



## Instructions And Setup Guide

[About](#)

[Installation](#)

[Rigid Collider Creator](#)

[Quick Start](#)

[UI](#)

[Tools UI](#)

[Hulls UI](#)

[Auto Hull Settings UI](#)

[Defaults UI](#)

[Settings UI](#)

[Assets UI](#)

[Hull Inflation](#)

[Skinned Collider Creator](#)

[Quick Start](#)

[UI](#)

[Tools](#)

[Bone Hierarchy](#)

[Properties \(Bone\)](#)

[Properties \(Auto Hull or Manual Hull\)](#)

[Defaults](#)

[Settings](#)

[Assets](#)

[Extra Tools](#)

[API Example / Bulk Collider Generation](#)

[Generate Colliders From Selection](#)

[Span Physics](#)

[Under The Hood](#)

[Disconnecting From Editor Data](#)

[Reconnecting Editor Data](#)

[Install Location](#)

[Hull Data Folder](#)

[Hull Types](#)

[Max Planes for hulls](#)

[Hull Precision](#)

[Keyboard Shortcuts](#)

[Tips](#)

[Known Issues](#)

[Problems? Feature Requests? Bugs?](#)

[Licences](#)

# About

Technie Physics Creator is a fast, efficient way to create accurate colliders for your objects. For rigid objects (such as crates, weapons, food) then use the Rigid Collider Creator. For skinned objects (such as characters and animated objects) then use Skinned Collider Creator to create and assign colliders to animated bones.

If you prefer to watch a video demonstration then there are tutorial videos on the Asset Store page.

# Installation

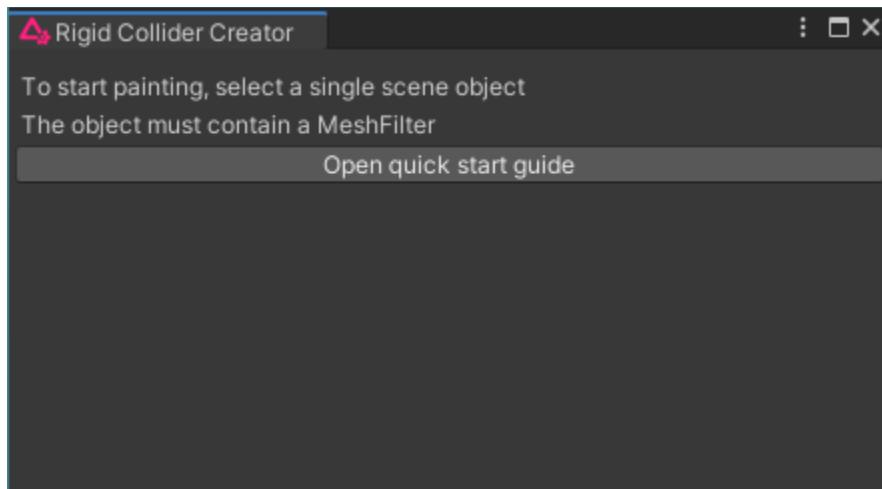
Simply extract the package into your project via the package manager. If you are upgrading between minor versions then you can leave the previous version and extract over the top of it. If you are upgrading between major versions, or you receive errors after importing, then it is recommended to delete the previous version (found in Assets/Technie/PhysicsCreator/ and then install a fresh copy.



# Rigid Collider Creator

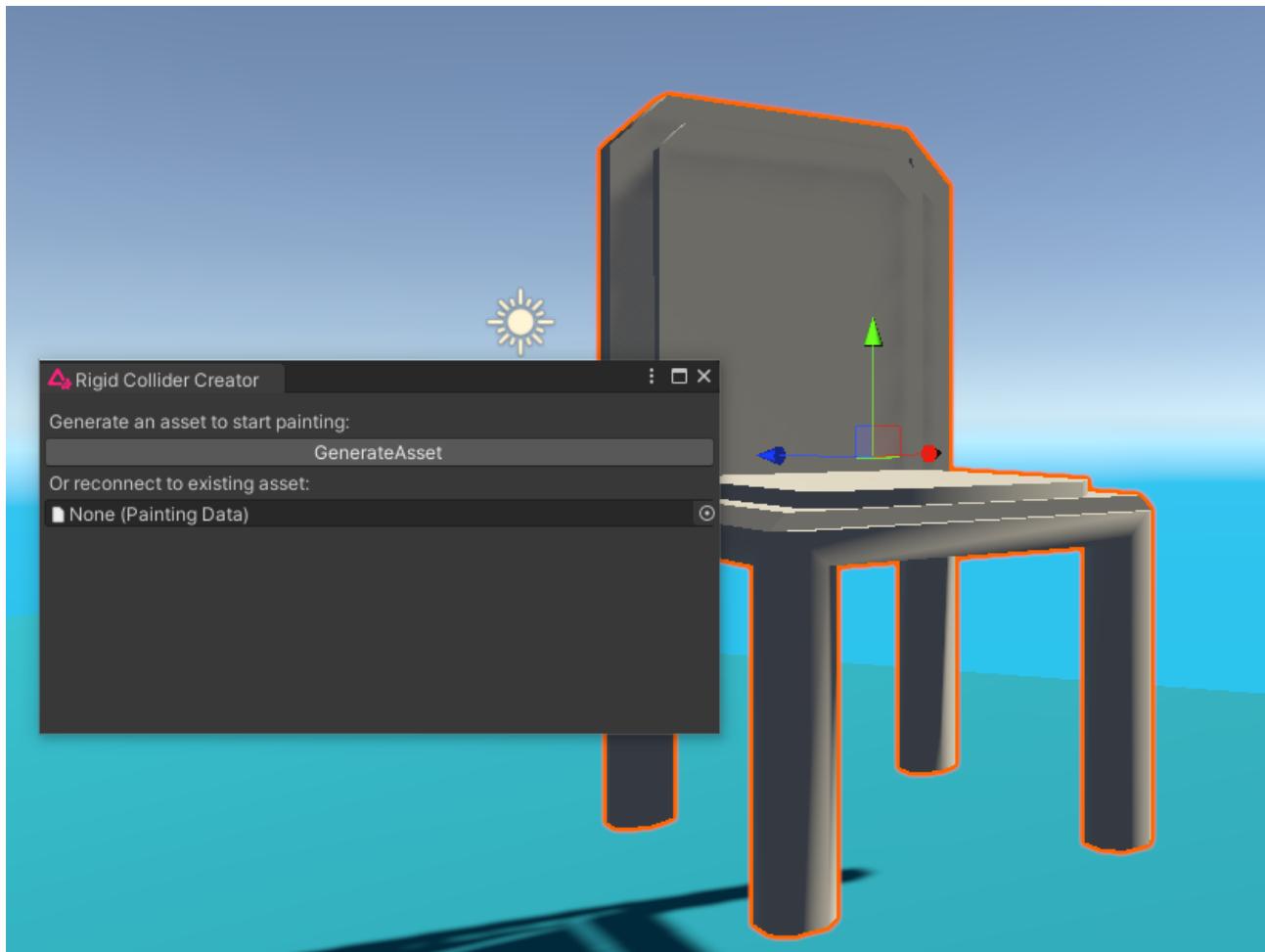
## Quick Start

Once installed, open the window by the Window->Technie Collider Creator->Rigid Collider Creator menu.. Move this somewhere handy as you go through the quick start guide.



*Rigid Collider Creator window with no object selected*

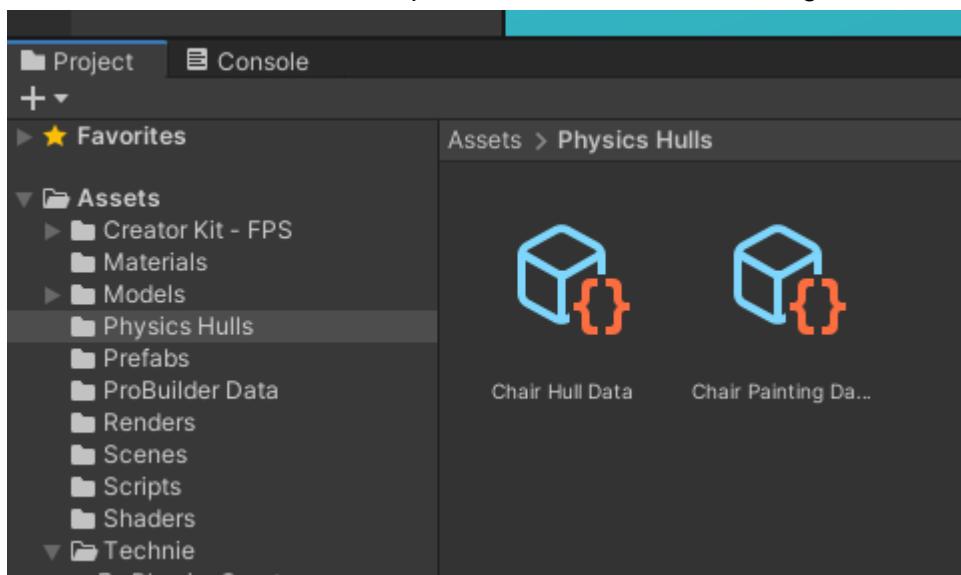
Now select a single object in your scene that you want to paint physics colliders for. The object must have a MeshRenderer component on it, this is what we'll paint onto. The Rigid Collider Creator window will change to show that this can be used to paint on.



