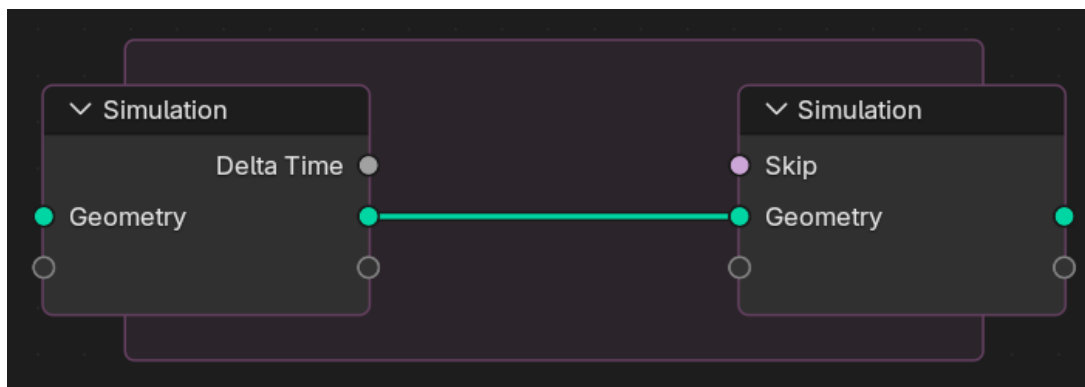


[Skip to content](#)

# Simulation Zone

Simulation zones allow the result of one frame to influence the next one. That way even a set of simple rules can lead to complex results, with the passing of time. The most common type of them is physics simulation, with specific solvers for physical phenomena.



Initial simulation nodes and simulation zone.

When adding a simulation, two nodes are added, defining between them a “Simulation Zone”.

The inputs that are connected to the Simulation Input node are evaluated only once, at the beginning of the simulation, passed to the next simulation state and eventually outputted. Other nodes can be linked inside the simulation region from the outside. Those are re-evaluated every step based on their value at the given frame.

It is not possible to have any link going towards outside. The result of the simulation can only be accessed via the Simulation Output node. This also allow sub-frame interpolation for motion blur.

## Note

This node cannot be used in the [Tool context](#)—only in the *Modifier* context.

## Note

Anonymous attributes are not propagated by the simulation nodes unless they are explicitly stored in the simulation state. This is because detecting which anonymous attributes will be required for the simulation and afterwards would require looking into the future to see what data is necessary.

## Clock

The simulation is tied to the animation system, with support for sub-steps. It will only be evaluated while the animation frame changes, and is cached like the existing physics simulations in Blender.

## Properties

In the Node Editor the inputs can be renamed, shuffled and removed. This is also the place where sub-steps can be defined for a simulation.

## Inputs

### Geometry

Standard geometry input, which is available by default to input geometry into the simulation zone. More bake items can be added by dragging sockets into the blank socket or in the *Simulation State* panel. Items can be renamed by `Ctrl - LMB` on the socket name or in the nodes *Properties* panel.

### Delta Time

The time in seconds between frames. Essentially this the inverse of the render [Frame Rate](#).

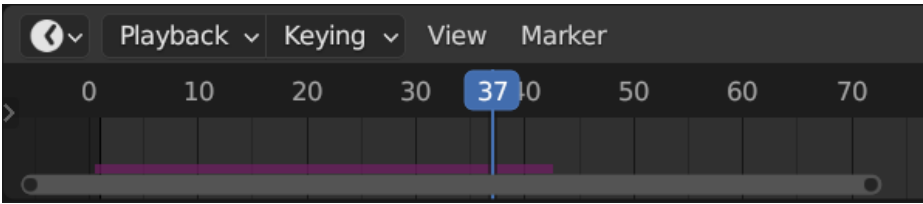
This delta is used to drive the simulation by connecting it node setups that depend on a rate. This will keep the simulation playback consistent when the frame rate changes.

**Skip**

Forward the output of the simulation input node directly to the output node and ignore the nodes in the simulation zone.

**Baking**

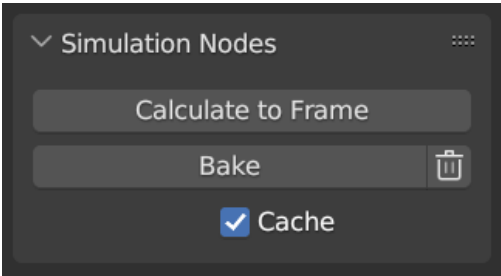
The simulation is automatically cached during playback. The valid cache can be seen as a strong yellow line in the timeline editor. This allows for animato to quickly inspect all the previous frames of a simulation.



Cached frames in the Timeline.

For the cases where the current frame is the only one relevant, users can opt-out of“Cache” to save memory.

