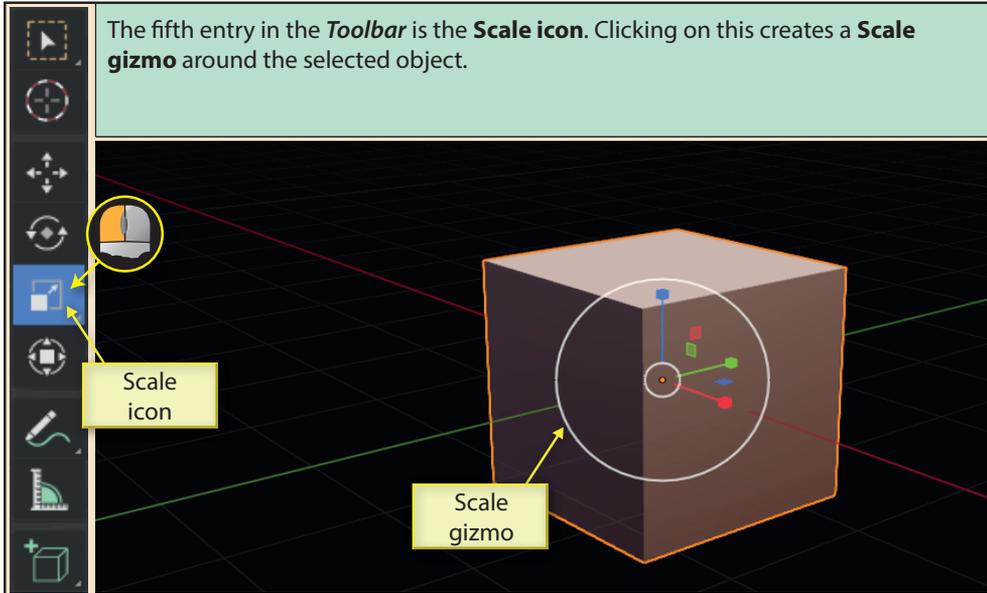


Object Scaling

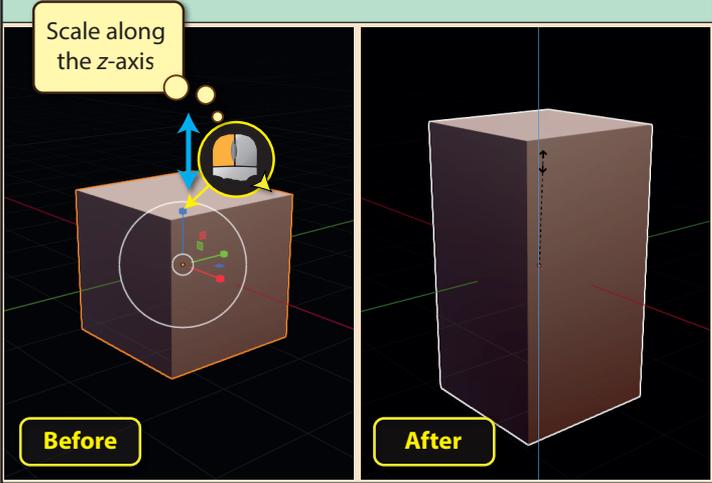
Although we can specify the size of an object when it is first created, we can also resize it later in the design process. We have options to scale the object equally in all directions or to adjust its size along one or two directions only.

Both a *Toolbar* icon and keyboard shortcuts are available to initiate scaling.

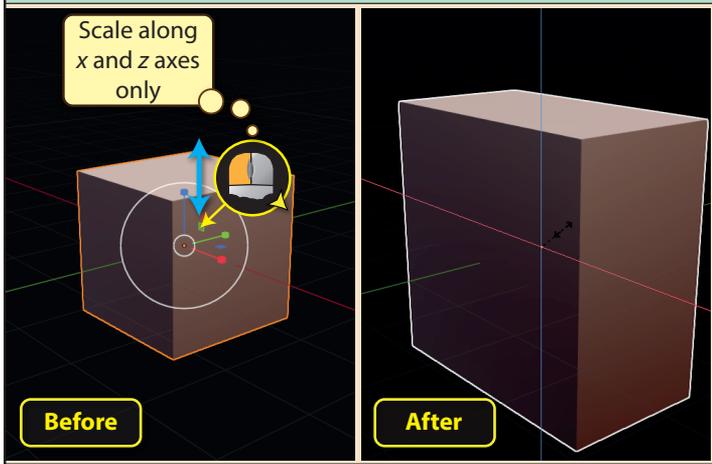
The fifth entry in the *Toolbar* is the **Scale icon**. Clicking on this creates a **Scale gizmo** around the selected object.



