

Measure Tool

EpsilonDelta

1 Introduction

Measure Tool displays dimensions and distances of the gameobjects in the scene. One editor script with no setup lets you know how much space gameobjects take in worldspace units (in local or world directions). Can be used as persistent Sceneview GUI or as a new Unity feature - custom EditorTool.

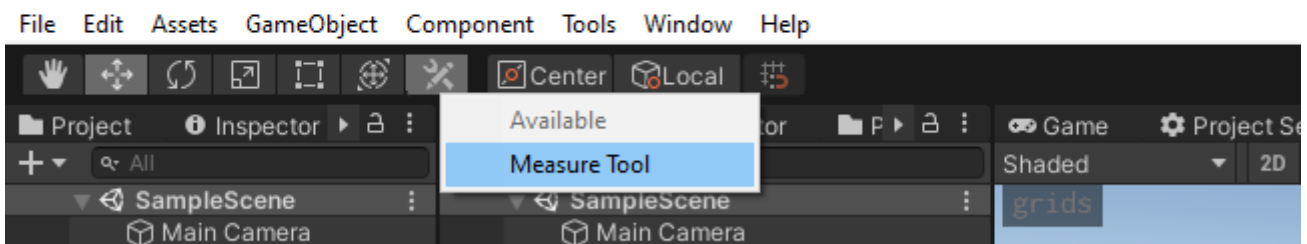
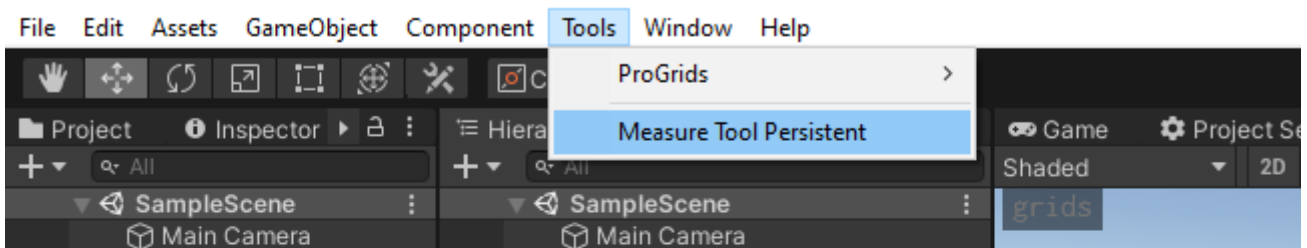
Measure Tool can be useful in many different scenarios. I, for example, always wanted to see real dimensions of gameobjects and UI elements in Canvas for better orientation, but Unity has no means to display it. Therefore I created this tool.

2 Requirements

Can be used with Unity 2019.3 or newer. No other requirements.

3 Setup

There isn't any technical setup, just import the asset from Unity Asset Store. You can start using Measure tool as a persistent tool going to Tools → Measure Tool Persistent or as an Editor Tool next to other Tools (move, scale, rotate...).



You might want to assign **keyboard shortcuts**: open Shortcut Manager (Edit → Shortcuts) find Measure Tool Persistent and Measure Tool and assign shortcuts. Defaults are alt+s and alt+a respectively. That's it.

measure		
Category	Command	Shortcut
All Unity Commands	Main Menu/Tools/Measure Tool Persistent	Alt+S
Binding Conflicts	Measure Tool	Alt+A

4 Measuring

This tool measures dimensions in world space units. When Local is set to false, it measures the worldspace aligned bounding box. When Local is set to true, it measures the bounding box aligned with local space axes. Beware that this tool takes scale into account, therefore local space does not need to be selected with move tool handles, but might be sheared/skewed (e.g. if gameobject is rotated and scaled and gameobject's parent is scaled). Some objects like 3D colliders transform in a special way, which is also accounted for.

