

The Complete Guide to Blender Graphics

Supplement-02

Cell Fracture 1

The **Cell Fracture** tool performs a similar operation to the Object Fracture tool. Cell Fracture is more sophisticated having additional control options.

The tool resides in the **User Preferences window, Add-ons, Object category** and requires activation before use.



Figure 2.1

When activated you will find the **Cell Fracture** button in the **Tool Panel, Tools tab** under **Transformation** (Figure 2.1). Clicking the button displays a panel containing the Cell Fracture controls (Figure 2.2).

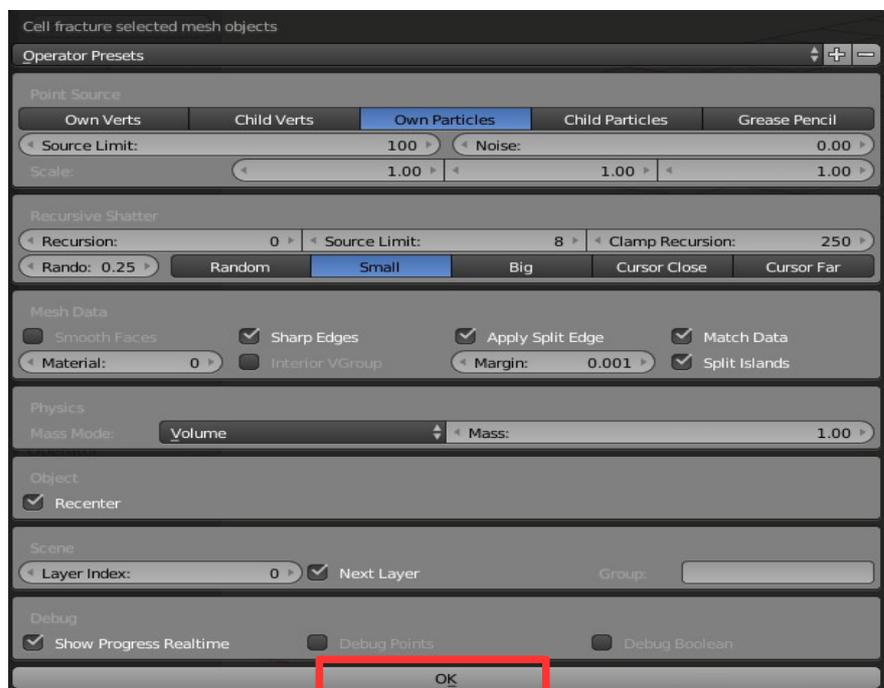


Figure 2.2

To demonstrate the functions, at the very basic level, have the default **Cube** selected in the **3D window**.

Note: In the 3D window header, the orange dot in the **Layers** buttons shows that objects are present.



Figure 2.3

3D window Layers buttons

Make special note that the Cube is on **Layer 1** in the 3D window. Click the **Cell Fracture** button in the **Tool Panel** to display the **Cell Fracture** panel (Figure 2.2).

In the Cell Fracture panel, leave the default settings as they are and click the **OK** button at the bottom of the panel. You will observe that something happens to the Cube in the 3D window and it becomes deselected. The Layers buttons in the 3D window header now indicate that there are objects on Layer 2 (Figure 2.4).



Figure 2.4

Click on the Layer 2 button and you see what appears to be a Cube on Layer 2. It is selected as shown by a red outline and there are several orange dots scattered within the profile.

