Easy Breakables

A object breaking script for Unity.

Made by Gampixi

Documentation

Thank you for buying Easy Breakables. Easy Breakables is designed to let you add breakable objects to your game in an easier way.

In the demo included with the project, you can find a script which should give you an idea on how to set up Easy Breakables with raycast shooting, for even more convincing breakables.

If you have any questions or problems to report, feel free to e-mail me at rudolfsagris@gmail.com

I hope you have a fun and smooth experience using Easy Breakables.

The sounds included are from freesound.org and use the Creative Commons Zero(no attribution) license.

Overview
Easy Breakables decides what damage level your broken object should be at by checking the damage the last hit did. Look at it this way – you can have one giant boulder smash a wall and break it into 20 pieces, or have a ping-pong ball hit a glass panel and break it into 5 pieces after having hit the panel 20 times.
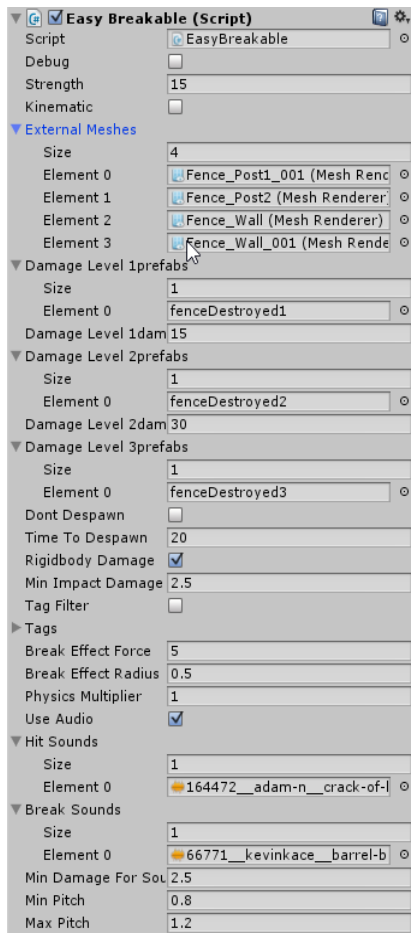
Variable explanation

Debug-This will enable messages about damage taken and breaking.

Strength-Self explanitory. Health of breakable.

Kinematic-Whether the rigidbody attached to the breakable is kinematic or not.

External Meshes-Assign any meshes not on the main object here, e.g. the included fence prefab.

Damage Level 1-3 Prefabs-Assign the broken prefabs for the damage level here.

Damage Level 1-3 Damage-The maximum last hit damage for that damage level. Going over that will go to the next damage level.

Dont Despawn-Whether debris from the broken object should despawn over time.

Time To Despawn-Time until debris despawns.

Rigidbody Damage-Whether this breakable gets damaged from impacts from other rigidbodies.

Min Impact Damage-Minimum damage from a rigidbody to take damage.

Tag Filter-Choose which tags to take rigidbody damage from.

Tags-Tags for the Tag Filter.

Break Effect Force-The force multiplier for the breaking impact effect (AddExplosionForce at impact point)

Break Effect Radius-The radius of the AddExplosionForce at the impact point.

Physics Multiplier-Multiplier for damage taken from rigidbodies.

Use Audio-Whether the breakable should emit sound.