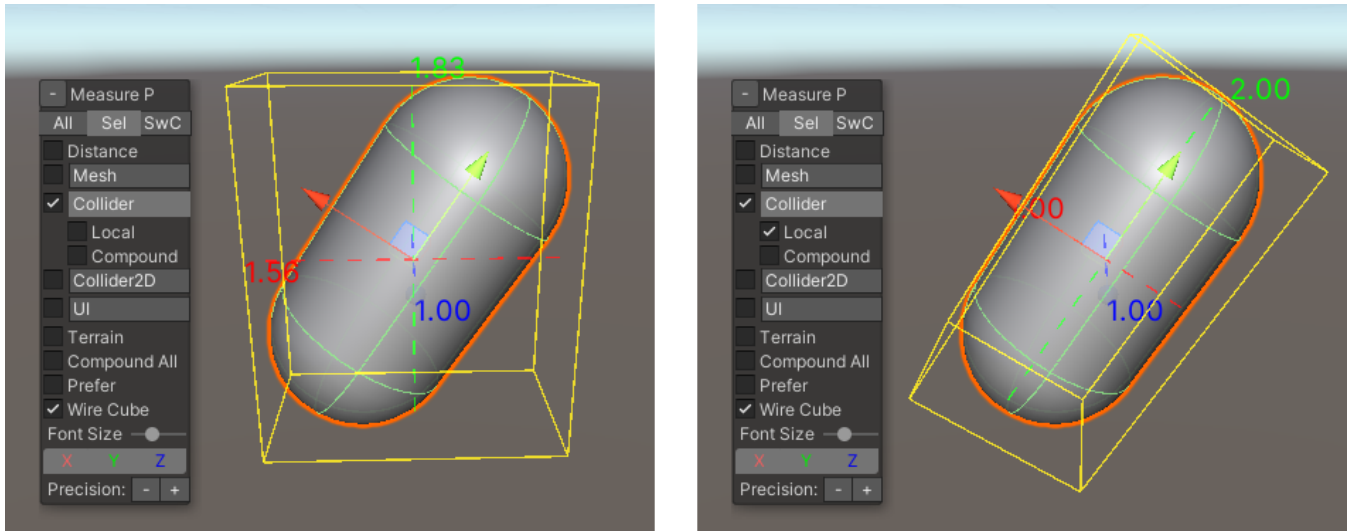


Figure 1: World (left) vs Local (right) measures

Measure Tool measures in edit time (works also in prefab mode) and in run time inside the editor (if you leave it on - beware of overhead). It does not alter the gameobjects in any way and is not included in the build.

Measure Tool can measure almost any basic game elements. Explicitly:

- Mesh, Sprite, statically batched mesh, procedurally generated mesh
- 3D Colliders, mesh colliders
- 2D Colliders
- UI (RectTransforms)
- Terrain

There are some limitations:

- Skinned mesh is not measured right now. I plan to add this feature
- When you change rotation of static batched mesh in runtime, measurement will rotate, but mesh not
- Composite 2D and Tilemap collider are only measured in world space
- Edge radius of 2D Colliders is not measured
- WheelCollider is not measured, it seems there is a bug in Unity that prevents it
- Does not work with DOTS (I haven't tested it)

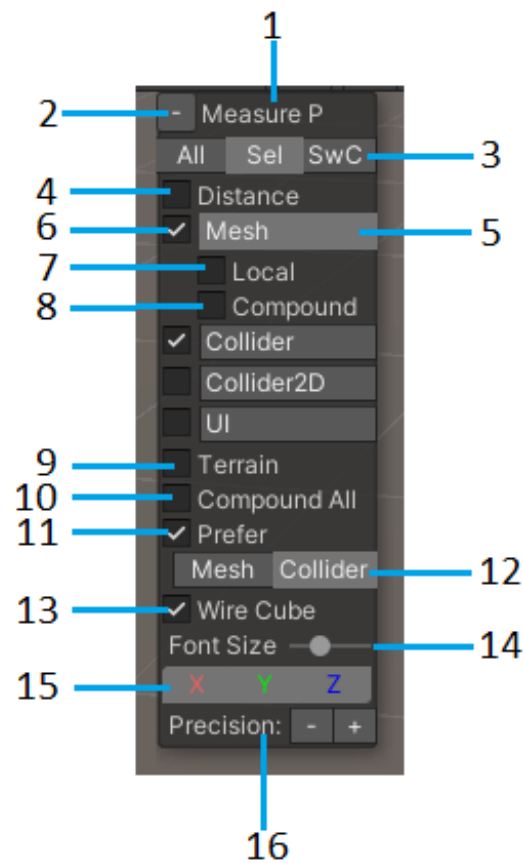
Unity is a huge engine and there might be many different scenarios that I didn't think of. If you find such a scenario (whether it is a bug or you feel that Measure Tool should measure something that it does not) and is not already listed here or in the roadmap, please contact me via email or create a github issue (links at the end of the document).

