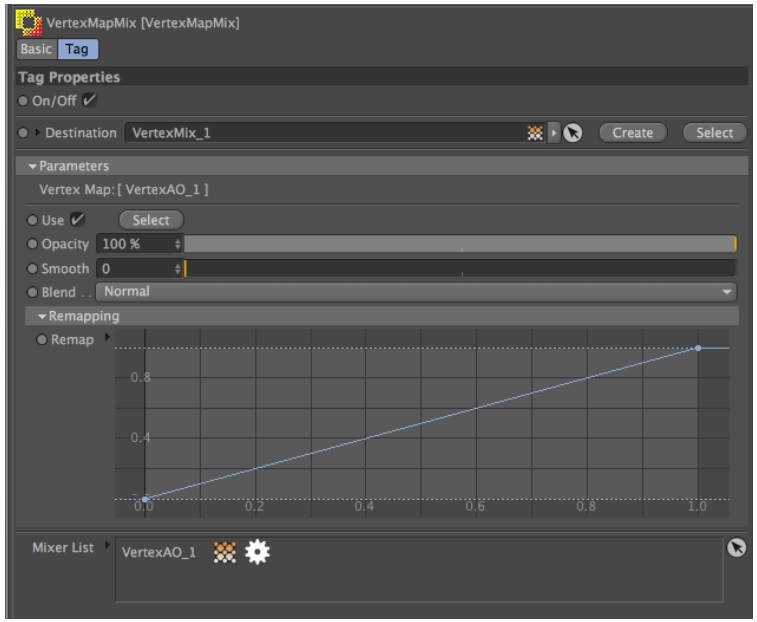
VertexMapMix

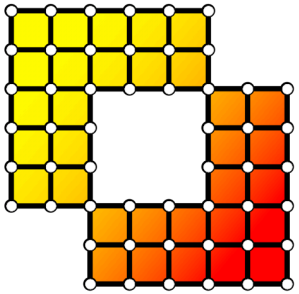
VertexMapMix is a tag that will create a new vertex map based on operations performed on another vertex map or several vertex maps.

This tag, of course, will only work on polygonal editable objects since Vertex Maps are only valid for those types of objects.



The **On/Off** option can be turned off if the calculation needs to be calculated only once (no dynamic generation of vertex maps) or as soon as a successful calculation is achieved.
This will speed up Cinema 4D.

The Vertex Map tag that will store the final result should be dragged into the **Destination** field. If there is still no Vertex Map in the object, a new one can be created by clicking the **Create** button.
To see the result of the calculations applied to the Vertex Map tag, the Vertex Map tag must be selected and, for that, the easiest way is to click the **Select** button at the right of the **Destination** field.



The problem with this is that, if the Vertex Map tag is selected, the Attribute Manager shows the parameters of the Vertex Map tag, instead of the parameters of the **VertexMapMix** tag. To solve this, do the following:

- Select the **VertexMapMix** tag.
- Click the **Lock** icon in the Attribute Manager. 
- Click the Vertex Map tag or click the **Select** button at the right of the **Destination** field.

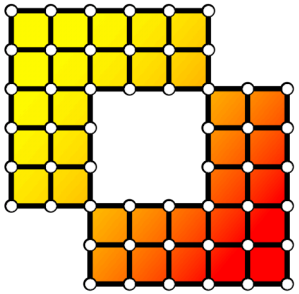
At the bottom of the Attribute Manager, there is a **Mixer List** field where all the Vertex Map tags (it can be only one or several) that will be used for the calculation should be dragged.
In this list, all the Vertex Map tags will show their name, a Vertex Map tag icon and a cogwheel icon.

This cogwheel icon is very important, as we will see in a moment.

The Vertex Map tags in this list are evaluated from top to bottom and they can be dragged up and down to rearrange them and modify the result.

All your Vertex Map tags should be named with useful and coherent names. Otherwise, it will be harder to understand to which Vertex Map tag the elements in the **Mixer List** refer to, if you have a list of items, all named *Vertex Map*.