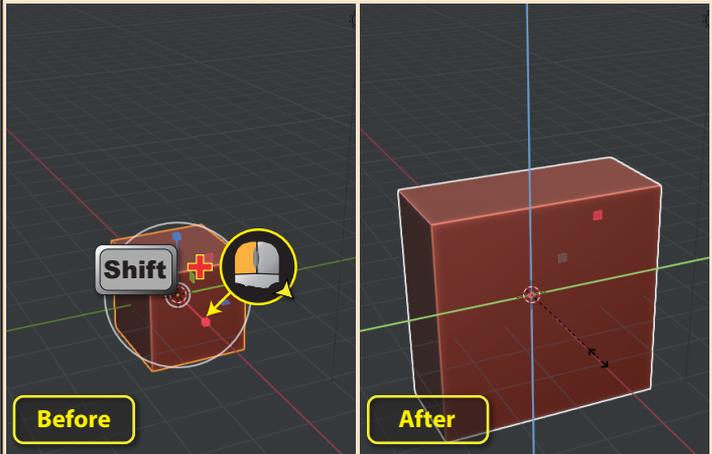
Dragging on one of the coloured lines of the *Scale gizmo* resizes the object along the corresponding axis.



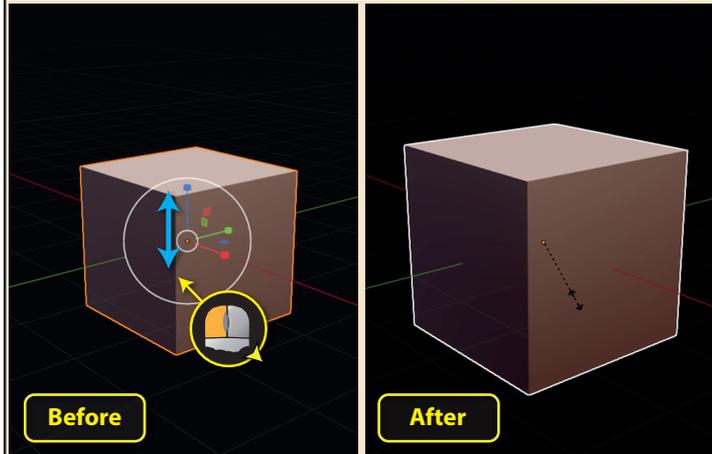
Dragging on one of the discreet coloured rectangles scales the object in all but the direction indicated by the rectangle's colour (e.g. dragging on the green square would scale the object in the X and Z directions only).



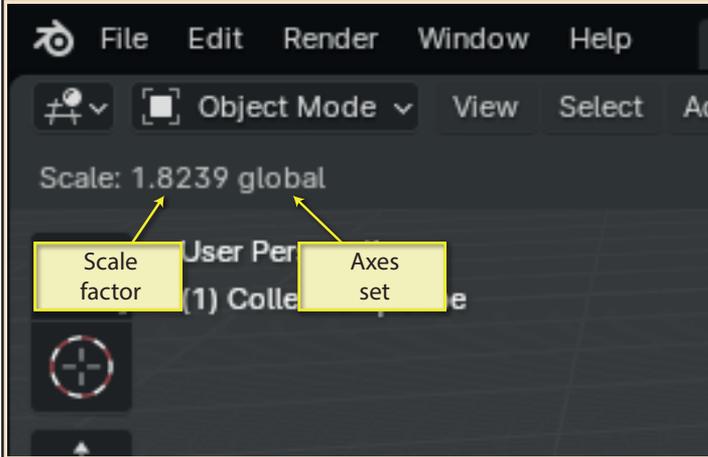
The same effect can be achieved by holding down the **Shift** key and dragging on one of the lines. Below we can see the effect of holding down **Shift** while dragging on the red line.



