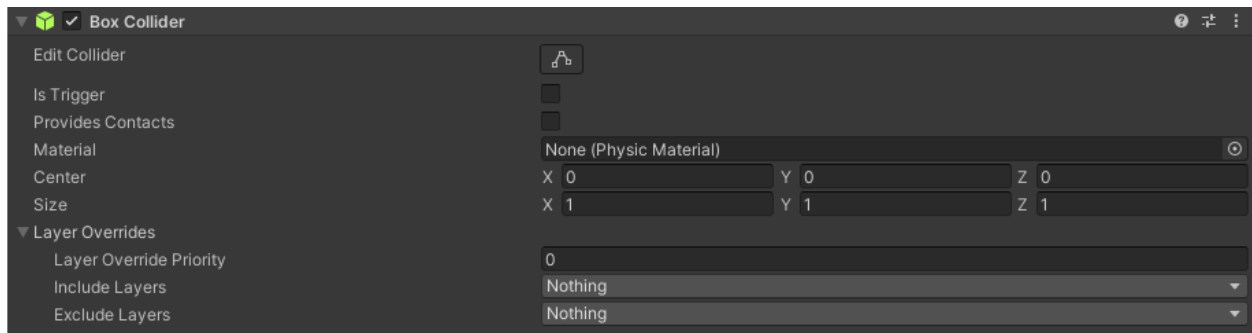


# Box Collider

Evan Svendsen & Devon Shields



The [BoxCollider](#) is part of the `UnityEngine.PhysicsModule` library.

It can be added to any game object from the **Add Component** menu by clicking **Add Component -> Physics -> Box Collider**

The [BoxCollider](#) component inherits from the base class `Collider` which is used to define all 3D collider components. There is an additional [BoxCollider2D](#) component that can be added from the **Physics 2D** menu but should not be used in 3D games. For 3D games, only use colliders from the **Physics** menu.

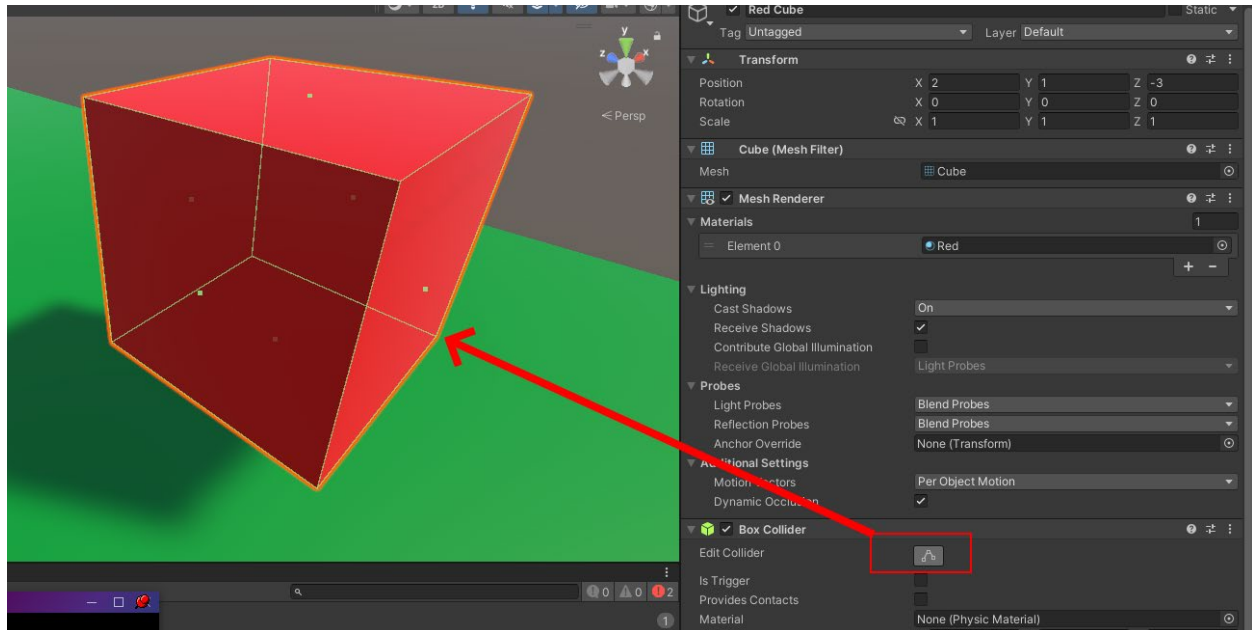
Box colliders will also be automatically added to Cube objects and Volumes. They can also be removed from a cube object for cubes that are visual only and have no interaction with the player. Although it should be noted that Volumes require a collider of any type to work, so the [Box Collider](#) is good enough for most use cases.

