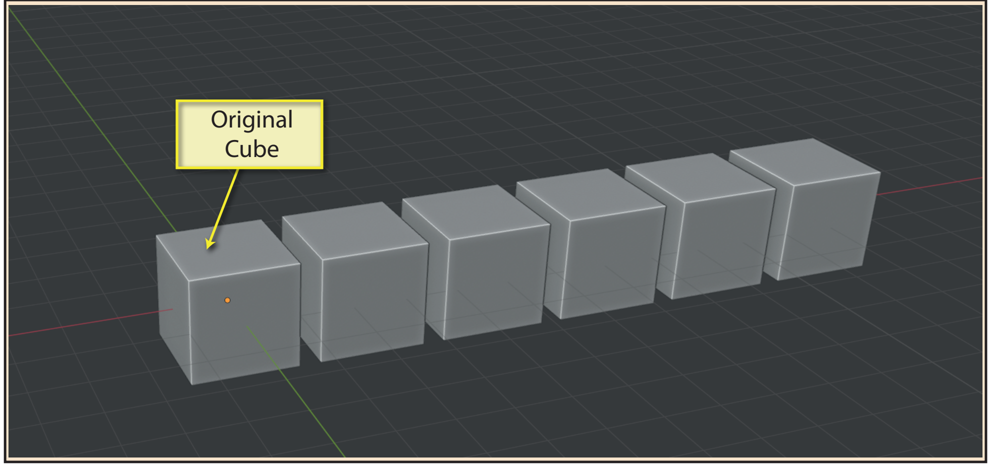
The Array Modifier

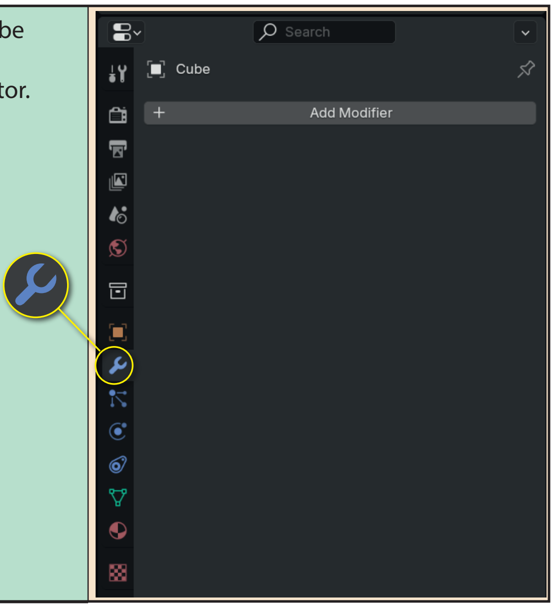
A modifier adjusts the look of a scene object "on the fly". A modifier can be deleted at any time, making the effect it creates vanish.

Normal changes to an object cannot be undone once the project is saved and hence these types of changes are known as "destructive" operations. However, since we can always remove the effects of a modifier, its effects are termed "non-destructive".

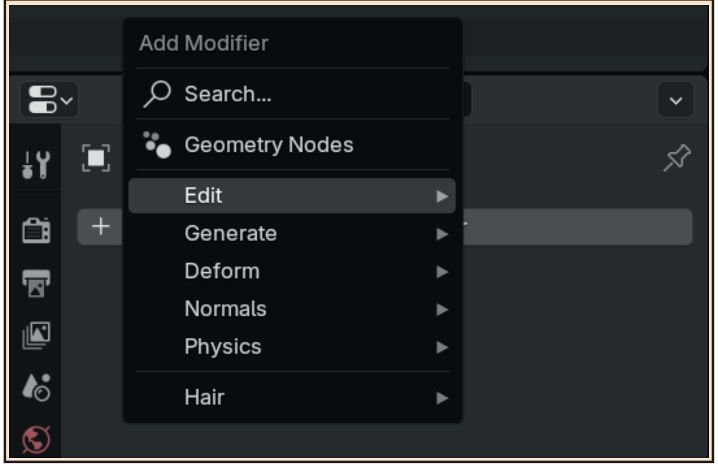
The **Array modifier** allows us to create many copies of the object to which it applied as well as control the positioning of these duplicates. Below we can see the results of making 5 copies of a Cube.



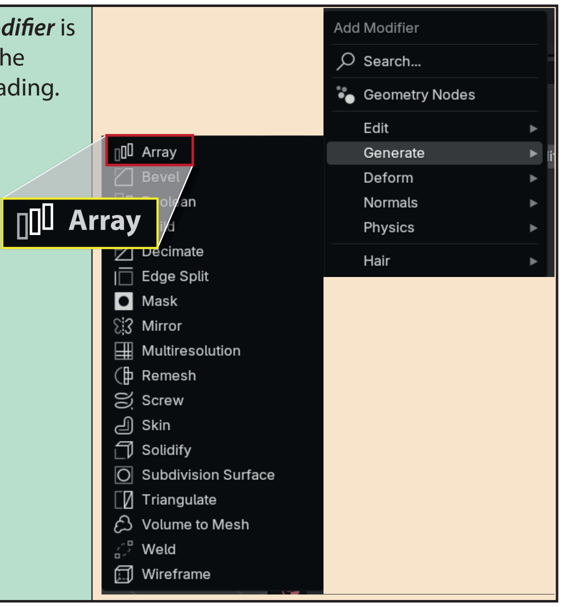
Modifiers can be found in the Properties Editor.



