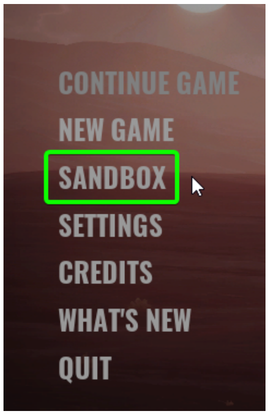
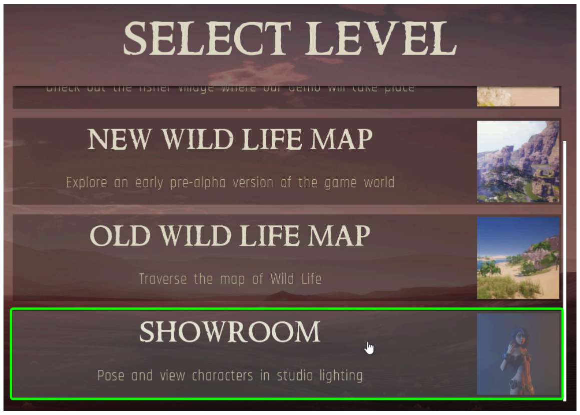
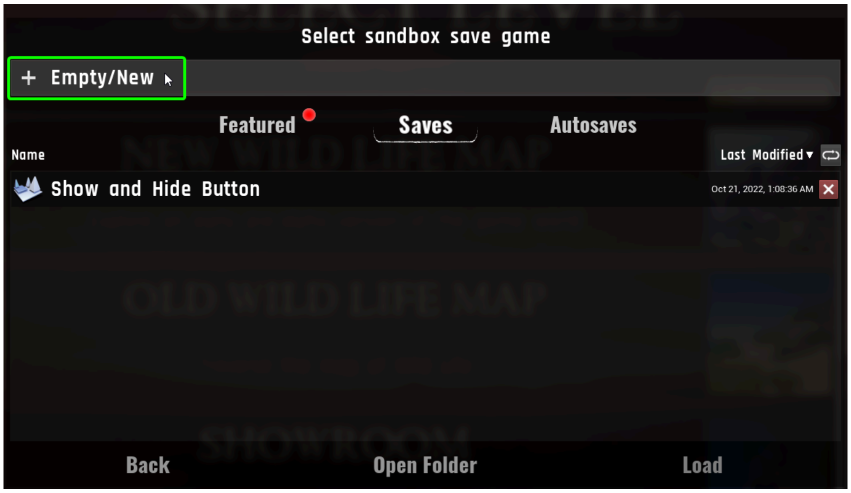
Sandbox Basics

This tutorial is nowhere near a complete documentation of all sandbox features, but hopefully it will help get you started with the essentials!

Let's get started by opening a fresh sandbox scene. We will choose the showroom for our Tutorial:



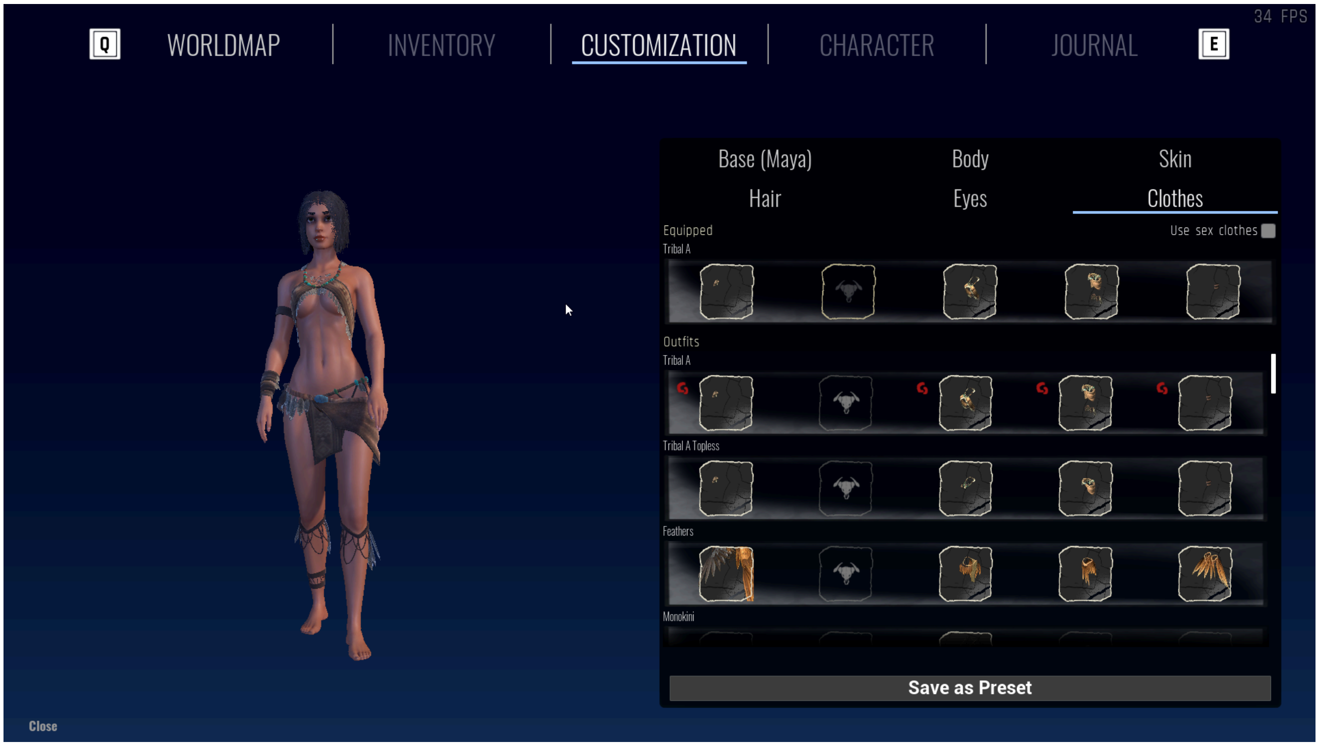

SELECT LEVEL
Check out the fisher village where our demo will take place
NEW WILD LIFE MAP
Explore an early pre-alpha version of the game world
OLD WILD LIFE MAP
Traverse the map of Wild Life
SHOWROOM
Pose and view characters in studio lighting



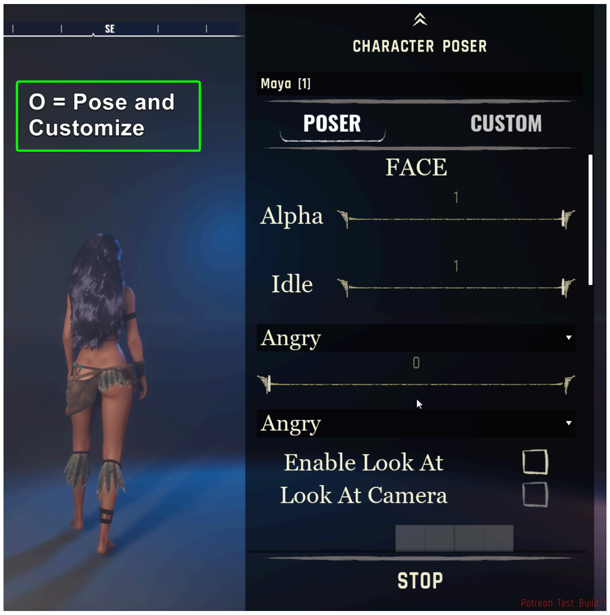
Once in there, we have a few important hotkeys. (You can change these under Input Settings).