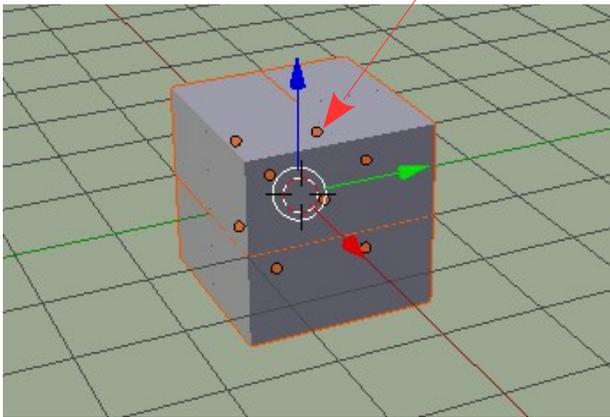


Figure 2.5

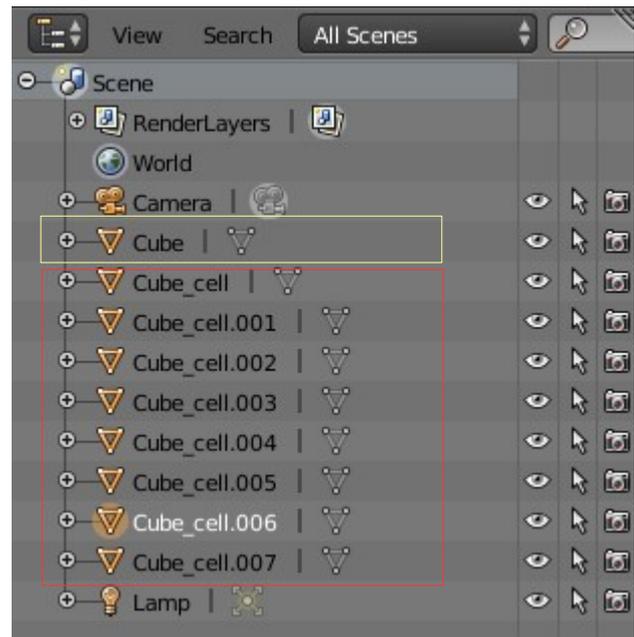


Figure 2.6

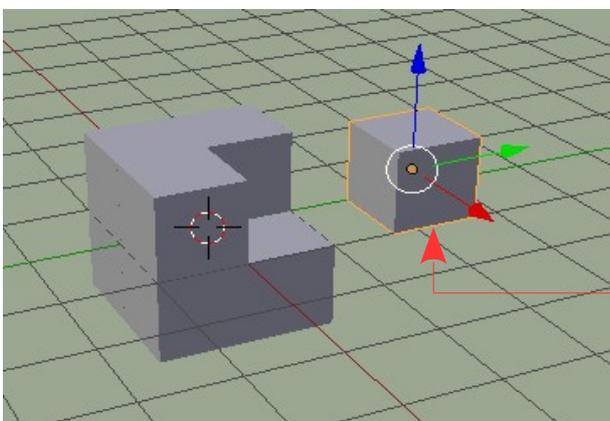


Figure 2.7

Cube_cell.006 selected and moved aside.

Look in the Outliner window (Figure 2.6) and observe the listing, Cube, Cube_cell, Cube_cell.001, Cube_cell.002 etc.

The first entry, Cube, is the cube on Layer 1 (the original default Cube). The remaining entries show that the Cube on Layer 1 has been subdivided into eight new objects (Shards). The Shards have been placed on Layer 2 in a cluster taking the shape of the original Cube. The red dots in the cluster are the centers of each Shard.

Note: In the Properties window, Object Data buttons, in the Browse Mesh Data to be linked menu, there are eight entries. Mesh, Mesh.001, Mesh.002 etc. Eight entries, eight vertices on the Cube, therefore, eight Shards (Figure 2.8).

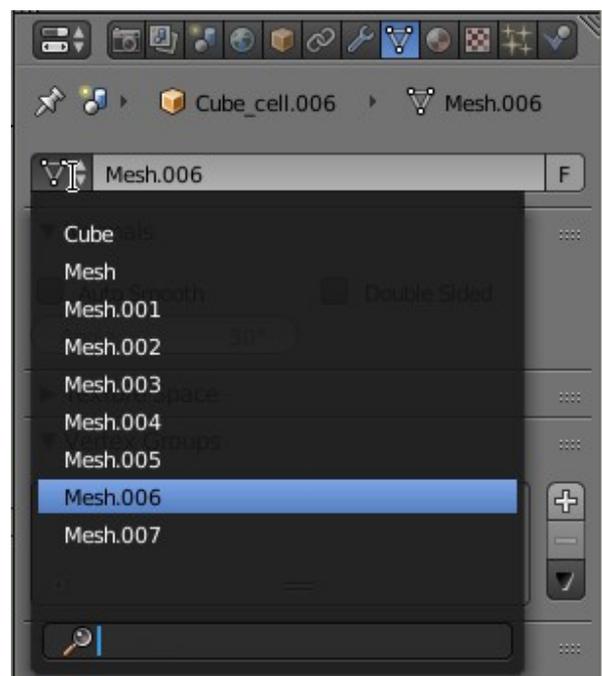
In Layer 2, deselect then RMB click on one of the Shards and using the manipulation Widget translate it away from the cluster (Figure 2.7). **Note:** The shape of the Shard, a cube.