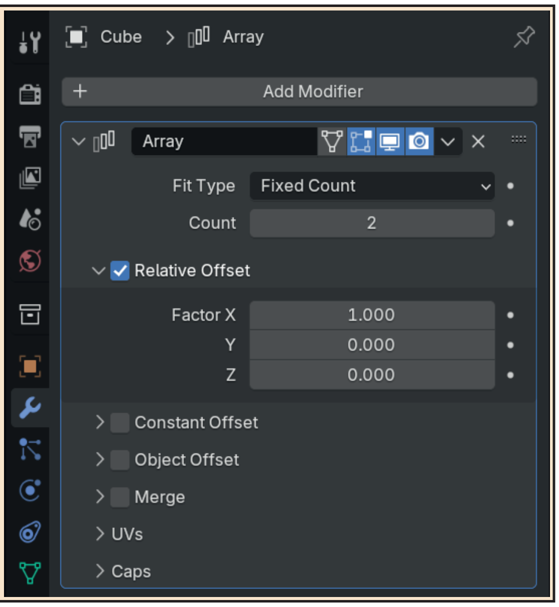
WIth the object in the scene selected, we need to click on **Add Modifier** to produce a list of modifier groupings.



The *Array modifier* is listed under the **Generate** heading.

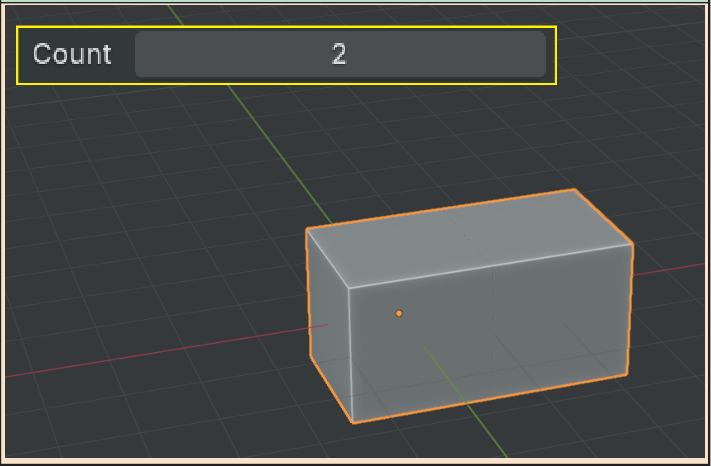


Once we've clicked on **Array**, the ***Modifier page*** displays the various options available for this modifier.



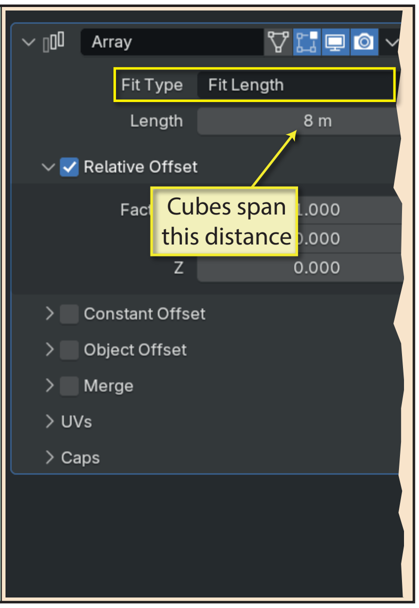
Fit Type: Fixed Count
Count: 2
✓ Relative Offset
Factor X: 1.000
Y: 0.000
Z: 0.000

Count determines the number of copies made, but this value also includes the original object. So, with the default value, 2, only one copy of the Cube is made.

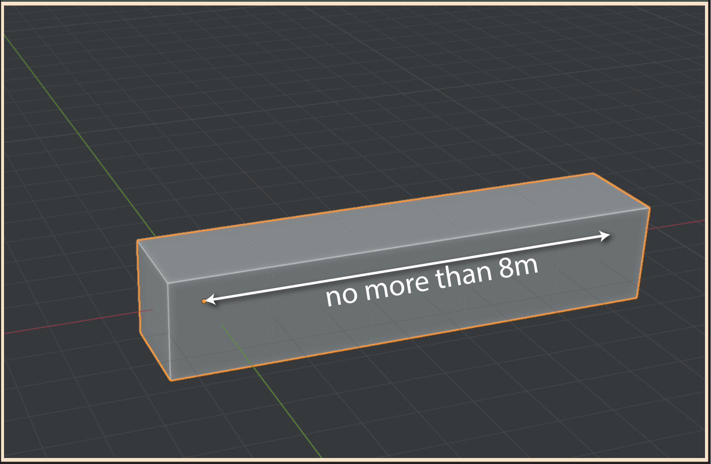

Count 2

Moving back to **Fit Type**, this drop down list determines how the number of copies is calculated. The default, **Fixed Count**, creates the number of copies needed to match the value in **Count**.

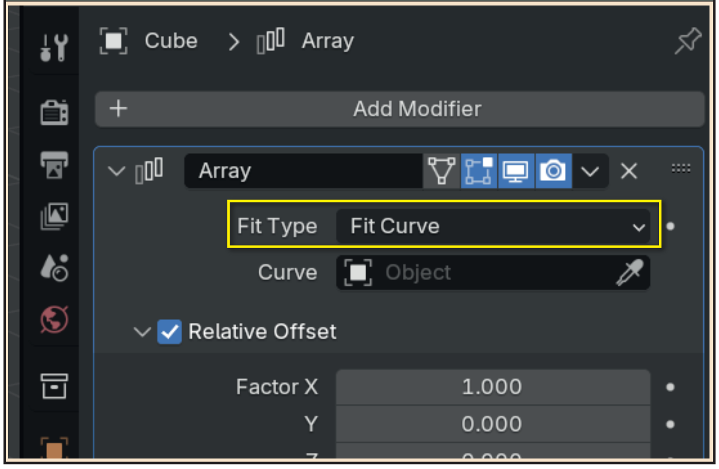
But if we choose **Fit Length**, the next field changes from ***Count*** to **Length**, where we must specify the length over which the original and its copies must stretch.