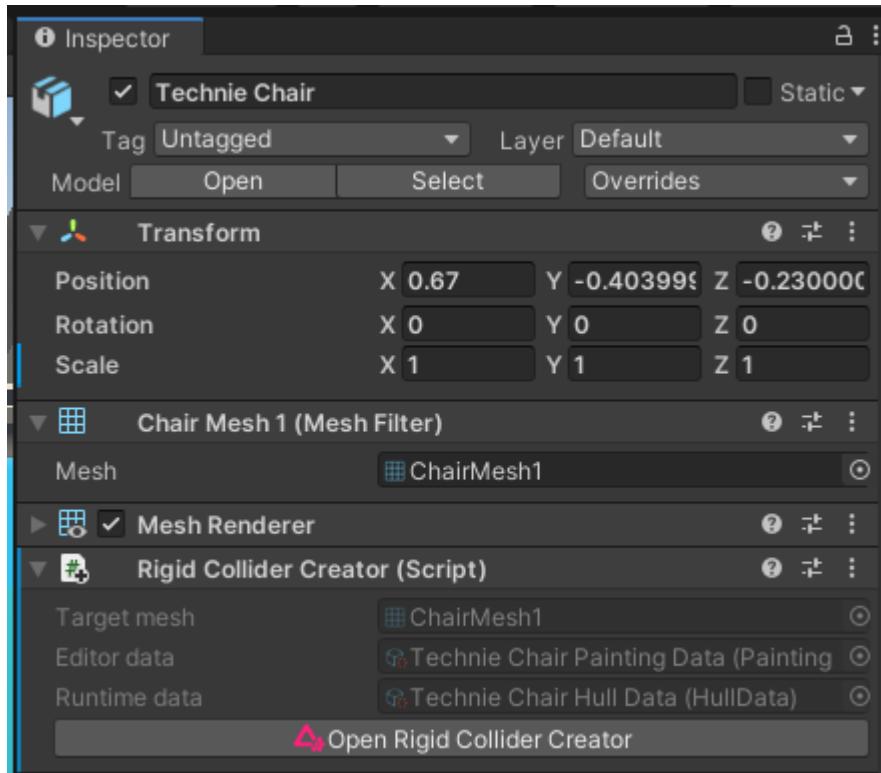
*Rigid Collider Creator window with valid selection*

Now click Generate Asset to set this object up for painting. This will do three things:

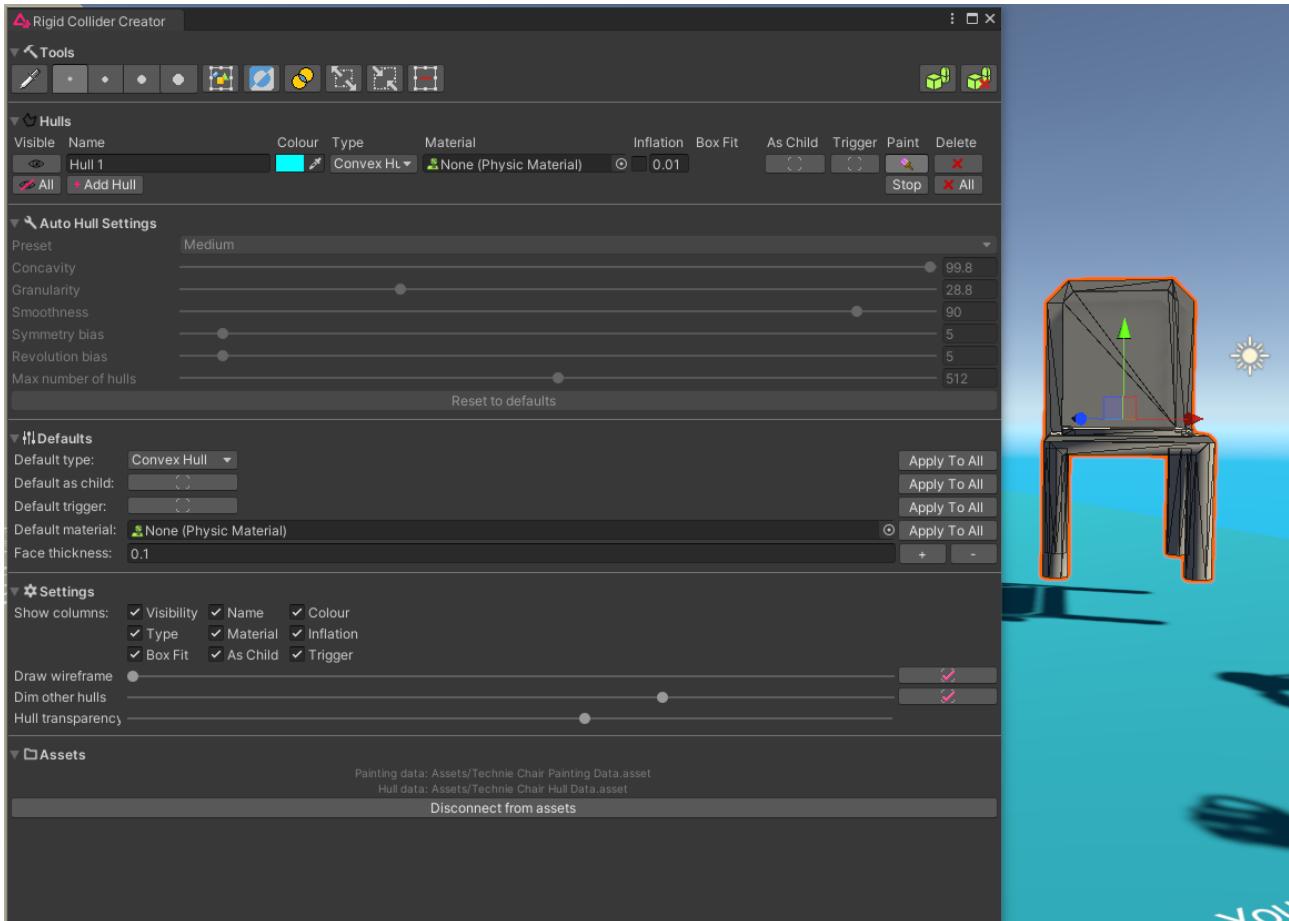
- Assets for painting and hull data will be created in your project in Assets/Physics Hulls. These store the data about the painted surfaces and also the generated collision meshes.



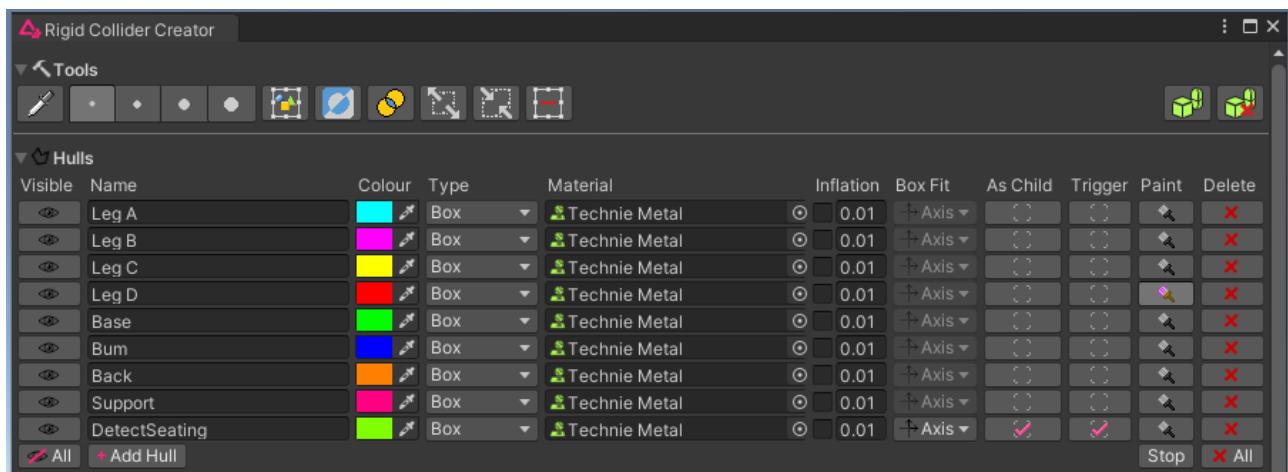
- A RigidColliderCreator component will be added to the object. This is what connects the assets to this particular object so you can paint on it.



After the assets are generated the window will show you have a single hull ready for painting.



Click and drag on the surface of the object to paint faces to include in the collider. To remove faces click or drag on a face that is already marked. You can also set the name, colour and type of the hull in the window.



