

5.5 Modeling - Metas

Metas.....	1
Primitives.....	2
Visualization.....	2
Meta Ball Options.....	2
Resolution.....	3
Threshold (Influence).....	4
Update.....	4
Meta Structure.....	5
Technical Details.....	5
Underlying Structure.....	5
Editing Metas.....	6
Meta Shape.....	7
Stiffness.....	7
Negative Influence.....	8
Hiding Elements.....	9
Deleting Elements.....	9
Conversion.....	9
Object Families.....	9
Examples.....	10
Hints.....	11

Metas

Reference

Mode: *Object* or *Edit* modes

Menu: Add ▸ Meta

Meta objects are *implicit surfaces*, meaning that they are *not explicitly* defined by vertices (as meshes are) or control points (as surfaces are): they exist *procedurally*. Meta objects are literally mathematical formulas that are calculated on-the-fly by Blender.

A very distinct visual characteristic of metas is that they are fluid *mercurial*, or *clay-like* forms that have a “rounded” shape. Furthermore, when two meta objects get close to one another, they begin to interact with one another. They “blend” or “merge”, as water droplets do, especially in zero-g (which, by the way, makes them very handy for modeling streams of water when you don’t want to do a fluid simulation). If they subsequently move away from one another, they restore their original shape.

Each of these is defined by its own underlying mathematical structure (Technical Details), and you can at any time switch between them using the *Active Element* panel.

Typically *Meta* objects are used for special effects or as a basis for modeling. For example, you could use a collection of metas to form the initial shape of your model and then convert it to another object type (well, only meshes are available...) for further modeling. Meta objects are also very efficient for ray-tracing.

Note that *Meta* objects have a slightly different behavior in *Object* mode.

Primitives

There are five predefined meta “primitives” (or configurations) available in the Add ▸ Meta sub-menu:

Meta Ball

adds a meta with a point underlying structure.

Meta Tube

adds a meta with a line segment underlying structure.

Meta Plane

adds a meta with a planar underlying structure.

Meta Ellipsoid

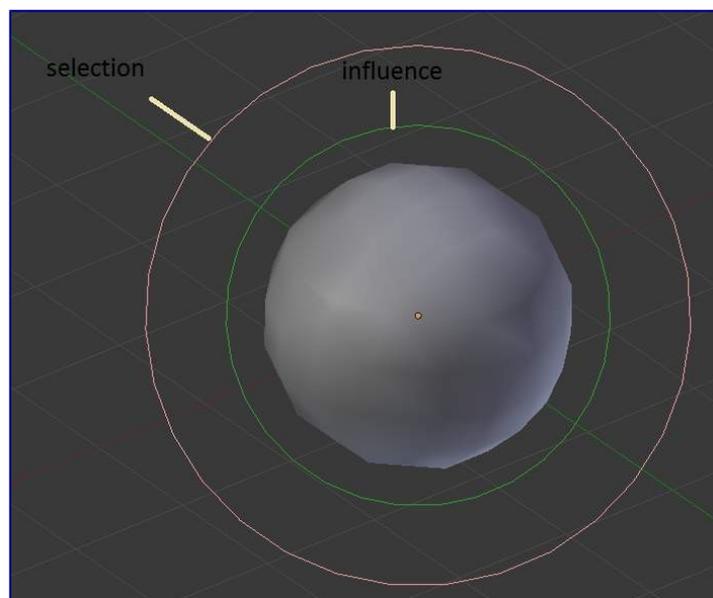
adds a meta with an ellipsoidal underlying structure.

Meta Cube

adds a meta with a volumetric cubic underlying structure.

Visualization

In Object mode, the calculated mesh is shown, along with a black “selection ring” (becoming pink when selected).