When the result is ready to be sent to a render-farm, it can be baked to disk. This allows for the simulation to be rendered in a non-sequential order.



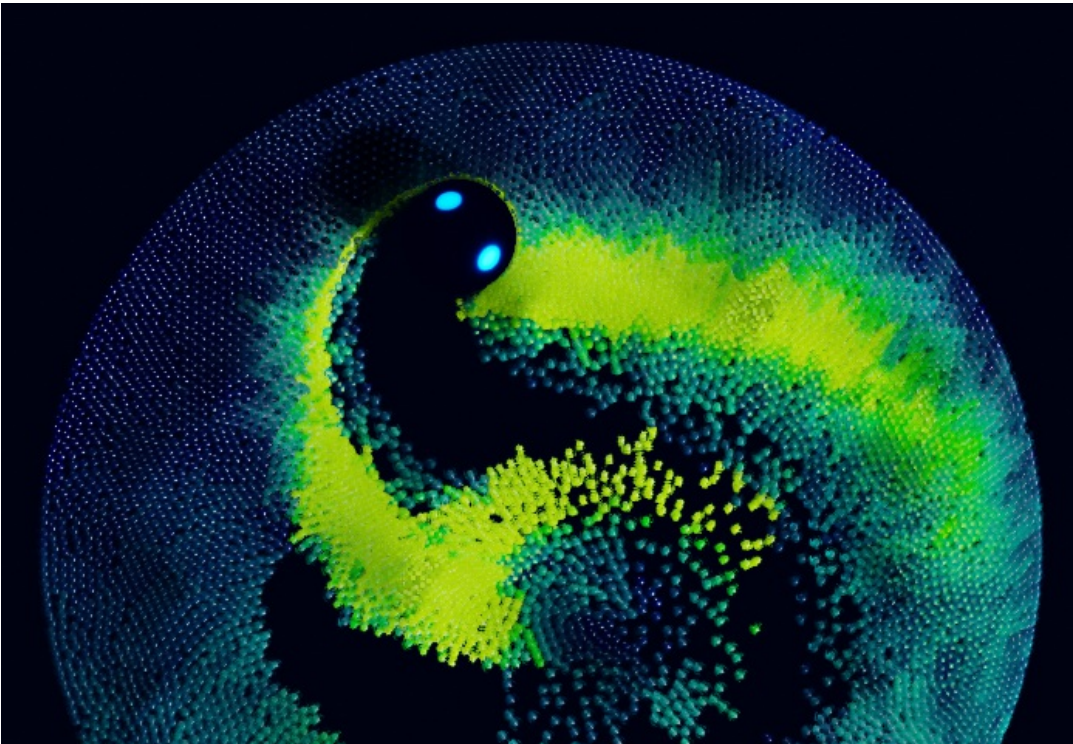
Simulation and Physics, Simulation Nodes user interface.

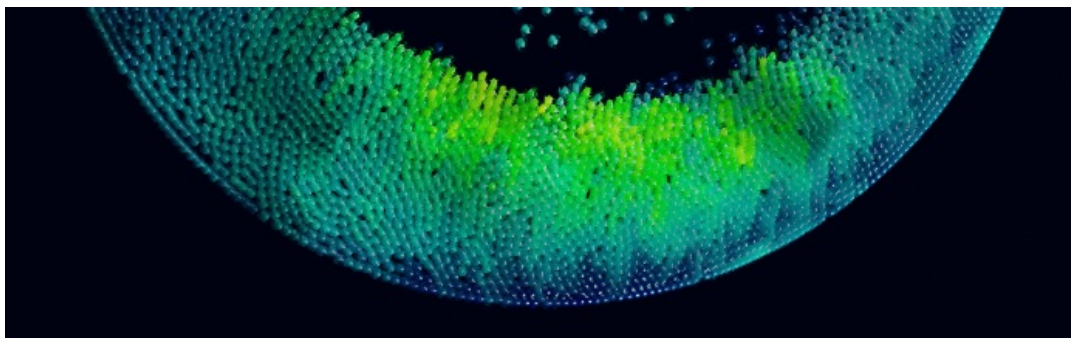
Note

Baking the simulation will bake all the simulations in all modifiers for the selected objects.

**Examples**

Combined with the [Index of Nearest](#), this can be used for a number of sphere-based simulations.





Index of Nearest sample file CC-BY Sean Christofferson.

[Previous](#)  
[Volume Cube Node](#)

[Copyright](#) © : This page is licensed under a CC-BY-SA 4.0 Int. License

Made with [Furo](#)

Last updated on 2025-05-10

[View Source](#)  
[View Translation](#)  
[Report issue on this page](#)

[No](#)  
[Material Noc](#)