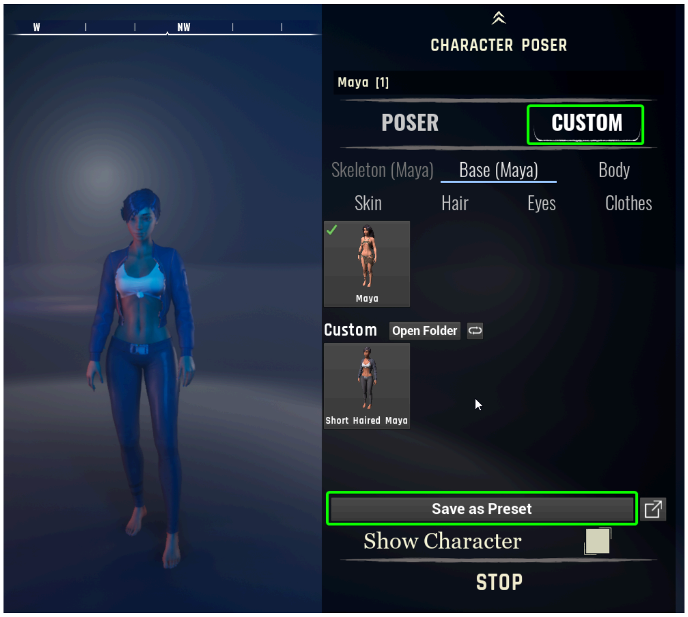
The game has a little focus bug sometimes: If nothing happens when you press these after loading the scene, try clicking your mouse once!

"**I**" opens your Inventory, where at the moment you can change outfits, customize your character and look at the worldmap

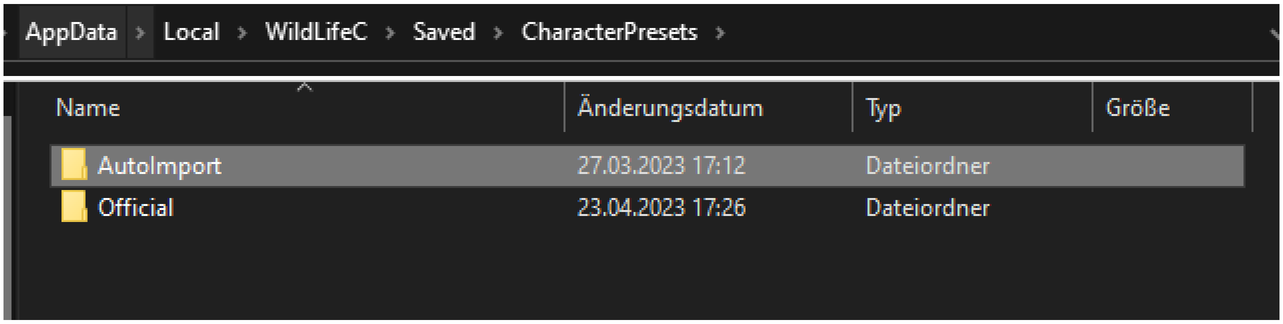


“**O**” opens the character Customizer and Poser. You can blend two poses together with the sliders - don’t forget to move the “Idle” slider to zero to see your new pose! In the custom tab you can also change a character's appearance and save it as a preset.





Your character presets are saved under %localappdata%\WildLifeC\Saved\CharacterPresets. If you drop a downloaded preset into the AutoImport folder there, it will automatically get sorted into the right folder when you start the game.