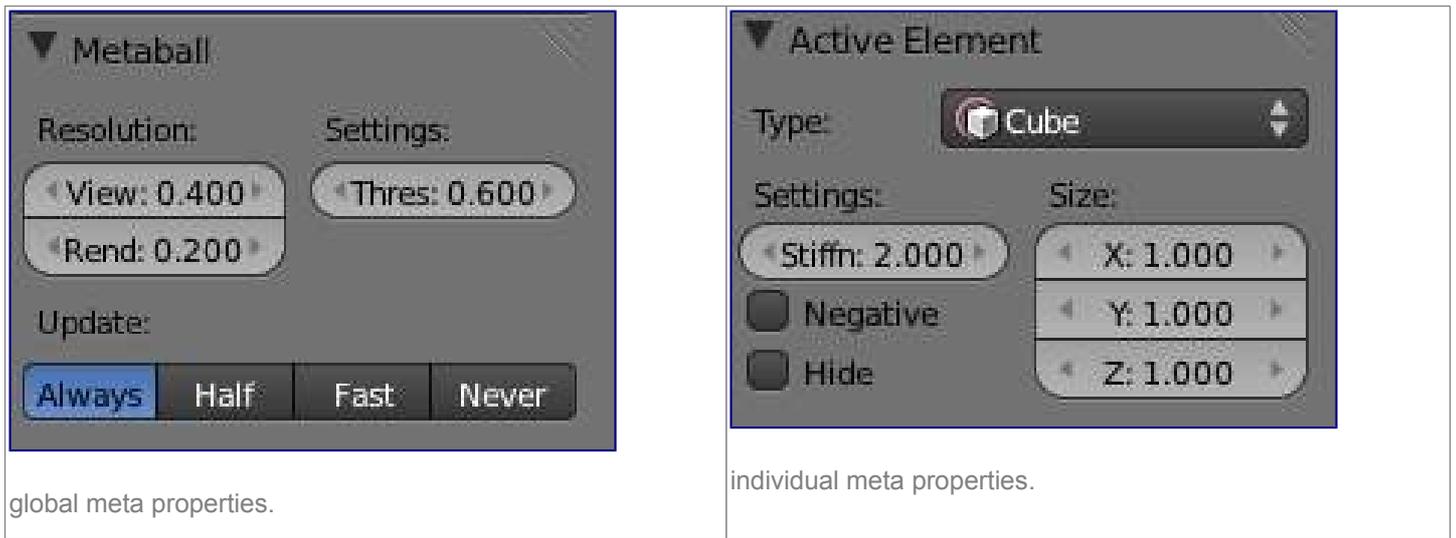


Meta Ball example.

In *Edit* mode (*Meta Ball example*), a meta is drawn as a mesh (either shaded or as black wireframe, but without any vertex of course), with two colored circles: a red one for selection (pink when selected), and a green one for a direct control of the meta’s stiffness (light green when active). Note that except for the *Scale* (S) transformation, having the green circle highlighted is equivalent to having the red one.

Meta Ball Options

All Meta objects in a scene interact with each other. The settings in the *MetaBall* section apply to all meta objects. In *Edit* mode, the *Active Element* panel appears for editing individual meta elements.



Resolution

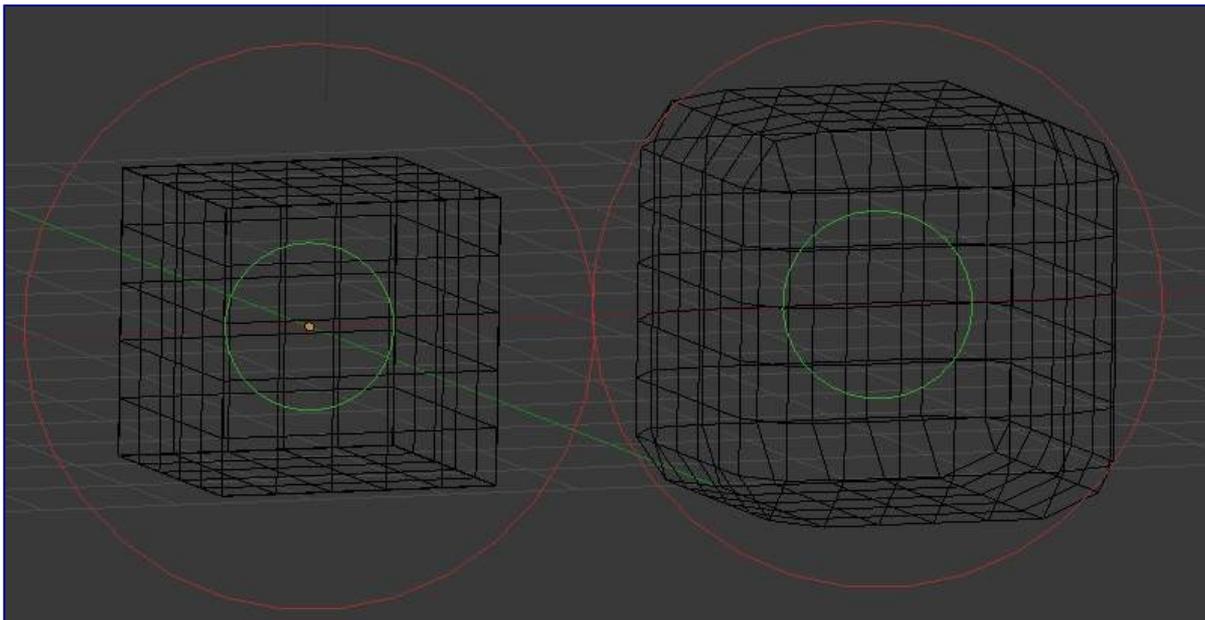
The *Resolution* controls the resolution of the resultant mesh as generated by the