Fit Type: Fit Length
Length: 8 m
Relative Offset ✓

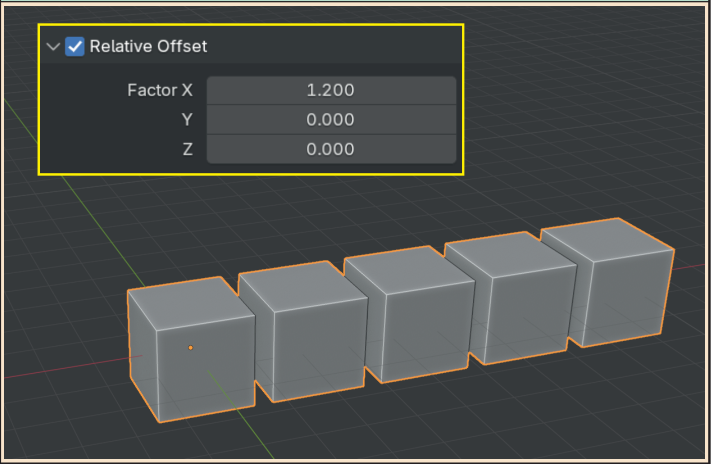
The **Length** value is measured from the middle of the origin to the middle of the final copy. The actual length of this distance will not exceed the entered value.



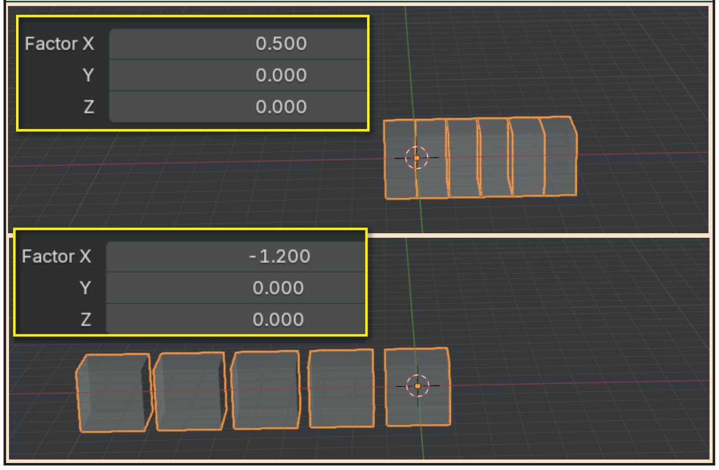
The final option for **Fit Type** is ***Fit Curve*** which adds copies to a maximum length equal to the length of a Curve object. But since we have yet to cover Curves, we'll ignore this option for now.


Cube > Array
Fit Type: Fit Curve
Curve: Object
Relative Offset ✓
Factor X 1.000
Y 0.000

The ***Array modifier's*** **Relative Offset** defines the distance between the middle of each Cube relative to its length in that direction. For example, if **Factor X** is set to 1.2 the space between the centre of each copy is 120% of the width of the Cube.

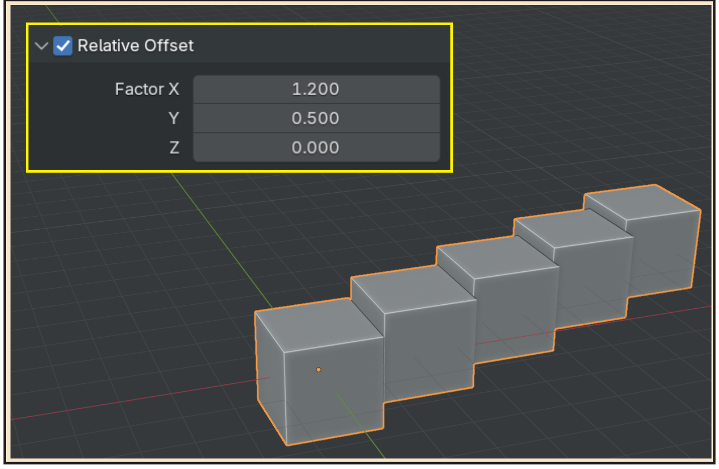

Relative Offset ✓
Factor X 1.200
Y 0.000
Z 0.000

A value between 0.0 and1.0 will cause the Cubes to overlap. And a negative value will cause the Cubes to position in the opposite direction.


Factor X 0.500
Y 0.000
Z 0.000

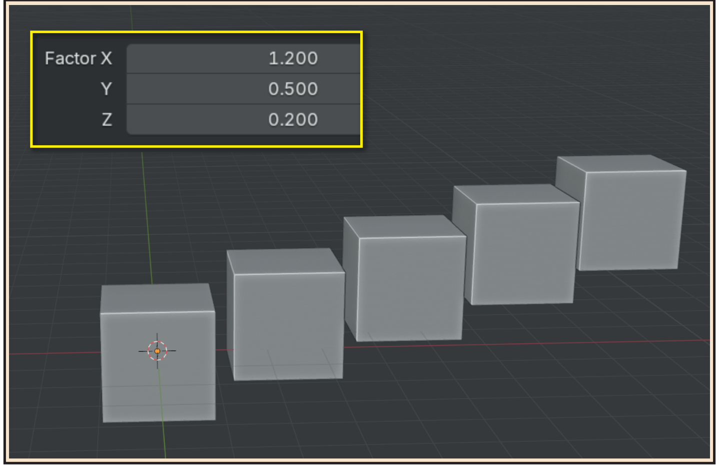
Factor X -1.200
Y 0.000
Z 0.000

And if we add a **Factor Y** value of **0.5**, the Cubes also displace by 1m in the Y direction.