The Shard is, in fact, a separate cube object.

The Cell Fracture tool has divided the original Cube based on the number of vertices on the Cube's surface. Eight vertices = Eight Shards.

If you have the Shard (cube object) moved away from the cluster on layer 2 and selected, then repeat the Cell Fracture operation, the Shard will be subdivided and new Shards (objects) will be placed on layer 3.

Figure 2.8



Employing the Cell fracture operation using a UV Sphere object (32 segments, 16 rings) which has 512 vertices you would expect 512 Shards to be produced but this is not the case.

In the Cell Fracture panel, under Point Source, you see that the Source Limit has been (by default) set at 100 therefore only 100 points (vertices) are used when calculating the number of Shards.

You will observe that the Cell Fracture operation produces Shards of a geometric shape and that the fracture is identical each time the operation is performed (**providing an identical object is being fractured**).

Cell Fracture in Practice

The foregoing explains what occurs when the Cell Fracture tool is run but the tool is designed to simulate breaking, shattering or exploding when animating. There are many effects which can be produced. As an introduction, the following example will get you started.

Fracturing a UV Sphere – Dropping

In a new Blender Scene scale the default Cube object producing a ground plane. Scale the Cube to produce the Ground Plane or use a Plane object and extrude on the Z axis to produce thickness.

Add a UV Sphere and position it above the plane.

The objective will be to simply have the Sphere fall and break apart when it lands on the Plane.

In this demonstration, you may be in **Blender Render mode** or have **Cycles Render** active.

Activate the Cell Fracture Add-on in the User Preferences window.

Select the Ground Plane and in the Properties window, Physics buttons, add Physics Type: Rigid Body and change the Type to Passive mode. This ensures that the Plane will not move when an animation is played.

Select the UV Sphere and in the Physics buttons add Rigid Body Physics Type: Active. The UV Sphere will interact with the Ground Plane.

With the UV Sphere selected, click Cell fracture in the Tool Panel to open the Cell fracture Operator Presets Panel.

Note: Point Source is set to **Own Particles** but if you check in the Particles buttons you will see that a Particle System has not been generated. This being the case, the Point Source will revert to **Own Verts** when Fracture occurs. This takes place whether you change the Point Source setting or not.

Also by default, Source Limit is set at 100, therefore, the number of Shards produced is limited to 100.

Click OK and see 100 Shards created and placed on Layer 2 (the next layer) in a cluster taking the shape of the original UV Sphere.

For the Shard cluster to interact with the Plane in the Game, the Plane must be on the same layer. Move the Plane to Layer 2. To see what happens and render an image you must also move the Camera and the Lamp to Layer 2 and be in Layer 2 when you press the Play key in the Timeline window to play the animation.

In Layer 2, RMB click on one of the Shards in the cluster to select. In the Properties window, Physics buttons you will find that Physics has not been applied when the Shards were generated. Remember Rigid Body Physics was applied to the original UV Sphere. Add Rigid Body Physics, Type: Active to the selected Shard.

You want all the Shards in the cluster to have this same Physics type, therefore, box select the Shard cluster, then in the Rigid body Tools tab at the LHS of the Screen click Copy From Active to copy the Physics to all Shards.

Press Play in the Timeline window header to play an animation and see the Shard Cluster descend under the influence of gravity and shatter on impact with the Ground Plane.

Wow! What happened? The cluster exploded.