Meta
object.

View
The 3D View resolution of the generated mesh. The range is from **0.05** (finest) to **1.0** (coarsest).

Render
The rendered resolution of the generated mesh. The range is from **0.05** (finest) to **1.0** (coarsest).

One way to see the underlying mathematical structure is to lower the *Resolution*, increase the *Threshold* and set the *Stiffness* (see below) a fraction above the *Threshold*. (*Underlying structure*) is a (*Meta cube*) with the above mentioned configuration applied as follows: *Resolution* of **0.410**, *Threshold* of **5.0** and *Stiffness* a fraction above at **5.01**.



Left: Underlying structure, Right: the shape.

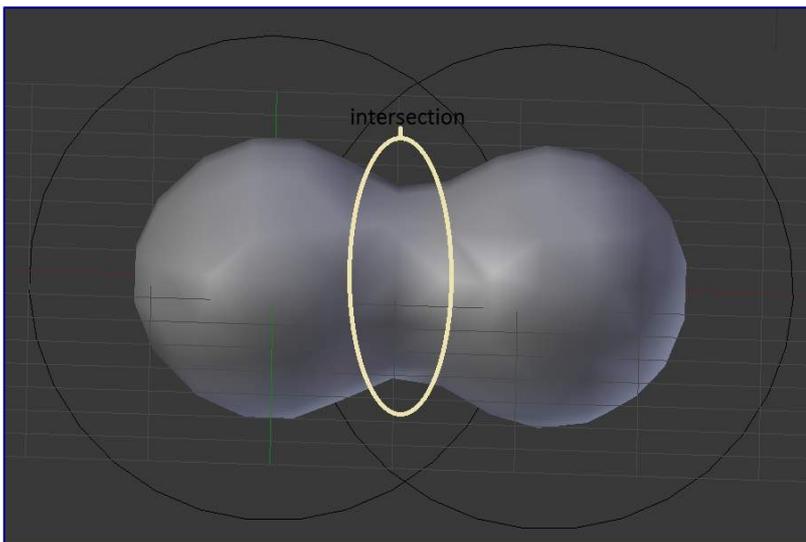
You can clearly see the underlying cubic structure that gives the meta cube its shape.

Threshold (Influence)

Reference
Mode: <i>Object</i> or <i>Edit</i> modes
Panel: <i>MetaBall</i> (<i>Editing</i> context)

Threshold defines how much a meta's surface "influences" other metas. It controls the *field level* at which the surface is computed. The setting is global to a group of *Meta* objects. As the threshold increases, the influence that each meta has on each other increases.

There are two types of influence: **positive** or **negative**. The type can be toggled on the *Active Element* panel while in *Edit* mode, using the *Negative* button. You could think of **positive** as attraction and **negative** as repulsion of meshes. A negative meta will push away or repel the meshes of positive *Meta* objects.



Positive.

A *positive* influence is defined as an attraction, meaning the meshes will stretch towards each other as the *rings of influence* intersect. (*Positive*) shows two meta balls' *rings of influence* intersecting with a *positive* influence.

Notice how the meshes have pulled towards one another. The area circled in white shows the green *influence* rings intersecting.

