Grab System

Grab and throw objects in Unity.

V 1.1.1

- Incomplete documentation parts will be improved over time.
- Get the most up to date documentation by clicking here.
- Remember you can hover over fields in the “Inspector” window in Unity’s editor to read tooltip explanations of each field.
- If you have any questions or need assistance email support at intuitivegamingsolutions@gmail.com.

Table of Contents

NOTE: See ‘API Reference.pdf’ (online) if you are looking for source code documentation.

Grabbers

2.a. The Grabber Component

- Documentation coming soon!

2.b. The RayGrabber Component

- Documentation coming soon!

2.b. The DistanceGrabber Component

- A component that provides the core functionality for making GrabbableObjects grabbable from a distance. Attached to the same GameObject as your Grabber.
- Documentation coming soon!

GrabbableObjects

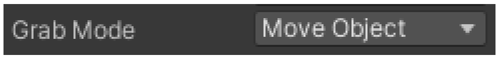
3.a. The GrabbableObject Component

3.a.i. The Component

- The GrabbableObject component is responsible for making any object, with or without a Rigidbody component grabbable by any Grabber.

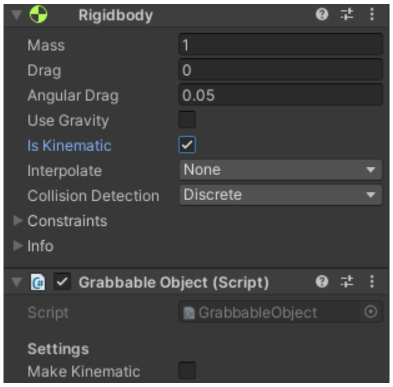
3.a.ii. Making a 'Pinned' Grabbable (Kinematic)

- Make sure your GrabbableObject component is using the GrabbableObject.GrabMode.MoveObject grab mode.



Grab Mode: Move Object

- To make a 'pinned' GrabbableObject that stays in whatever position and rotation it is released in you will need to make a GrabbableObject that has a kinematic Rigidbody component **and** has GrabbableObject.makeKinematic disabled (as this would undo the kinematic state of the grabbable on release.)



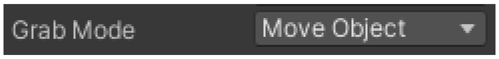
Rigidbody
Mass: 1
Drag: 0
Angular Drag: 0.05
Use Gravity: □
Is Kinematic: ☑
Interpolate: None
Collision Detection: Discrete
Constraints
Info

☑ Grabbable Object (Script)
Script: GrabbableObject
Settings
Make Kinematic: □

A screenshot showing the described settings where the GrabbableObject has 'make kinematic' disabled and the Rigidbody has 'is kinematic' enabled already.

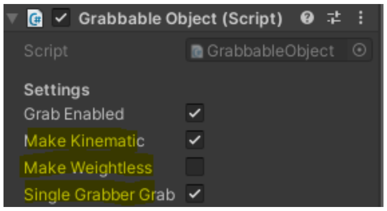
3.a.iii. Making a 'Weightless' Grabbable

- Make sure your GrabbableObject component is using the GrabbableObject.GrabMode.MoveObject grab mode.



Grab Mode: Move Object

- **Single Hand Grab - Weightless Grabbable**
  - To make a 'weightless' GrabbableObject that may only be grabbed by a ***single hand*** you will want to ensure that the GrabbableObject: (1) has a Rigidbody component, and (2) has 'Make Kinematic' enabled.



☑ Grabbable Object (Script)

Script: GrabbableObject

Settings

Grab Enabled ☑

Make Kinematic ☑

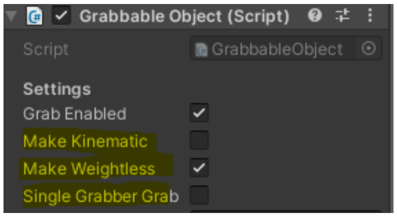
Make Weightless □

Single Grabber Grab ☑

A screenshot showing which settings should be enabled and disabled..