Relative Offset

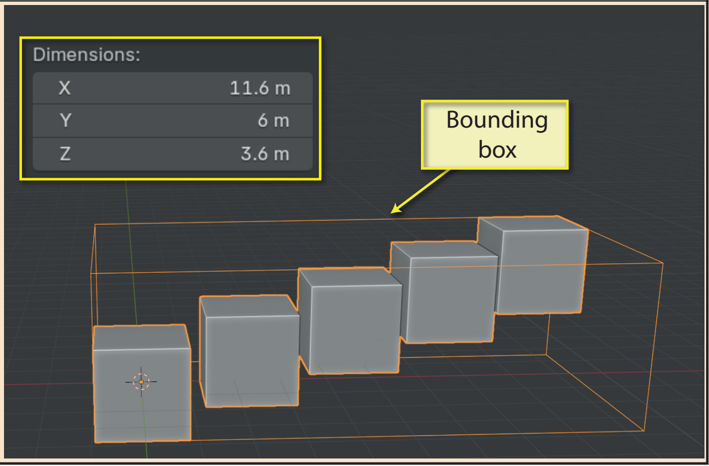
Factor X	1.200
Y	0.500
Z	0.000

Of course, adding a Z value will adjust the height of the copies.



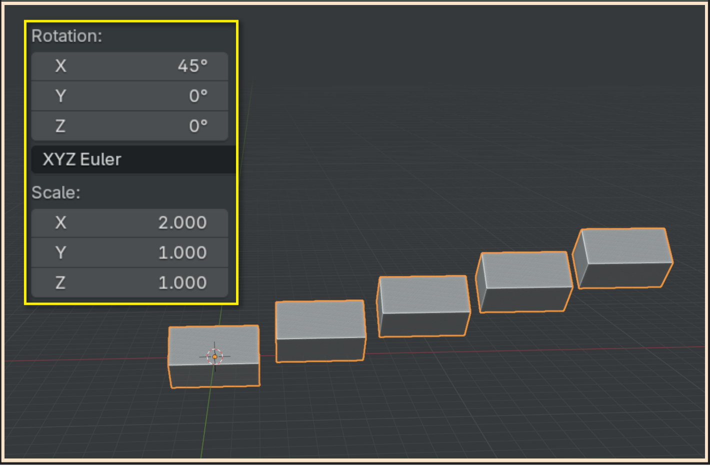
Factor X	1.200
Y	0.500
Z	0.200

Looking at the dimensions of the Cube in the ***Sidebar***, we can see that it includes the copies that have been created. In fact, the dimensions given represent the size of the bounding box that includes all 5 Cubes.


Dimensions:

X	11.6 m
Y	6 m
Z	3.6 m

Changing the scale or rotation of the original Cube will also effect the copies. Below we can see the result of setting the scale in the X direction to 2.0 and rotating the Cube 45° about its x-axis.


Rotation:

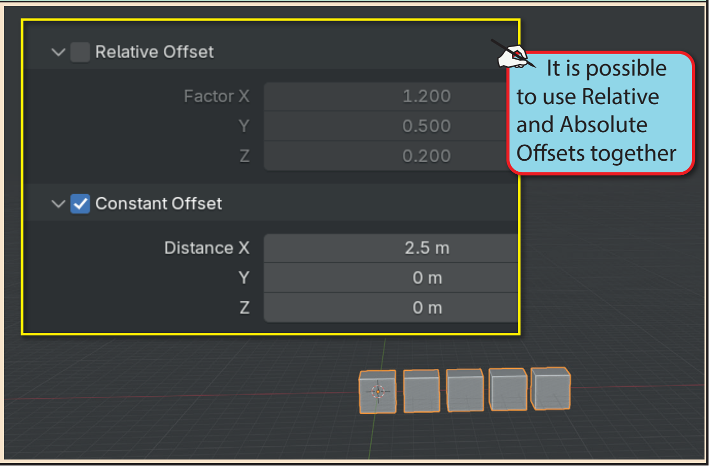
X	45°
Y	0°
Z	0°

XYZ Euler

Scale:

X	2.000
Y	1.000
Z	1.000

Returning to the Array modifier's page in the Properties Editor, we can use **Constant Offset** as an alternative to ***Relative Offset***. Here we can specify absolute distances rather than ones dependent on the dimensions of the Cube.


Relative Offset

Factor X	1.200
Y	0.500
Z	0.200

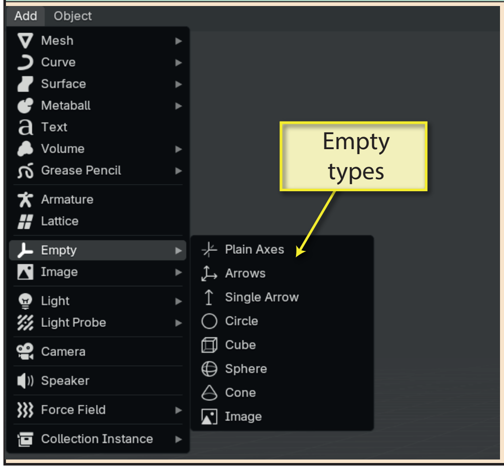
Constant Offset

Distance X	2.5 m
Y	0 m
Z	0 m

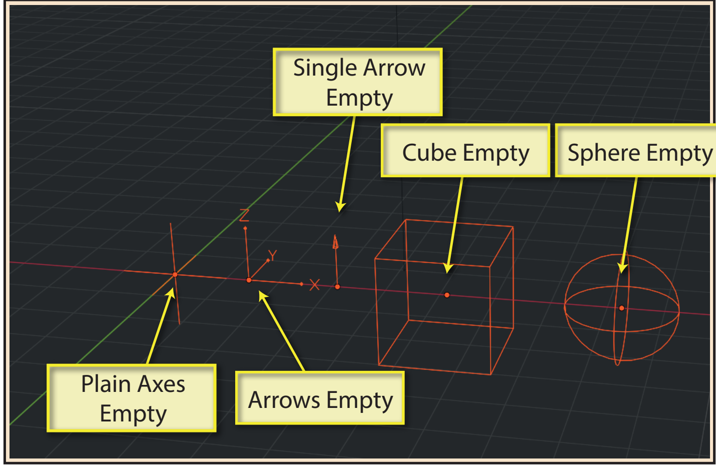
It is possible to use Relative and Absolute Offsets together

A less intuitive, but more powerful way of controlling the positioning of the copies we make is to use the **Object Offset** option where a second object controls the characteristics of the copies.