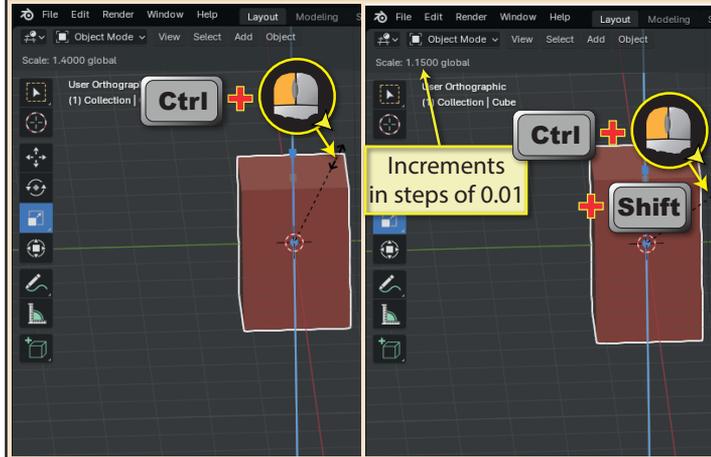
Dragging in an empty area between the two white circles of the gizmo scales the object equally in all three dimensions.



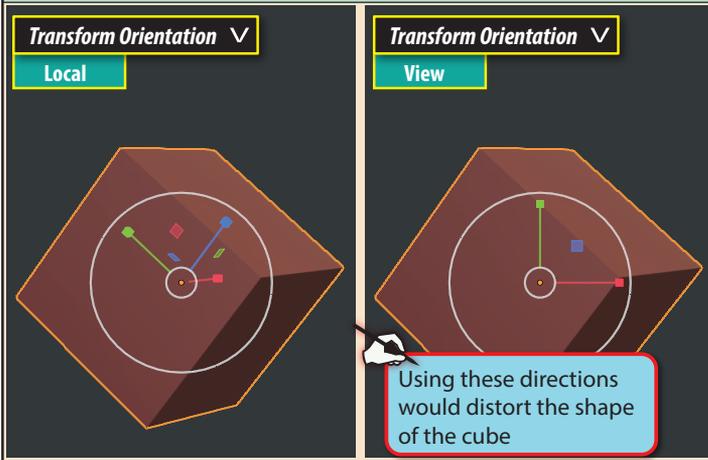
As we drag to change the size of the selected object, in the top-left corner of the **3D Viewport** we can see the current change in scale and the axes set that we are scaling along.



If, while scaling, we hold down the **Ctrl** key, scaling will be in incremental steps of **0.1**. And if we then include holding down the **Shift** key, scaling will reduce to steps of **0.01**.



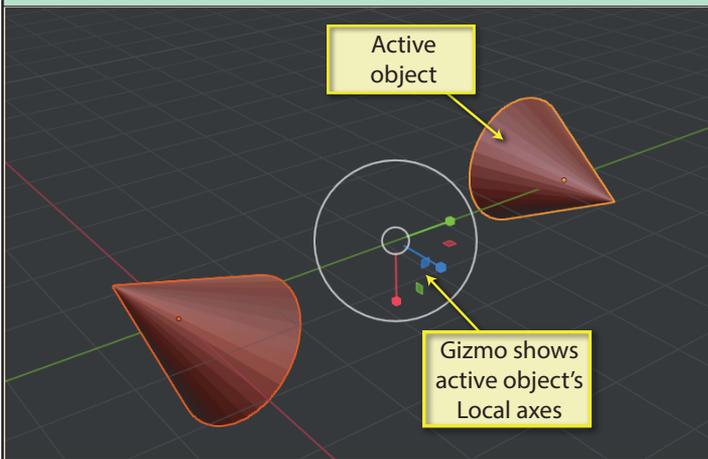
If we first rotate an object so that its **Global** and **Local** axes are no longer in alignment, then move to the top-middle of the **3D Viewport** and change the **Transform Orientation** to **Local** or **View**, we'll see the coloured lines of the **Scale Gizmo** change orientation.



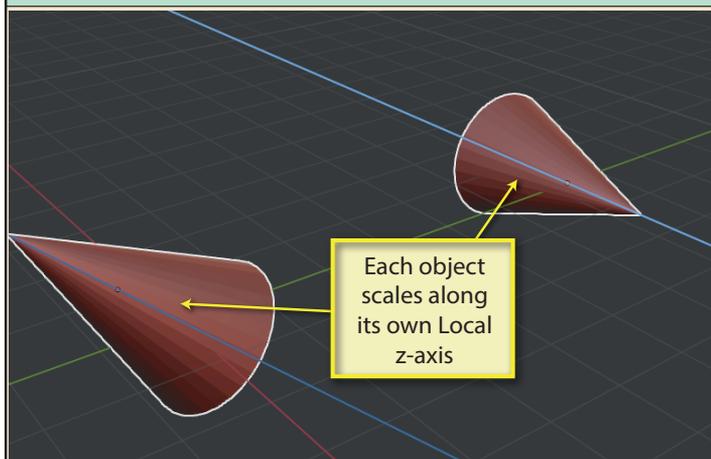
When we select the **Scaling gizmo**, an **Orientation** field appears in the top-left of the **3D Viewport**. The value here (when other than **Default**) overrides any given in the **Transform Orientation** field. For example, although **Transform Orientation** is set to **Global**, with the **Orientation** field set to **Local**, it's the **Local** axes that appear in the Scaling Gizmo.



