1. Gizmo set up

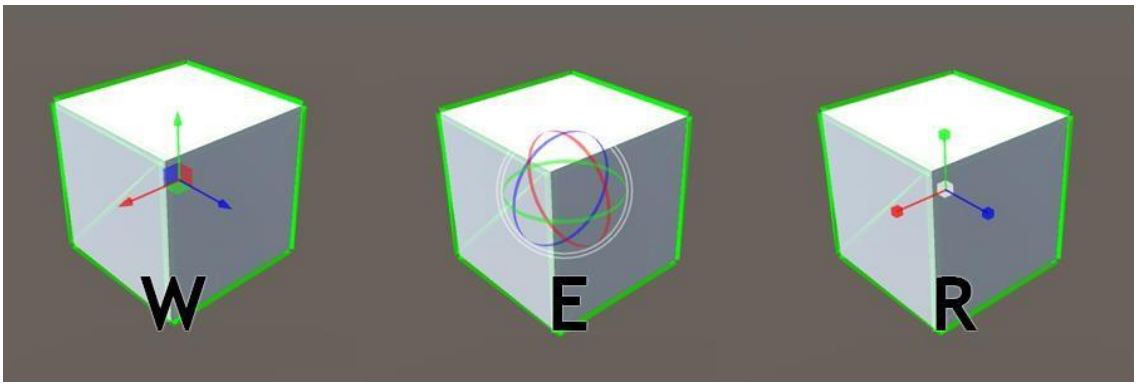
To start using moving, rotating gizmo, you will need to complete these steps:

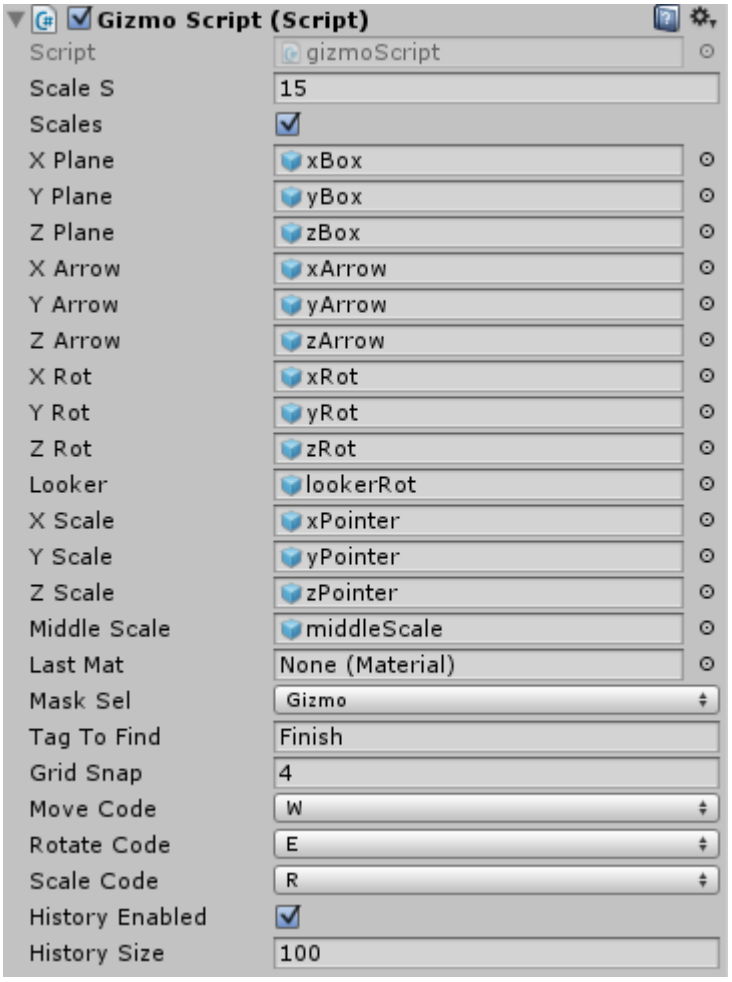
- Place „Gizmo“ prefab in to the scene
- Set Gizmo object of the scene layer to „Gizmo“, or anything else
- In Gizmo object of the scene in the script parameters set „Mask Sel“ to same layer, in this case „Gizmo“
- In Gizmo object of the scene in the script parameters edit „Tag To Find“. All objects in the scene with this tag can be controlled with gizmo, and objects without this tag will be ignored.