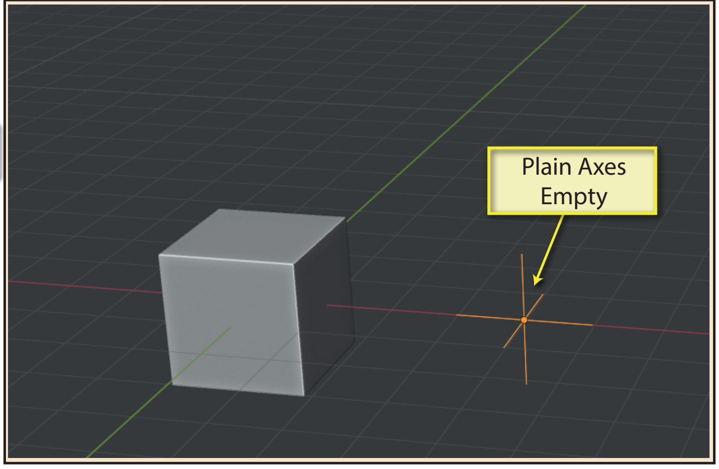
Although any object can be used for this task, the most useful are **Empty** objects (`Add>Empty`).



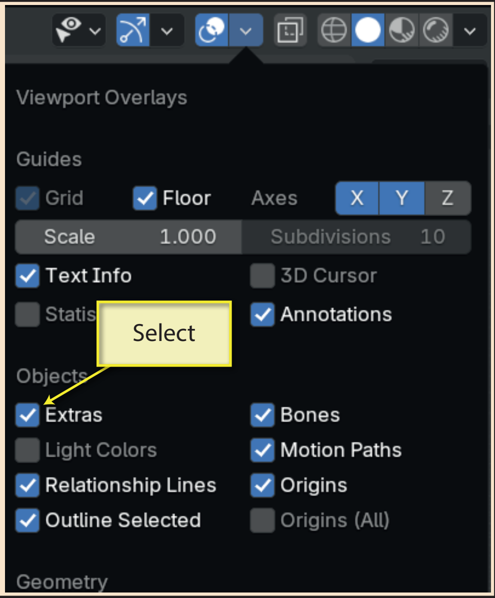
Although we have a choice of **Empty** types, in reality all of these are used to represent a single point in 3D space. An **Empty** will never appear in a rendered image and the various types only differ in how they are shown in the ***3D Viewport***.



Often an ***Empty*** is used in controlling certain types of animation, but it can also be used here to control the positioning of copies made using the ***Array*** modifier. The first step is to add a ***Plain Axes Empty*** to our scene, moving it along along the x-axis.



If the ***Empty*** doesn't appear in the ***3D Viewport***, it's because the **Extras** checkbox isn't selected in the ***Viewport Overlays*** panel.


Viewport Overlays

Guides

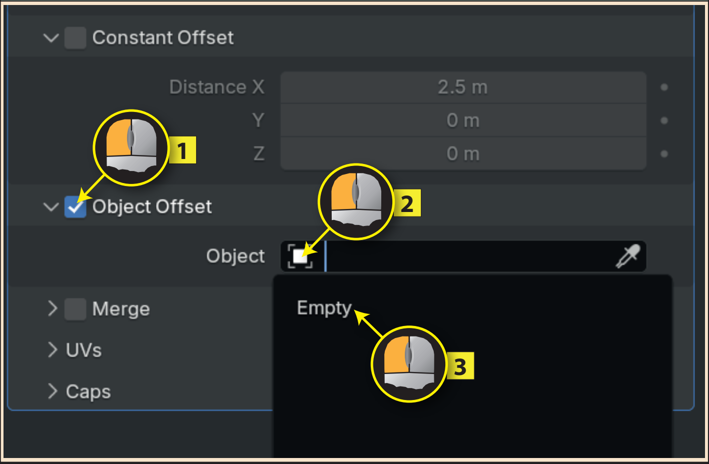
- ✓ Grid
- ✓ Floor
- Axes: X Y Z
- Scale 1.000
- Subdivisions 10
- ✓ Text Info
- ☐ 3D Cursor
- ☐ Statistics
- ✓ Annotations

Objects

- ✓ Extras
- ✓ Bones
- ☐ Light Colors
- ✓ Motion Paths
- ✓ Relationship Lines
- ✓ Origins
- ✓ Outline Selected
- ☐ Origins (All)

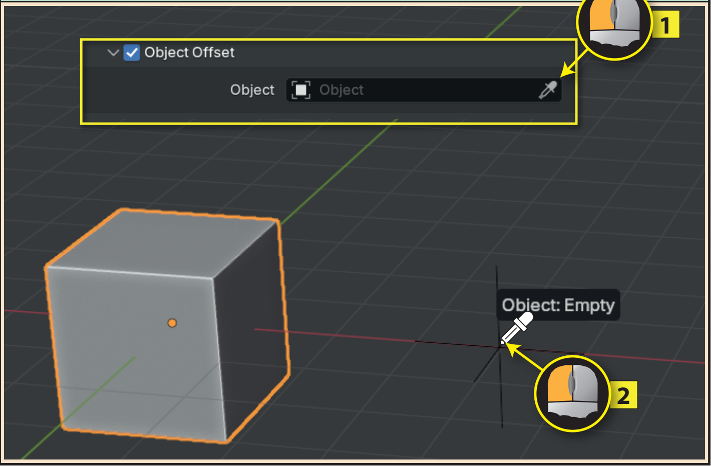
Geometry

Back on the ***Array modifier*** page, **Object Offset** offers two ways of linking the modifier to an object. Method one is to select the object from the dropdown list in the **Object** field.