*Multiple hulls painted on one object*

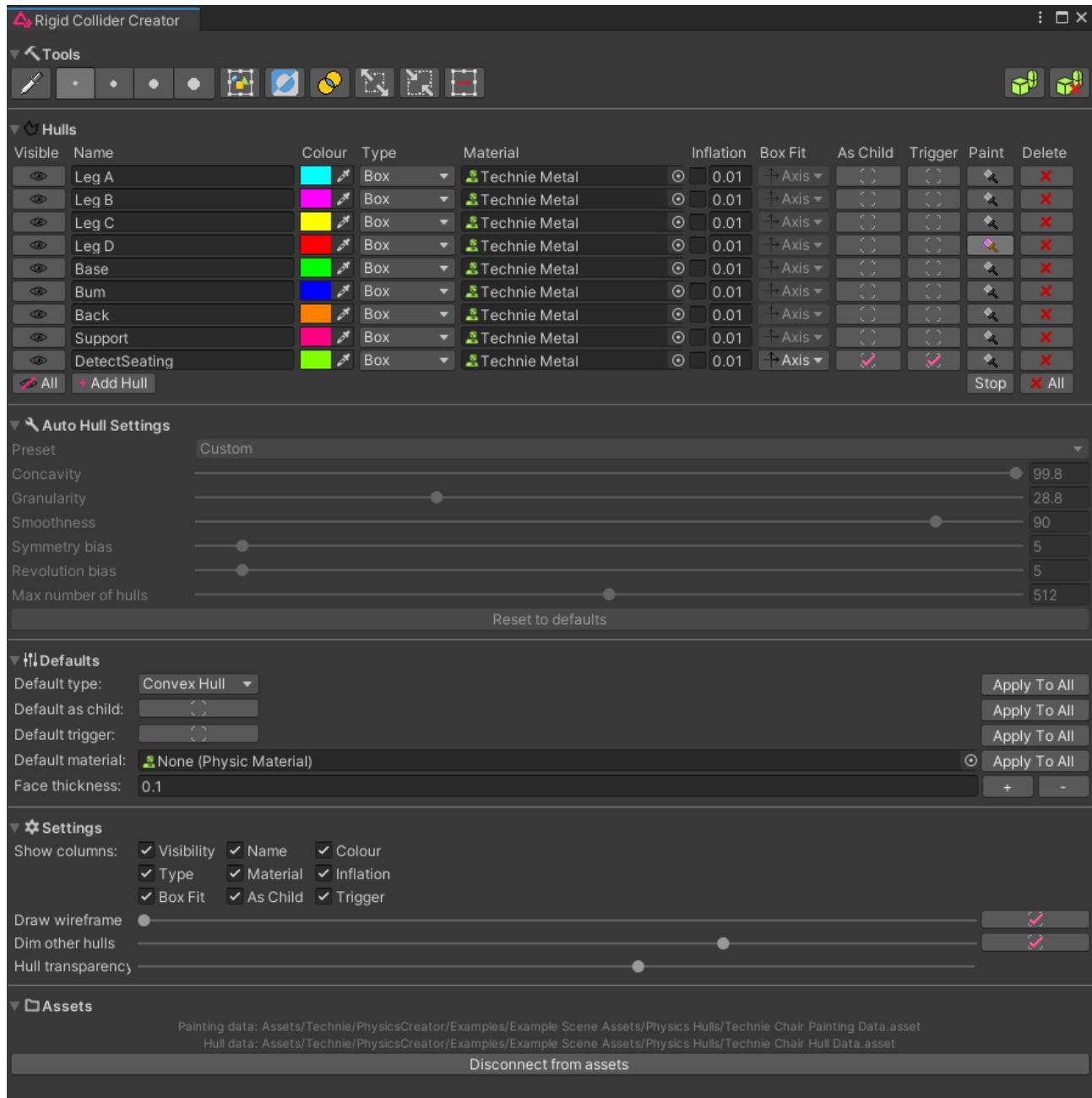
More hulls can be added by clicking Add Hull, if you have multiple hulls on an object then choose which to edit by pressing the Paint button.

Once you're happy with your painting, hit Generate Colliders and colliders will be created on the object.



You can now refine your painting data, add more hulls if you need more accuracy and fine tune the properties of each hull. Just press Generate Colliders to update the collides and see the changes.

# UI



**Hulls**

Visible	Name	Colour	Type	Material	Inflation	Box Fit	As Child	Trigger	Paint	Delete
Leg A	Leg A	Blue	Box	Technie Metal	0.01	Axis				X
Leg B	Leg B	Magenta	Box	Technie Metal	0.01	Axis				X
Leg C	Leg C	Yellow	Box	Technie Metal	0.01	Axis				X
Leg D	Leg D	Red	Box	Technie Metal	0.01	Axis				X
Base	Base	Green	Box	Technie Metal	0.01	Axis				X
Bum	Bum	Dark Blue	Box	Technie Metal	0.01	Axis				X
Back	Back	Orange	Box	Technie Metal	0.01	Axis				X
Support	Support	Magenta	Box	Technie Metal	0.01	Axis				X
DetectSeating	DetectSeating	Green	Box	Technie Metal	0.01	Axis	✓	✓		X
All	+ Add Hull									Stop

**Auto Hull Settings**

Preset	Custom
Concavity	99.8
Granularity	28.8
Smoothness	90
Symmetry bias	5
Revolution bias	5
Max number of hulls	512

**Defaults**

Default type:	Convex Hull	Apply To All	
Default as child:		Apply To All	
Default trigger:		Apply To All	
Default material:	None (Physic Material)	Apply To All	
Face thickness:	0.1	+	-

**Settings**

Show columns:	<input checked="" type="checkbox"/> Visibility <input checked="" type="checkbox"/> Name <input checked="" type="checkbox"/> Colour <input checked="" type="checkbox"/> Type <input checked="" type="checkbox"/> Material <input checked="" type="checkbox"/> Inflation <input checked="" type="checkbox"/> Box Fit <input checked="" type="checkbox"/> As Child <input checked="" type="checkbox"/> Trigger
Draw wireframe	✓
Dim other hulls	✓
Hull transparency	

**Assets**

Painting data: Assets/Technie/PhysicsCreator/Examples/Example Scene Assets/Physics Hulls/Technie Chair Painting Data.asset  
Hull data: Assets/Technie/PhysicsCreator/Examples/Example Scene Assets/Physics Hulls/Technie Chair Hull Data.asset

Disconnect from assets

## Tools UI



**Pipette** - used for selecting a hull from the scene view. Select this then click on a hull in the scene view to select it for painting.

**Brush size** - choose a brush size to paint faces for specific hulls.

**Select All** - Adds all faces as painted to the current hull.

**Select Inverse** - Inverts the painted faces on the current hull

**Select Unpainted** - Sets the painted faces to those which are not painted in any other hull

**Grow Selection** - Expands the currently painted faces by also selecting adjacent faces.



**Shrink Selection** - Shrinks the currently painted faces by unpainting faces at the edges.

**Select None** - unpaints all faces for the current hull

**Generate** - regenerates the collider components for this hull

**Delete Generated** - deletes all generated objects - colliders, and child objects and child colliders

## Hulls UI

Hulls		Visible	Name	Colour	Type	Material	Inflation	Box Fit	As Child	Trigger	Paint	Delete	
<input checked="" type="checkbox"/>	Leg A			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	Leg B			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	Leg C			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	Leg D			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>		
<input checked="" type="checkbox"/>	Base			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	Bum			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	Back			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	Support			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/>	DetectSeating			Box			<input checked="" type="checkbox"/>	<input type="checkbox"/> 0.01			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	All		Add Hull										Stop

**Visibility** - toggle whether this hull is shown in the scene view or not

**Name** - enter a name here so you can easily keep track of different hulls

**Colour** - choose the colour this hull will use in the scene view

**Type** - choose the type of collider this hull will generate

**Material** - drag in a physics material and it will be assigned to the generated collider

**Inflation** - tick the box to enable hull inflation, and specify an amount to inflate by

**Box Fit** - what algorithm is used to fit Box hulls to the data.

*Axis* is aligned to the object's axis and unrotated

*Tight* creates the smallest possible box around the painted faces

*Face* matches the boxes faces to the painted faces

Note that *Tight* and *Face* are only possible when *As Child* is ticked.

**Max Faces** - the maximum number of faces this hull is allowed to have.

See 'Max Faces' section below.

**Trigger** - tick this box to make the generated collider a trigger

**As Child** - tick this box to create this collider as a child object

**Paint** - this selects which hull is the active hull for painting

**Delete** - click to delete this hull

And on the bottom row:

**Visibility All** - Toggles visibility of all hulls at once

**Add Hull** - Adds a new hull

**Stop** - Stops painting and deselects active hull

**Delete All** - Deletes all the hull authoring and painting data you have created on this object



## Auto Hull Settings UI



These settings are used when the AUTO hull type is chosen. These control how the model is broken down into separate hulls and affect the accuracy and time required when pressing Generate.

**Preset** - choose from one of the quality presets, or choose 'custom' to adjust the individual settings manually

**Concavity** - Controls how sensitive to concave regions the output is. A low value will ignore smaller bumps and grooves. A higher value will follow the surface more accurately but generate more hulls.

**Granularity** - Affects the internal resolution of the calculations. Too low and fine details like corners and grooves will not be captured. Too high and the algorithm can take a very long time so start low and only increase it when needed.

**Smoothness** - Affects how smooth the output hulls will be. High values create very accurate hulls with many faces. Low values create blockier hulls which will be quicker at runtime.

**Symmetry bias** - Affects how the model is cut into individual pieces around symmetry planes.

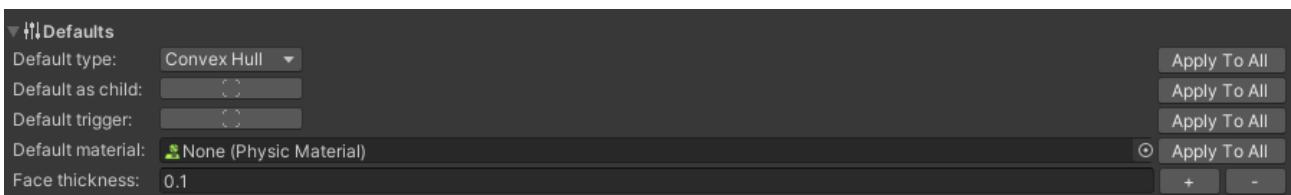
**Revolution bias** - Affects how the model is cut into pieces around revolution axes.

**Max number of hulls** - Control the maximum number of hulls each AUTO hull can produce. Note this controls a target and the exact number of output hulls may be slightly different.

**Reset to defaults** - resets the settings for 'custom' back to their default values.

Auto hull settings are stored as part of the painting data, so when you return to the object and press 'generate' the same settings will be applied as last time.

## Defaults UI



**Default type** - this type will be set to any newly created hull (from Add Hull)

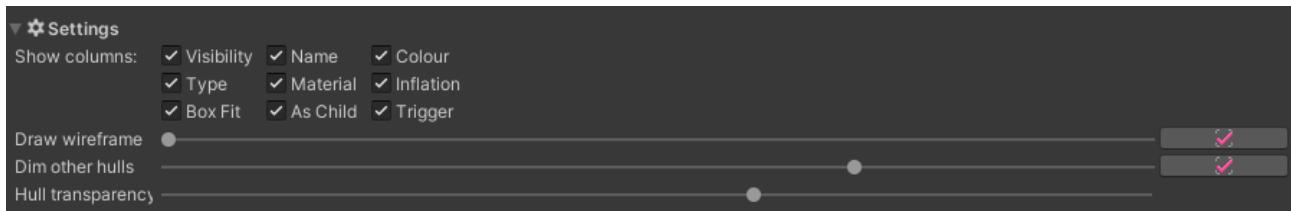
**Default trigger** - should new hulls be marked as a trigger by default?

**Default material** - drag a physics material to use this as the default for new hulls

**Face thickness** - when generating face colliders, this sets how 'deep' the faces are

**Apply to All** - will apply this setting to all the Hulls you have already created.

## Settings UI



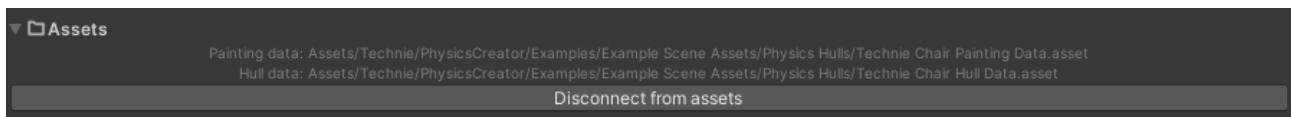
**Show columns** - toggle the columns in the Hulls section to slim down the UI

**Draw wireframe** - toggle drawing the wireframe for the current painting mesh. The slider controls the brightness of the wireframe lines.

**Dim other hulls** - when ticked this keeps the hull you're painting bright and darkens the other hulls so it's easier to see which is which. The slider controls how much to dim the other hulls by.

**Hull Transparency** - adjust the overall transparency of the scene overlay

## Assets UI



Displays the assets linked to this hull. See 'Disconnecting from Editor Data' section.

## Hull Inflation

Hull inflation can be turned on by ticking the 'Inflate' check box in the row for the hull. This will expand the hull outwards in all directions by the entered amount. This is particularly useful for convex hulls, as inflating it will reduce the amount of faces needed and therefore increase the performance for only a small loss in accuracy. If you have highly smooth geometry then you should definitely enable this.

For sphere and box hull types, this will simply expand their size by the entered amount. This can be useful if you want your collision to be larger than the object (eg. to make it easier to shoot/hit an enemy).

For face hull types then the inflation amount will be how far backwards the face is projected to give it its thickness. In these cases you may want the thickness to be somewhat larger than the default to prevent physics objects 'leapfrogging' very thin face hulls if they're moving very quickly.

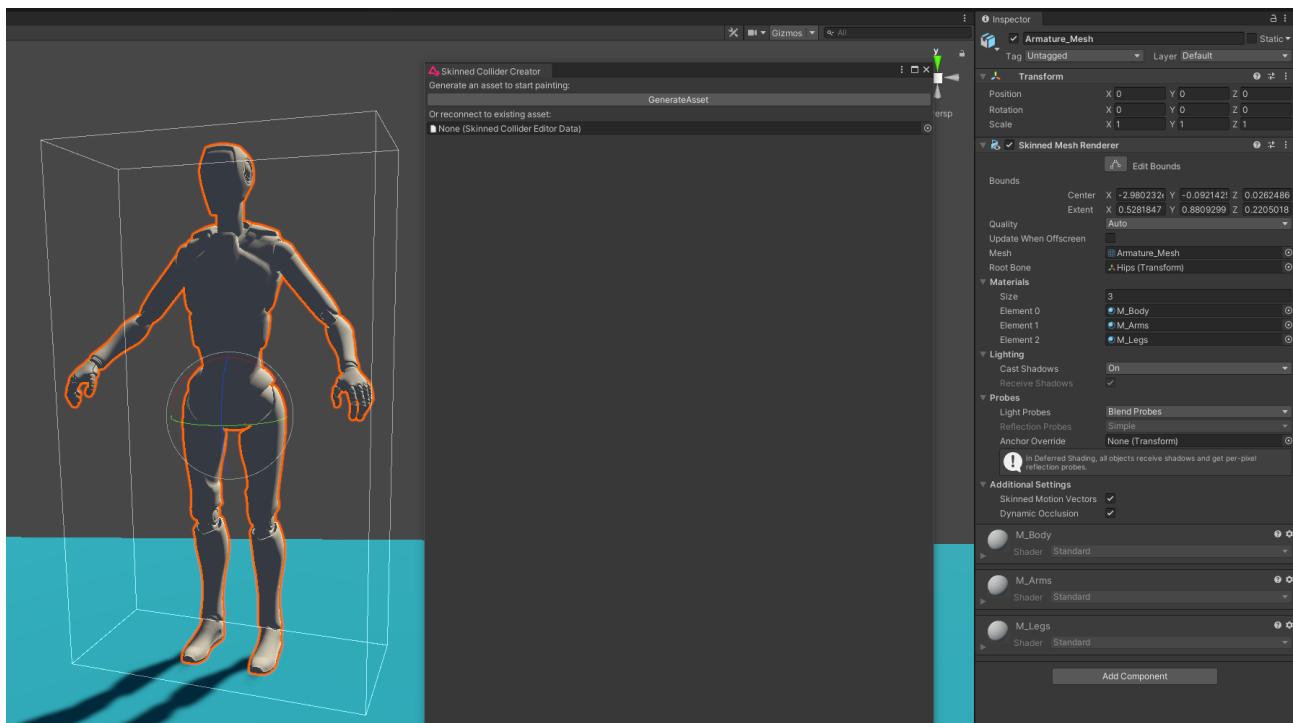
Due to change in the Unity physics engine in 2019.3, hull inflation is currently disabled in this version.

# Skinned Collider Creator

## Quick Start

Adding physics to a skinned model works mostly the same as for a rigid object. Start by opening the window with Window->Technie Collider Creator->Skinned Collider Creator window.

Then select the SkinnedMeshRenderer component you want to add physics onto. This is often nested underneath the root of your animated model.



Now press “Generate Asset” and it’ll create the painting and runtime data files needed, and change to show the skinned collider editor.

By default this will show the bone hierarchy which you attach colliders to, but no colliders have been created yet. Click ‘Expand All’ to see them all.

**Skinned Collider Creator**

**Tools**

**Bone Hierarchy**

```

    ▼ Hips
        ▼ Left_UpperLeg
            ▼ Left_LowerLeg
                ▼ Left_Foot
                    ▼ Left_Toes
                        △ Left_ToesEnd
        ▼ Right_UpperLeg
            ▼ Right_LowerLeg
                ▼ Right_Foot
                    ▼ Right_Toes
                        △ Right_ToesEnd
    ▼ Spine
        ▼ Chest
            ▼ UpperChest
            ▼ Left_Shoulder
                ▼ Left_UpperArm
                    ▼ Left_LowerArm
                        ▼ Left_Hand
                            ▼ Left_IndexProximal
                            ▼ Left_IndexIntermediate
                            ▼ Left_IndexDistal
                            △ Left_IndexDistalEnd

```

**Properties**

**Defaults**

**Rigidbody**

Mass	1	Apply to all
Linear Drag	0	Apply to all
Angular Drag	0.05	Apply to all
Linear Damping	0	Apply to all
Angular Damping	0	Apply to all

**Collider**

Material	None (Physic Material)	Apply To All
Collider type	Convex	Apply To All

**Settings**

Draw wireframe	Off
Dim other hulls	On
Hull transparency	0.5

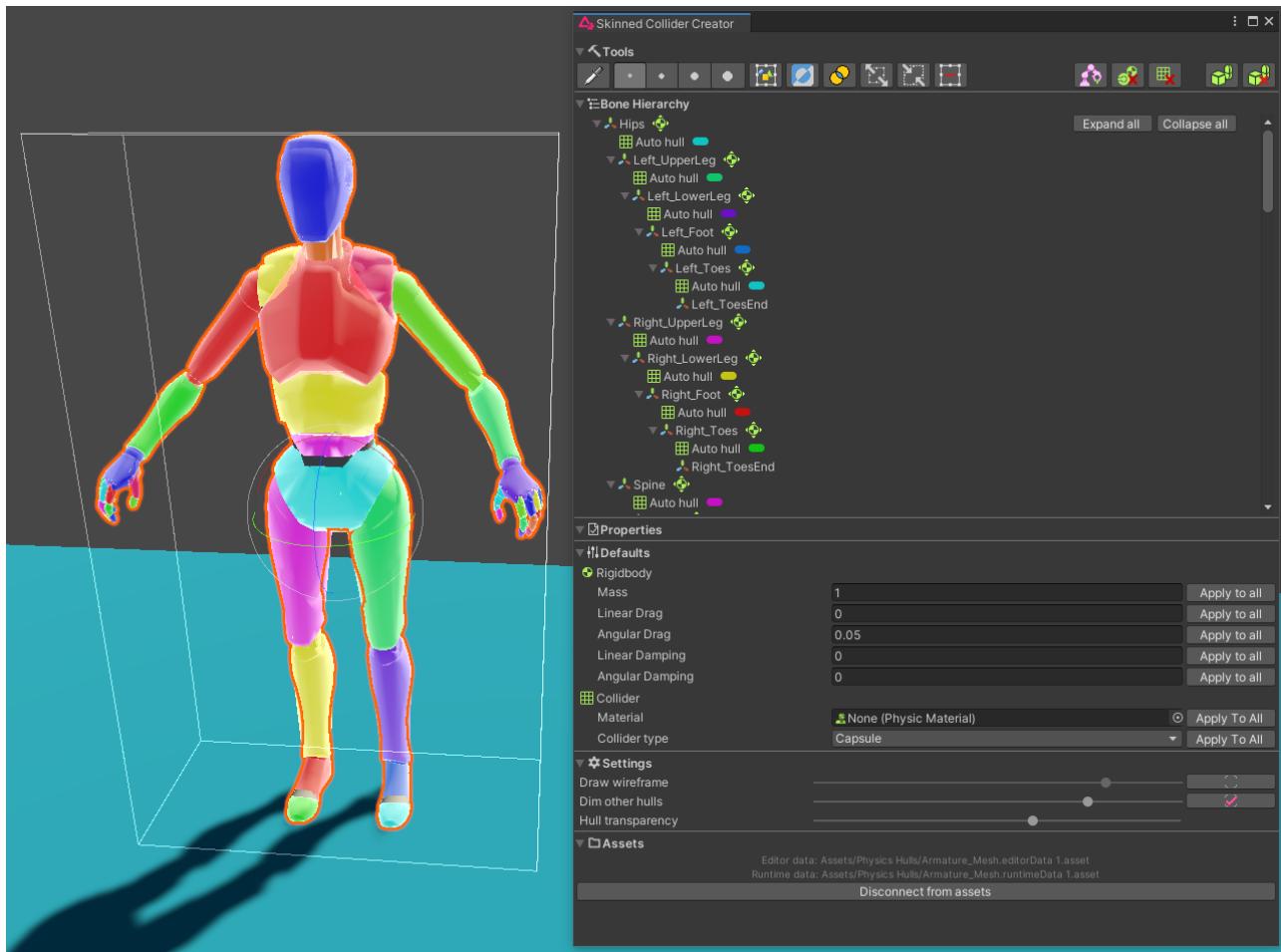
**Assets**

Editor data: Assets/Physics Hulls/Armature\_Mesh.editorData 3.asset  
Runtime data: Assets/Physics Hulls/Armature\_Mesh.runtimeData 3.asset

Disconnect from assets

Click the 'Auto Setup' button in the toolbar, and hulls will be created for all bones which have geometry attached to them.



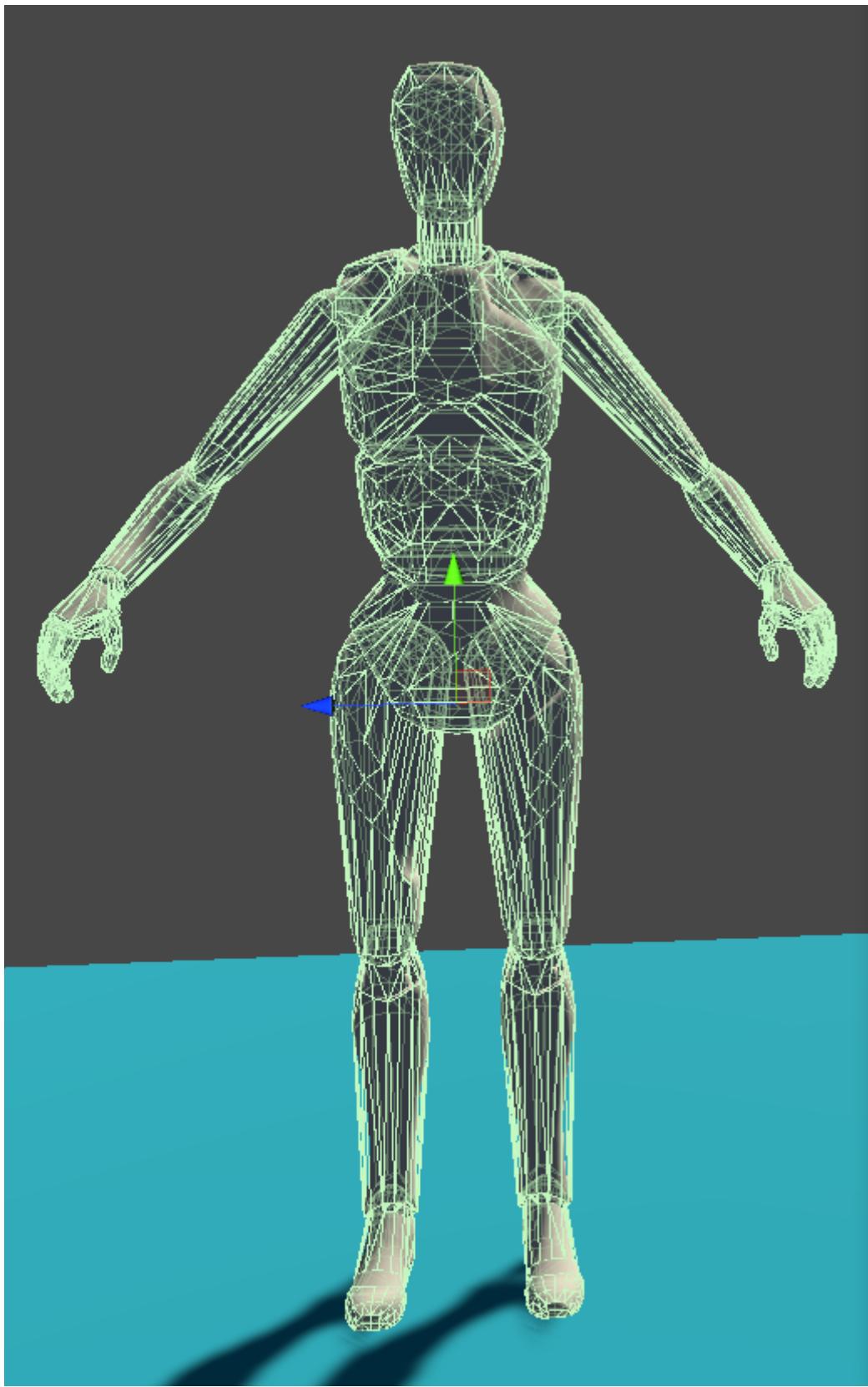


Each hull will show as a coloured overlay in the scene view, and a selectable ‘Auto Hull’ entry in the bone hierarchy.

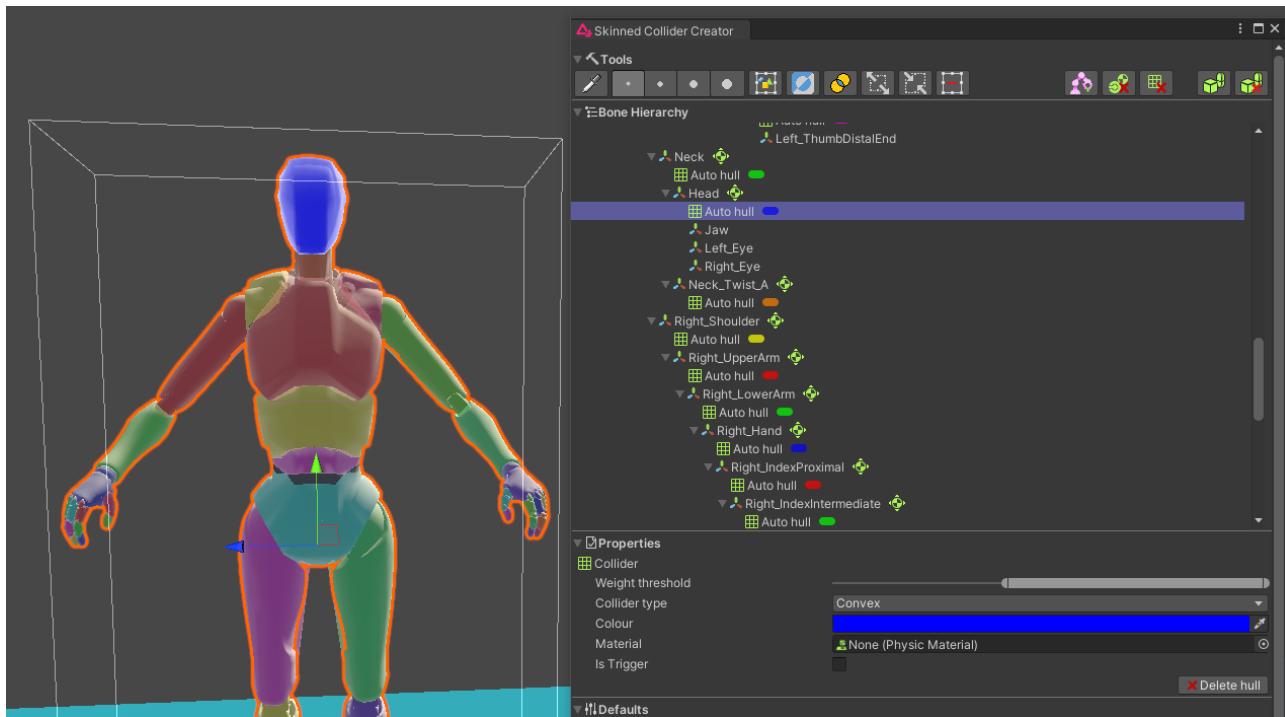
Then you just have to press ‘Generate Colliders’ to create collider components attached to your character:



Selecting the skeleton root in the Unity Hierarchy will now show you the generated colliders:



To edit the colliders further, select the Skinned Mesh Renderer, then select a hull to edit and change its properties. At first you will want to adjust the Weight threshold which controls how far along each bone the collider reaches. You may also want to change the collider type - Convex is often the most accurate but depending on your model you may want to use Capsule or Box instead.



Don't forget to press 'Generate Colliders' again to update the colliders after you've edited the hulls.

## Ui

### Tools



**Pipette** - used for selecting a hull from the scene view. Select this then click on a hull in the scene view to select it for painting.

**Brush size** - choose a brush size to paint faces for specific hulls.

**Select All** - Adds all faces as painted to the current hull.

**Select Inverse** - Inverts the painted faces on the current hull

**Select Unpainted** - Sets the painted faces to those which are not painted in any other hull

**Grow Selection** - Expands the currently painted faces by also selecting adjacent faces.

**Shrink Selection** - Shrinks the currently painted faces by unpainting faces at the edges.

**Select None** - unpaints all faces for the current hull

**Auto Setup** - Creates colliders for all bones that have geometry attached to them, and creates kinematic rigidbodies for all bones that contain colliders.

**Remove All Bodies** - Removes all Rigidbody setup data from the bones of the model. Note you must click 'Generate' for this to propagate to the actual components on the model.

**Remove All Hulls** - Removes all Hull setup data from the bones of the model. Note you must click 'Regenerate' for this to actually remove the collider components from the model.

**Generate** - regenerates the collider components for this hull

**Delete Generated** - deletes all generated objects - colliders, and child objects and child colliders

## Bone Hierarchy



The bone hierarchy shows the transforms which make up the skeleton and the hulls attached to those transforms. Icons next to the bone names indicate whether it has a rigidbody or not. Colliders are shown with the green collider icon, and a colour swatch to indicate the colour of the hull in the scene overlay.

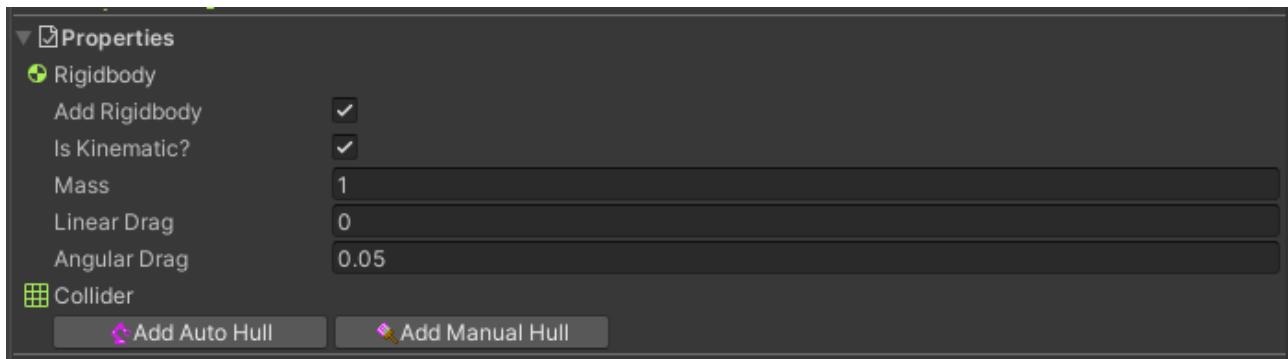
The currently selected element is highlighted. This is the one that is shown in the Properties ui.

**Expand all** - will expand the entire bone hierarchy so that it is all visible.

**Collapse all** - will collapse the entire bone hierarchy so that only the root is visible.

## Properties (Bone)

The properties ui shows the settings for the current selection in the Bone Hierarchy. When a bone is selected it looks like this:



**Add Rigidbody** - tick to create a rigidbody on this bone.

**Is Kinematic?** - tick to mark the rigidbody as kinematic. To have a collider follow an animated bone you want to tick both this and the 'add rigidbody' boxes.

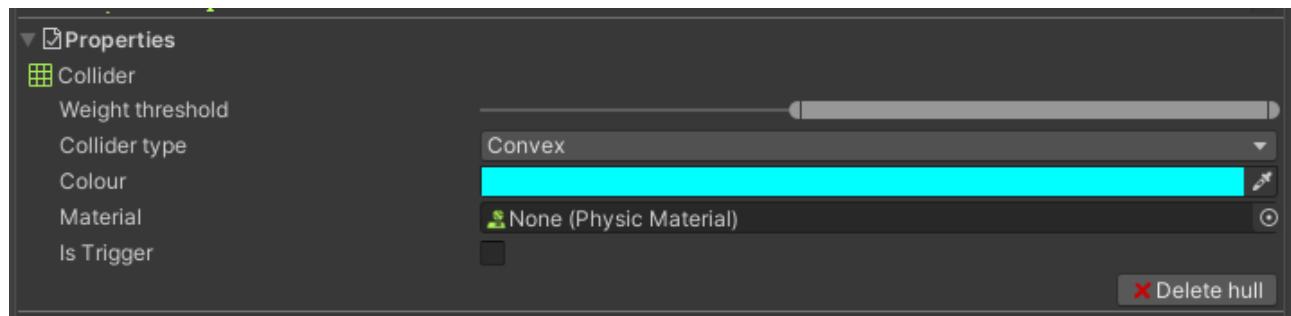
**Mass** - the mass for the generated rigidbody

**Linear drag** - the x/y/z drag for the rigidbody.

**Angular drag** - the rotational drag for the rigidbody.

## Properties (Auto Hull or Manual Hull)

When a hull is selected in the bone hierarchy the following properties are shown:



**Weight Threshold** - this is only shown for 'Auto' hulls. Drag the ends to set how much of the bone the collider will cover.

**Collider type** - the type of collider (Convex, Capsule, etc.) that will be created. See the notes on collider types earlier in the documentation.

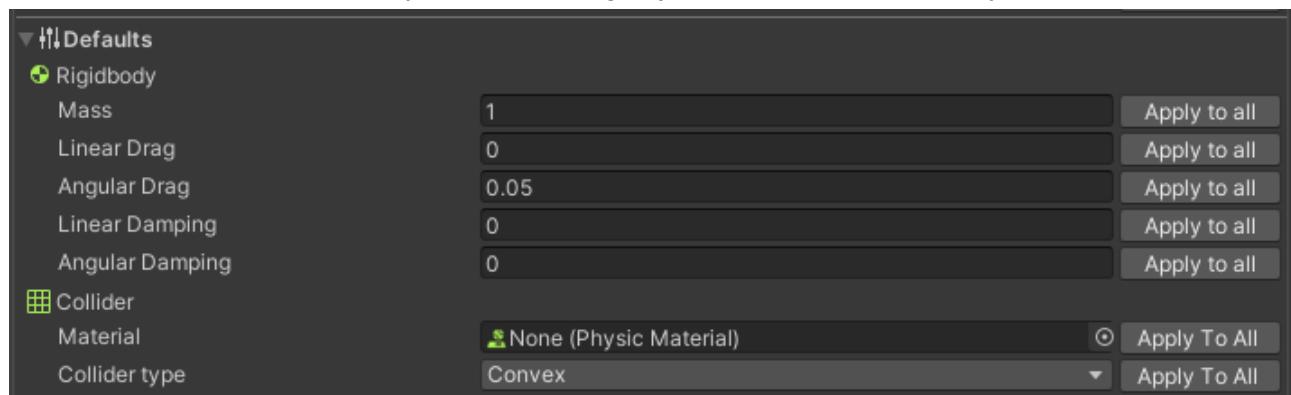
**Colour** - the colour of the hull in the scene overlay.