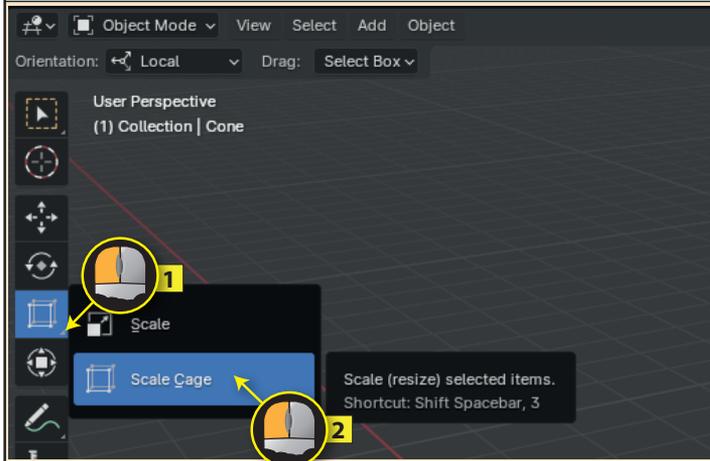
If more than one object is selected when using the **Local** axes for scaling, the **Scaling gizmo** displays the **Local** axes of the **active object**.



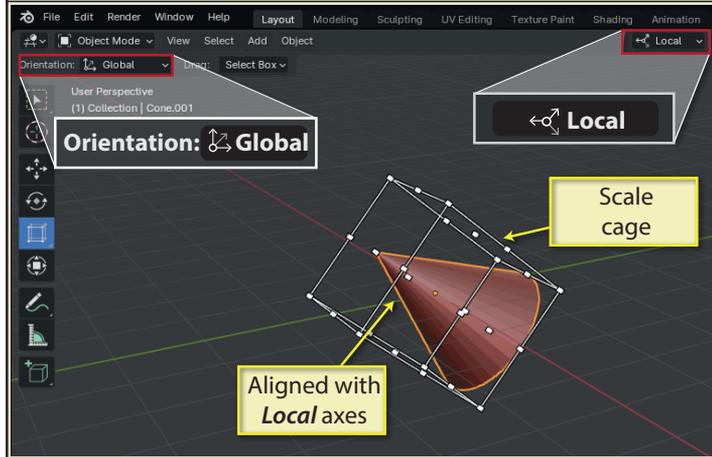
However, when we actually perform the scaling, each selected object scales along its own **Local** axes set.



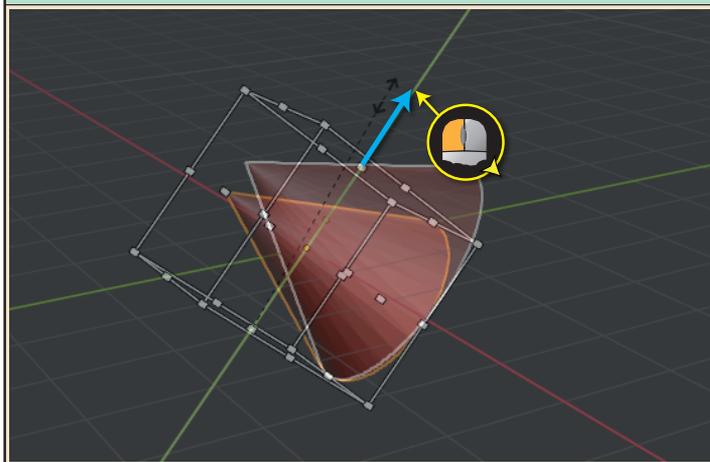
A second scaling tool is available from the *Toolbar*. This is known as the **Scale Cage**.