Vertex Map tags created with **VertexMapMix** and **VertexAO** have default names of **VertexMix**_*n* and **VertexAO**_*n*, respectively.
Change the names of the Vertex Map tags as soon as possible to more descriptive names.



If the **Mixer List** is empty, the resulting Vertex Map tag will be also empty (all red).

If the **Mixer List** contains only one Vertex Map tag, the resulting Vertex Map tag will be, by default, identical to the Vertex Map tag in the **Mixer List** (this can be changed).

If the **Mixer List** contains more than one Vertex Map tag, the resulting Vertex Map tag will be, by default, identical to the last (bottom) Vertex Map tag in the **Mixer List** (this can be changed).

This works like this because, by default, the blending mode of each Vertex Map is set to **Normal**, with and **Opacity** of 100%. So, each Vertex Map will simply replace the one before (above) it.

To change the way each Vertex Map in the **Mixer List** interacts with the one above it, click its correspondent **cogwheel icon**, to access its parameters.

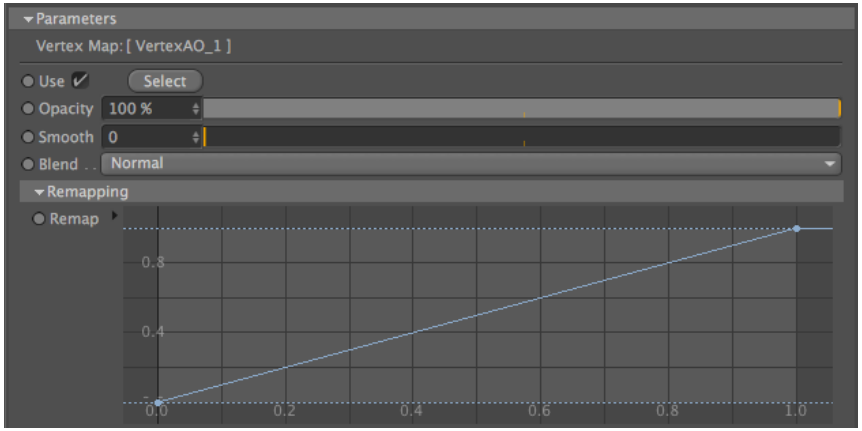
WARNING!

Due to limitations in the way lists are provided to programmers, simply clicking the name or icon of the Vertex Map in the list will **NOT** load its parameters to the relevant fields.

For that to work, the **cogwheel icon** must be pressed.

Once the correspondent cogwheel icon is clicked, the **Parameters** fields are populated with the relevant values associated with that particular Vertex Map.
To make sure that the correct parameters are being shown, check the name of the **Vertex Map** that is shown, as the first item in the **Parameters**. Hence, the importance of correct naming of Vertex Maps.

The parameters are stored in the Vertex Map tags (invisible to the user). So, if a Vertex Map tag is used in more than one **VertexMapMix** tag, with different parameters, some unpredictable results may occur.
If you need to use a Vertex Map more than once with different parameter, use an additional **VertexMapMix** tag to create a duplicate of that specific Vertex Map first and then use that duplicate to define different parameters.

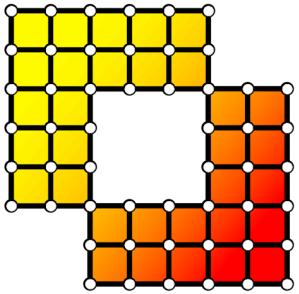


Turning off the **Use** option will exclude this Vertex Map from the calculation. It will simply be ignored.

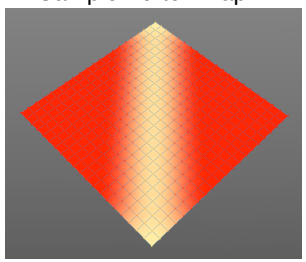
Adjusting the **Opacity** parameter will determine the “transparency” of the Vertex Map. Making it less than 100% will make the previous Vertex Map (or result of previous Vertex Map calculations) start to show through.
The “previous” Vertex Maps are the ones above, in the **Mixer List**.