  - This will ensure that all physics properties of the grabbable object are ignored while the object is being grabbed.
  - This differs from making a 'pinned' grabbable in that the 'Make Kinematic' setting will automatically restore the non-kinematic state of the grabbable object once it is released whereas the steps from the 'pinned' grabbable section create a grabbable object that is always kinematic, even after being released.
  - ***If you do not have 'single hand grab' enabled then you will have made an 'averaged grabbable' instead of a weightless single-hand grabbable.***

- **Multi Hand Grab - Weightless Grabbable**
  - To make a multi-hand grab 'weightless' GrabbableObject you will want to ensure that the GrabbableObject: (1) has a Rigidbody component, (2) has 'Make Kinematic' ***disabled***, and (3) has 'Make Weightless' enabled.



☑ Grabbable Object (Script)

Script: GrabbableObject

Settings

Grab Enabled ☑

Make Kinematic □

Make Weightless ☑

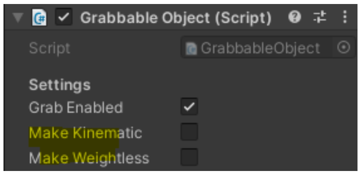
Single Grabber Grab □

A screenshot showing which settings should be enabled and disabled..

  - It is important to note that 'multi hand weightless grabables' are not actually weightless, they just have their mass temporarily overridden with a smaller mass while held. This allows the grabbable to seem weightless while still remaining an active Rigidbody.

3.a.iv. Making a 'Weighted' Grabbable

- To make a 'weighted' GrabbableObject you will want to ensure that the GrabbableObject both: (1) has a Rigidbody component, (2) does ***not*** have 'Make Kinematic' enabled, and (3) does ***not*** have 'Make Weightless' enabled..



Grabbable Object (Script)

Script GrabbableObject

Settings

Grab Enabled ☑

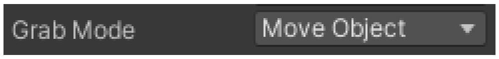
Make Kinematic □

Make Weightless □

A screenshot showing which settings should be disabled..

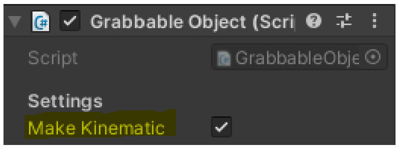
3.a.v. Making an 'Averaged' Grabbable

- It is important to note that 'Averaged' grabables are ***rarely the correct choice***, averaged grabables use the average position and rotation between all grabbers that are currently holding them.
- Averaged grabables are useful for making things such as 2-handed spell effects
- Make sure your GrabbableObject component is using the GrabbableObject.GrabMode.MoveObject grab mode.



Grab Mode: Move Object

- To make an 'averaged' GrabbableObject you will want to ensure that the GrabbableObject both: (1) has a Rigidbody component, and (2) has 'Make Kinematic' enabled.



Grabbable Object (Scri

Script GrabbableObje

Settings

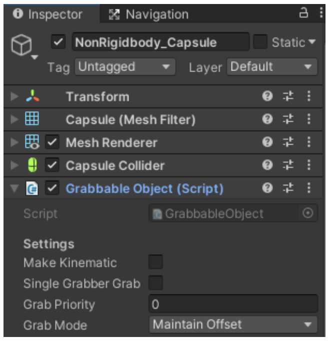
Make Kinematic ☑

A screenshot showing which setting should be enabled..

  - This will ensure that all physics properties of the grabbable object are ignored while the object is being grabbed and the position and rotation will be averaged based on all grabbers currently grabbing the grabbable..
  - This differs from making a 'pinned' grabbable in that the 'Make Kinematic' setting will automatically restore the non-kinematic state of the grabbable object once it is released whereas the steps from the 'pinned' grabbable section create a grabbable object that is always kinematic, even after being released.

3.a.vi. Making a 'Stationary' Grabbable

- There are many different ways to make a stationary grabbable including using the 'is kinematic' property of a Rigidbody and using the GrabbableObject.GrabMode.MaintainOffset grab mode.
- The recommended way to make a 'stationary' grabbable is to have a GrabbableObject with **no** Rigidbody component and the 'Grab Mode' set to 'Maintain Offset'.