Update

While transforming metas (grab/move, scale, etc.), you have four "modes" of visualization, located in the *Update* buttons group of the *MetaBall* panel:

Always

fully draw the meta during transformations.

Half Res

During transformations, draw the meta at half its *Wiresize* resolution.

Fast

Do not show meta mesh during transformations.

Never

Never show meta mesh (not a very recommended option, as the meta is only visible at render time!).

This should help you if you experience difficulties (metas are quite compute-intensive...), but with modern computers, this shouldn't happen, unless you use many metas, or very high resolutions...

Meta Structure

Technical Details

A more formal definition of a meta object can be given as a *directing structure* which can be seen as the source of a static field. The field can be either positive or negative and hence the field generated by neighboring directing structures can attract or repel.

The implicit surface is defined as the surface where the 3D field generated by all the directing structures assume a given value. For example a meta ball, whose directing structure is a point, generates an isotropic (i.e. identical in all directions) field around it and the surfaces at constant field value are spheres centered at the directing point.

Meta objects are nothing more than mathematical formulae that perform logical operations on one another (AND, OR), and that can be added and subtracted from each other. This method is also called **Constructive Solid Geometry** (CSG). Because of its mathematical nature, CSG uses little memory, but requires lots of processing power to compute.

Underlying Structure

Reference
Mode: <i>Edit mode</i>
Panel: <i>MetaBall tools (Editing context), Transform Properties</i>

Bforartists has five types of metas, each determined by its underlying (or directing) structure. In *Edit mode*, you can change this structure, either using the relevant buttons in the *MetaBall tools* panel, or the drop-down list in the *Transform Properties* panel (N). Depending on the structure, you might have additional parameters, located in both *Transform Properties* and *MetaBall tools* panels.

Ball (point, zero-dimensional structure)

This is the simplest meta, without any additional setting. As it is just a point, it generates an isotropic field, yielding a spherical surface (this is why it is called *Meta Ball* or *Ball* in Blender).

Tube (straight line, uni-dimensional structure)

This is a meta which surface is generated by the field produced by a straight line of a given length. This gives a cylindrical surface, with rounded closed ends. It has one additional parameter:

dx

The length of the line (and hence of the tube - defaults to **1.0**).

Plane (rectangular plane, bi-dimensional structure)

This is a meta which surface is generated by the field produced by a rectangular plane. This gives a parallelepipedal surface, with a fixed thickness, and rounded borders. It has two additional parameters:

dx, dy

The length, width of the rectangle (defaults to **1.0**).

Note that by default, the plane is a square.

Ellipsoid (ellipsoidal volume, tri-dimensional structure)

This is a meta which surface is generated by the field produced by an ellipsoidal volume. This gives an ellipsoidal surface. It has three additional parameters:

dx, dy, dz

The length, width, height of the ellipsoid (defaults to **1.0**).

Note that by default, the volume is a sphere, producing a spherical meta, as the *Ball* option...