**Material** - the physics material that the generated collider will use.

**Is Trigger** - tick this to set the generated collider as a trigger.

## Defaults

These are the default settings used when a new bone or hull is created. Clicking 'Apply to all' will take the default value and apply it to all existing objects in the bone hierarchy.



## Settings

These settings control how the overlay in the scene view is drawn.



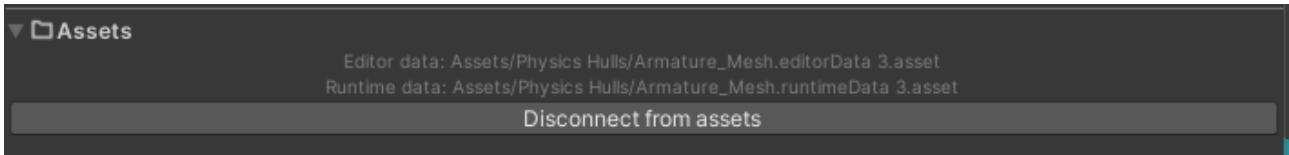
**Draw wireframe** - show the triangle wireframe of the current object.

**Dim other hulls** - When a hull is selected other hulls are dimmed to show which one you're editing. This slider affects the strength of this effect.

**Hull transparency** - how transparent the hulls in the scene view are shown.

## Assets

This shows what assets the window is currently editing.



Pressing 'Disconnect from assets' will remove the link and stop editing them. The colliders will stay on the object, and the markup data will remain in these assets, ready to be reconnected and edited further if needed.

## Extra Tools

### API Example / Bulk Collider Generation

Window > Technie Collider Creator > Run API Example (Batch Generate Colliders)

Select one or more components in a scene with a MeshRenderer in the hierarchy view, then choose this option to automatically create and generate colliders for them. You can use this to quickly create colliders for large amounts of objects, and then later go back and refine them further if you need more precision or the model changes.

This script also illustrates how to use the Collider Creator API to automate the process. It is located at `Technie/PhysicsCreator/Examples/ApiExample.cs` and is a good starting point if you want to write custom automation scripts for your specific needs. For more details refer to the comments in the example file.

### Generate Colliders From Selection

Window > Technie Collider Creator > Generate Colliders From Selection

A useful starting point when you're authoring physics for a level. Clones the current selection and wraps colliders around any object with a MeshRenderer on it. Use this to quickly fill in physics before hand adjusting the trickier areas.

### Span Physics

Window > Techie Collider Creator > Span From Selection

Tool for guided creation of colliders, especially for level geometry. Select several objects (such as the corners of a wall) and this will try and wrap the tightest possible box collider around them. Very useful for rapidly authoring collision data for walls, floors, ceilings, etc.





# Under The Hood

## Disconnecting From Editor Data

Physics Creator has been deliberately written to have zero CPU overhead when your game is actually running. The RigidColliderCreator/SkinnedColliderCreator component itself does no work, and only uses a tiny amount of memory for the painting data. Colliders are all created at editor-time and not game startup. So while building your game you can leave the RigidColliderCreator/SkinnedColliderCreator component on your objects for when you want to further edit your hulls.

If you want zero CPU and memory overhead (such as when you ship your game) you can remove the component. This means that the painting data asset will no longer be loaded, but all colliders will still work as before.

When choosing this option you will also get the option to delete the child components or leave them. The child components are just markers for each hull, and aren't used otherwise. Leaving them means that if you reconnect the data later on it'll know what hull the colliders were generated from so it's easier to carry on where you left off. Deleting them removes all Technie components so you have a completely clean object.

## Reconnecting Editor Data

If you disconnect an object from the painting data you can later reconnect it by dragging the painting data asset into the hull painter window. This will let you paint the hulls again.

## Install Location

Collider creator will auto-detect its install location, so once you've added it to your project you can move it to any folder in your Assets folder.

## Hull Data Folder

By default hull and painting data will be stored in `Assets/Physics/Hulls`. This will be created automatically if it does not exist, along with a marker file to identify it.

If you want your data to be stored elsewhere, then simply rename or move this folder - just make sure you move the marker file (`PhysicsCreatorHullFolder.asset`) as well. New data will be created wherever you have moved the marker file to.

## Hull Types

When creating hulls, you have a few types available:

**CONVEX HULL** - creates a convex hull around the selected faces. Highest accuracy and still fast and allows for rigidbodies. It is recommended to use this by default.



BOX - generates BoxColliders. Very efficient but axis-aligned (unrotated) by default. If you need rotated boxes tick the *As Child* option and choose an appropriate *Box Fit* method.

SPHERE - generates the smallest possible SphereCollider that encloses the hull. Very efficient, use this when you need a smoothly curved surface.

FACE - takes the selected faces, and gives them a thickness (from the 'face thickness' setting). Useful for flat surfaces such as floors, walls, ceilings, etc. The generated collider is a convex hull.

FACE AS BOX - the same as FACE, but the generated collider is created as a BoxCollider instead.

CAPSULE - Halfway between a box and a sphere, this is very efficient and great for long, thin volumes (like table legs) where you want smooth surfaces. Like BOX this is axis-aligned by default so use the *As Child* option for a tighter fit on angled volumes.

AUTO - Automatically creates convex hulls for the selection. The whole mesh is converted into hulls, and then trimmed down to only the volume defined by the painted sections. Use when you want results fast but don't mind small inaccuracies. You can use the *Select All* button to quickly apply AUTO to the whole mesh.

## Max Planes for hulls

While convex hulls are stored as triangle meshes in your project, when used by the physics engine they are internally converted into a set of planes which make up the hull (similar to the [UnityEngine.Plane](#) class). This means that the runtime performance is affected by the amount of planes in a hull, **not** the amount of triangles. For example, a cube with 12 triangles converts into 6 planes. A highly tessellated cube with 1200 triangles would still convert into 6 planes and have the same runtime performance.

Also, the physics engine supports a maximum of 255 planes per convex hull. Any hulls with more planes than that will be reduced to the maximum.

As an example, here is the default Unity sphere with different max planes settings:

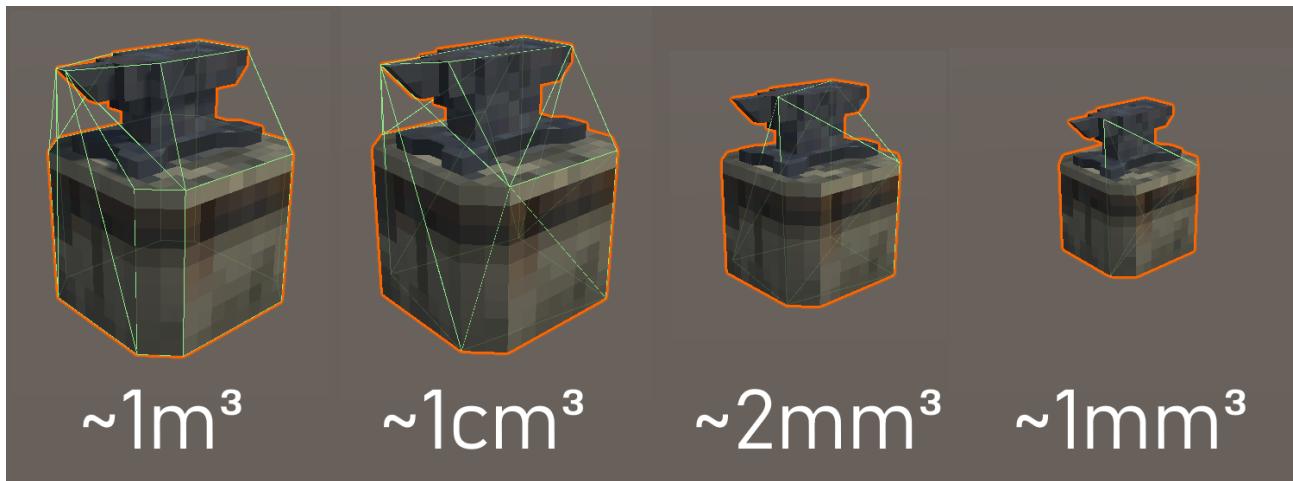


Notice that even a very smooth surface can often be reduced to 64 planes with very little difference in silhouette and accuracy but a significant performance difference, and depending on their in-game size, can often be reduced even further. Most objects (which are a mix of flat and curved surfaces) can often be set to 32 as an initial starting point.