## Purpose

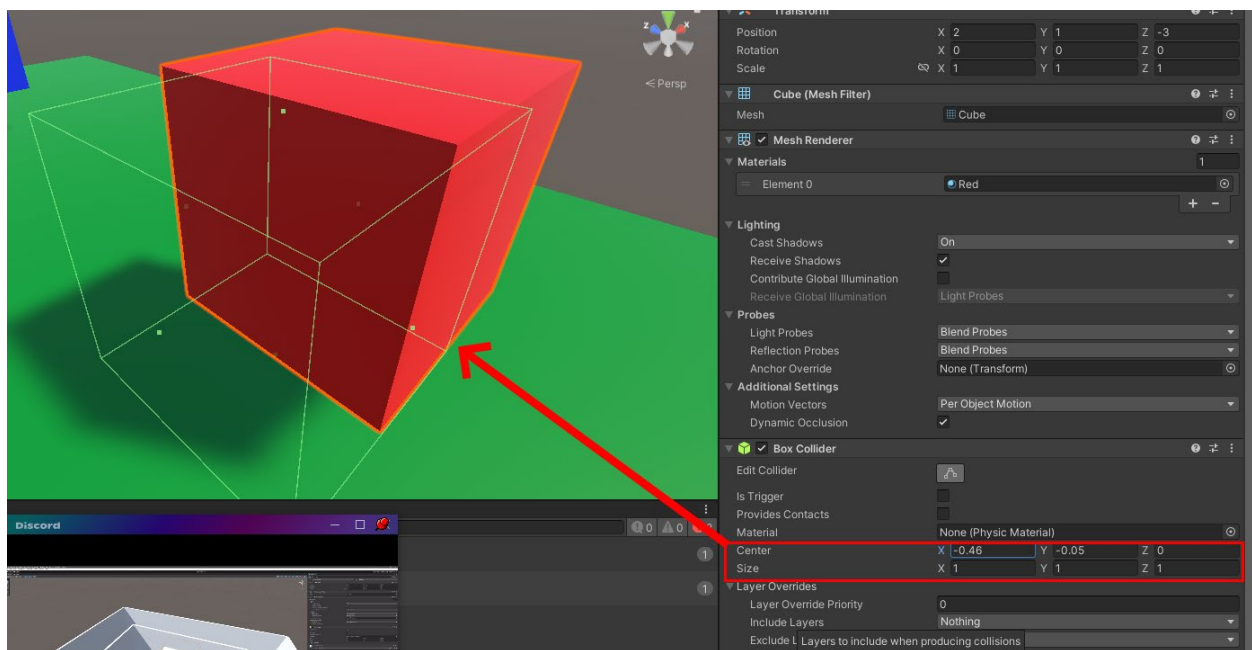
The [Box Collider](#) serves 2 primary purposes in Unity. The first is creating areas that player can not access (walls, floors, doors) preventing movement. The second is detection of game state changes with the [OnTriggerEntered](#) method inherited from [MonoBehaviour](#). For Volumes, [Box Colliders](#) define the area that volumetric effect is applied to.

## Configuration

A **BoxCollider** component exposes several properties to the Unity Inspector window. The first option is a button that allows you to resize the collider independent of the object it is attached to. **BoxColliders** automatically resize when their parent object is scaled.



The scale of the box can also be set with numbers in the input box as well. The resize button is useful for moving only the edge of the box you want. Otherwise, scaling happens evenly on either side of the central pivot point of the object.