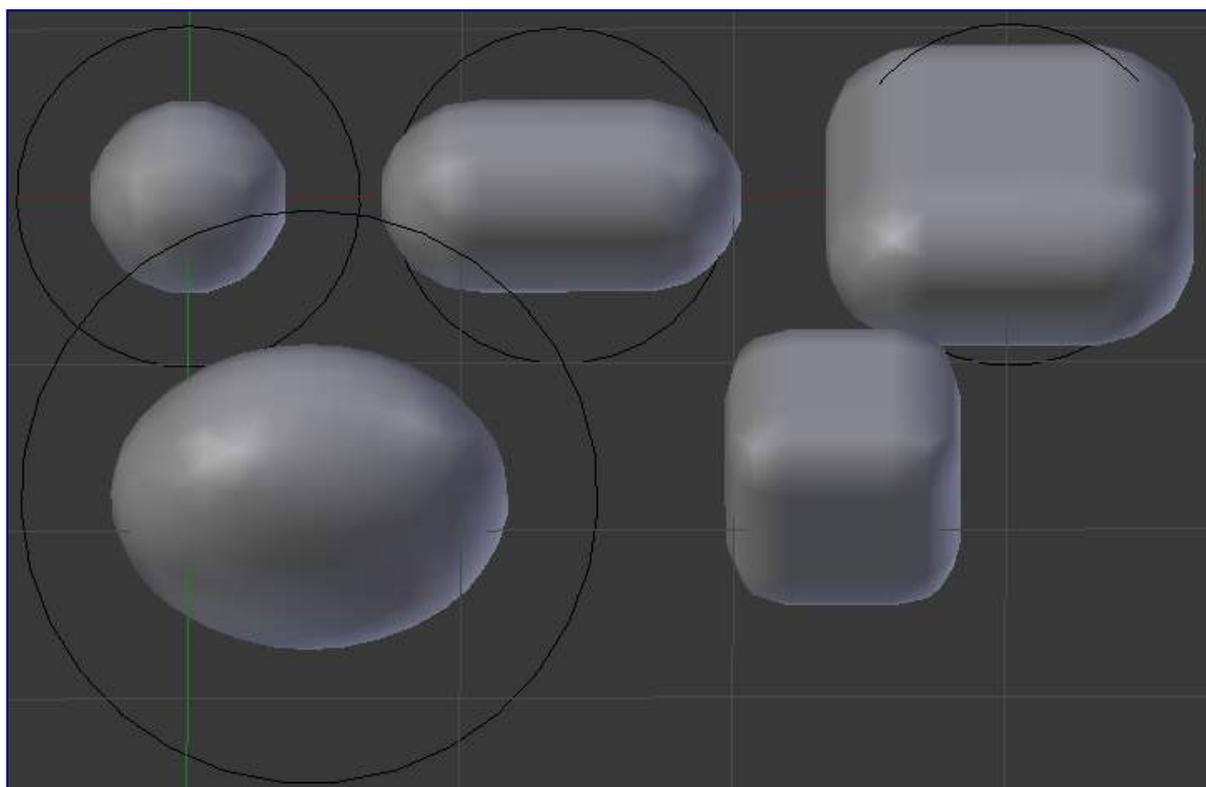
Cube (parallelepipedal volume, tri-dimensional structure)

This is a meta which surface is generated by the field produced by a parallelepipedal volume. This gives a parallelepipedal surface, with rounded edges. As you might have guessed, it has three additional parameters:

dx, dy, dz

The length, width, height of the parallelepiped (defaults to **1.0**).

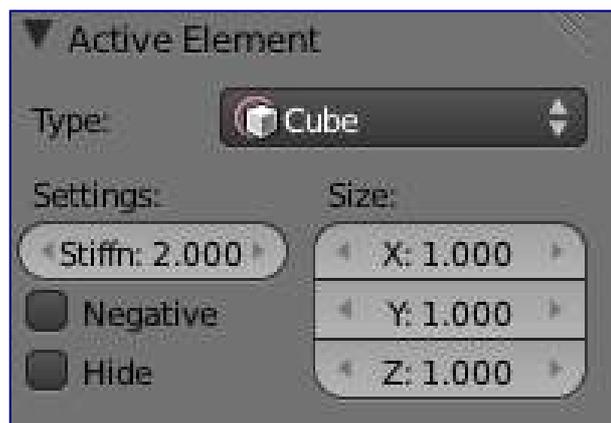
Note that by default, the volume is a cube.



the 5 meta primitives.

Editing Metas

When in *Edit* mode, the *Active Element* panel appears. These settings apply only to the selected meta element.



the active element panel.

Meta Shape

The *Type* menu lets you change the shape of the meta object, as explained above.

