

3.2.5.5.1 Editors - 3D View - Transforms - Duplication - Introduction

Duplication.....	1
Duplication.....	1
Instancing.....	2
Duplicate.....	2
Examples.....	3
Linked Duplicates.....	3
Examples.....	4
Procedural Duplication.....	4
Copying & Linking Objects Between Scenes.....	5
Linked Library Duplication.....	5

Duplication

- Duplication
 - Instancing
 - Duplicate
 - Linked Duplicates
 - Procedural Duplication
 - Copying & Linking Objects Between Scenes
 - Linked Library Duplication
- DupliVerts
 - DupliVerts as an Arranging Tool
 - DupliVerts as a Modeling Tool
 - See also
- DupliFaces
 - Basic usage
 - Scale
 - Limitations / Considerations
- DupliFrames
 - Examples
- DupliGroup
 - Basic Usage
 - DupliGroup and Dynamic Linking
 - Making a DupliGroup Object Real

Duplication

There are two types of object duplication, being Duplicate and Linked Duplicates which instance their object-data.

Instancing

Each Blender object type (mesh, lamp, curve, camera *etc.*) is composed from two parts: an *Object* and *Object Data* (sometimes abbreviated to *ObData*):

Object

Holds information about the position, rotation and size of a particular element.

Object Data

Holds everything else. For example:

Meshes: Store geometry, material list, vertex groups... etc.

Cameras: Store focal length, depth of field, sensor size... etc.

Each object has a link to its associated object-data, and a single object-data may be shared by many objects.

Duplicate

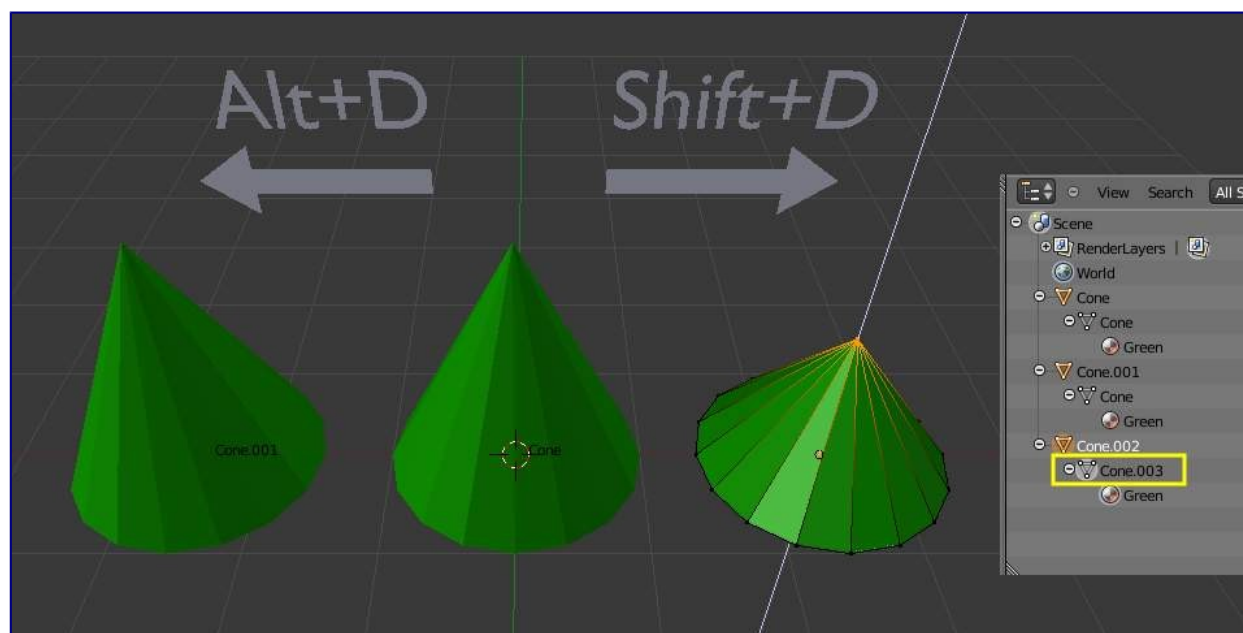
Reference
Mode: <i>Edit</i> and <i>Object</i> modes Menu: <i>Object</i> → <i>Duplicate</i> Hotkey: Shift-D

This will create a visually-identical copy of the selected object(s). The copy is created at the same position as the original object and you are automatically placed in *Grab* mode. See the example below.