With this simple setup gizmo is ready for control, next we need to set up objects to control:

- Put any object in the scene and add „gizmoSelectable“ script
- Change this object tag to gizmo „Tag To Find“ tag (this is optional if you work only with one gizmo)

That is all with gizmo, now you should be able to control objects in the scene. To change how gizmo controls, press „W“ for move, „E“ for rotate, „R“ for move. For history, press ctrl + Z in standalone and in editor per only Z. These buttons can be changed.





Scale S: Defines how big gizmo is in view.

Scales: Toggles if objects are scaled by size.

Last Mat: Material that appears when you select axis to control.

Mask Sel: Gizmo mask, you need to have the same mask on gameObject.

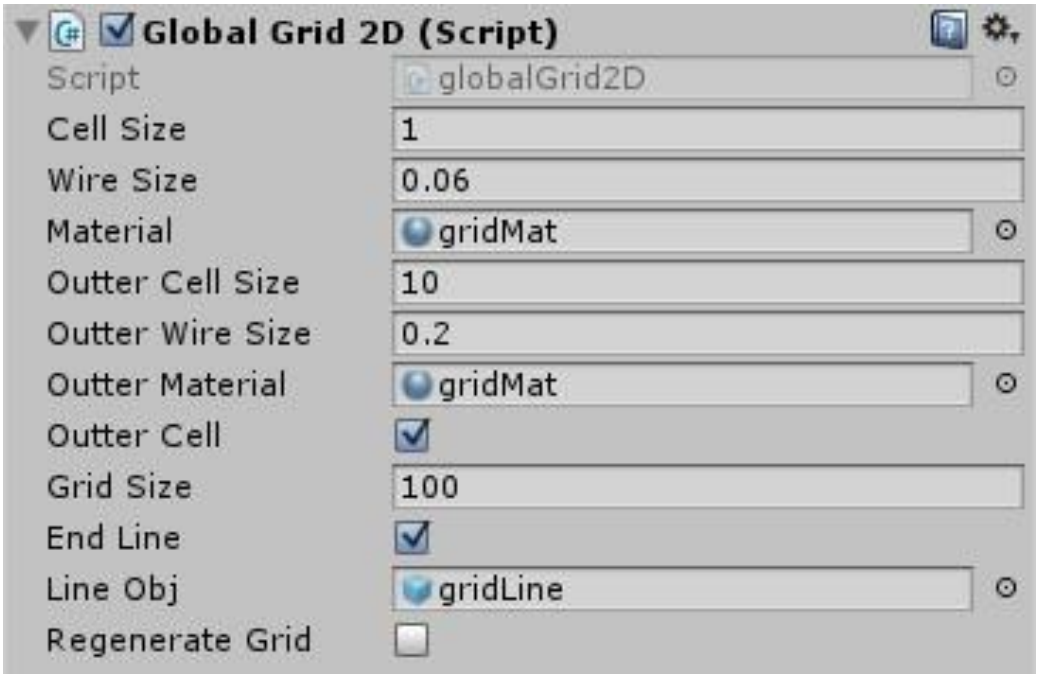
Tag To Find: With this gizmo you can control all objects with this tag.

Grid Snap: Snaps to 3d grid. 4 means „1/4“ which is 0.25. So every 0.25 steps controled object will snap.

History Enabled: If false, when pressing ctrl + z nothing happens.

History Size: How many moves does history record.