- ☐ Constant Offset
  - Distance X 2.5 m
  - Y 0 m
  - Z 0 m
- ✓ Object Offset
  - Object
- ☐ Merge
- UVs
- Caps

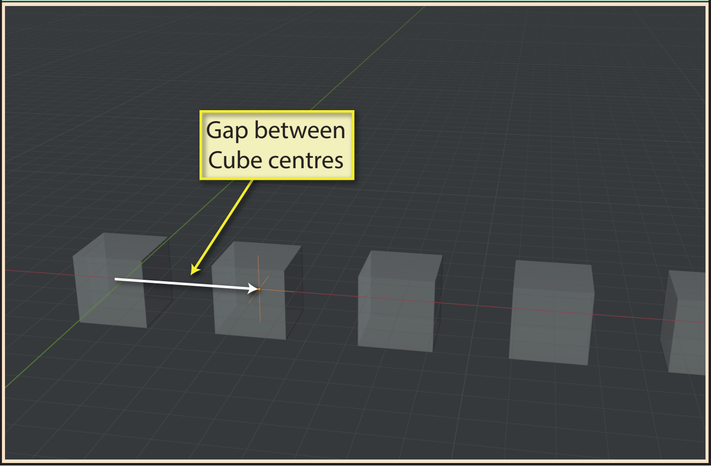
Empty

Method two is to click on the eyedropper and then select the object in the ***3D Viewport***.

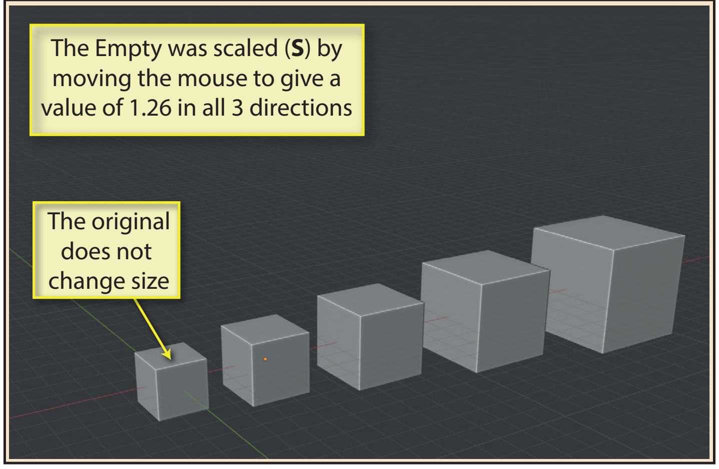

- ✓ Object Offset
  - Object: Object

Object: Empty

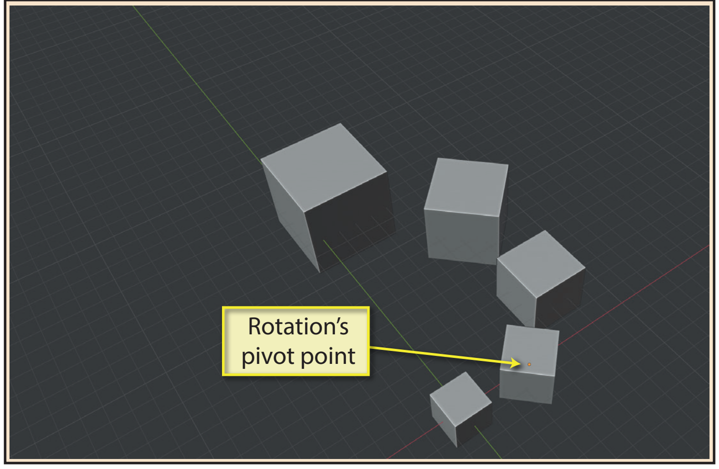
Once the ***Empty*** is selected, its distance from the original Cube's origin determines the distance between the each copy of the Cube. This means that the ***Empty*** is usually in the same location as the first copy produced by the modifier.



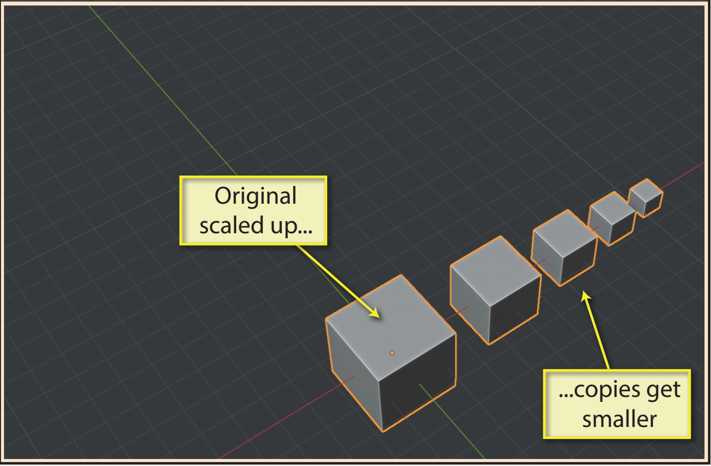
If we scale the ***Empty***, the Cube copies also scale, each one changing size relative to the previous one, creating an incremental change.