This copy is a new object, which shares some data-blocks with the original object (by default, all the Materials, Textures, and Ipos), but which has copied others, like the mesh, for example. This is why this form of duplication is sometimes called “shallow link”, because not all data-blocks are shared; some of them are “hard copied”!

Note that you can choose which types of data-block will be linked or copied when duplicating: in the User Preferences, Editing Page.

Examples



The mesh Cone.006 of object Cone.002 is being edited. The mesh's Unique data-block ID name is highlighted in the Outliner.

The cone in the middle has been (1) link duplicated to the left and (2) duplicated to the right.

- The duplicated right cone is being edited, the original cone in the middle remains unchanged. The mesh data has been copied, not linked.
- Likewise, if the right cone is edited in object mode, the original cone remains unchanged. The new object's transform properties or data-block is a copy, not linked.
- When the right cone was duplicated, it inherited the material of the middle cone. The material properties were linked, not copied.

See above if you want separate copies of the data-blocks normally linked.

Linked Duplicates

Reference

Mode: *Object mode*

Menu: *Object → Duplicate Linked*

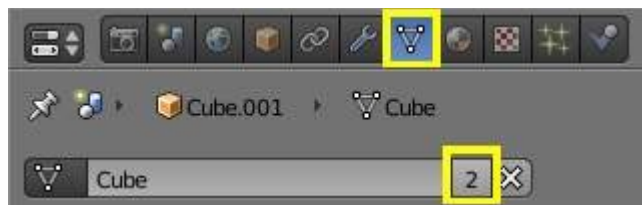
Hotkey: *Alt - D*

You also have the choice of creating a *Linked Duplicate* rather than a *Duplicate*; this is called a deep link. This will create a new object with **all** of its data linked to the original object. If you modify one of the linked objects in *Edit mode*, all linked copies are modified. Transform properties (object data-blocks) still remain copies, not links, so you still can rotate, scale, and move freely without affecting the other copy. Reference (*Duplicate Example*) for the discussions below.

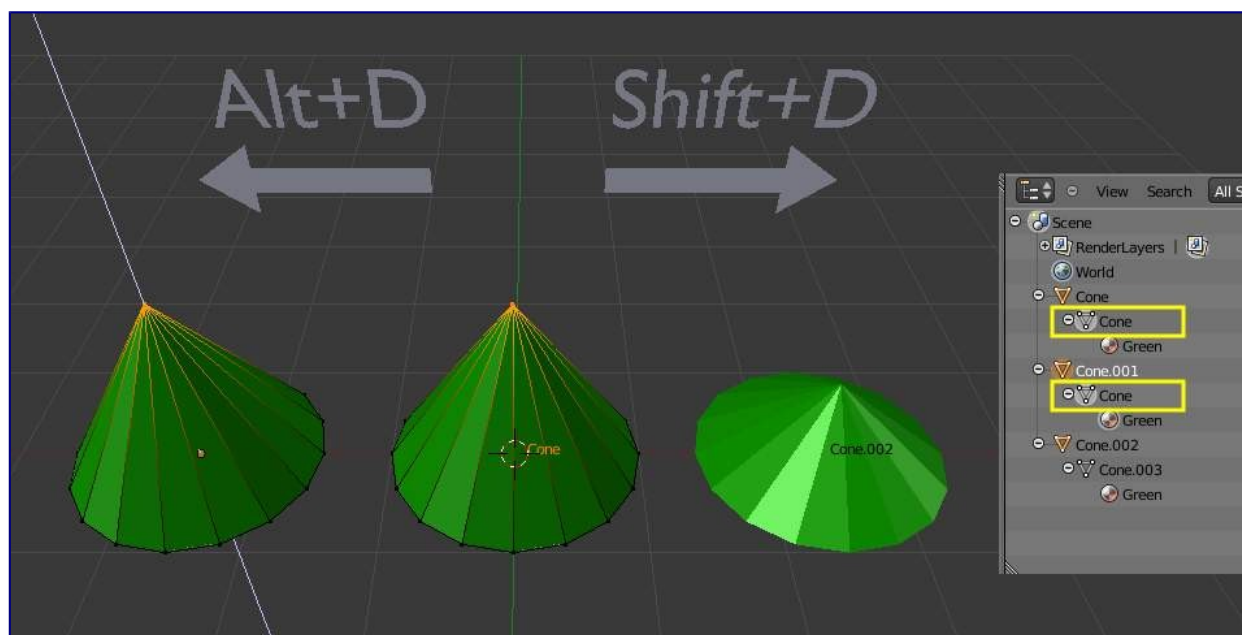
Hint

If you want to make changes to an object in the new linked duplicate independently of the original object, you

will have to manually make the object a “single-user” copy by LMB the number in the Object Data panel of the Properties Window.



Examples



The object Cone.001 was linked duplicated. Though both these cones are separate objects with unique names, the single mesh named Cone, highlighted in the Outliner, is shared by both.

The left cone is a *Linked Duplicate* of the middle cone (using **Alt - D**).

- As a vertex is moved in *Edit mode* in one object, the same vertex is moved in the original cone as well. The mesh data are links, not copies.
- In contrast, if one of these two cones is rotated or rescaled in object mode, the other remains unchanged. The transform properties are copied, not linked.
- As in the previous example, the newly created cone has inherited the material of the original cone. The material properties are linked, not copied.

A common table has a top and four legs. Model one leg, and then make linked duplicates three times for each of the remaining legs. If you later make a change to the mesh, all the legs will still match. Linked duplicates also apply to a set of drinking glasses, wheels on a car... anywhere there is repetition or symmetry.

Procedural Duplication

Reference

Mode: *Object mode* and *Edit mode*

Panel: *Object settings*

There are currently four ways in Blender to procedurally duplicate objects. These options are located in the *Object* menu.

Verts

This creates an instance of all children of this object on each vertex (for mesh objects only).

Faces

This creates an instance of all children of this object on each face (for mesh objects only).

Group

This creates an instance of the group with the transformation of the object. Group duplicators can be animated using actions, or can get a Proxy.

Frames

For animated objects, this creates an instance on every frame. As you'll see on this topic's subpage, this is also a *very* powerful technique for arranging objects and for modeling them.

Copying & Linking Objects Between Scenes

Sometimes you may want to link or copy objects between scenes. This is possible by first selecting objects you want to link and then using: Object ▸ Make Links ▸ Object to Scene.

This makes the same object exist in 2 different scenes at once, including its position and animation data. You can tell this is a *multi-user* object by the blue color of its center-circle

If you don't want the objects to be shared between the scenes, you can make them *Single-User* by using: Object ▸ Make Single User ▸ Object.

Further information on working with scenes can be found *here*.

Linked Library Duplication

Reference

Menu: *File → Link Append*

Hotkey: *Shift-F1*

Linked Libraries :Linked Libraries are also a form of duplication. Any object or data-block in other *.blend* files can be reused in the current file.

Hint

- If you want transform properties (i.e. object data-blocks) to be “linked”, see the page on *parenting*.
- Material Transparency will not display when instancing dupli-groups; this is a known limitation of Blender's view-port.

