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 Contact

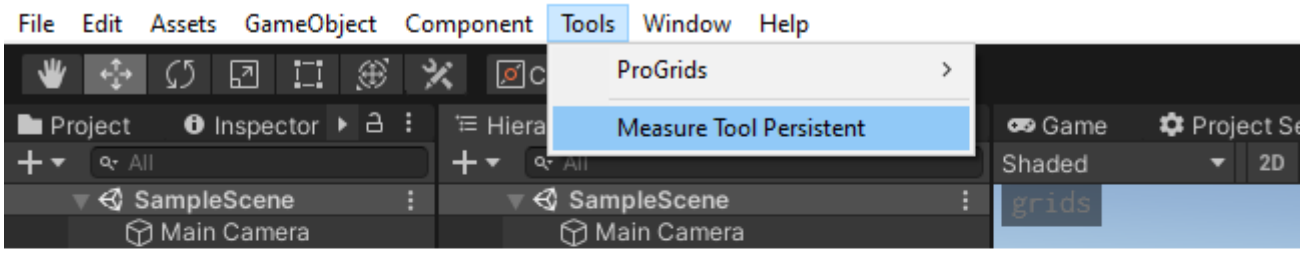
Please, **rate and leave a review** in the Asset Store if you are satisfied with the product or if you have a feature request or any other problem. It will help me to improve this asset. You can also contact me via:
Email: epsildel@gmail.com
Github repo for issues: https://github.com/EpsilonD3lta/Measure-Tool

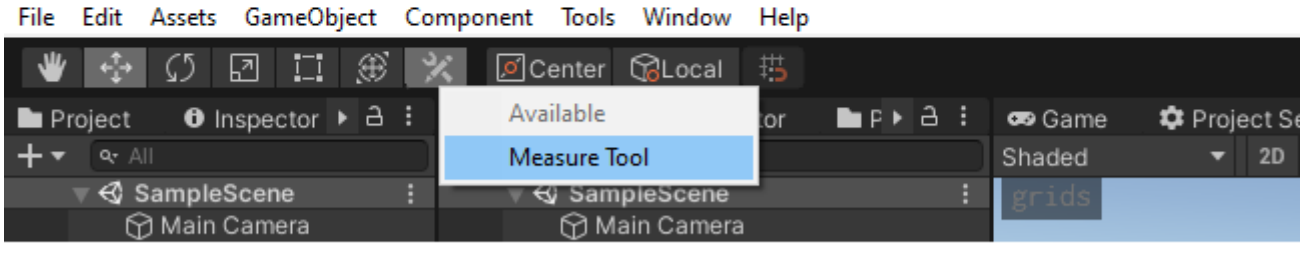
3 Requirements

Can be used with Unity 2019.4 or newer. No other requirements.

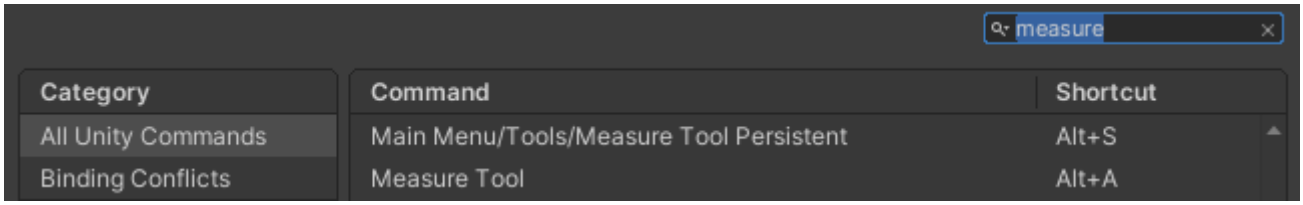
4 Setup

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...). Persistent tool persists through other tool selection (rotation, scale, move etc.), while Editor Tool does not.





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.



5 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 aligned 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.