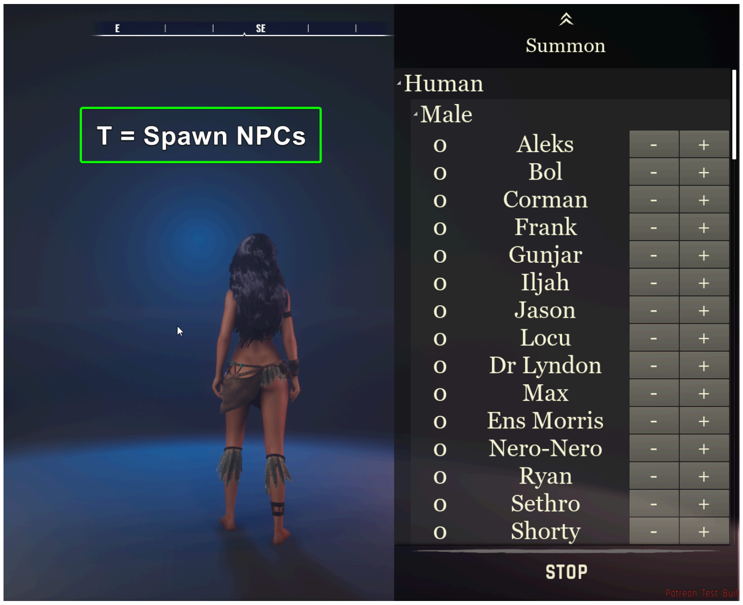


AppData > Local > WildLifeC > Saved > CharacterPresets >

Name	Änderungsdatum	Typ	Größe
AutoImport	27.03.2023 17:12	Dateiordner	
Official	23.04.2023 17:26	Dateiordner	

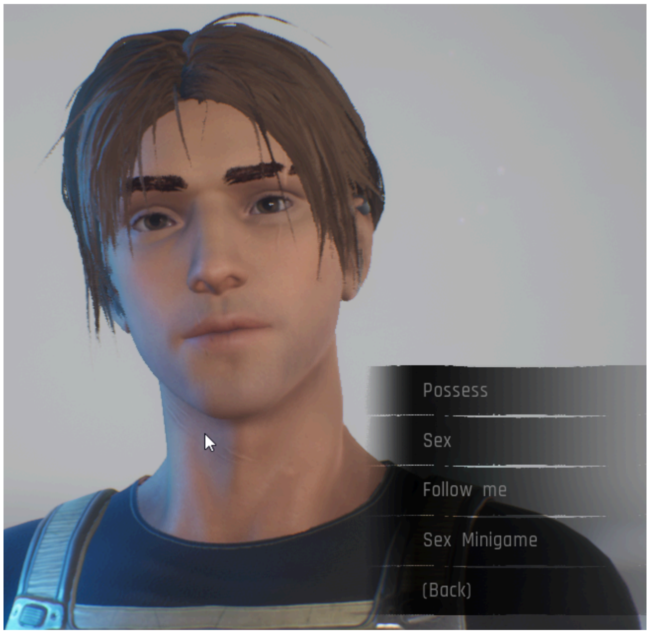
To switch to a different character, we will need this next menu:

“**T**” opens the spawner, where you can add interactable NPCs to your scene.



Click the “**+**” next to a name to spawn a copy of that character. The “**-**” removes them again.