The **Is Trigger** checkbox is the control that determines whether the box collider is solid (objects do not pass through) or a trigger object that objects can pass through. Below that, the **Provides Contacts** checkbox will force contact points to be sent in the subscribable **Physics.ContactEvent** method. Otherwise contact points are only injected based on 1 of 3 conditions:

- If the Collider or its Rigidbody have a script with a [MonoBehaviour.OnCollisionStay](#) method, all contacts will be generated for [Physics.ContactEvent](#).
- If the Collider or its Rigidbody has a script with either [MonoBehaviour.OnCollisionEnter](#) or [MonoBehaviour.OnCollisionExit](#) but not [MonoBehaviour.OnCollisionStay](#), enter and exit contacts will be generated for [Physics.ContactEvent](#), but not stay contacts.
- If the [PhysicsVisualizationSettings.showAllContacts](#) property is set to true, all Colliders will generate all contacts for visualisation purposes!

Reference: <https://docs.unity3d.com/ScriptReference/Collider-providesContacts.html>

Entering and exiting a Trigger Box Collider can be detected by using the [OnTriggerEnter](#) and [OnTriggerExit](#) methods of a mono behaviour script. The [OnTriggerStay](#) method is called every frame that a physics object remains inside of a collider.

**NOTE:** These methods work for every type of 3D collider, but **Box Collider** as well given its inheritance from **Collider**.

The **Physics Material** field of the component is used to assign a physics material to the collider that controls friction and bounciness on the surface.

The Layer Overrides collapsable contains 3 options.

**Layer Override Priority:** If 2 colliders are conflicting on the priority of the layer, the collider with the highest priority is used.

**Include Layers:** Additional layers to include for collision detection not defined in the Physics matrix of the **Project Settings**.