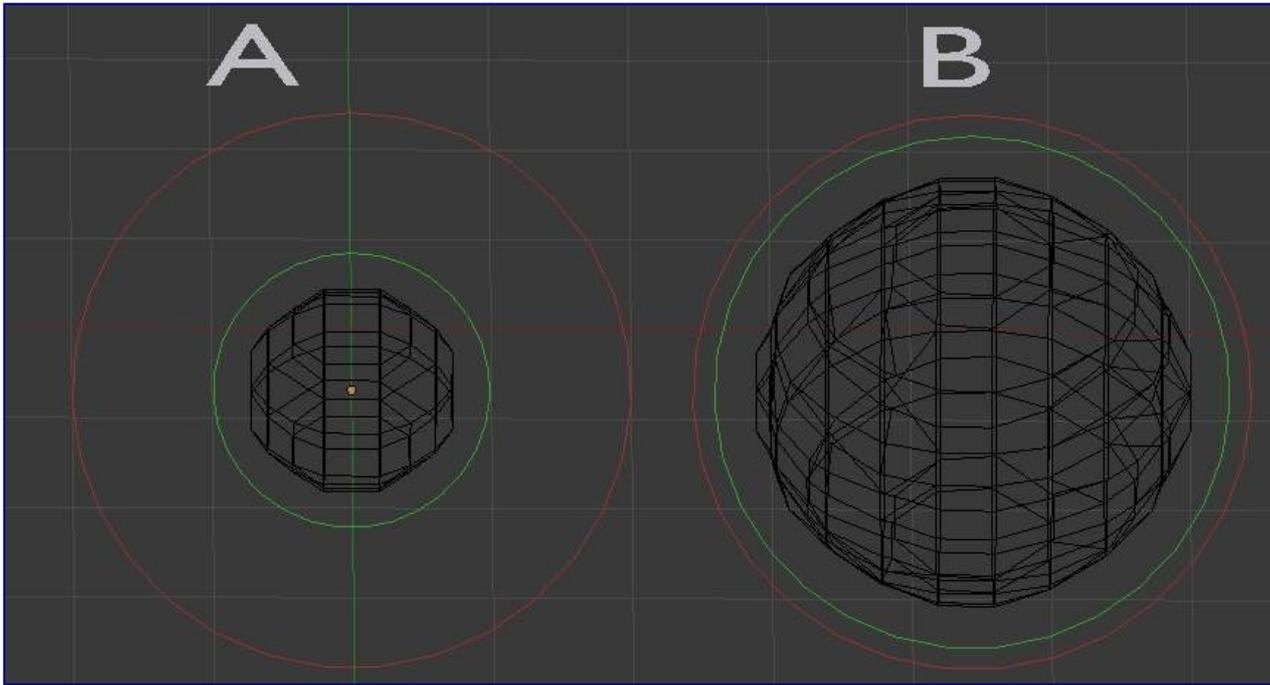
Stiffness

Together with *Threshold*, *Stiffness* controls the influencing range. While the threshold is common to all metas in the same object (or even the same Object Families), the stiffness is specific to each meta.

Scaling the inner green circle changes the *Stiffness* value. Stiffness defines how much the meta object is filled. This essentially defines how sensitive a meta is to being affected by other metas. With a low stiffness, the meta will begin to deform from further away. A higher value means the meta needs to be close to another one to begin merging.

When a *Meta* object comes within “range” of another meta, the two will begin to interact with each other. They don’t necessarily need to intersect, and depending on the *Threshold* and *Stiffness* settings, they most likely won’t need to. *Stiffness* is materialized by the *green ring*

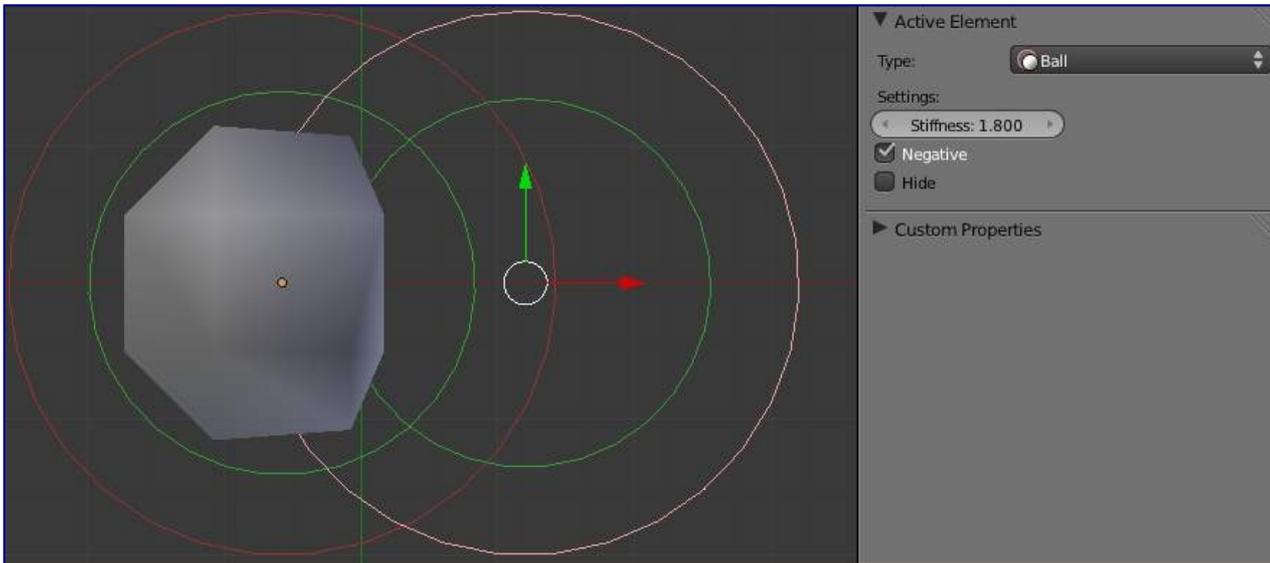
The range is from **0.0** to **10.0**. But to be visible, the *Stiffness* must be slightly larger than the *Threshold* value. You can also visually adjust the *Stiffness* ring by using the RMB to select it and activate *Scale* mode with S.