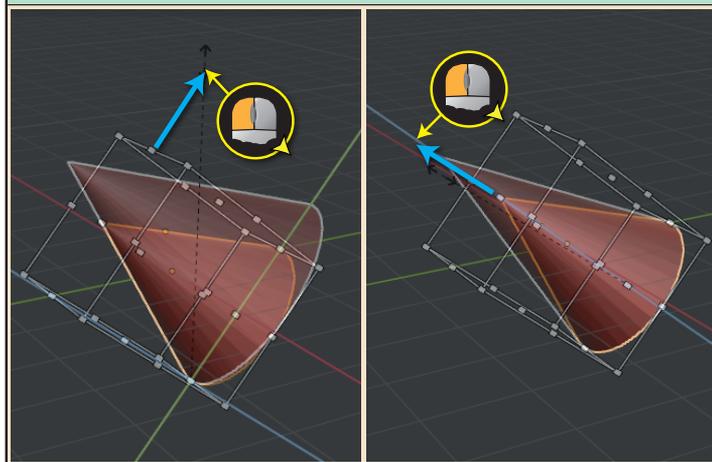
This creates a box, or cage around the selected object. The alignment of the cage is controlled by the **Transform Orientation** (the *Orientation* field being ignored - at least in the current version of Blender).



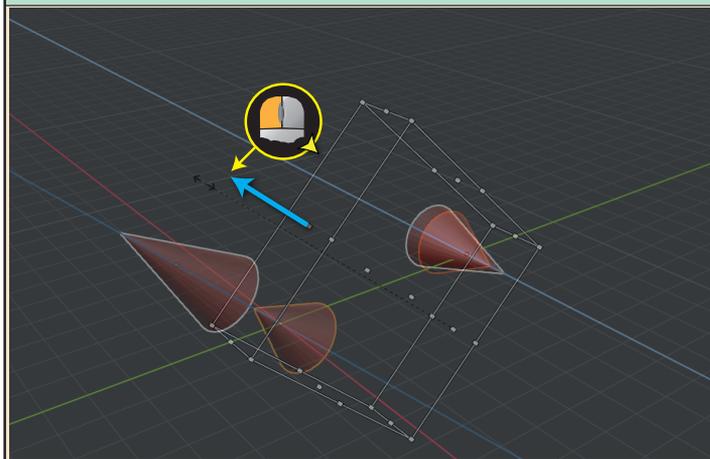
The cage has white control points in the middle of faces, the middle of edges, and on each vertex. Dragging of a face control point will scale the object along the corresponding axis.



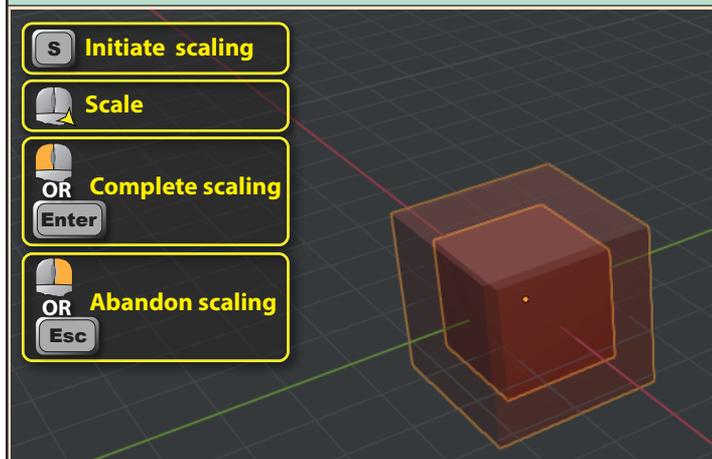
Dragging on an edge control point, scales in two directions and dragging on a face control point scales in one direction.



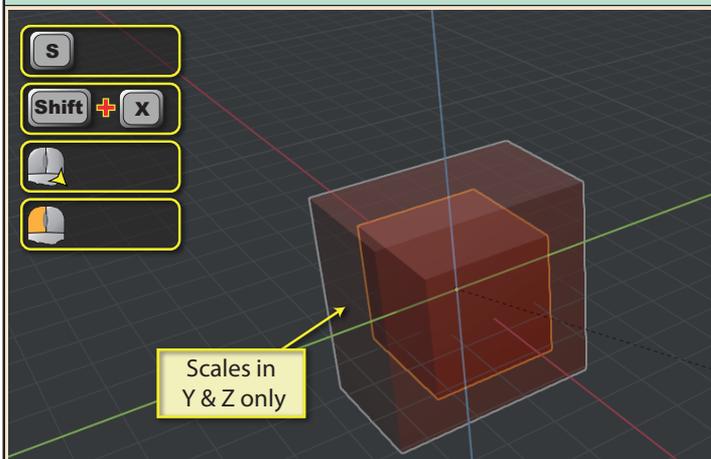
Like the *Scaling* gizmo, the *Scale Cage* displays along the axes of the *active object* when more than one object is selected, but scales the objects along their own axes when scaling begins.