The Empty was scaled (**S**) by moving the mouse to give a value of 1.26 in all 3 directions
The original does not change size

If we rotate the ***Empty***, the copies rotate too. Rotation is about the ***Empty's*** origin. Below the ***Empty*** is rotated 45° about its z-axis. Notice that each rotates from a starting orientation of the previous Cube.



Strangely, if we scale the original Cube when its linked to the ***Empty***, the scale of the copies is inversely proportional to the scale of the original. Making the original larger makes the copies smaller!

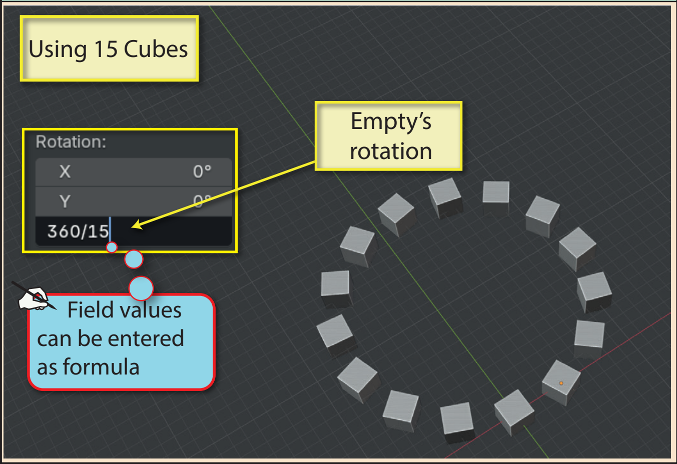


In fact, Blender seems to use the following formula to determine the size and rotation of the next copy...

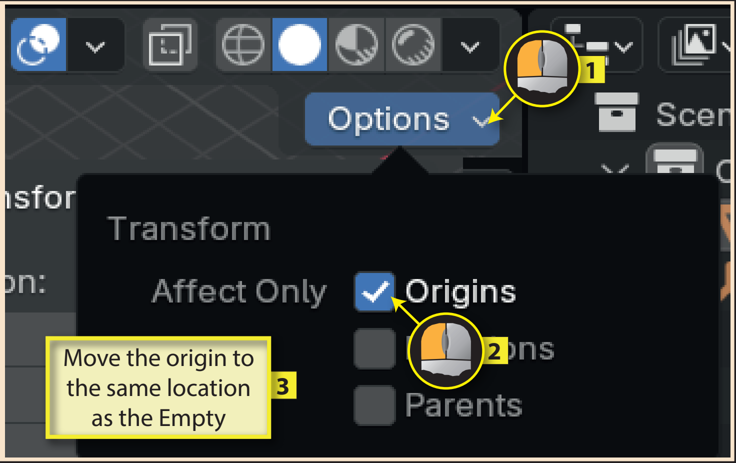
$$\textbf{Size of previous Cube} \times \frac{\textbf{Empty's scale factor}}{\textbf{Original's scale factor}}$$

Rotation of previous Cube +
Empty's rotation - Original's rotation

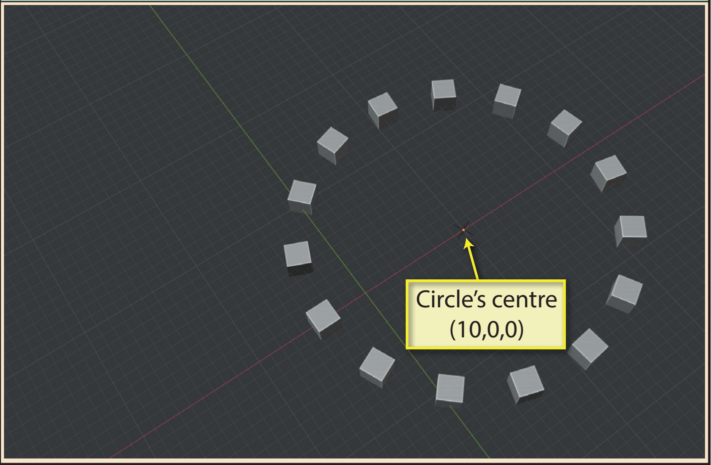
If we want the rotation of the copies to form a complete circle on the XY plane, then we need to rotate the Empty using the following formula: ***Rotation Angle = 360/Number of Cubes***.



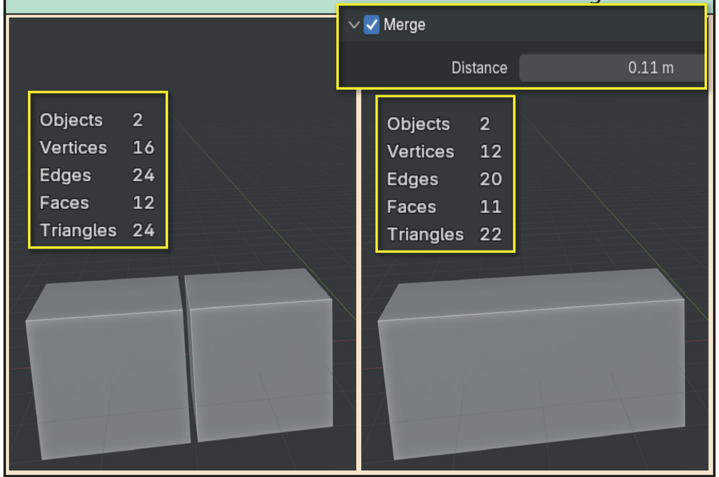
To have the centre of the created circle at the position of the ***Empty***, the ***Empty*** and the origin of the original Cube must be at the same location. Once the original Cube has been selected, we can move its origin using either **Object>Set Origin> Origin to 3D Cursor** or... (see below)



Below, we can see the result with the ***Empty*** and Cube's origin at location (10,0,0).