Stiffness.

In (*Stiffness*), the meta ball labeled A, has a smaller *Stiffness* value than the one labeled B. As you can see, the *green ring* radius is different for each of them.

Negative Influence



Negative.

The opposite effect of a *positive* influence would be a *negative* influence: the objects repel each other. (*Negative*) shows a meta ball and a meta plane where the first is negative and the second, positive. Notice how the negative meta is not visible: only the surrounding circles appear. This is how Bforartists indicates that the object is negative.

Moving the sphere to the plane causes the plane's mesh to “cave in” or collapse inward. If you move the plane away from the sphere, the plane's mesh will restore itself.

To make a meta *negative*, just select the meta in edit mode, and check *negative* in the *active element* panel.

Hiding Elements

As in *Object* mode, you can hide the selected meta(s), and then reveal what was hidden. This is very handy for cleaning your views up a bit... Note that the two red and green rings always remain visible in *Edit* mode, as well as the select circle (in *Object* mode...).

To hide the current selection, use **H**, the *Hide* toggle button in the *MetaBall tools*, or the Metaball ▸ Hide MetaElems ▸ Hide Selected menu option.

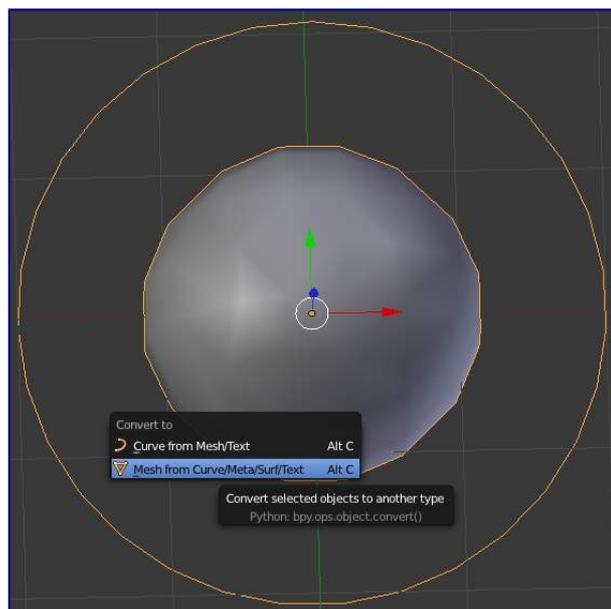
To hide everything but the current selection, press **Shift - H** or use Metaball ▸ Hide MetaElems ▸ Hide Deselected.

To reveal what was hidden, use **Alt - H**, or the relevant option in the same Metaball ▸ Hide MetaElems menu. You can also un-toggle the *Hide* button in the (*MetaBall tools* panel).

Deleting Elements

There is no *Erase* menu for metas, just a confirmation pop-up asking you if you want to delete the selected metas. Clear and simple!