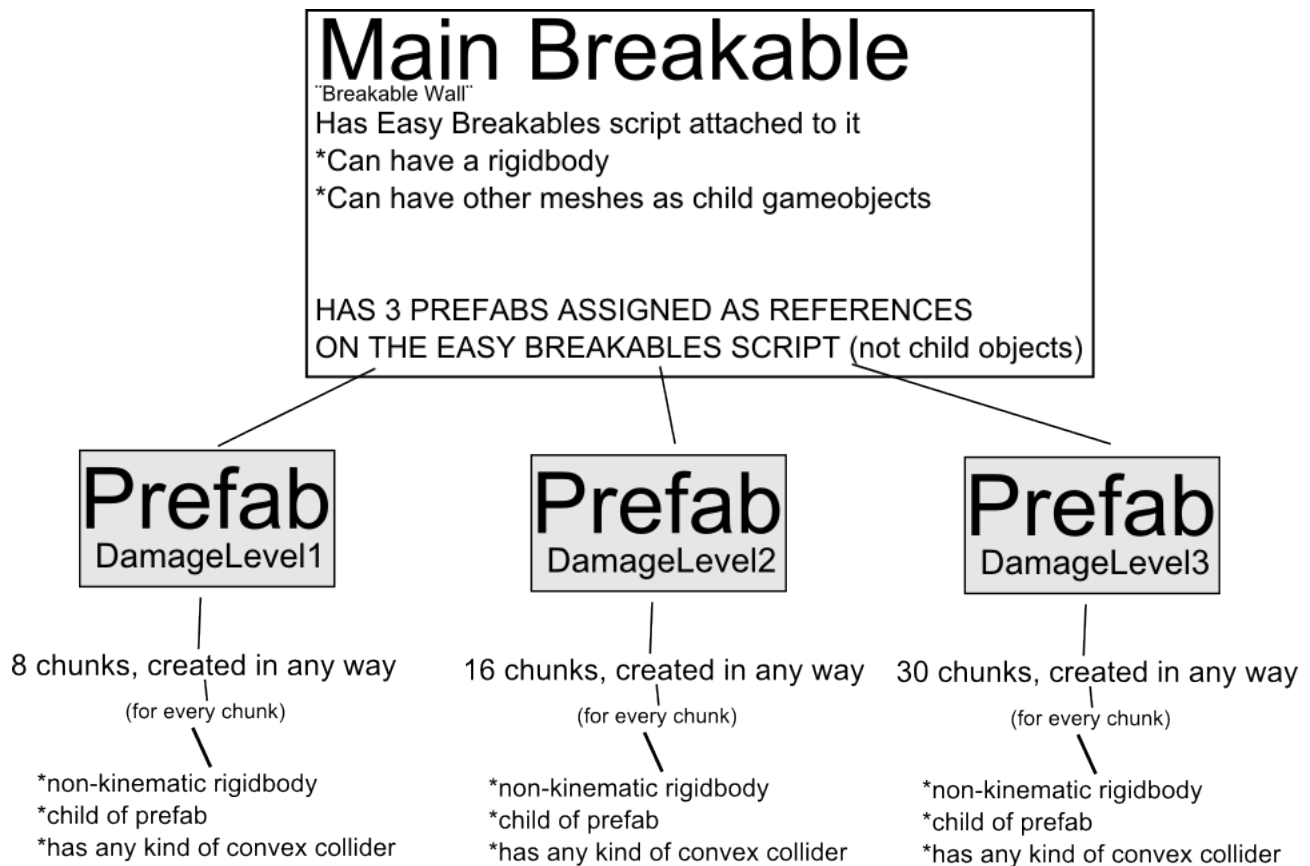**Hit Sounds**-List of sounds to emit when the breakable takes damage.

Break Sounds-List of sounds to emit when the breakable breaks.

Min Damage For Sound-Minimum damage to emit a hit sound.

Min/Max Pitch-Minimum and maximum random pitch for emitted sounds.

Basic setup overview

In my opinion, the best way to learn how a breakable is set up is to study it yourself, but this flowchart should give you a better understanding on how a breakable is set up.

Main Breakable
¨Breakable Wall¨

Has Easy Breakables script attached to it

*Can have a rigidbody

*Can have other meshes as child gameobjects

HAS 3 PREFABS ASSIGNED AS REFERENCES
ON THE EASY BREAKABLES SCRIPT (not child objects)

Prefab
DamageLevel1

8 chunks, created in any way

(for every chunk)

*non-kinematic rigidbody
*child of prefab
*has any kind of convex collider

Prefab
DamageLevel2

16 chunks, created in any way

(for every chunk)

*non-kinematic rigidbody
*child of prefab
*has any kind of convex collider

Prefab
DamageLevel3

30 chunks, created in any way

(for every chunk)

*non-kinematic rigidbody
*child of prefab
*has any kind of convex collider

The chunks in the damage level prefabs can be created in any way. The can be made with the Blender's Cell Fracture tool, or created by hand. All they need to be is separate objects with separate rigidbodies and separate colliders.

Damaging the breakable with a script (Gun example)

The most common way to damage a breakable is by shooting it with a gun. Easy Breakables has a public function to make it easy to damage breakables from an external script.

```
public void Damage(float damage)
public void Damage(Vector3 pos, float damage)
```

If you call Damage without the pos overload, then pos will default to Vector3.zero, which is the middle of the scene. This isn't desirable for most cases, so if you damage the breakable with a raycast, you should provide the position of the hit too.

The best way to do this is like this:
```
RaycastHit hit;

if(Physics.Raycast (transform.position, transform.TransformDirection (Vector3.forward), out hit)) {
        if(hit.collider.tag == "Breakable") {
        hit.collider.gameObject.GetComponent<EasyBreakable>().Damage (hit.point + (hit.normal *
0.25f), force);
        }
}
```

In this example we assume that the breakable has a tag Breakable on it. Also, notice that in addition to the hit point, we also add the hit normal * 0.25 to the hit point. This is so the hit isn't like almost inside the breakable and this makes a better impact effect/explosion once the breakable fractures.

Known problems

My breakable goes all over the place when it breaks
Make sure that your forces are all reasonable. Make sure that the damage prefab chunk colliders do not overlap over each other too much.

My damage prefab chunks fall through everything
Do they have colliders? If so, are the colliders Convex? In the case of the Mesh Collider, you have to set the Convex checkbox to enabled.

My not-broken mesh is still visible once it breaks
Make sure that Easy Breakables knows that there is a mesh. If you have a breakable consisting of multiple meshes in one, make sure that you have assigned all the meshes under External Meshes.

Nothing collides with my breakable
Make sure that your breakable (the object with Easy Breakables script on it) has a collider.

I get massive frame drops when something breaks on my phone/tablet
Easy Breakables relies on having lots of rigidbodies acting as debris. Mobile devices do not handle that very well, you might want to consider having your damage prefabs have less chunks.