GRID

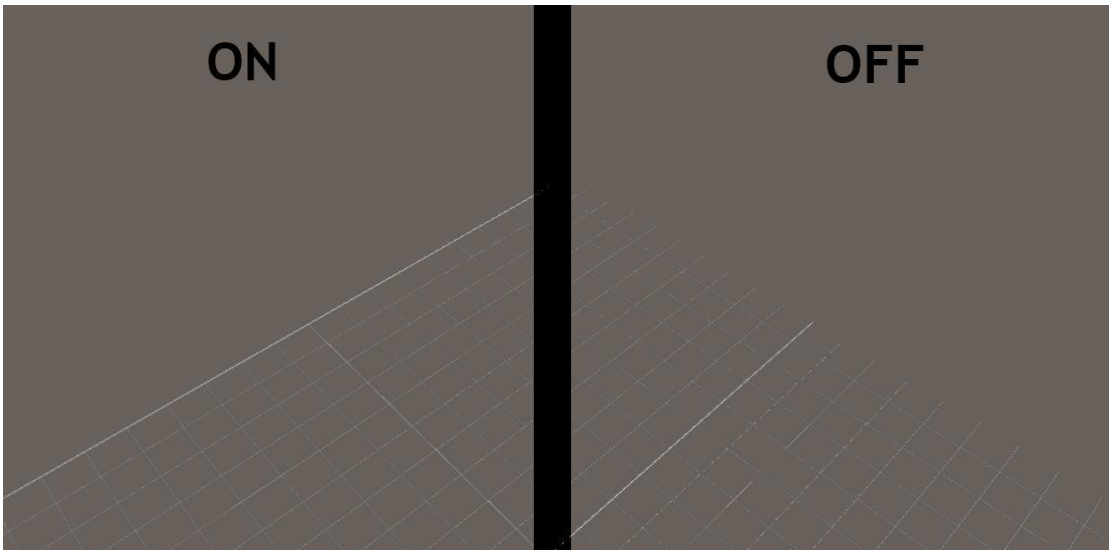


Grid has inner and outter cell. „Cell Size“ defines how big squares are, „Wire Size“ defines how thich lines are. Note, always use even numbers to avoid problems.

Outter Cell: Toggles between showing and hiding outter cells.

Grid Size: Overal size of the grid.

End Line: Toggles between showing and hiding end lines.



Line Obj: Objects of what grid is made. You can experiment with different objects, but pivot should be at the start and object facing forwards (to Z axis).

Regenerate Grid: Does what it says. Grid doesn‘t regenerate dinamically.

FOR SCRIPTING

globalGrid2D.cs

```
public float cellSize = 1f;
public float wireSize = 0.06f;
public Material material;
public float outterCellSize = 10f;
public float outterWireSize = 0.2f;
public Material outterMaterial;
public bool outterCell = true;
public float gridSize = 12f;
public bool endLine = true;
public GameObject lineObj;
public bool regenerateGrid = false;
```

gizmoSelectable.cs

```
[HideInInspector]
public bool selected = false;
public bool showHighlight = true;
public float wireframeSize = 0.06f;
public Color highlightColor = Color.yellow;
[HideInInspector]
public bool highlighted;
```

gizmoScript

```
[HideInInspector]
GameObject[] control = null; // Objects which gizmo controls
public float scaleS = 15; // Scale based on main camera distance
public bool scales = false;
[HideInInspector]
public Vector3[] offset; // Offset from controlled objects
Vector3 thisOffset = new Vector3 (0, 0, 0); // Offset from gizmo selected obj to mouse
public GameObject xPlane, yPlane, zPlane, xArrow, yArrow, zArrow;
public GameObject xRot, yRot, zRot, looker;
public GameObject xScale, yScale, zScale, middleScale;
[HideInInspector]
public GameObject selected; // Selected gizmo part
[HideInInspector]
public Material mat; // Used to assign material when selected
public Material lastMat; // Used to assign right material to gizmo when deselecting
public LayerMask maskSel; // This gizmo mask for selection. Dont change it unless you know what you are doing
Vector3 screenPoint; // Mouse position on screen
[HideInInspector]
public static gizmoScript GS;
public string tagToFind; // With what tag objects to find
public float gridSnap = 4; // Grid size to snap to, it means 1/4 = 0.25 units will be snapped
[HideInInspector]
public int type = 0; // 0 for transform, 1 for rotate and 2 for scaling
public KeyCode moveCode = KeyCode.W;
public KeyCode rotateCode = KeyCode.E;
public KeyCode scaleCode = KeyCode.R;
Vector2 mos;
```