☑ NonRigidbody_Capsule □ Static

Tag Untagged Layer Default

Transform

Capsule (Mesh Filter)

☑ Mesh Renderer

☑ Capsule Collider

☑ Grabbable Object (Script)

Script GrabbableObject

Settings

Make Kinematic □

Single Grabber Grab □

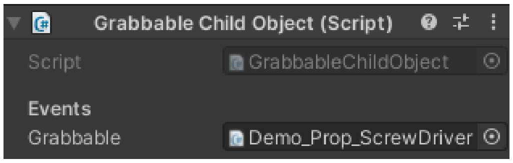
Grab Priority 0

Grab Mode Maintain Offset

A screenshot showing a situation where the 'stationary' GrabbableObject does not have a Rigidbody component.

3.b. The GrabbableChildObject Component

- The GrabbableChildObject component should be attached to any child object of a GrabbableObject that has a Collider that you want to be made grabbable.
- After adding the GrabbableChildObject component to a child collider it should automatically find the GrabbableObject reference on the parent, if not make sure to manually drag the reference into the component.

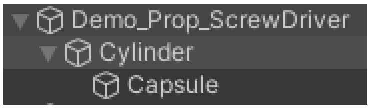

Grabbable Child Object (Script)

Script: GrabbableChildObject

Events

Grabbable: Demo_Prop_ScrewDriver

A screenshot showing which setting should reference the parent GrabbableObject..



A screenshot showing what GameObject contains the GrabbableChildObject component in the hierarchy.

3.c. The DistanceGrabbable Component

- A component that makes a GrabbableObject instantaneously grabbable from a distance by a DistanceGrabber. Attached to the same GameObject as your GrabbableObject.
- Documentation coming soon!

3.d. The ForceDistanceGrabbable Component

- A component that makes a GrabbableObject grabbable via physics forces (or instantaneously) from a distance by a DistanceGrabber. Attached to the same GameObject as your GrabbableObject.
- Documentation coming soon!

3.e. The DistanceCatchAssist Component

- A component that assists a DistanceGrabber (by reference, not attached to the same GameObject) in making catching easier.
- The '***snap grab***' setting makes it so a large trigger area can be used and still have partially believable catching but can lead to clipping.
- When not using snap grab the component will briefly force a grab when the grabbable hits the trigger giving an extra moment to complete the grab manually.
- Documentation coming soon!

Modules

4.a. TimeSystem Module

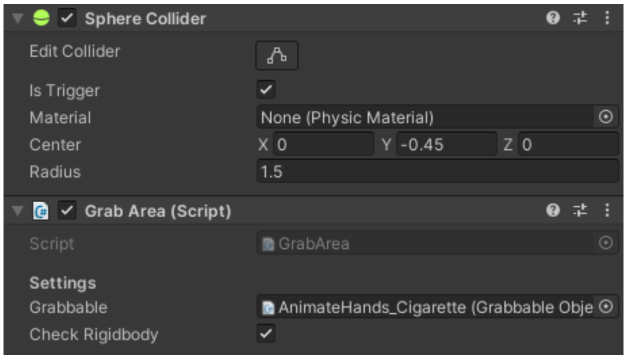
- The module that provides the current time.
- Why does this module exist? This module exists to allow you to easily change what 'current time' retrieval method is used. This lets you use realtimeSinceStartup, unscaledTime, or time depending on what your game requires.
- API Reference: Time System - API Reference

4.b. PhysicsTools Module

- A module that provides useful components and helper functions for many different collision-related, velocity tracking, rigidbody processing use cases and more.
- Documentation: Physics Tools - Documentation
- API Reference: Physics Tools - API Reference

Extras

5.a. The GrabArea Component



☑ Sphere Collider

Edit Collider

Is Trigger ☑

Material None (Physic Material)

Center X 0 Y -0.45 Z 0

Radius 1.5

☑ Grab Area (Script)

Script GrabArea

Settings

Grabbable AnimateHands_Cigarette (Grabbable Obje

Check Rigidbody ☑

- A component that allows you to make grabbing easy by allowing ConditionalGrabber grabbers to automatically grab a GrabbableObject while the ConditionalGrabber grabs while in the grab area trigger.
- The GrabArea component should be attached to the same GameObject as some ***trigger*** Collider. This allows a Grabber to easily grab the referenced 'Grabbable' whenever it attempts a grab in this area.