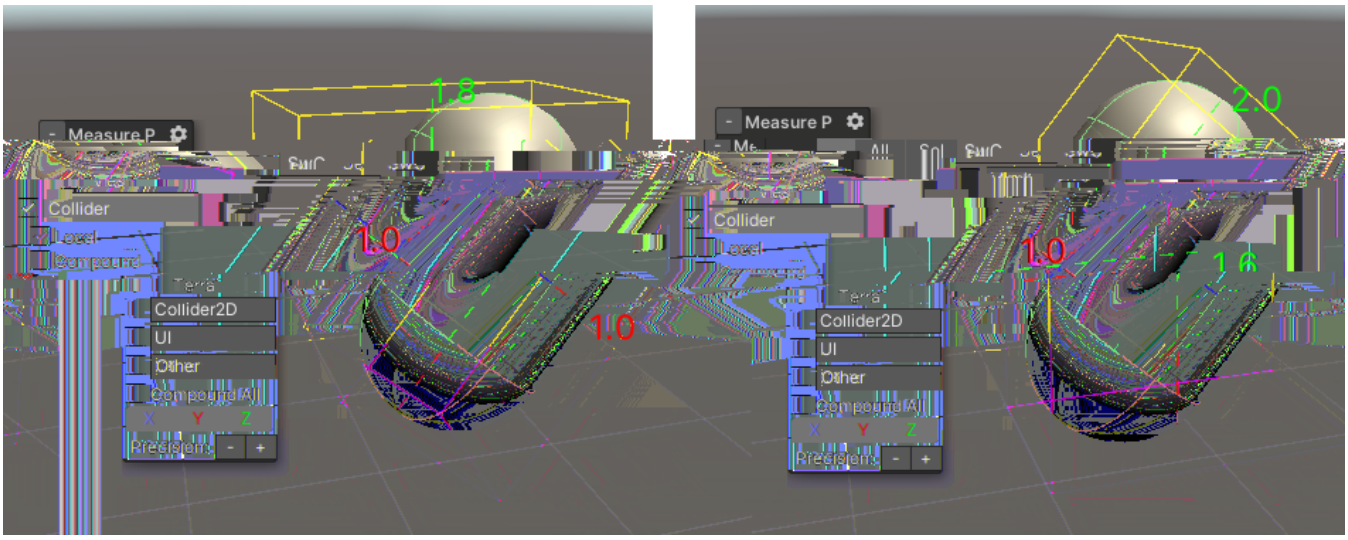


Figure 1: World (left) vs Local (right) measures

Measure Tool measures in edit time (works also in prefab mode) and in runtime 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, Skinned Mesh, Sprite, statically batched mesh, procedurally generated mesh
- 3D Colliders, Mesh Colliders
- 2D Colliders
- UI (RectTransforms)
- Terrain
- Miscellaneous: Particle system bounds, Camera frustum
- Distances, world positions

There are some limitations:

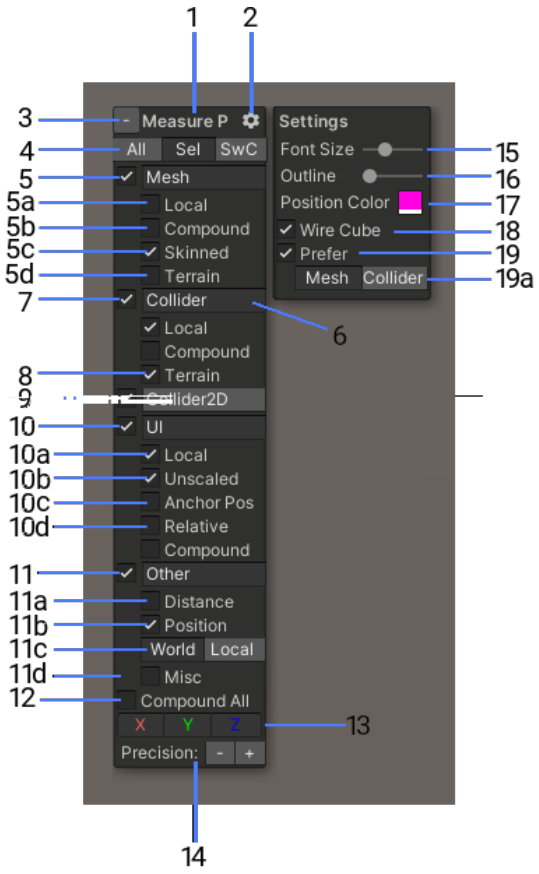
- When you change rotation of statically batched mesh in runtime, measurement will rotate, but mesh not
- Composite 2D and Tilemap collider are only measured in world space
- Skinned Mesh without assigned root bone in Skinned Mesh Renderer component is only measured in world space
- Edge radius of 2D 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)

