a **"Timeshift"** node into the graph and setting its frame to the first frame of your particle sequence, usually it's the start of the frame range. To do this delete the expression there and put in a frame, often it is the frame zero ("0").
Then you can wire things up like seen here:



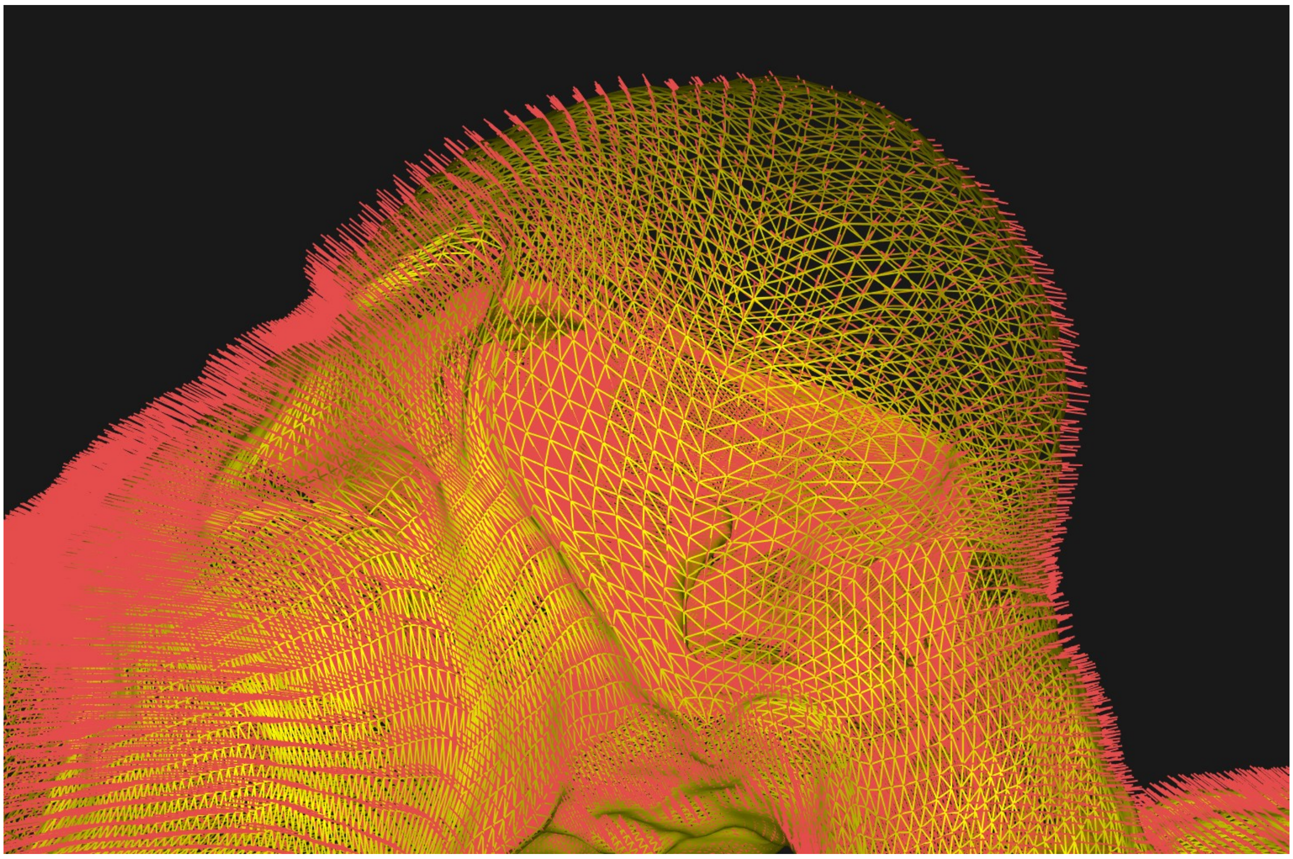
Usually you should see the deformed geo in the viewport and by dragging the timeslider or hitting the play button you should see the animation.



The deformer currently doesn’t produce velocities, but this is easily fixed:
Look for the **“Calc-Vel”** and connect and place it below the deform node.
This node will function when running the sim (not by dragging the timeslider manually).



You can visualize the velocities in the viewport by hitting **“v” –** note this is currently experimental and still has some know issues and performance drops.



The "Calc-Vel" node will take care of adding back in the velocities of the mesh.

You can write out the sim by placing a **"Write"** node into the graph and connecting it to the deformer node.

Note how all connection wires are gray and none is red – this indicates a purely procedural graph as opposed to the upressing scene in the last tutorial. Here we used the upressing solver node with added it's own dynamics, so the sim had to be run from the first to last frame, dragging the timeslider wouldn't lead to the same results.

STØRM

THE SIMULATION TOOL