

# Creating quite nice explosions in Blender

A quite nice guide by Rhysy

Firstly, I apologise for the awful format of these tutorials. I hate PDF, but it was either this or learn HTML coding. Never going to happen. Secondly, although .blends are included, don't expect to be able to simply download them and append everything into your own scene. You'll almost certainly have to alter halo sizes and material properties, especially with the more advanced one. The colours used may not necessarily integrate well with your own scene, so expect some work. Thirdly, I'd like to take the opportunity to rant about the bizarre lack of any good HTML editor that doesn't entail looking at the code, but I won't.

This tutorial will be in two parts. The first shows a basic explosion, very quick to setup and render, and taking very little system resources. The second is far more complex, looks much better but is not so easy to animate. Users already comfortable with particle effects may want to skip ahead to the second explosion, which does not use conventional particles. I will assume a basic working knowledge of Blender throughout. And I may occasionally delve into pointless sarcastic remarks like this one, for no reason. Also, I won't go into animating the explosions, as that's largely a matter of tweaking particle motion and material settings... and okay, I've not actually animated these. I could if I wanted too though. Probably.

## Part 1 : A small, simple explosion

Download the .blend :

<http://rhysy.plexersoft.com/blends/Explosion%20small.blend>



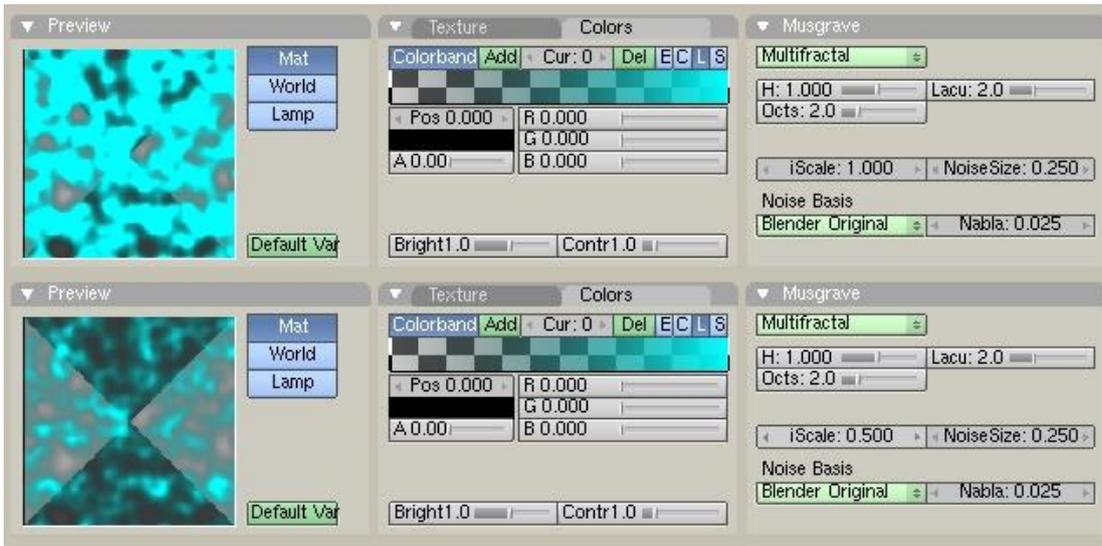
This explosion is very suitable if required for only a small part of a scene. It takes moments to setup and moments to render. This simplicity makes it easy to animate. It's not at all good for close-ups though. It may not be too impressive here, but for shots of this technique employed in fully-developed scenes and not just against a black background, see the Conwy Castle and War Of The Worlds galleries<sup>1</sup>.

There are 3 basic components to this : the inner, bright fireball, the outer, radial "trails", and the debris. The trails help convey a sense of expansion without the need for motion blur. In fact, it looks horrid with motion blur. Both parts of the flame are made with ordinary particles with halo materials. These are textured with musgrave textures. You can use procedural clouds instead, but musgrave is better because it enables more detail. Crank up the "Octs" setting and lower the "H" setting. Also, it's useful to

---

<sup>1</sup> This is MY tutorial and I'll blow my own trumpet if I want to.

turn down the intensity setting. See what happens with the default parameters of a musgrave texture with the intensity at 1.0 and 0.5 :



The second one is much less saturated with blue. This is good, because it means any other colourband settings will display much more clearly and not be so swamped by the far-right hand marker.

But first thing's first. Delete the default plane (or cube or whatever it is these days) and add an icosphere. Make it emit particles (see all the wonderful documentation on blender.org if you don't know how to do this). Let's start with the inner part of the fireball. Simply add some normal and random speed to the particles. The spacing between the particles will affect how large the halo setting needs to be, so here's what I used. Note the particle settings also :



You should have a rough sphere of particles on screen. Now assign a material and select the halo setting. Add a texture. Sadly you can only have one texture on halos (we'll find a way around that in the second part of the tutorial). Now, logically enough, go into the textures setting and choose musgrave. Set the H to 0.323 and the Octs to 7.415 (experiment later with these, see what you like best). Also set the intensity to 0.3 and raise the noise size to maximum, otherwise it'll look grainy. Now turn on the colourband. Make colours like this :



See the .blend for further details. Now, go back into the materials settings. Make the texture affect the colour and the alpha (choosing Multiply instead of Mix may give better results), don't forget to turn on HaloTex or nothing will happen. Set the hardness to max and turn the add setting up quite high, but not to full (I used 0.75). This adds the colour of the halo to whatever's behind it, faking a quite convincing glow effect. You may want to lower this value later, if you want to use the sequencer glow effect. Or, you may not. In any case I lowered the halo size to 0.2 and turned on X-alpha. There's probably no need for X-alpha though.

That's the main fireball done. Now for the glowing spike things. Duplicate the icososphere and rotate it so that the particles from the two emitters do not overlap each other. For the spikes I only used 500 particles with a much higher normal speed and a bit more random speed. I also turned on vect, which basically displays the particles as lines along the direction they're travelling. Musgrave textures seem to work particularly well for this effect, I find. Here's the particle settings I used :



Now this mesh will have the same material as the previous one. Either delete the material, or better yet make a single-user copy of the old one (i.e. click on the number next to the name of the material in the materials panel). Only minor changes to make here. Lower the alpha to 0.5 and raise the add to 1.0. This will make the merge quite nicely with the main fireball. Now, go into the textures panel and make the texture a single-user copy too. No colour changes need be made, but changing the yellow marker to orange or even red can look quite nice. More importantly though, raise the intensity to 0.4 or 0.5 so that they'll display more clearly (since we have fewer particles here). I also raised the H setting to 0.569, but this may not be necessary.

That should take care of the fireball. Now for the debris, that's even easier. Duplicate an icososphere – any one – and move it to a new layer, to make it easier to work with. Create a simple piece of debris. A plane with one vertex moved around will do. Parent this to the icososphere (select the debris, shift-select the sphere, press ctrl+p). In the object panel (not the particle section, but the object part), turn on dupliverts and rot. In the particles tab turn on vect, if it isn't already. This will make the bits of debris rotate. Now it's just playing with particle motion settings to get a decent distribution of debris. By now this shouldn't be too hard. Don't use too many particles though, or repetition may become obvious. I used two bits of debris, each attached to a different emitter. You can just select both the sphere and debris (at the same time !), duplicate and rotate. Different speed settings for the second emitter also helps.

Your debris may initially be arranged in a very weird pattern. Select the debris mesh and go into the object panel. In the "Anim settings" panel, play with the "TrackX, Y, Z" and "UpX, Y, Z" buttons until you get something better. Finally, to illuminate the debris, which helps fake the 3D appearance of the particles, I just used an ordinary lamp at the centre of the spheres, with a yellow colour. It may be helpful to use the "Sphere" lamp option in a real scene.

## Explosions with bells and whistles

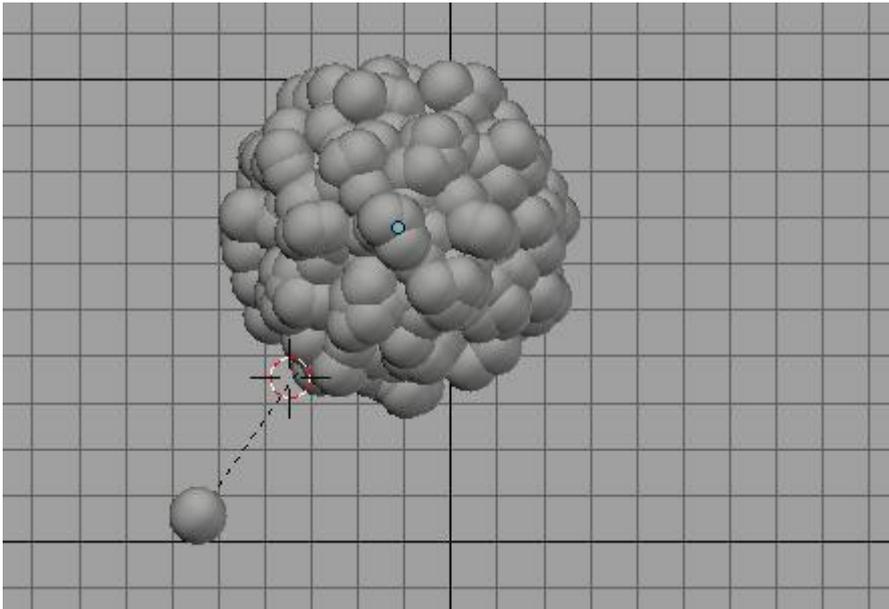
Download the .blend :

<http://rhysy.plexersoft.com/blends/Explosion%20animated.blend>

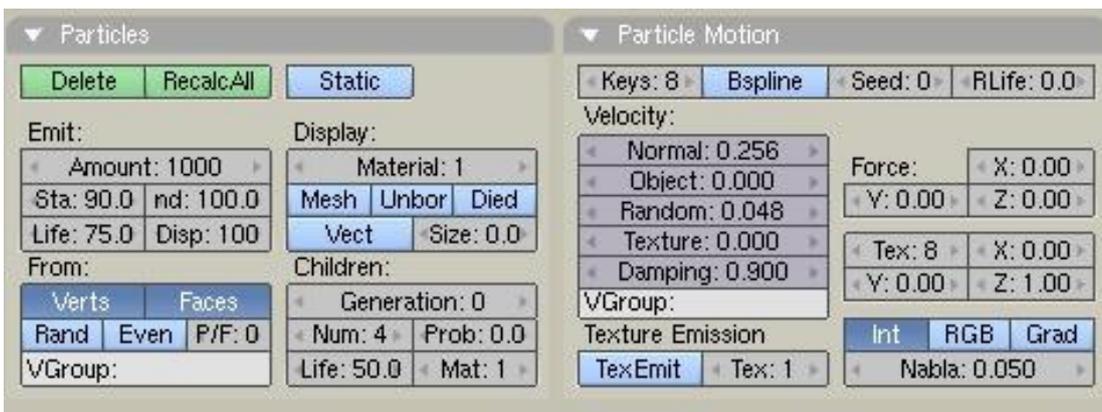


The last tutorial pretty much took particle explosions to their limit. Oh, sure, you can play with shaded particles with the unified renderer if you like, and probably even get some good results. But this can lead to some extremely long render times and may mean adjusting your whole light setup in a real scene. In this part I explore a technique developed by [Macke](#). It's far more powerful, gives reasonably good faked volumetric effects and is suitable for large renders or close ups – if anything it looks *better* in a larger render. The downside is that it's a lot more complicated, uses up far more memory and is much harder to animate.

First things first then, create a particle emitter from an icososphere, and get a nice rough sphere of particles. Let the particles be emitted over several frames, not all at once as in the previous example, as we're going to be making a complete "solid" explosion. Find a frame at which you have a nice rough sphere of particles. Don't skimp on the random speed setting, but don't turn it up too high either. Next add a second icosphere, level 2, move it out the way of the particles, and parent it to the emitter. Turn on dupliverts. Now adjust the size of the dupliverted sphere, and if necessary work on the particle motion settings too. You're aiming to get something like this :



For reference, here are the settings I used :

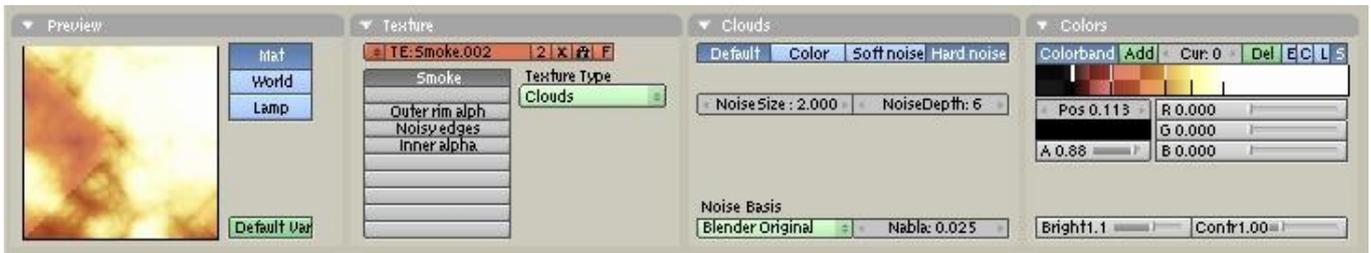


Now we'll give this mesh a material. It's far easier to just append this from the .blend, but let's take it step-by-step for the sheer hell of it. Firstly, set the colour of this material to white. Turn on z-transparency. Leave the reflectivity as it is for now, but you may need to adjust it later. Turn OFF traceable and shadow an all that stuff. You can turn them on if you use transparent shadows, but that will mean a load more work, and a major slowdown. You'll also want to lower the ambient slider to 0 (stops AO from working on the material) and I found it useful to turn translucency up to 1. Also turn off specularly. I used a ramp, can't remember why though. Here's the material settings I ended up with, excluding textures :



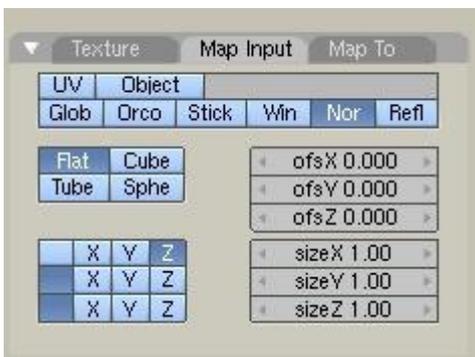
Now add the first texture. This is where the real work begins. For now, don't worry about the exact texture settings too much. Everything in the scene to some extent affects everything else, so at this stage there's really little point worrying if things are exactly right, because they're almost certainly not.

Anyway make this first texture a clouds texture, noise size 2.0, hard noise, depth 6. Though I said earlier musgrave gives more detail, I couldn't get it looking so nice with this method – but in any case it won't be necessary for extra detail. Adjust the colours so that you get something like you see below :



In the material panel, make this texture affect the colour (value 1.0), normal (value 25, but you may prefer to lower this), and displacement (value 0.5, may need to adjust). Using bumpmapping gives a massive amount of extra detail, or at least it will when we add lamps. Displacement is not strictly required, but it helps. A subsurf modifier will make the displacement effect look better if you do. Also choose Multiply instead of Mix, and in the Map Input panel choose Global instead of Orco and Sphere instead of Flat. Global coordinates are, I think, a bit like using object-orientation, so that the texture spreads across the particle spheres seamlessly.

Now for Macke's extremely clever trick. With just one texture, the fire colour will be there but nothing else. To fake the appearance of a volumetric, 3D explosion, Macke uses 3 more clouds textures. The first fades the outer rim of each sphere so you don't get a sharp cutoff. This is a blend sphere affecting the alpha (value 1.0) with Subtract. Note the settings in the texture panel, these ensure it makes the edge of the sphere fade correctly :



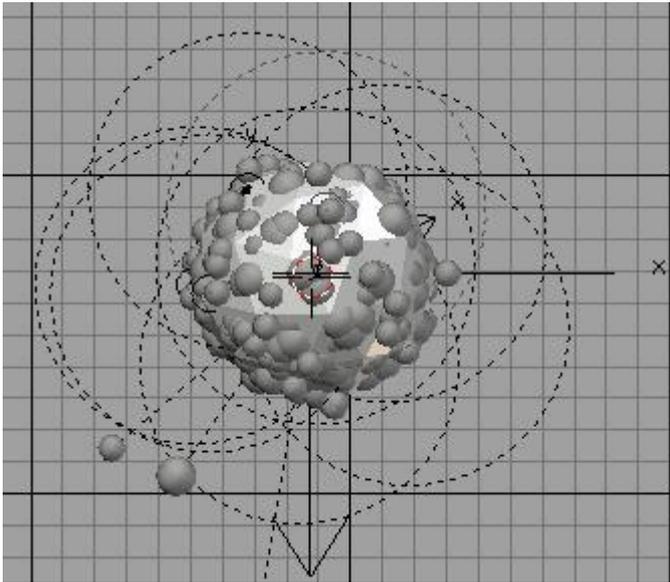
The second texture makes the ragged appearance you get with fireballs, rather than a smooth fade off for each sphere. I found this unnecessary. Basically, It's just a clouds texture with two colourband markers close together for a quick change in the alpha value. Consult the .blend for more information.

The final texture is very important, playing a similar role to the first. It has the same mapping settings as above, but in the Map To panel Alpha, Negative, and Multiply are selected. In the textures panel, contrast is set to 2.0.

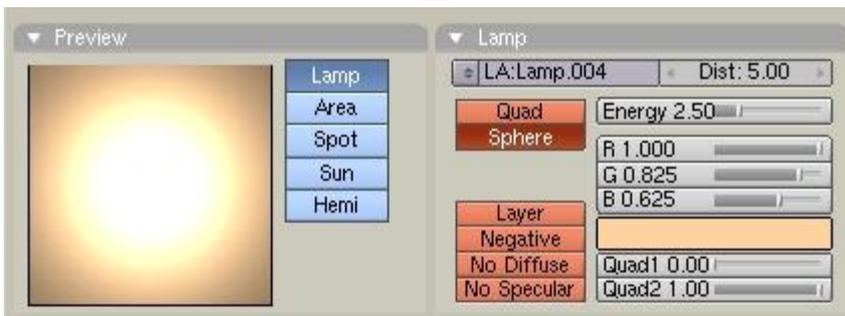
That completes the basic fireball. To give some more variation, I added a second emitter with a second, smaller icosphere as its child, using a different texture to affect the colour. I gave the particles slightly more random variation with this one.

Now, these fireball materials don't have traceable on, so they can't cast shadows. But they're not shadless, so they must receive light from somewhere. To make the cloud look like its glowing from within, I created another icosphere particle emitter on another layer, this one emitting lamps set to "Sphere"

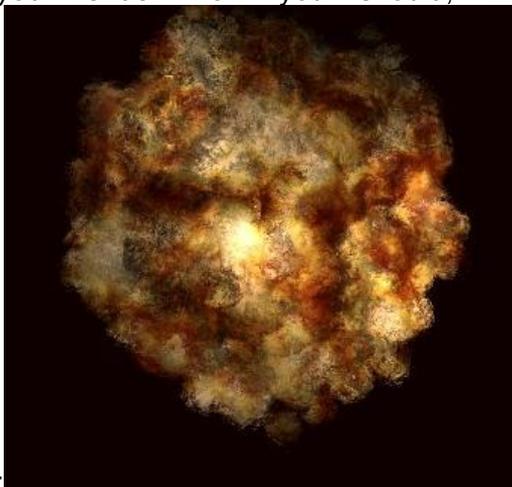
instead of more icospheres. I made this emitter quite big, about the same size of the particle–fireball “blob” :



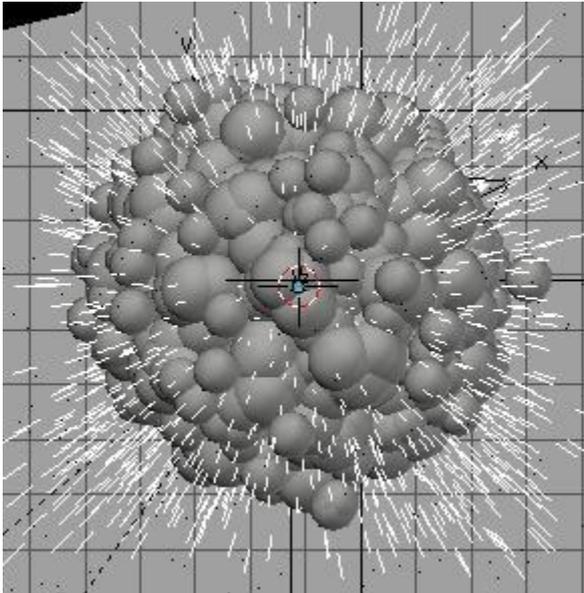
I used a few (8 particles) bright lamps rather than many dim ones. This is better because it gets a less even glow, which helps to show the bumpmapping more and creates the appearance of several bright, hot regions and several darker, cooler areas. I made the lamps a sort of orange–gold colour :



The sphere setting restricts the influence of the lamps to within a specified radius. It also means the energy decays quickly with distance, so they need to be bright. Settings were found largely through trial and error. This gives us the basic explosion effect. We’re not at the bells–and–whistles stage yet, but if you render now you should, with perhaps a little work, get something like this :



Absolutely lovely. A little dim for my liking, but from here on in, it's finishing touches. Firstly, let's add some regular particle halos as trails like we did in the first tutorial, to help give a sense of expansion. This time I used lots of low-alpha particles rather than a few bright ones. The exact settings were again arrived at through lots of trial and error. I used the texture I used for the fire, but did not set halo tex. Oddly this still seems to have some affect on the colour. Halo size was 0.3 and alpha was initially 0.3 too, set to fade over 25 frames (i.e., one quarter of each particle lifetime. Halo material from frames 0 to 100 is mapped onto the lifetime of each particle). Add effect was 0.5. Play with the particle settings so that they appear to be emerging from the cloud, like so (consult the .blend if needed) :

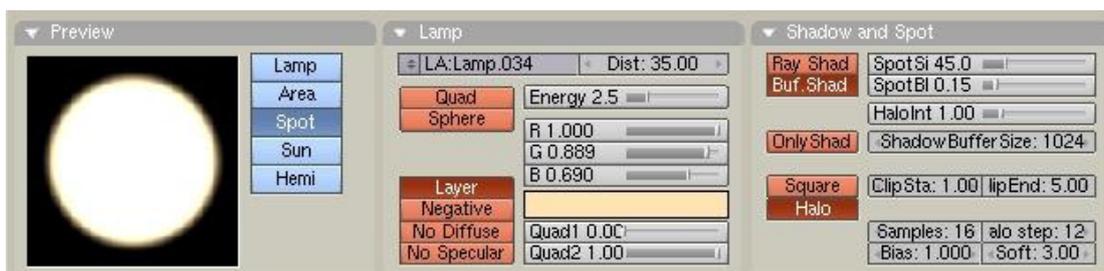


Note that these particles on their own won't look as good as when rendered with the main fireball, because the add effect can't work on a black background – there's no colour to add to. So for test rendering you'll need to render them together. I recommend border select rendering : shift+b and left-click and drag in the camera view to define the area you want to render.