**Exclude Layers:** Layers to exclude for collision detection.

**NOTE:** If the same layer is both included, and excluded, the excluded one takes priority.

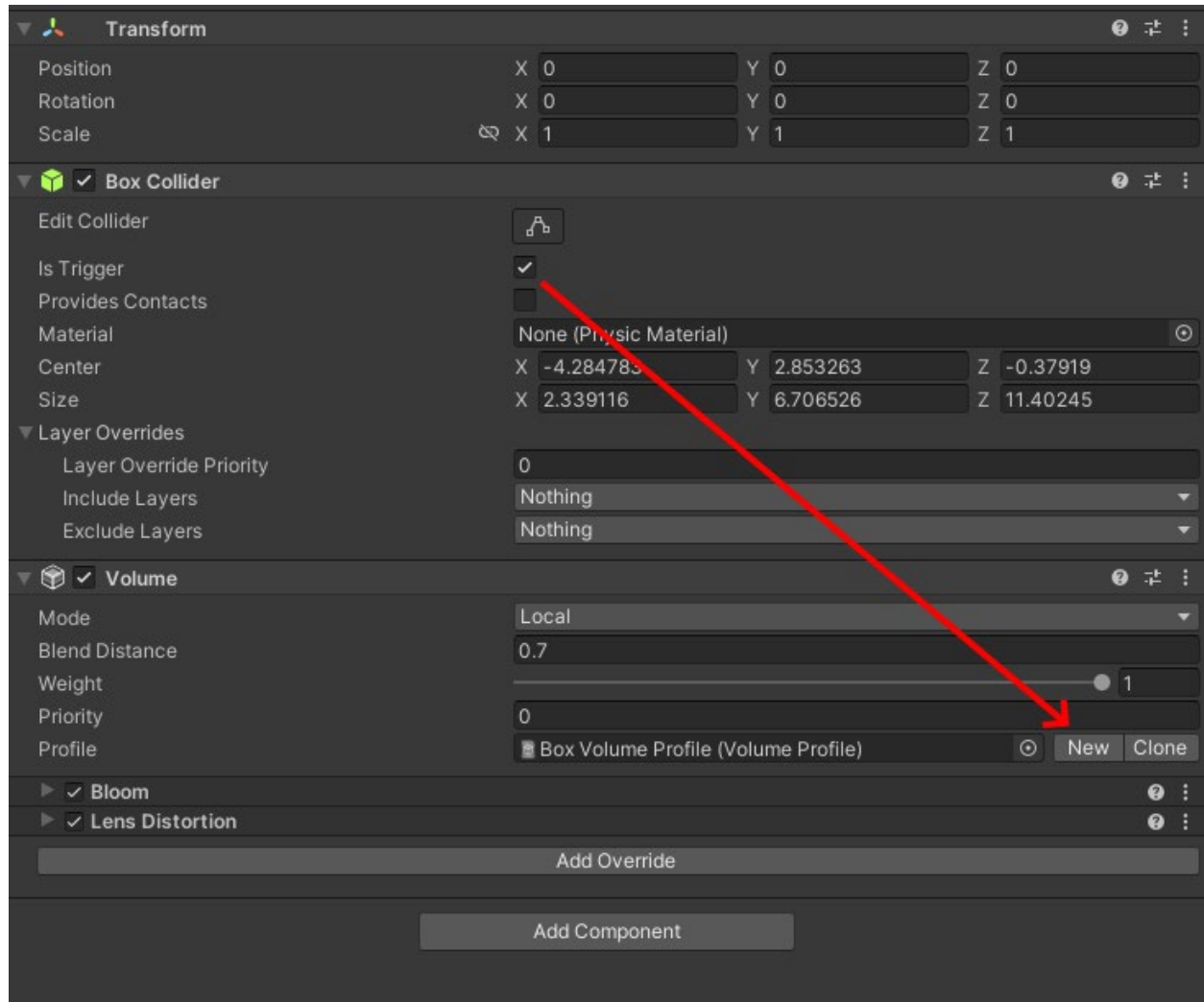
## **Code**

All the documentation for properties and methods of the box collider can be referenced for any modifications that can be made at runtime through code:

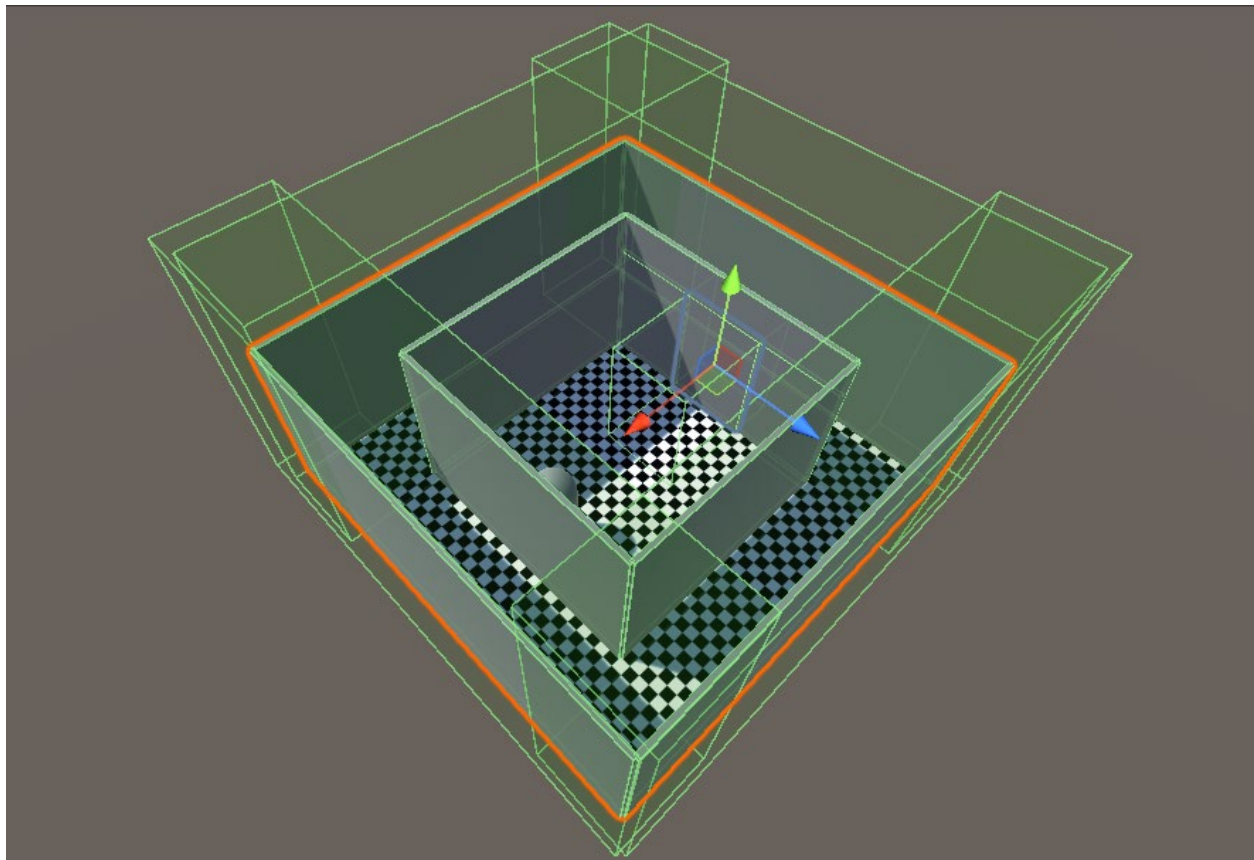
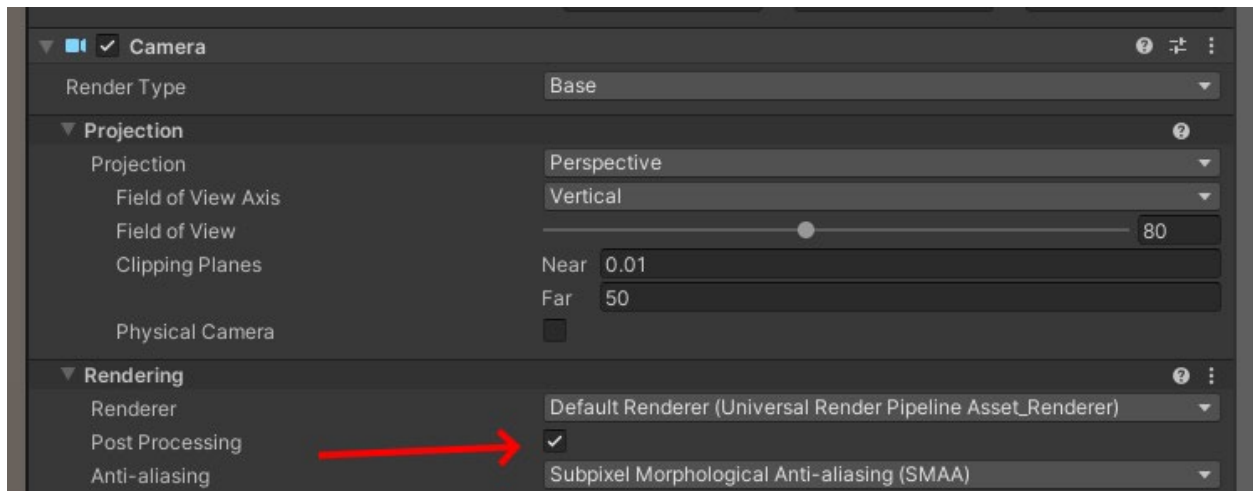
<https://docs.unity3d.com/ScriptReference/BoxCollider.html>