- When you have "Auto Sync Transforms" turned off in Project Settings ! Physics (or Physics 2D), the measures can be in rare cases a bit off (values of transform from previous frame)

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).

6 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. Settings toggle - open/close settings window
3. Foldout Button. Use to fold/expand Measure Tool window. Hold alt to open/close all subsections
4. 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
5. Measure Mesh Toggle - should meshes be measured? (similar for Collider/Collider2D/UI)
   a 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)
   b Show world space aligned bounding volume (dimensions and wirecube) of all measured meshes combined in world space (similar for Collider/Collider2D/UI)
   c Should measure also skinned mesh? This option is here, because skinned mesh measuring is a bit slower and if there are too many of them it might slow the editor.
   d Should measure also terrain mesh?
6. Foldout Button - fold/expand measuring options of given category

7. Collider measuring
8. Should measure also Terrain Collider?
9. Collider2D measuring
10. UI measuring
    a Locally oriented bounding box vs World axis aligned bounding box of UI
    b Use root (top level) canvas coordinate system for measuring. This is useful when working with CanvasScaler component. This will effectively unscale the reported sizes/dimensions so the numbers will stay the same in any Game window resolution.
    c Show anchored position (this replaces local/world position if it is turned on)
    d RectTransform distances relative to the parent RectTransform (if any)
11. Other measuring
    a Should distances be measured? (Between All/Selected/SelectedWithChildren gameobjects)
    b Should show positions?
    c World or local position
    d Measuring miscellaneous objects - Particle systems, Camera frustum (more in the future)
12. Show bounding volume (dimensions and wirecube) of everything currently measured
13. 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
14. How many decimals should be shown?
15. Font size of labels in the SceneView
16. Thickness of the optional outline
17. Text color of the world position label
18. Should a wirecube of the measured dimensions of gameobjects be shown? (compound bounding volumes will always show wirecube)
19. If gameobject has Mesh and Collider components, should I measure both, or prefer only one of it
    a If prefer, then what?

7 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).

8 Known Issues

1. All the listed limitations in Measuring section

9 Roadmap

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

1. Angles
2. Point and click/drag measuring
3. CompositeCollider2D local measuring
4. WheelCollider measuring
5. EdgeRadius measuring
6. Miscellaneous measuring: Cloth, windzone, spriteshape, linerenderer, navmesh, bilboardRenderer
7. Closest point distance
8. DOTS support (it's too soon)

10 Changelog

v1.4.1

- Minor bug fixes

v1.4.0

- **New Feature:** Enhanced UI measuring: RectTransform relative position, unscaled measuring, anchored position
- **New Feature:** Add world/local option to position
- **New Feature:** Add Camera plane distance measuring
- Significant performance optimization
- Minor bug fixes and refactoring

v1.3.0

- Add optional outline
- Improve readability of compounds
- Minor bug fix and refactoring

v1.2.0

- **New Feature:** Added measuring of Camera frustum
- **New Feature:** Added option to display world position
- Add Settings window, create "Other" group and miscellaneous category
- Remember window position
- Fix clamping window to SceneView on scaled Monitor
- Minor bug fixes and refactoring

v1.1.0

- **New Feature:** Added measuring of Skinned Mesh
- Minor optimization and refactoring

v1.0.0

- First release