

Energy Explosion Framework documentation

Energy Explosion Framework is a system for easily and conveniently creating spectacular explosions in your game.

Explosions have different visual and sound effects and different behavior. From the usual big bang, scattering objects falling into the affected area, to a sci-fi energy explosion, after which there is a collapse, pulling all the objects that fell into the affected area into the explosion point.

Quick start

1. Open the "DemoScene" and click Play in the Unity editor window. Click on the wall to create an explosion.
2. Using the UI buttons located on the screen, you can select one of the three explosion modes, switch the effects of explosions and explosion types (for more details on all modes and types, read further in the detailed description).
3. Copy the controller prefab of the explosion mode you like to your project and place it in the root of the scene.
4. Copy the prefabs of the explosion effects you like to your project and add links to them in the previously copied controller in the same way as in the demo scene.
5. If necessary, you can change the available parameters of the controller or the effects of the explosion such as: radius or force of the explosion, a layer for displaying decals, ray length and others.
6. After the explosion effect has finished playing, GameObjects are deleted by themselves.

Detailed description

There are three modes of explosions available:

- *explosions using raycast* - the explosion occurs at the point at which the mouse was clicked. For this mode to work, the presence of the **ExplosiveRayMaker** prefab on the stage is required;
- *falling balls* - the object appears at the point of clicking with the mouse and under the influence of gravity falls down, exploding when it touches the floor or after the timer ends. For this mode to work, the presence of the **ExplosiveFallingBallsController** prefab on the stage is required;

- *shooting with explosive balls* - the ball shoots from the camera position in the direction indicated by the mouse cursor and explodes when touched with objects or by timer. For this mode to work, an **ExplosiveShotController** prefab must be present on the scene.

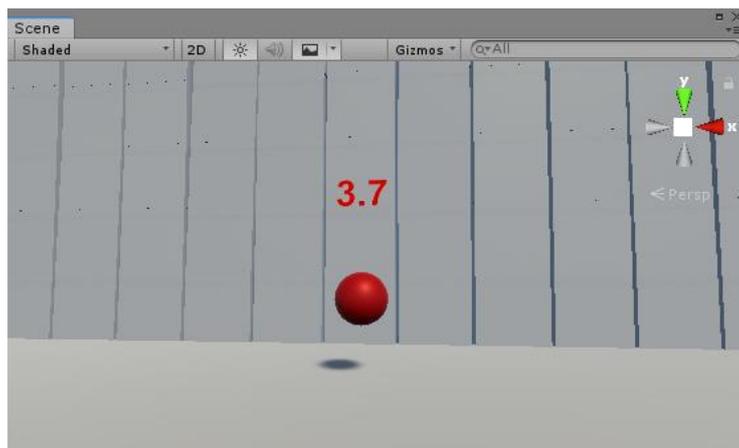
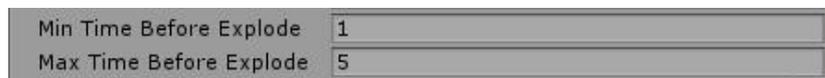
In the demo scene, the modes are switched by pressing the corresponding button. Required controllers turn on and off automatically.

The framework implements a timer explosion system.

The timer can be turned on and off, started when spawning a sphere, or when hitting an object.



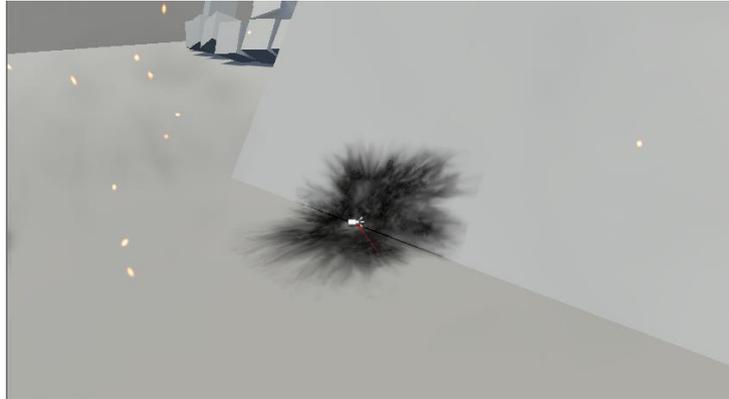
You can also set the timer time by changing the corresponding fields in the Exploder.cs component in the prefab of the corresponding explosive object. The time will be selected randomly in the specified range. If you need to set a constant time - you should specify it in two fields at the same time.



Decals.

Explosions leave decals on the surface near the explosion at any angle, using the customizable spotlight technology. The surface must be marked with a specific, pre-selected layer. The demo scene uses the Ground layer, but you can create and use any layer and any number of layers.

For clarity, in the editor, on the “scene” tab, the normal is drawn from the point of explosion to the planes onto which the decal “shines”.



The following types of explosions are available in the framework:

Big Explosion - A powerful explosion with a strong blast wave, scattering objects over a long distance.