## Hull Precision

Internally Unity's physics engine has limited precision for colliders, and will snap vertices that are very close together. Often this is helpful and improves robustness, but if you have very small objects then it can accidentally collapse the collision more than it should do. Note this can happen with regular mesh colliders, or those generated from Collider Creator.

As an example, here is the same anvil model with a single MeshCollider component (not generated by Collider Creator, but added by hand):



This model (when imported with the default import scale) is about 1m<sup>3</sup>. As the import scale is reduced the physics engine loses the definition of the shape. Note that this does **not** happen if you keep the import scale at default, and scale the model down using the Transform component - the vertex collapsing is done in model space, not world space.

So if you have very small objects which need precise colliders, consider making them bigger via import scale, then shrinking them again with Transform's scale.

# Keyboard Shortcuts

Most common actions can also be done with keyboard shortcuts. To view or rebind the shortcuts, open the shortcuts manager from the menus in **Edit > Shortcuts...**

Category	Command	Shortcut
All Unity Commands	Add Hull	Shift+Q
Binding Conflicts	Generate colliders	Shift+W
Main Menu	Delete generated colliders	Shift+E
3D Viewport	Stop painting	Shift+S
Android	Paint all faces	Shift+A
Animation	Unpaint all faces	Shift+Z
Collider Creator	Cycle brush size	Shift+1
Curve Editor	Select Pipette	Shift+`
Grid	Cycle hull type	Shift+T
ParticleSystem	Select next hull	Shift+=
Profiling	Select previous hull	Shift+-
Scene Picking	Toggle as child	Shift+Y
Scene View	Toggle is trigger	Shift+U
Scene Visibility	Delete active hull	Shift+Backspace
Snap		
Stage		
Terrain		

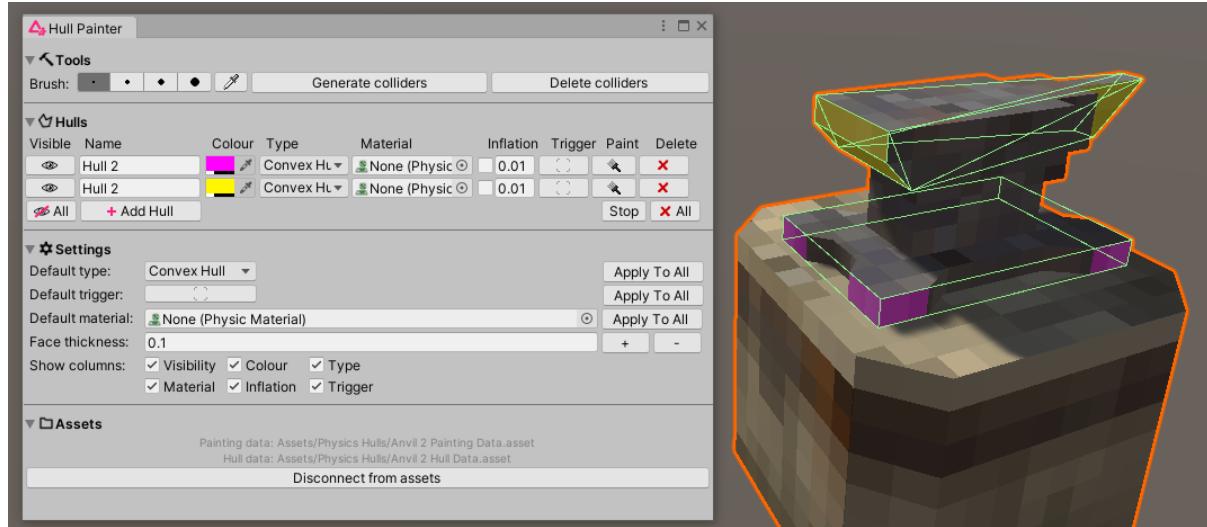
*Rebind keyboard shortcuts here*

By default all shortcuts use SHIFT to avoid conflicts with other commands, but they can be mapped to different keys here.

As this requires the shortcut manager, this feature is only available if you are using Unity 2019 or later.

# Tips

Remember that you do not need to paint all the triangles! The hulls will be made as small as possible around those that are painted. Often you only need to paint a few surfaces to get the shape you want, so experiment.



*These hulls only have a few faces painted but still produce a suitable collider*

If your model is small or contains small, difficult to select triangles then scale the whole model up while hull painting by scaling the transform. Your mouse / camera movement will be more precise. Then just return to normal scale at the end and generate the colliders again.

When you start painting you will add faces if you start your drag over an unpainted face, and subtract if you start over an already painted face. If you need to explicitly add or subtract then hold SHIFT (for add) or CONTROL (for subtract) when clicking.

While Box hulls are technically the most efficient, they are often much less accurate. I start by making everything as Convex Hulls and only change them to boxes if the difference is minimal. I have yet to see a real-world difference in performance in an actual game.

A lot of people are worried about the performance of MeshColliders in their game. It's important to understand that this component is really two different collider types in the actual physics engine, just exposed via one component. When 'convex' is not ticked it is a triangle mesh collider, which are typically expensive at runtime, and suffer from robustness problems because it's a surface representation, not a volume representation. Convex Collider does not use these. However when 'convex' is ticked then it is actually a convex hull, represented as a list of planes, which defines a volume. These are very efficient to process, and because they are a well defined volume (like Box, Sphere or Capsule colliders) they are much more robust as well.

Since colliders are generated to always fully enclose painted faces, you do not have to paint every face. Often you only need to paint the faces at the far extremes of the shape you want. This can be quicker to edit in some situations.

The Generate Colliders button will update existing colliders after the first usage, so use it often.

If you edit a generated collider manually then the painter will decide you have taken ownership of it and will no longer update it. Delete the collider and press Generate Colliders to return ownership.

When using AUTO hulls, start with the preset settings first and see if they're suitable, if not then start changing the values with the Custom preset. In general:

- If auto hulls aren't lining up with the edges of your model, increase the Granularity slowly
- If the hulls appear in the wrong places or the topography is wrong, adjust the symmetry and revolution biases
- If the hulls are too chunky, raise the Smoothness
- If you're trying to optimise the output hulls to be more efficient, lower the Smoothness, Concavity or Max Number Of Hulls.

## Known Issues

If you encounter errors when upgrading then delete the contents of Assets/Technie/PhysicsCreator, restart unity, and then reimport the package again. Your created data will be safe as it is stored separately in Assets/Physics Hulls.

If you see errors like 'Can't destroy Transform component of XXX' when opening or closing a scene in the editor, you're seeing this unity bug:

<https://issuetracker.unity3d.com/issues/cant-destroy-transform-component-error-is-thrown-when-closing-a-scene-that-contains-a-dontsave-prefab-with-a-child>

If you're trying to paint but instead are seeing the selection rectangle then you need to enable gizmos (press the Gizmos button at the top right of the Scene view). This is fixed for 2019.1 onwards.

If you have this error in the console after importing the project:

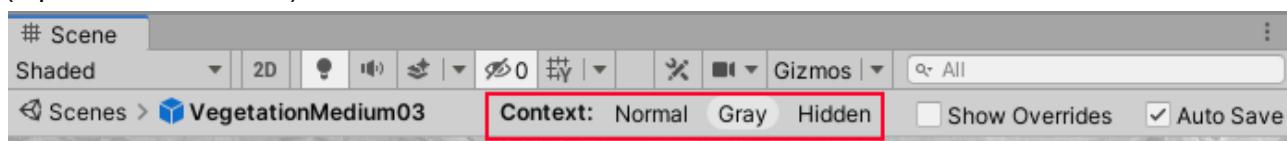
Instance of HullPainterEditor couldn't be created. The script class needs to derive from ScriptableObject and be placed in the Assets/Editor folder.

Then close and restart the editor and it will fix itself. This is a known bug in Unity 2017 LTS (but fixed in 2018 onwards).

If you receive a DllNotFoundException: libvhacd in the console, then you are probably missing the VS 2019 Redistributables on your computer. The installer can be found on the Microsoft website at <https://visualstudio.microsoft.com/downloads/>

Or download them directly from here: [https://aka.ms/vs/16/release/VC\\_redist](https://aka.ms/vs/16/release/VC_redist)

If editing a prefab within a scene and you cannot see the hull overlay, make sure your 'Context' (top of the scene view) is set to 'Normal' or 'Hidden'.



# Problems? Feature Requests? Bugs?

If you need support or have a bug you can reach us via Discord in our support channel here:  
<https://discord.gg/DrQnsmZC2c>

Or send an email to [technie@triangularpixels.com](mailto:technie@triangularpixels.com)

Please include your Unity version, and OS in any support emails. If reporting a bug then if you include a (small!) reproduction project with instructions on how to reproduce your bug then we'll be able to fix things *much* quicker. Please also remember to include meta files as these contain import settings for meshes that are often important. Thanks!



## Licences

Technie Collider Creator is provided under the Unity Asset Store Licence

[https://unity3d.com/legal/as\\_terms](https://unity3d.com/legal/as_terms)

Collider Creator uses the V-HACD library by Khaled Mamou, licensed under the BSD 3 licence.

See the 'Third Party Notices.txt' in the Plugins folder for more details.

The Humanoid model in the example scene is included from the 'Unity Starter Assets - Third Person Character Controller' package, under a Unity Companion Licence.