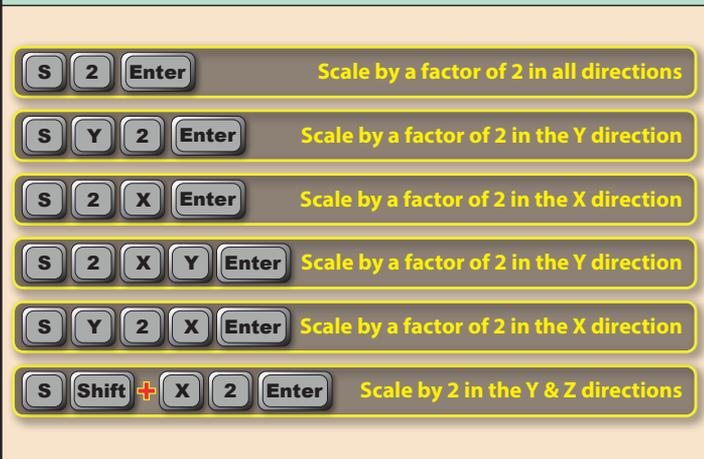
As an alternative to using the *Toolbar*, we can scale the selected object by pressing the **S** key and moving the mouse. A left click or **Enter** key completes the scaling. To cancel the scaling, click the right mouse button or press the **Esc** key.



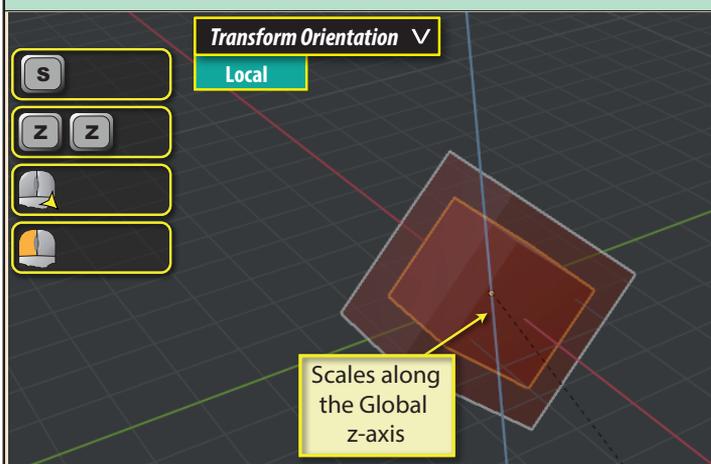
To scale in two directions using the keyboard, hold down the **Shift** key while entering the axis letter. For example, **S Shift X** would scale in the **Y** and **Z** directions only.



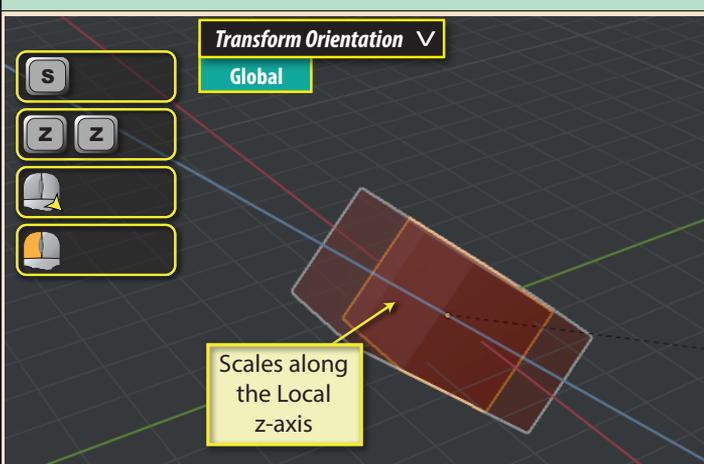
The **S** key can be followed by an axis letter (X, Y or Z) and a numeric scaling factor. The only restriction in the key order is that the **S** comes first. Scaling will be along the axes set given in *Transform Orientation*. The examples below are all valid.