This may be the effect required but if not go back and select a single Shard. In the Physics buttons, Rigid body Collisions tab under Sensitivity, you will see Collision Margin with a Margin value of 0.040. When the Shard cluster is generated, a gap is created between the Shards. You will observe that the Rigid body Collisions, Shape is Type Convex Hull.

Convex Hull refers to the shape of the bounding volume Blender uses for calculating reactions in Rigid Body Physics (there are seven options). Convex Hull is a shape roughly representing the shape of the object, which in this case is the shape of a Shard. The Collision Margin value extends the Convex Hull shape outside the shape of the Shard. In this particular case, the default value 0.040 extends the shape in excess of the gap between the Shards, therefore, when the animation is played the individual shards fight with each other for space and force each apart. The cluster explodes.

To cancel this action check (tick) **Collision Margin** and reduce the Margin value to 0.000. Remember you have only one Shard selected. Box select all Shards. In the Rigid Body Tools tab, click Copy from Active then play the animation.

This time the cluster descends due to gravity and shatters on impact with the Ground Plane.

If you wish to maintain the mid air explosion you may control the effect by adjusting the Margin value. Don't forget to copy the value to all shards.

Note: The methodology works with the 3D window in Blender Render mode and with Cycles render active. At the time of writing with Blender version 2.78c in Blender Game, there doesn't appear to be the facility to copy the Collision Bounds value to all shards. The Shard cluster, therefore, disintegrates in mid air when the Game is played. To create a simulation of a sphere shattering on impact, in Blender Game, a keyframe animation has to be created which is another story.

Cell Fracture Options

The Cell Fracture panel contains optional settings for affecting how the object is fractured.

Operator Presets

At this point, it is worth while knowing the function of some of the presets. There are many options in the panel and it will be up to you to experiment and discover the full extent. I have been criticised for taking this approach but to cover everything would take a very long time.

The following will explain a few of the main functions.

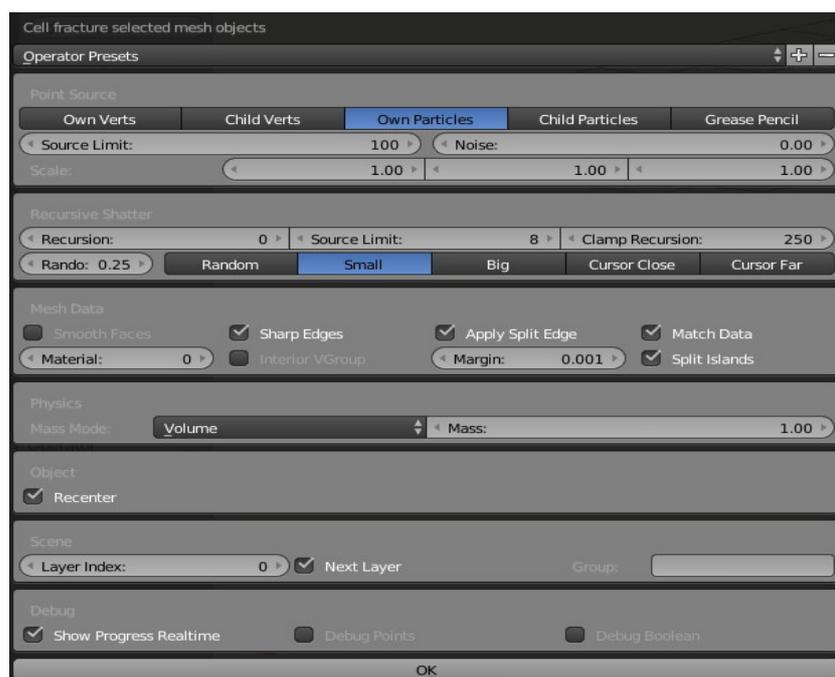


Figure 2.9

Point Source: Determines the number of Shards generated.

Own Verts: The number of Shards will equal to the number of vertices on the surface of a mesh object.

Note: With all the Point Source options the Source Limit value limits the number to the value set. For example a UV Sphere with 32 segments and 16 rings has 32 x 16 vertices (512) plus one at each pole = 514. When fractured only **approximately** 100 Shards will be generated.