Dust Explosion - A light explosion without fire, not leaving traces of an explosion on the surface and having a weak blast wave.

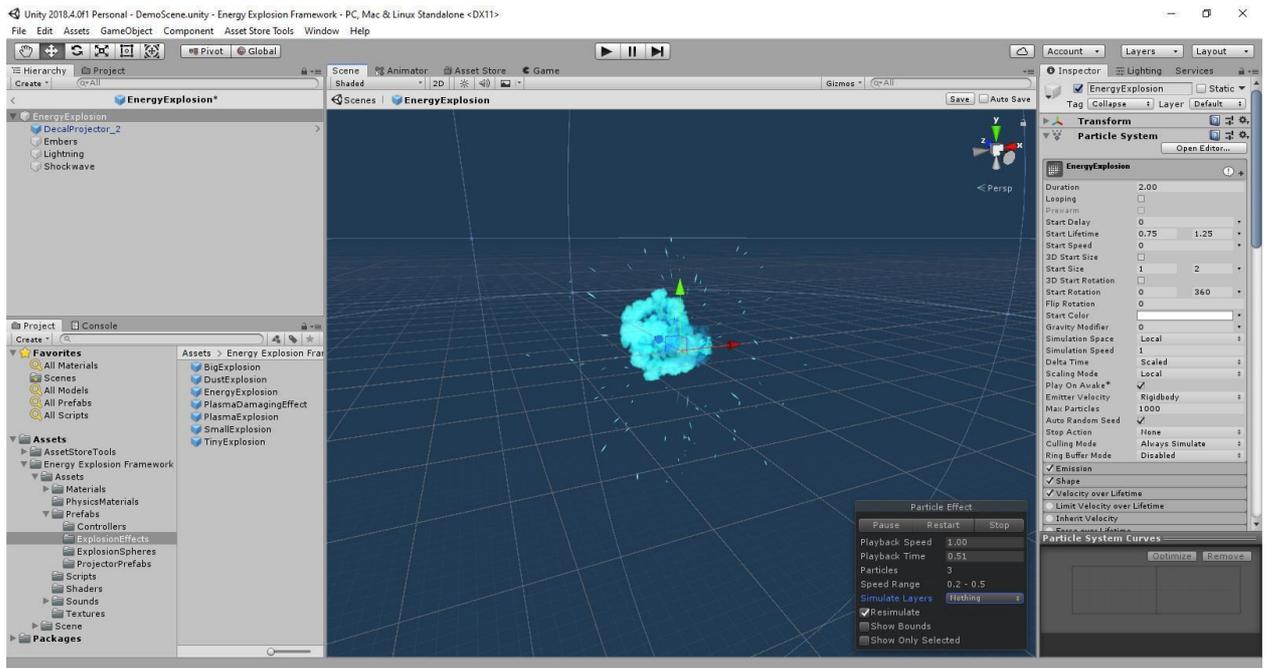
Energy Explosion - A cool energy explosion, leaving behind a collapsing point, attracting nearby objects.

Plasma Explosion - A plasma explosion that repels and imposes a damage effect in the form of electrical impulses on the affected objects.

Small Explosion - A small fiery explosion.

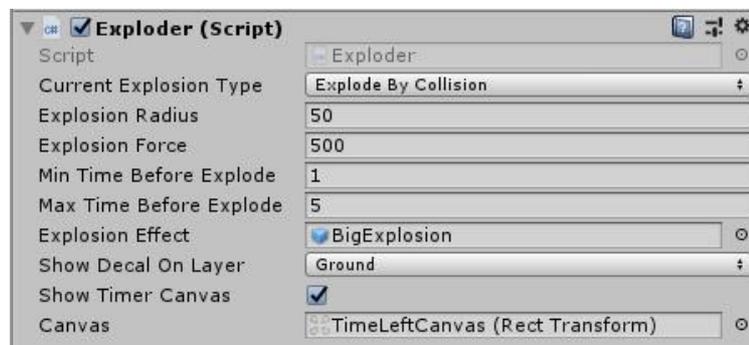
Tiny Explosion - A tiny explosion, with a small blast wave and an interesting visual effect.

All the visual effects of the explosions are available in the form of customizable prefabs that can be easily customized in the editor.

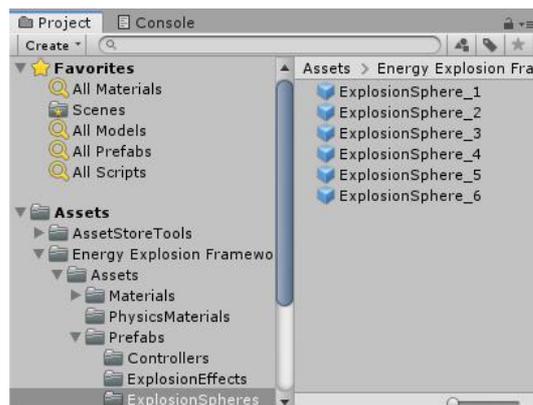


Exploding balls (or any other objects).

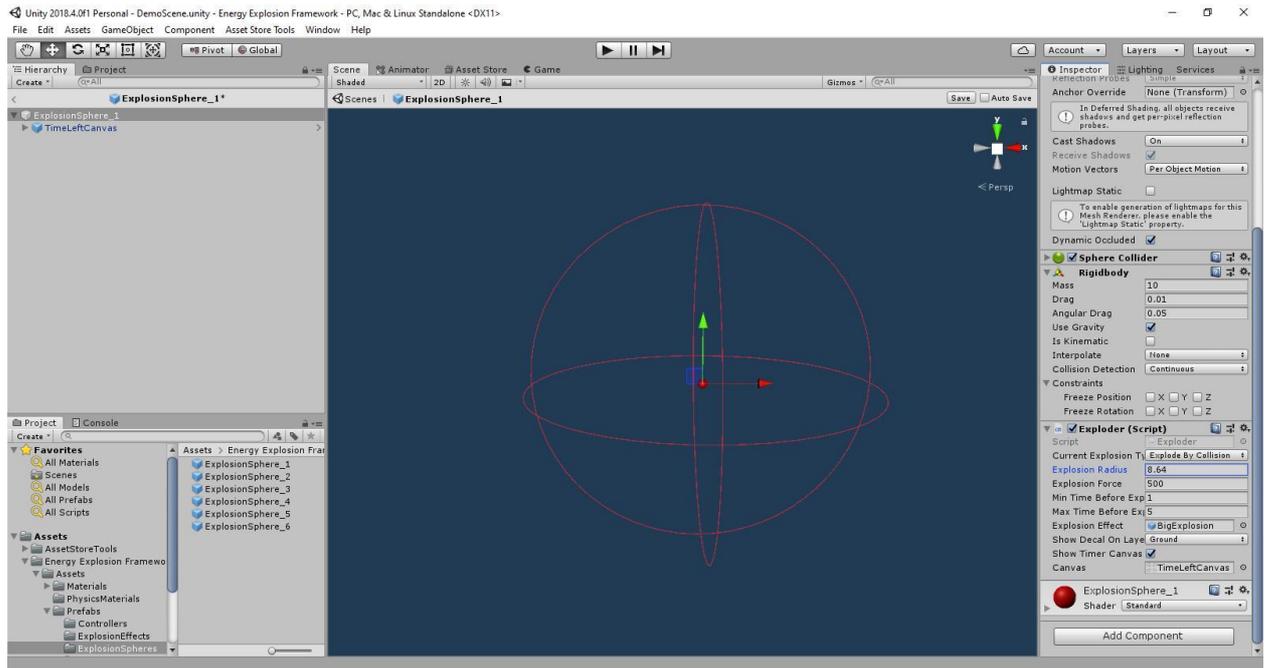
To create a new exploding object, just attach the Exploder.cs component to any object that has a rigidbody and collider, and configure it. All fields have a meaningful and understandable name.



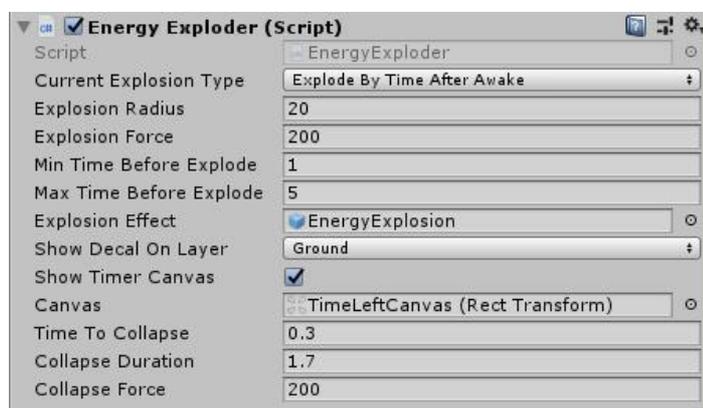
In the framework, prefabs of exploding balls are available for each visual effect of the explosion.



Each prefab can be visually edited by changing, for example, the radius of the explosion, set the strength, type and time of the explosion, as well as change the visual effect to any existing in the package or to any of your customized effects.



This framework is expandable and, if necessary, you can easily supplement it with your own types of explosions with non-standard behavior, simply inheriting from the Exploder base class and adding the changed behavior, as, for example, an energy explosion was made.



All scripts contain comments to facilitate understanding of what is happening.

The type of explosions using raycasts is not logically connected with the types of explosions using balls for the convenience of using these types in different projects.

The visual effects used in this framework are presented for demonstration! They were taken from the **Unity Particle Pack**, slightly modified and added sounds found on **freesound.org** and mixed in **Audacity**.

In case of problems or wishes - contact me s_tim@bk.ru, and I will answer you!

Thank you for your purchase!