## Volumes

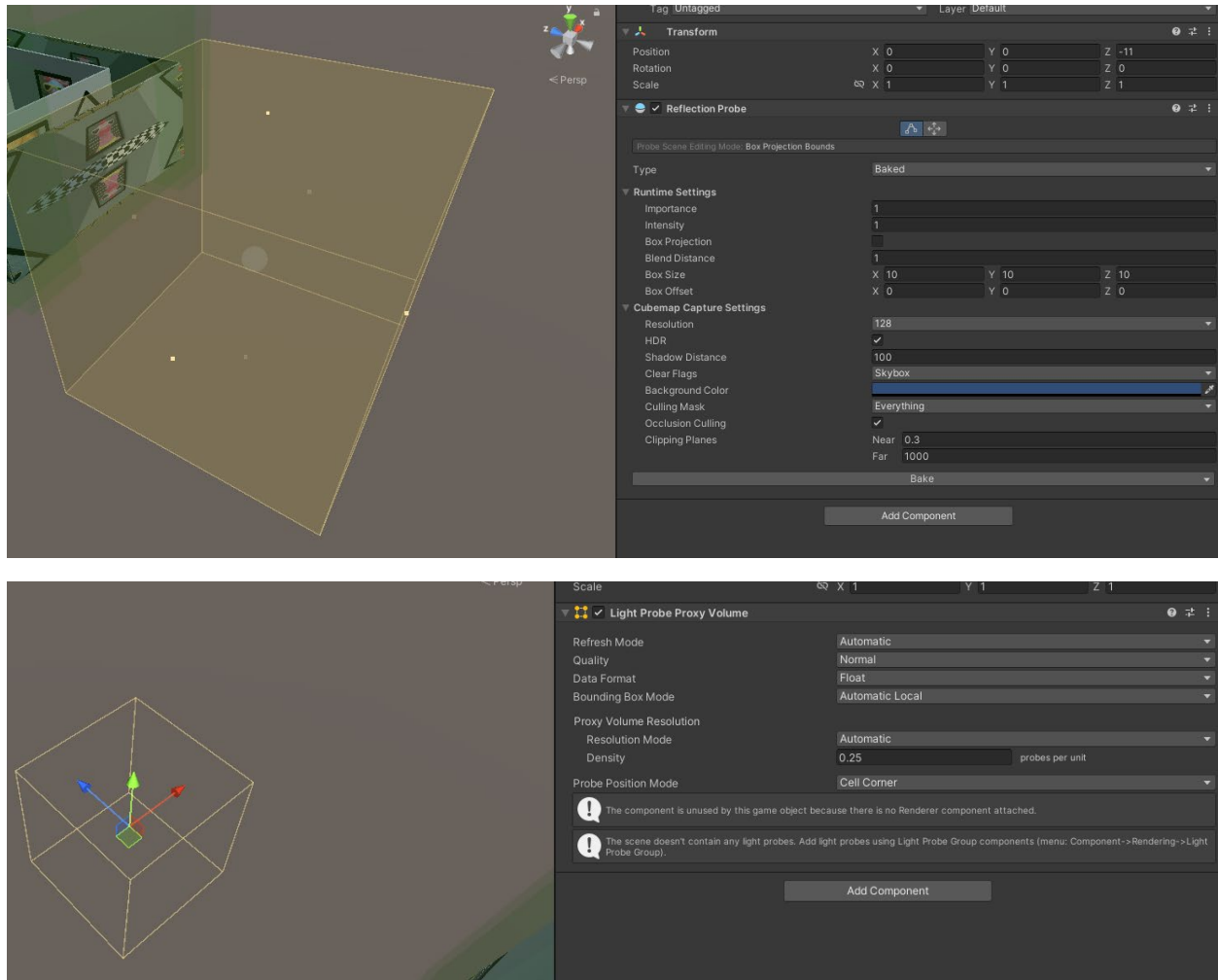
Volumes require a box collider that is a trigger to define a region of post processing. This can be used to add camera effects and color corrections to URP projects. And water and fog volumes for HDRP render pipeline.



To create a post processing volume, you need only add a volume, create a new post processing settings file. Define the area of impact using the box collider boundaries. And ensure to set the post processing checkbox to True on the camera being rendered.



## Light Probes & Reflection Boxes



**Light Probes** and **Reflection Probes** are another form of the box collider component that do not interact with the player or other physics objects. They instead are used to control how lighting and rendering happen in your scene. For example, a mirror can be made using a reflection probe collider if the mirror has a material applied with a 1.0 smooth value and it exists within the volume of the reflection probe box collider.