5 Manual

Toggle Measure Tool on (as described in Setup paragraph) and window in Scene View (Scene tab) will appear. This window let's you decide what to measure and how.

Description:

1. Name. "Measure P" or "Measure T" indicates if **P**ersistent or Editor **T**ool is turned on. You can position window by dragging the name
2. Foldout Button. Use to fold/expand Measure Tool window



3. Choose what should be measured in the scene. "All": all gameobjects are measured, be aware that measuring too many gameobjects can slow down the editor. "Sel": only selected gameobjects are measured. "SwC": selected gameobjects with all their children are measured
4. Should distance be measured? (Between All/Selected/SelectedWithChildren gameobjects)
5. Mesh Foldout Button - fold/expand Mesh measuring options (similar for Collider/Collider2D/UI (i.e. Rect-Transforms))
6. Measure Mesh Toggle - should meshes be measured? (similar for Collider/Collider2D/UI)
7. Should measure in Local space directions? (or World space). Note that Local space means all combined transformations of all parents (some objects like Colliders are transformed in a special way, which is accounted for)
8. Show world space aligned bounding volume (dimensions and wirecube) of all measured meshes combined in world space (similar for Collider/Collider2D/UI)
9. Should Terrain be measured (TerrainCollider is measured as when Collider option is On)
10. Show bounding volume (dimensions and wirecube) of everything currently measured
11. If gameobject has Mesh and Collider components, should I measure both, or prefer only one of it
12. If prefer, then what?
13. Should a wirecube of the measured dimensions of gameobjects be shown? (compound bounding volumes will always show wirecube)
14. Font size of labels in the SceneView
15. Turn on/off individual axis measurement. Note that if two-dimensional object (like Collider2D or RectTransform) is measured in local directions, Z axis is never shown
16. How many decimals should be shown?

6 Asset structure

There is only one editor script in MeasureTool/Scripts/Editor folder called MeasureTool.cs and two icons in MeasureTool/Icons/ folder. Editor folder contains assembly definition file (.asmdef). You can move MeasureTool folder wherever you like (though you should of course avoid some special Unity folders like Streaming assets etc.). The MeasureTool.cs should be kept together with its asmdef in Editor folder, best keep it where it is relative to MeasureTool folder due to icons loading (icons are loaded with GUIDs, only if failed, relative path is used).

7 Known Issues

1. All the listed limitations in Measuring section
2. If your monitor scales UI, then the SceneView boundaries cannot be calculated properly (probably a bug in Unity) and Measure window can be dragged out of SceneView (I can't imagine this happen, but if the window is lost in the void, widen SceneView or restart Unity)

8 Roadmap

Prioritized list (1. - highest priority) of things I want to implement:

1. SkinnedMesh measuring
2. LineWidth option (will be probably possible from 2020.2)
3. CompositeCollider2D local measuring
4. WheelCollider measuring
5. EdgeRadius measuring
6. Miscellaneous measuring: Cloth, particle system bounds, windzone, spriteshape linerenderer, navmesh, billboardRenderer
7. Closest point distance
8. DOTS support (it's too soon)

9 Contact

Email: epsildel@gmail.com

Github repo for issues: <https://github.com/EpsilonD3lta/Measure-Tool>