The **Smooth** parameter will “blur” the result. It can go from 0 (zero, meaning “no blur”) to 10 interactions of the “blur” effect.
If more than 10 interactions are needed, an intermediate Vertex Map can be used to store the “blurred” result and to serve as starting point for another calculation in another **VertexMapMix** tag (several can be used in an object).

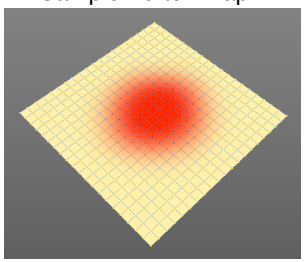
The **Blend** parameter will determine how the Vertex Map is "mixed" with the previous Vertex Map (or result of previous Vertex Map calculations).
The "previous" Vertex Maps are the ones above, in the **Mixer List**.
The available blend modes are as follows.



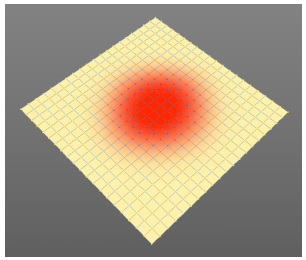
Sample Vertex Map 1



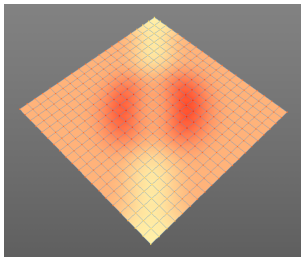
Sample Vertex Map 2



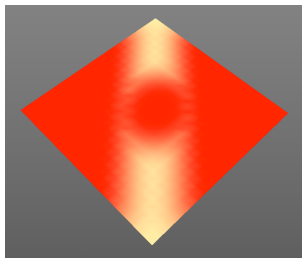
Normal



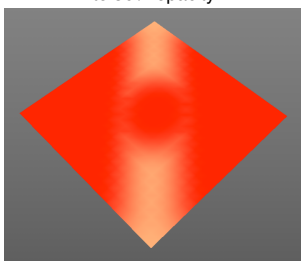
Normal - Vertex Map 2 set to 50% opacity



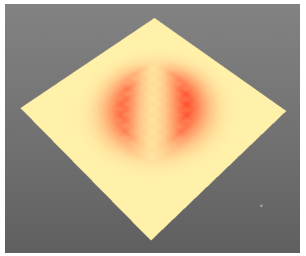
Minimum



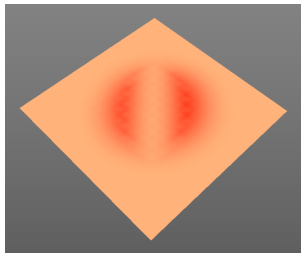
Minimum - Vertex Map 2 set to 50% opacity



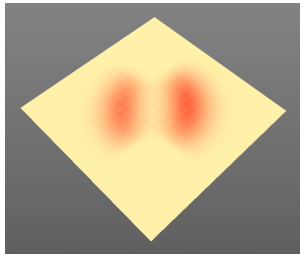
Maximum



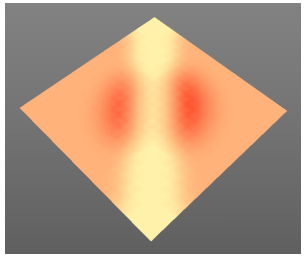
Maximum - Vertex Map 2 set to 50% opacity



Add

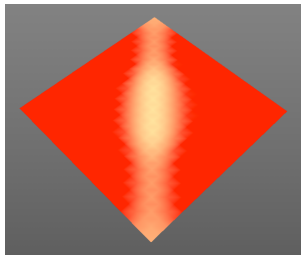


Add - Vertex Map 2 set to 50% opacity



Subtract

Subtract - Vertex Map 2 set to 50% opacity



Finally, using the Remap spline (closed, by default, inside the Remapping group) it is possible to change the way values are interpreted. The contrast and brightness can be adjusted. Changing the slope of the curve, the result can even become inverted.

Pressing the **Select** button at the right of the **Use** option, will select the relevant Vertex Map, showing its values in the editor.