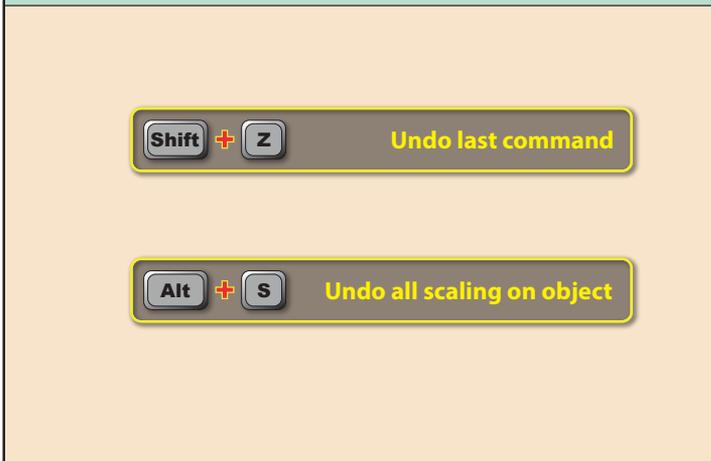
With *Transform Orientation* set to anything other than *Global*, pressing the axis letter twice will force the scaling to be along the corresponding *Global* axis.



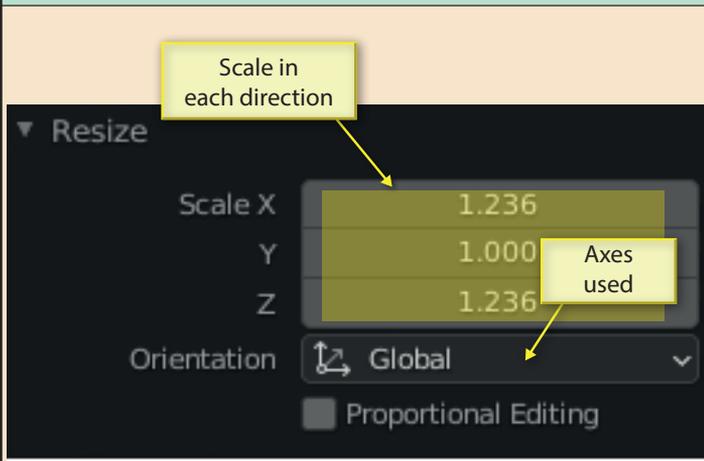
However, if the *Transform Orientation* is already set to *Global*, the double letter will cause the scaling to be along the *Local* axis.



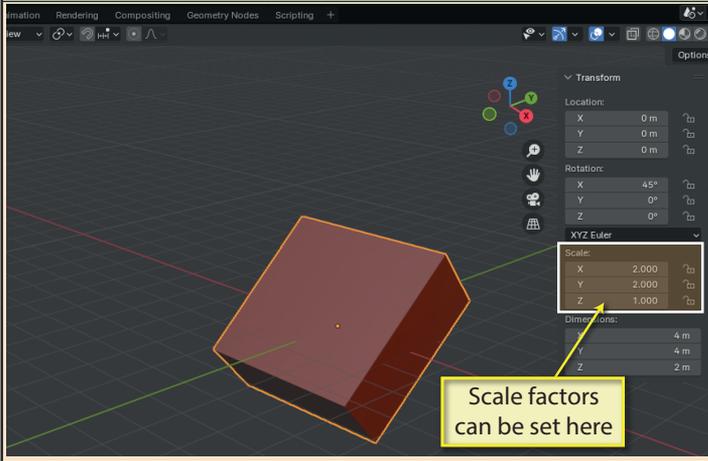
If we've already completed a scaling command that we now want to undo, we can enter **Ctrl Z** which will undo the last command. But a more powerful option is to enter **Alt S** which undoes ALL scaling that has been performed on an object.



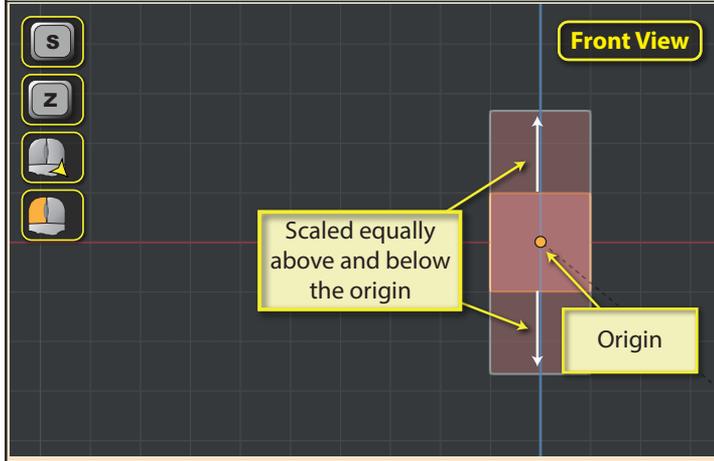
Immediately after performing the scaling, the *Last Op* panel at the bottom-left of the *3D Viewport* allows us to adjust the scaling along each axis and also to choose which set of axes are being used when scaling (this overrides the previously selected axes set).