Conversion



the convert menu

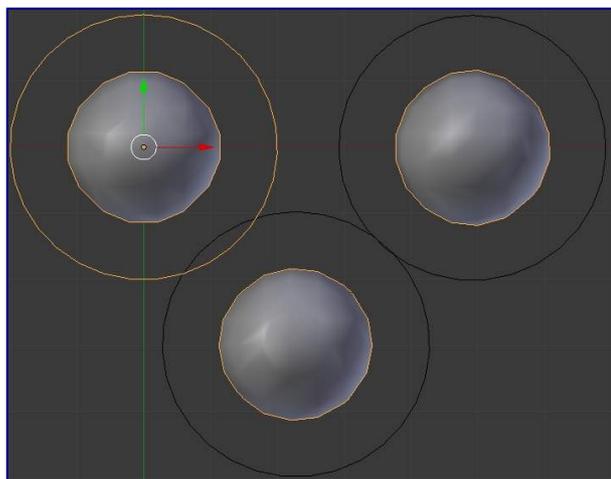
You can only convert metas to meshes, but here you have the option to keep the original *Meta* object (i.e. create a new *Mesh* one, instead of a “real” conversion...). Note that the resolution used for the new mesh is the *Wiresize* one, not the *Renderize* one.

Object Families

Meta objects have different behavior in *Object* mode than other object types - they can be “regrouped” into so-called “families”.

A “family” is a way to regroup several meta objects, producing something very similar to having several metas inside the same object.

A family is defined by the left part of an object’s name (the one before the dot). Remember, an object’s name is the one in the *OB* field, in most panels, **not** the *MB* field, which is the meta data-block’s name... For example, the *family* part of *MetaPlane.001* is *MetaPlane*. Each meta object in the same “family” is associated with one another as discussed below.



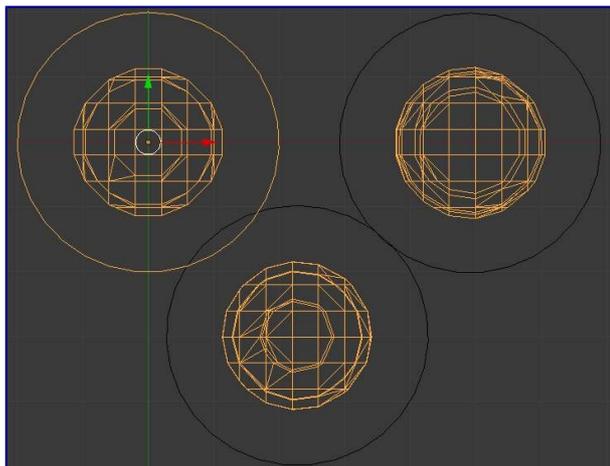
Meta ball base.

Families of metas are controlled by a *base Meta* object which is identified by an *Object* name **without** a right part. For example, if we have five metas called *MetaThing*”, *MetaThing.001*, *MetaThing.002*, *MetaThing.003* and *MetaThing.004*, the *base Meta* object would be *MetaThing*.

The *base Meta* object determines the basis, the resolution, the threshold, *and* the transformations. It also has the material and texture area. The *base meta* is effectively the parent of (or perhaps a better word to use is “the owner of”) the other metas in the group (i.e. it is as if the other metas were “included” or joined into the base one).

Examples

(*Meta ball base*) shows the *base meta* labeled *B*. The other two *Meta* objects are *children*. Children’s selection rings are always black, while the group’s mesh is orange. Because the metas are grouped, they form a unified mesh which can always be selected by selecting the mesh of any meta in the group. For example, in the example (*Meta ball base*), only the lower sphere (the parent) has been selected, and you see that both the parent’s mesh *and* all of the children’s meshes are now highlighted.



Scaling the “base”.

The *base Meta* object controls the **polygonalization** (mesh structure) for the group, and as such, also controls the polygonalization for the children (*non-base*) metas. If we transform the *base* meta, the children’s polygonalization changes. However, if we transform the children, the polygonalization remains unchanged.

Hints

This discussion of “polygonization” *doesn’t* mean that the various meshes don’t deform towards or away from each other (meta objects always influence one another in the usual way, whether or not they are members of the same family). Rather, it means that the underlying mesh structure changes only when the *base* object transforms. For example, if you scale the *base*, the children’s mesh structure changes. In (*Scaling the “base”*), the *base* has been scaled down, which has the effect of scaling the mesh structure of each of the children. As you can see, the children’s mesh resolution has increased, while the *base* decreased. *The children did not change size!*