5.b. The IgnoreCollidersWhileGrabbing Component

- A component that allows ignore collisions between all colliders in a Grabber and some specified colliders while the Grabber is grabbing the relevant GrabbableObject.

5.c. The DelayUnignoreCollidersOnRelease Component

- A component that works hand-in-hand with the IgnoreCollidersWhileGrabbing component to allow you to specify an unignore collision (with grabber) delay from the time the IgnoreCollidersWhileGrabbing component was initially supposed to unignore collisions.
- Why does this component exist? Why not build it into the IgnoreCollidersWhileGrabbing component?
    - In many use cases, like in the XR gun demo, grabbable objects are deactivated once they (or other grabbable objects) are released. This would prevent delayed unignore collision events from ever occurring. To work around this behavior this component was added which can be attached to any GameObject and simply reference the relevant IgnoreCollidersWhileGrabbing component ensuring this behavior works at all times.

5.d. The MaintainOffsetByGrabOffset Component

- A component that allows different 'maintain offset' Transforms to be specified based on the Grabber's offset relative to the GrabbableObject at the time of grab.
- This component is only useful when attached to the same GameObject as a GrabbableObject that is using the 'Maintain Offset' grab mode.

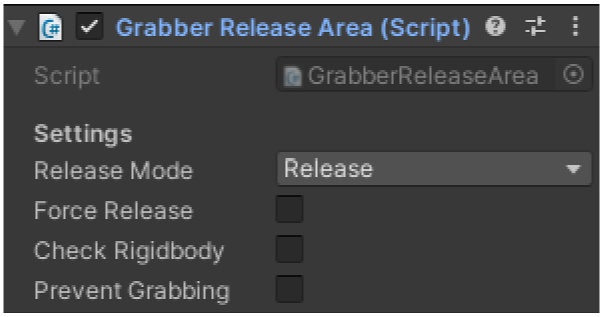
5.e. The MaintainOffsetByProjectedAngle Component

- A component that allows different 'maintain offset' Transforms to be specified based on the Grabber's angle around some local axis relative to the GrabbableObject at the time of grab.
- This component is only useful when attached to the same GameObject as a GrabbableObject that is using the 'Maintain Offset' grab mode.

5.f. The MaintainOffsetByRelativeAngle Component

- A component that allows different 'maintain offset' Transforms to be specified based on the Grabber's relative angle to the GrabbableObject at the time of grab.
- This component is only useful when attached to the same GameObject as a GrabbableObject that is using the 'Maintain Offset' grab mode.

5.g. The GrabberReleaseArea Component



Grabber Release Area (Script)

Script GrabberReleaseArea

Settings

Release Mode: Release

Force Release ☐

Check Rigidbody ☐

Prevent Grabbing ☐

A screenshot of the 'Inspector' pane for a GrabberReleaseArea component in v1.0.1

- A component that can force a Grabber to release whatever it is grabbing when it enters the area and/or prevent new grabs.
-

<table>
<tr><th>Setting</th><th>Description</th><th>Default Value</th></tr>
<tr><td>Release Mode</td><td>The release mode to use.
<table>
<tr><th>Mode</th><th>Description</th></tr>
<tr><td>None</td><td>No release.</td></tr>
<tr><td>No Throw</td><td>Release without throwing.</td></tr>
<tr><td>Release</td><td>Release with throwing.</td></tr>
</table></td><td>Release
<i>(GrabberReleaseArea.ReleaseMode)</i></td></tr>
<tr><td>Force Release</td><td>A boolean that controls whether or not a release is forced by this area. Forced releases cannot be prevented whereas non-forced releases can be prevented through various means.</td><td>false
<i>(bool)</i></td></tr>
<tr><td>Check Rigidbody</td><td>A boolean that controls whether or not any 'attachedRigidbody' to the triggering collider is checked for a Grabber component if none is found on the triggering Collider's gameObject.</td><td>false
<i>(bool)</i></td></tr>
<tr><td>Prevent Grabbing</td><td>Should this area prevent Grabbers in it from grabbing?</td><td>false
<i>(bool)</i></td></tr>
</table>

FAQ

(Frequently Asked Questions)

Q:
A: