




**Introduction–Build–WaveParticlesStatic**

**I n t r o d u c t i o n**

Blender offers a complete set of animation effects ready for use, only waiting for your creativity to bloom on your screen. We'll study here the use of the **Build**, **Wave** and **Particles** effects, in the ascending difficulty order, and give you many examples of use through full-length exercises. In any case, you can access these special effects by pressing the **F7-KEY**, or by clicking on the animation button . The following command buttons should then appear.



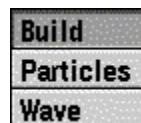
We'll pay special attention to the buttons on the right : **NEW Effect** and **Delete**. The other buttons will perhaps be studied in another tutorial.

**Basics**



All animation effects requires that you have a mesh on which they can act. **Select** the object you wish to animate (right click with the mouse) to make it appear in pink. Click now on the **NEW Effect** button. A small box toggled 'on' should appear by the side of the **Delete** button. Don't pay attention to it for now, but know that it works almost the same way than the small buttons used for controlling the layers of your scene.



You should also see appearing, more on the right, a **Build -** button with the minus sign at its end. It is a scrolling menu. Click and hold this button to see the animation effects available in Blender.



You can now choose **Build**, **Particles** or **Waves**, according to the effect you are looking for, and see that other buttons, which will be your effect parameters, appear in each case to tune your effects.

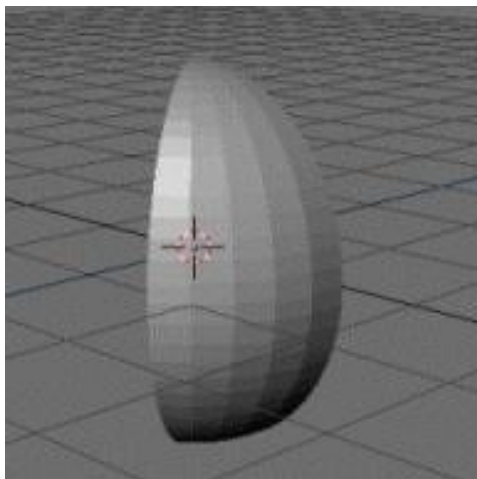
Please note also that, even if we'll not study this in this already promising tutorial, it is possible to assign many animation effects to the same objects, and that each of them appears as a small little box on the side of the  button. Click on the little box associated to the effect you want to change. Beware ! When you click on the  button, you erase all the animation effects associated to the active little box, and loose all its parameters !



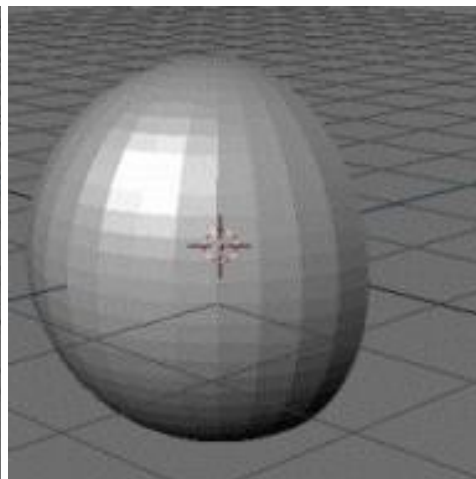
Introduction– Build –Wave –Particles –Static

**T h e B u i l d e f f e c t :**

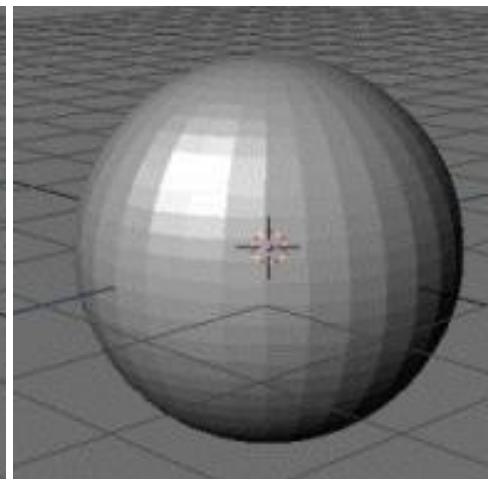
This effect is the simplest offered this day by Blender V1.80a. By applying it, you can, during an animation, render illusion that the object "build" itself step by step before your very eyes, face after face. Of course, you can specify the speed at which the building occurs, and also at which **Frame** number it starts. For example, this **UVSphere** builds itself during 100 **Frames**.



Frame No16



Frame No32



Frame No100

Let's give it a smooth start : open a new session (**CTRL+X**) and delete the default pink plane (**ALT+X**). Then add a UV Sphere (**SPACEBAR>ADD>MESH>UVSphere**) and validate all the default parameters. Make sure that the sphere is selected and shows in pink (if it is not the case, right click on it). Invoke the **Animation Buttons** (**F7**) and click one time on the **NEW Effect** button. The following interface should appear.



The **Build** function is only defined by those two parameters :

**Len: 100.00** (Length) : this button specifies the **length** of the animation (i.e. the number of **frames** during which the building will occur).

**Sfra: 1.00** (Starting Frame) : this button defines the **starting frame** of the building process.

This means that with **Len: 75.00** and **Sfra: 32.00**, your sphere will begin to build itself from **starting frame** No32, and will take 75 extra **Frames** to be complete. Thus, this also means that the sphere will only be complete on **Frame** No107.

Not sure ? After applying these parameters to your sphere, you'll have the surprise to see it disappearing from your screen. Press **ALT+A** to play the animation... You will soon see the first building parts of your sphere, until it is

fully completed...

## **P r a c t i c e s :**

Animating a lightning strike



## IntroductionBuild-Wave-ParticlesStatic

### The Wave effect :

This animation effect authorize you to crease a mesh in order to give it a wavy feel and touch that can be, of course, animated. Start by a new session (**CTRL+X**). In top view (**NUM7**), select the default plane. Size it with the **S-KEY** until it has seven times its nominal size (remember you can constrain the sizing to entire value by holding the **CTRL-KEY**). It is **highly** recommended to subdivide the mesh of your plane if you want to achieve good looking results with the **Wave** effect. Press **TAB** to enter edit mode. Select all the vertices (**A-KEY**) which will now appear in yellow. Hit five or six times the **W-KEY** in order to subdivide the mesh and validate with **ENTER** each time. You now have a plane with a rather tight mesh, and you can leave the edit mode (**TAB** again). In the Edit Buttons (**F9**), activate the **Axis** button to have a better understanding of the next parts of this tutorial. This will be our working basis.

As we already saw, invoke the Animation Buttons with the **F7-KEY**, and click on **NEW Effect**. Immediately, click on **Build** and hold the button to give you the time to select the **Wave** effect. Perfect. You should see now the following parameter buttons.



The **Wave** effect is far more complex to use than the **Build** effect because you have to define the wave that will disrupt your plane's geometry. By activating the **X** button, the wave will propagate only in the X direction of your plane (pay a look to the directions noted on your object since you activated the **Axis** button). Of course, if only the **Y** button is activated, it is in this direction that the wave will propagate, and in case both **X** and **Y** are activated at once, the wave will propagate in both directions at the same time (circular wave). The **Cycl** button defines the case where the wave repeats itself (button activated) or if it disrupts the mesh once only (button deactivated). Everything's all right ? Okay, let's go further.

**Time Sta** defines the **Frame** at which the wave is born. Please note that it can be a neagtive number, indicating that the wave started to propagate before your animation starts. **Lifetime** defines the number of **Frames** during which the waves will freely propagate (if **Cycl** is activated). **Damp time** represents some sort of inertia, or the survival time, once the wave **lifetime** is over. Thus, with a **Lifetime: 20.00** and a **Damp time: 50.00**, the wave will propagate freely during 20 **Frames**, and then will smoothly "die" during 50 **Frames**. Once this delay is over, the plane will have regained its perfect mesh structure.

**Sta x** and **Sta y** defines the location of the "hazard" giving birth to the wave (stone fall, water drop, etc.).

**Speed** defines of course the displacement speed of the wave (its frequency when **Cycl** is activated) and **Heigh...** its height. **Width** defines the wave length (the distance between two peak) and **Narrow** the "sharpness" of the wave (a very flat wave for a low value, a very sharp and narrow wave for a greater value)

In any case, you can stack different **Wave** effects to achieve complex results.

## **P r a c t i c e s :**

**Ocean :** this effect is of course ideal to build the waves of an ocean. Activate only **X** or **Y**, according to the direction facing the shore if you have one. Set the origin point of the disruption (**Sta x** and **Sta y**) far away and activate **Cycl**. To have beautiful waves, set **Time sta** to a negative value and **Lifetime** to a value greater than the number of **Frames** of your animation. In such a case, **Damptime** is quite useless, so you can forget about this parameter. Of course, **Speed**, **Heigh**, **Width** and **Narrow** should be adjusted to your needs.

**Stone in the pond :** in this precise case, **Sta x** and **Sta y** defines the falling point of your stone, and **Cycl** is also activated. We recommend a short **Lifetime** (like **Lifetime: 1.00** ?) and a **Damptime** slightly longer. The more the disrupted liquid by the stone is slimy, the more **Damptime** should be short (2.00 or 3.00 in the case of jelly or fresh cement, I would say –just try your own values). In the case of water, a **Damptime: 10.00** to 20.00 value appear correct most of the time. As the wave will propagate in all directions at once from the falling point, activate **X** and **Y**. The other parameters depend heavily on the mass of the stone (particularly **Speed** and **Heigh**).

**Water drop in a pool :** this effect is not much different than the earlier, apart that you'll certainly want to deactivate **Cycl** and set **Damptime: 0.00** (your water drop should not disrupt a lot your pool, should it ?). **Speed** will certainly be higher, but **Heigh** and **Width** so small that we shouldn't even care setting them ;o) Just note that many water drops should fall into your pool (rain...), so you should have to stack this effect many times to the same pool, with falling points (**Sta x** and **Sta y**) each time different !

**Matter disruption :** for a cool looking effect a la Matrix (the helicopter crash against the glass building), you should have the same settings than the stone in the pond practice, except that you will want to deactivate **Cycl** because you want a single wave... ;o)



Introduction – Build – Wave – Particles – Static

**T h e P a r t i c l e s e f f e c t :**

Particles are dimensionless items emitted by meshes. We give them Halo properties in order to simulate a simple way fire, smoke, water drops, clouds and even mist. When you activate the **DupliVerts** button, particules transform into objects the same as the emitter is parented to, which is very handy to simulate flocks of bugs, birds or even fishes. To start, we'll take again our good ol' **UVSphere**. Start a new session (**CTRL+X**) and erase the default plane (**ALT+X**). Then add a UVSphere (**SPACEBAR>ADD>MESH>UVSphere**) and validate all the default parameters. Make sure the sphere is selected and appear in pink. Then invoke the **Animation Buttons (F7)** and click one time on the **NEW Effect** button. Then click on **Build** and hold to have the time to select the **Particles** effect. You should see, then, the following.



It will get too long to explain with full details how each of these buttons act. If you are interested in a full description, please get the official Blender Manual. This tutorial will only explain the most used parameters and the one without which you could do anything. They are split in three main categories. Practices could go further in explanation, though.


**Population and lifetime of particles :**

**Tot** defines the number of particles. Be especially careful with this parameter, as it uses lots of memory and asks for more rendering time. **Sta** sets the **Frame** during which the first particle is emitted and **End** the **Frame** at which the object ceases to emit anything. It is very common to set **Sta** to a negative value (for the effect to stabilize to its "cruise rate") and **End** to a value greater than the **Frames** number of your animation (because you don't want your effect to dim before the end of your animation). **Life** is a measure of the lifetime of a particle (e.g., the number of **Frames** it will evolve before disappearing). **Keys** enhance the fluidity of your animation, at the price of an increase of both memory and rendering time. **Mult: 1.000** enables a particle to multiply itself by a number set by the **Child** parameter (these particles will have a lifetime set by the second **Life** button). **RandLife** and **Seed** apply to the lifetime of a particle a random variation (in fact, it is only a pseudo randomness, because for a given **RandLife** value, the obtained result will always be the same, repetitive and predictable, unless you also set to another value the **Seed** parameter).

**Speed of particles :**

The **Norm** button constrain the starting direction of the particle. The higher is the value, the more the particle will be emitted along the normal to the emitter. **Ob** defines how much the very own speed of the emitter will add to the speed of the particle, whereas **Rand** applies to the speed of the particle a random variation. At last, **Damp** simulates the inertia of the particle, and offers a mean to control its slowing along time.

### **Movement of particles :**

**Force** constrains the direction taken by the particles after they have been emitted, **X**, **Y** and **Z** defines the vector direction associated to this direction (remember to activate the  button to visualize better what your are setting parameters to). It is perfect to simulate the effect of wind upon flames or smoke, or even the angle water drops make with the vertical in the case of rain or waterfall.

### **P r a c t i c e s :**

[Animating a Fire with Blender](#)

[Animating Smoke with Blender](#)

[Modelling clouds or mist with Blender](#)

[Animationg a Flock of Bugs](#)





Introduction – Build – Wave – Particles – Static →

**T h e S t a t i c P a r t i c l e s e f f e c t :**

The static particles are a very clever and handy improvement of classical particles. In these, particles move and evolve along time, whereas the static particles are purely stationary. The trajectory they should have had if dynamic becomes a brand now that they are static. The uppermost interest is for the one who wants to model fur, hair or even grass ! Luckily, now that you master particle systems, you will discover that **static particles** behave quite exactly the same way and should prove easy to use. In fact, the parameter buttons are almost the same as the ones we saw just before ! Activate the **Static** button to get the following.



**Tot** sets the number of brands emitted from the object, and **Step** defines their continuity (**Step: 1** indicates a continued brand, while any greater value gives a discontinued brand, with greater space between two brands when **Step** closes to 100). **Life** will characterize, as you have already understood, the length of the brand. For most of the works (particularly grass and fur), you'll want to set **Keys** to 2, as greater values will add to the brands' susceptibility to the **Force X, Y and Z** parameters.

All other paramaters have the same use and logic than dynamic particles ! Now, it's up to you to play with this great function.

**P r a c t i c e s :**

Hair and fur

Grass