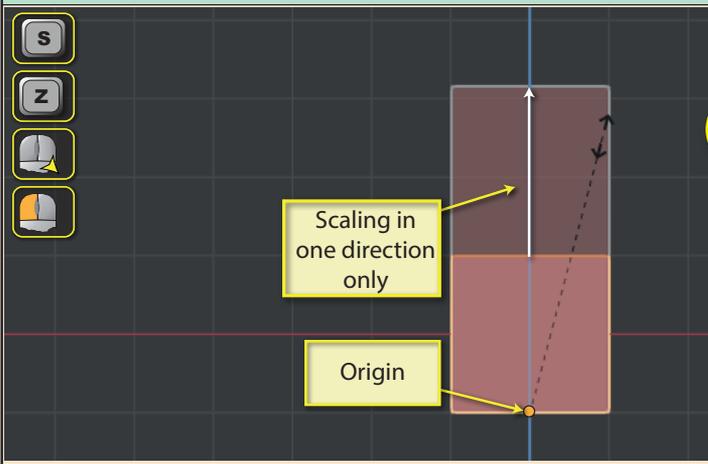
We can also specify a scaling factor for each dimension in the *Item* page of the *Sidebar*. Values entered here are implemented along the selected object's *Local* axes irrespective of any other settings.



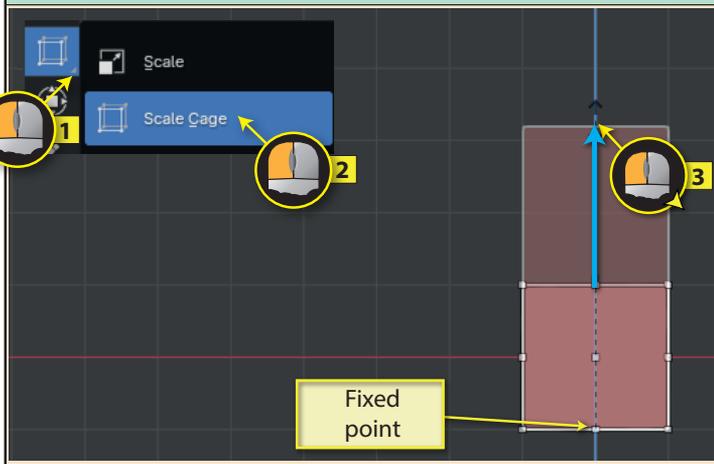
Normally, scaling is applied equally in all specified directions when the origin is at the middle of an object...



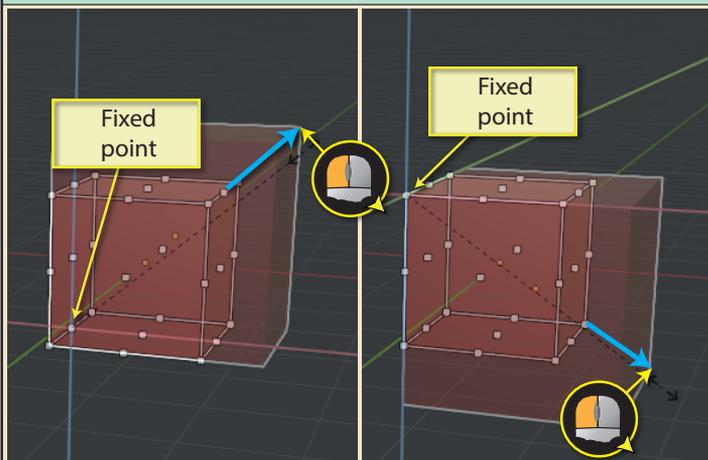
...but if we move the origin, then this will affect how the object is scaled.



If we want the object to scale from a point on its edge, but don't want to move its origin, we can do this by using the *Scale Cage*. When we drag on a control point of the cage, the opposite point remains in a fixed position.



Even when the controlpoint used is on an edge or on a vertex, we still have a fixed point.



When we have resized our object, we must apply the scaling. If we fail to do this we'll run into problems at later stages of our modelling process.

