# RTGizmo Version 1.3
# Documentation

## Overview

RTGizmo is a run time position, rotation and scale gizmo that can be used in your existing or new Unity3D projects. The goal of RTGizmo is a simple, but powerful gizmo control that can be used for manipulating objects at run time for level editors, player decorating or any other use you may find for it.

## Features

- Position, rotation and scale gizmo control
- Uses plane projection for accurate and smooth interaction
- Works with both Perspective and Orthographic cameras (Orthographic code provided by Alex Hill)
- Includes a 3D gizmo model (Can be replaced with your own)
- Includes custom gizmo handle shader
- Includes axis handles for x, y, z, xy, xz and yz
- Snapping options with configurable snap increments for all modes
- Keyboard shortcuts for switching between modes and snapping(Can be disabled)
- Gizmo control window for switching modes, manual value input and snap settings (Can be disabled)
- Included functions for controlling the gizmo settings from another script should you want to replace the built in gizmo control window
- Included with RTGizmo is the GUISpinner object class which is used in the gizmo control window for manual input. The GUISpinner object has built in functionality for right click to reset value as well a click timer on the plus/minus buttons. This means that if you hold down the plus or minus button for a moment it will start going up automatically instead of clicking repeatedly to increment the value.
- Core RTGizmo code available in Javascript and C# (C# translation assistance from Samuel Morais)

## Demo Scene

There are two included scenes under Gizmo Controller/Demo Assets which illustrate using C# or JS. Both scenes function the same, but Gizmo Controller Prefab and Box Selector scripts are either JS or C# depending on the scene.

The included demo scene contains 2 cubes that can be selected with the left mouse button. Once selected the RTGizmo object appears and you can manipulate the object. You'll also notice that the Gizmo Control window is only visible when the gizmo object is also visible.

In this example we also have a modified camera orbit script. Left mouse button + drag rotates the camera, right mouse button + drag pans the camera and mouse wheel zooms in and out.

The BoxSelector script demonstrates how to get the gizmo object, set the selected object and how to enable the gizmo controller. This is done quite simply with very few lines of code. Please see the Using The RTGizmo section below for specific code examples.

## Adding the RTGizmo To Your Existing Project

Adding the RTGizmo to your existing projects is quick and easy. After importing the RTGizmo package in to your scene simply drag the Gizmo Prefab, located under Gizmo Assets/Prefabs, in to your scene. The RTGizmo prefab will be invisible in the scene when you place it and should be on the top level of the scene hierarchy.

**IMPORTANT**
The gizmo object requires a special layer called gizmo. By default it uses layer number 8. If you are already using layer 8 for something else then after creating the gizmo layer you will need to change the LayerID field on the gizmo object. Click on the gizmo object in the scene hierarchy and change the LayerID field in the object inspector to the layer number matching the gizmo layer you created.

## Using the RTGizmo

Once the RTGizmo prefab has been added to your scene you will want to access it from your other scripts to set the selected object, set modes, make it visible etc..

The simplest and easiest way is to have a variable in your script like this...

JS:
var GC : GizmoController;

C#
public GizmoControllerCS GC;

Assign the RTGizmo object in the scene hierarchy to the variable in the Unity3D inspector. This will give your script direct access to the RTGizmo controller script.

To show the gizmo there are two steps. You must first set what object the gizmo should be controlling and then set it to show.

JS/C#:
```
//transform should be the transform of the object you want to manipulate
GC.SetSelectedObject(transform);
//GIZMO_MODE is an enum containing: TRANSLATE, ROTATE, SCALE
GC.Show(GIZMO_MODE.TRANSLATE);
```

## RTGizmo Public Variables

LayerID
Type: int
Default: 8
The scene layer the gizmo object uses. In order for the gizmo to work correctly you will need to create a layer called “gizmo”. Please ensure the layer number on the gizmo correctly associates to the layer number of the layer you created.

RotationSpeed
Type: float
Default: 250
Effects speed of object rotation when in rotate mode

SnappedRotationSpeed
Type: float
Default: 500
Effects speed of object rotation when in rotate mode with snapping enabled

Snapping
Type: boolean
Default: false
Whether snapping is enabled or disabled

MoveSnapIncrement
Type: float
Default: 1.0
Amount in units by which the object will move when snapping is enabled

AngleSnapIncrement
Type: float
Default: 5.0
Amount in degrees by which the object will rotate when snapping is enabled

ScaleSnapIncrement
Type: float
Default: 1.0
Amount in units by which the object will scale when snapping is enabled

AllowTranslate
Type: boolean
Default: true
Whether translate mode can be accessed by either shortcut keys or gizmo control window

AllowRotate
Type: boolean
Default: true
Whether rotate mode can be accessed by either shortcut keys or gizmo control window

AllowScale
Type: boolean
Default: true
Whether scale mode can be accessed by either shortcut keys or gizmo control window

EnableShortcutKeys
Type: boolean
Default: true
Determines whether the shortcut keys can be used for changing gizmo mode and snapping

TranslateShortcutKey
Type: String
Default: “1”
Shortcut key used to switch to translate mode. Will not work if EnableShortcutKeys is set to false

RotateShortcutKey
Type: String
Default: “2”
Shortcut key used to switch to rotate mode. Will not work if EnableShortcutKeys is set to false

ScaleShortcutKey
Type: String
Default: “3”
Shortcut key used to switch to scale mode. Will not work if EnableShortcutKeys is set to false

SnapShortcutKey
Type: String
Default: “s”
Shortcut key used to toggle snapping mode. Will not work if EnableShortcutKeys is set to false

ShowGizmoControlWindow
Type: boolean
Default: true
Whether or not to show the gizmo control window. Set to false to disable the gizmo control window.

GizmoControlButtonImages
Type: Texture2D[]
Default: null
This holds the icons for the the Position, rotation and scale buttons located in the Gizmo Control window. To set these you will need to select the gizmo controller in the scene hierarchy, set the GizmoControlButtonImages size to 3 and then drag the icons you whish to use to each open slot. The element0 is position, element1 is for rotate and element2 is for scale. The demo scene has these already setup as an example.

## RTGizmo Functions

Show(mode : GIZMO_MODE)
Makes the gizmo and control window visible and sets the active mode (TRANSLATE, ROTATE, SCALE). Must call SetSelectedObject function with a valid object before calling the show function.

Hide()
Makes the gizmo and control window invisible

IsHidden()
Returns true if the gizmo is hidden and returns false if the gizmo is visible

IsOverAxis()
Returns true if the mouse is current hovering over any of the axis handles

SetSelectedObject(ObjectTransform : Transform)
Set's the transform for the object the gizmo will effect. The passed in value should is the transform of an object.

SetControlWinPosition(position : Vector2)
Set's the Control Window position in screen pixel coordinates. Pass in a Vector2 contains the X and Y coordinates for the window

ToggleSnapping()
Toggles snapping on and off

SetSnapping(snap : boolean)
Set's snapping to to the passed in snap value

SetAllowedModes(move : boolean, rotate : boolean, scale : boolean)
Set's what modes can be used in while the gizmo is active. This effects both shortcut keys and the gizmo control window buttons

SetMode(mode : GIZMO_MODE)
Set's the active mode. mode is an enum:
GIZMO_MODE.TRANSLATE
GIZMO_MODE.ROTATE
GIZMO_MODE.SCALE

GetMode()
Returns the active mode as an enum value
GIZMO_MODE.TRANSLATE
GIZMO_MODE.ROTATE
GIZMO_MODE.SCALE