I also added some sparks, to represent very hot bits of debris, like you get off a bonfire. This is just another particle halo system. Small halo size, strong add effect, quite strongly yellow-coloured halos. Not necessary to animate their material.

We're almost there. A real explosion illuminates objects around it. I thought it would look quite nice to have some shafts of light coming out (and I was right, too). This is done with a volumetric halo spotlight, aimed straight at the camera. Now I can't abide spotlights, they're relics of an earlier Dark Age, but they are very useful for this. Aiming it at the camera fakes rather well the appearance of a spherical halo (normally this is better, in my opinion, than using multiple spotlights as the halos will overlap and cause weird effects). Be sure to put it on another layer, and set it to Layer, else it will illuminate the fireball and since the halo needs to be bright, this would be bad.

The exact settings I used are shown below. Key points to note are setting the samples to 16 and the halo step to 12 :

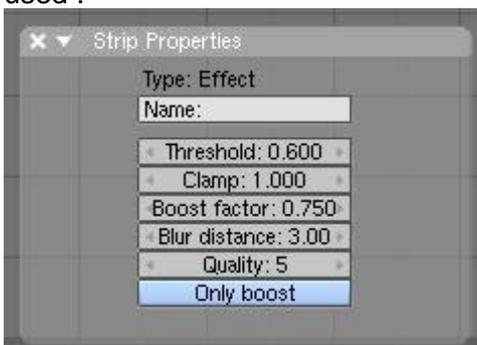


Now we need some objects to cast shadows. As shown above, reduce the size of the clipping (the region where shadows are calculated). Within it, place some small cubes randomly. The material settings for the cubes are not important as they should be completely obscured by the explosion, as long as they cast shadows.

Although it doesn't actually illuminate the fireball, it will still affect its appearance. This is because the fireball is partially transparent. We started off with a black background, but now the halo gives a yellow background to the fire. This tends to reduce the rather nice, fluffy appearance we had earlier. You have to strike a balance between a nice bright glowing explosion bursting with energy and a rather dim, but very volumetric looking explosion. There's plenty of room for improvement in this area.

Now the debris. I was getting bored and lazy by this stage, so I only used one emitter. I compensated by having two separate chunks of debris in the child object. I aimed to create rocks, which I did by proportionally editing icospheres and fractal subdividing. I also applied a subsurf modifier, so they wouldn't be too angular, and using auto smooth, so they would have some very sharp edges. I used these mesh editing techniques on the particle emitter too, so getting a more random distribution of debris. A simple procedural clouds texture on the rocks creates bumpmapping. The rocks are quite big, and some of them are visibly seen to be emerging from the cloud. This further enhances the appearance of a truly 3D cloud. Probably, it would be best to use a single lamp on the rock layer as in the first tutorial to illuminate them, or maybe even another spot halo, but I didn't find this necessary.

The very last step. Render your explosion now, and you may still find it somewhat dim. This is countered by taking the image and using the glow effect in the sequencer. Works better than simply upping the lamp energy, because it also blurs the glow, which is far more realistic. Here's the settings I used :



And that's it ! In practise, it's much easier to just append the whole shebang into a scene and tweak it. It can take an awful lot of tweaking, especially this second explosion which has many different aspects to it. Hopefully, this tutorial will explain more clearly what each component does.

One final note of caution : the second .blend is rather unstable ! I don't know why this is. Treat it with care.