Next on the ***Array modifier*** page is **Merge**. This is used to merge vertices that are close to each other. For example, If the Cubes' sides are just about touching each other, we can merge nearby vertices. **Distance** controls how close vertices must be to be merged.

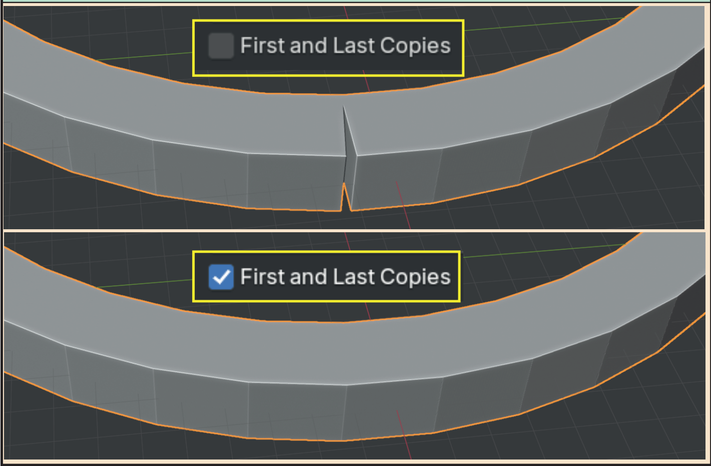


Merge ✓
Distance 0.11 m

Objects	2
Vertices	16
Edges	24
Faces	12
Triangles	24

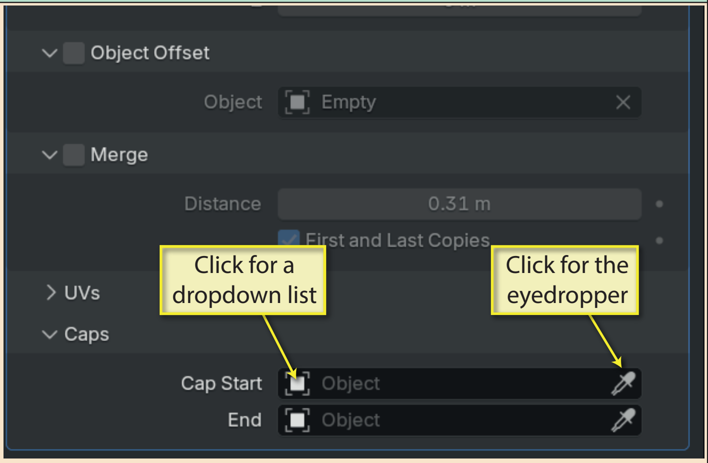
Objects	2
Vertices	12
Edges	20
Faces	11
Triangles	22

The other parameter under the **Merge** heading is the **First and Last Copies** checkbox. When a circular layout is created by the copies, Blender won't merge the vertices of the first and last copies unless this option is checked.



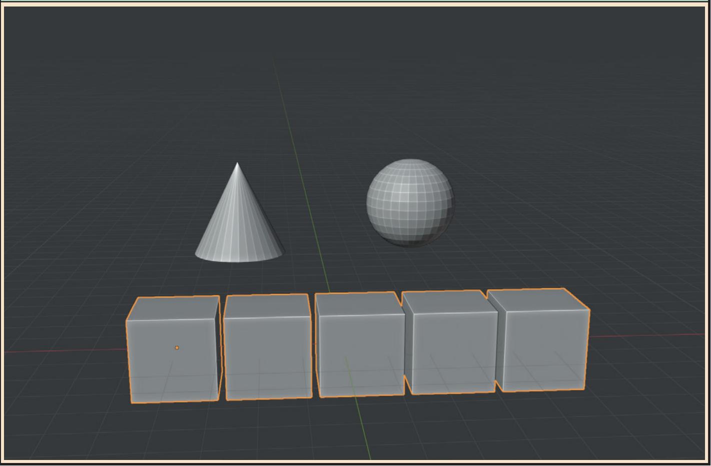
First and Last Copies
✓ First and Last Copies

UVs relates to assigning textures and is a topic for a later chapter. **Caps** is used to attach different obects to the original object (**Cap Start**) and the final copy (**End**). The objects can be selected from a dropdown list or using the eyedropper.

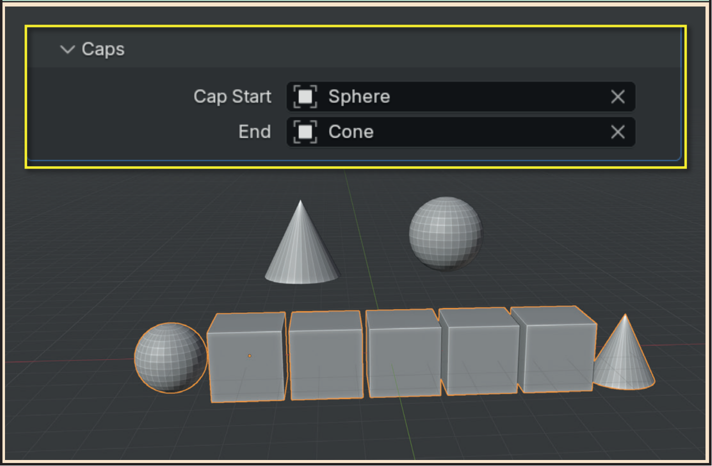


Object Offset
Object: Empty
Merge
Distance: 0.31 m
✓ First and Last Copies
UVs
Caps
Cap Start: Object
End: Object

With a Cone and UVSphere added to our scene...



... we can attach these to our original Cube and its final copy.



Caps
Cap Start: Sphere
End: Cone