Note: Approximately. The script appears to generate an approximate number only.

Child Verts: With one object parented to another the number of child's vertices is used to generate Shards.

Note: The script appears to have an anomaly in that it only uses half the number of child vertices.

Own Particles: The number of Shards determined by the number of Particles generated when a Particle System is applied.

Note: If no Particle System exists the value defaults to **Own Verts**.

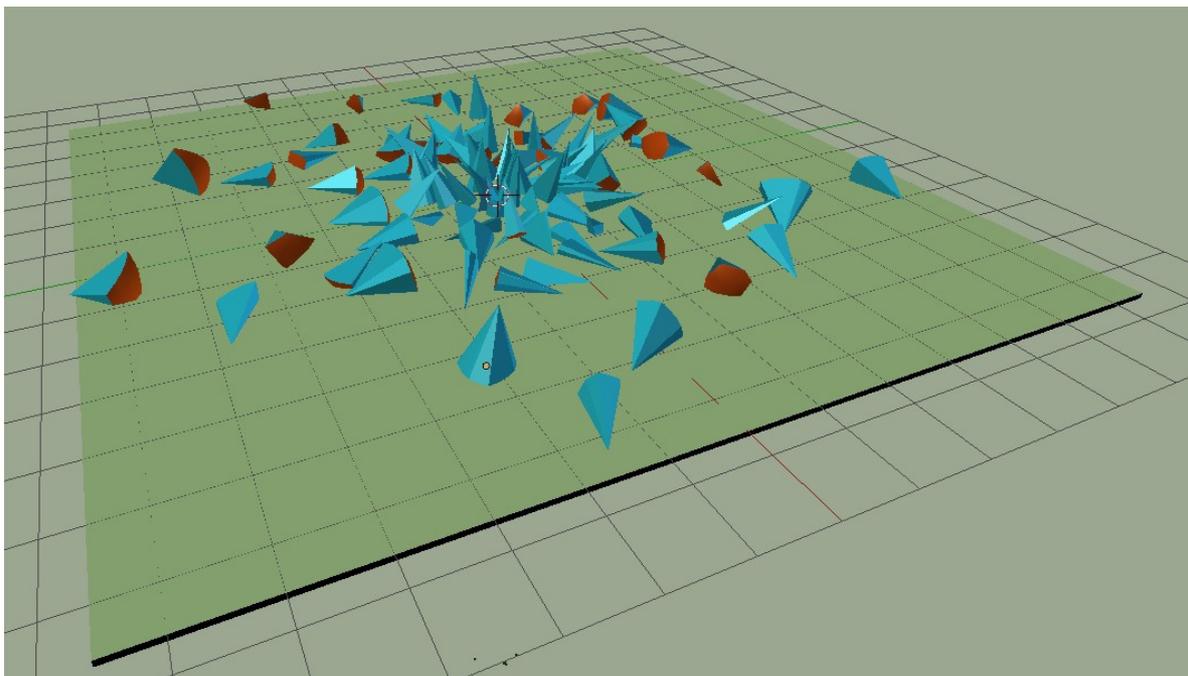
Grease Pencil: The number of Shards is determined by drawing lines on the object's surface using the Grease Pencil draw mode.

Recursion: Shards are generated according to Point Source then with a recursion value set the individual Shards are again subdivided producing additional Shards.

Mesh Data – Material: With a material applied to an Object (Material Channel 1) and a second Material Channel created, the Material color in Channel 1 is applied to the outer surfaces of Shards in a cluster and the Material color of Channel 2 is applied to the inner surfaces of Shards in a cluster.

Scene – Layer Index: When an object is fractured the Shard cluster is placed on Layer 2 with Next Layer is checked. Uncheck to have the cluster placed on Layer 1.

Disclaimer: Statement made claiming anomalies in the operation of the Cell fracture Add-on are made from observation only, using Blender version 2.78c. Other versions of Blender may produce different results.



UV Sphere Fractured on Ground Plane

Figure 2.10