Once you have spawned an NPC, talk to them by pressing “**E**”**.**

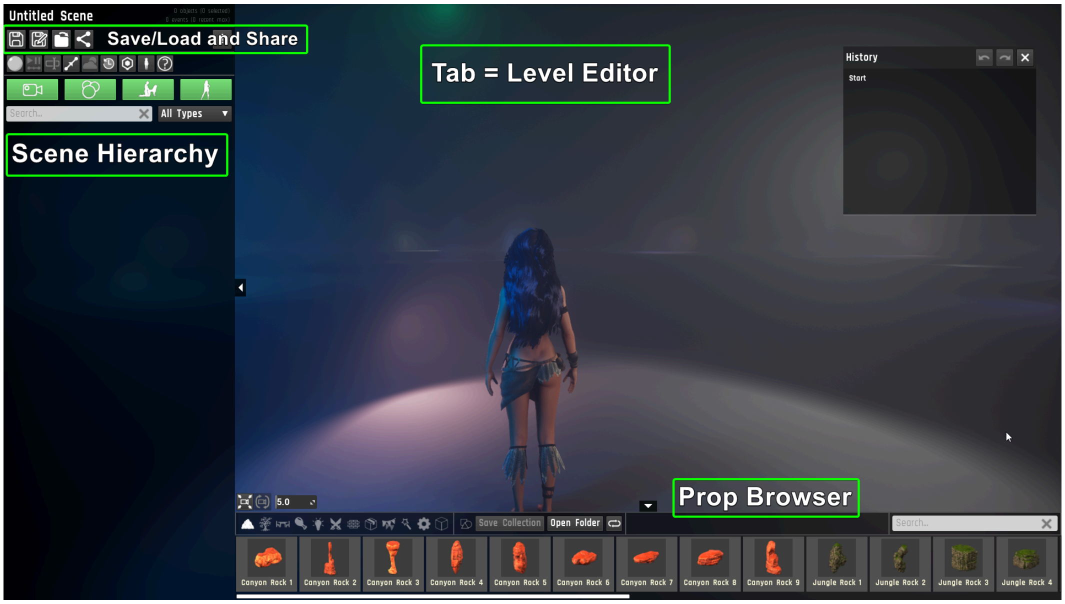


In the dialogue cheat menu you have various options:

Possess = Play as this character
Sex = Have sex with this character. Only available if a sex animation is available for that pairing
Sex (Group) = Only available when other characters are following you and a group animation is available for that combination
Follow Me = Have this character follow you. Note that this may not work everywhere, only in areas where the level has a navigation mesh.
This is also needed to start group scenes! First, ask all participants to follow you. Then when you talk to one of them, the Sex option will offer the group animations available for that combination. This also works for minigames.
Sex minigame = Starts a longer, interactive cinematic sex scene. Only available for the protagonists Max and Maya in combination with those characters that have a minigame with them.

The Level Editor

“**Tab**” opens the Level Editor. This menu allows you to create your own levels, place sex scenes, poses and characters in it and make interactive setpieces via events.



Save/Load and Share

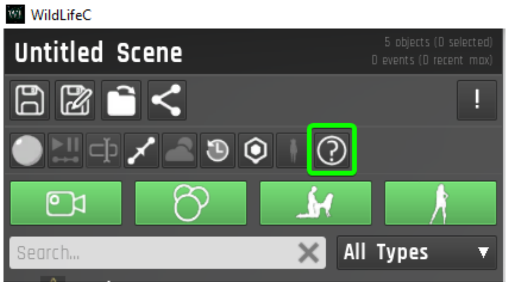
Tab = Level Editor

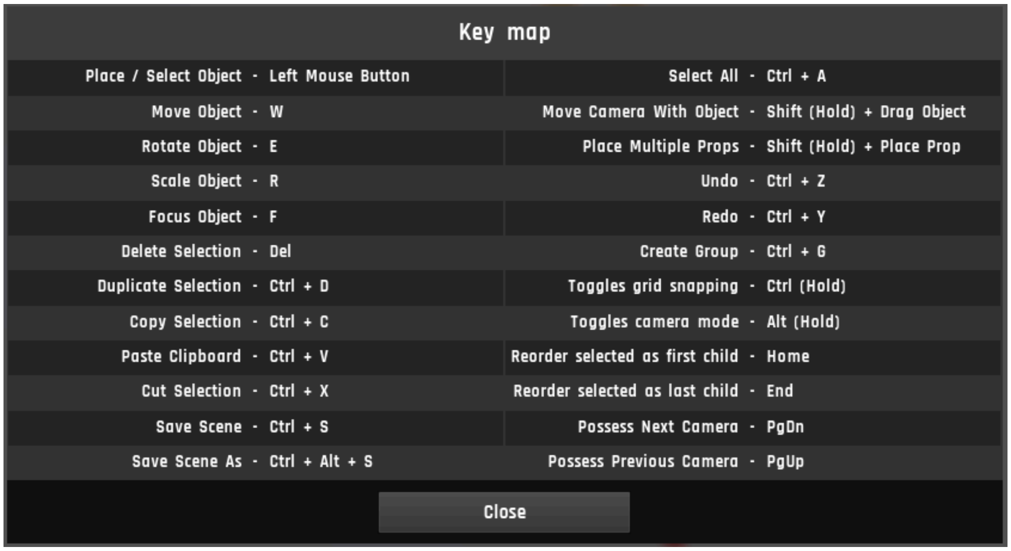
Scene Hierarchy

Prop Browser

To look around in this mode, hold the **Right Mouse Button.**
While holding it, **W, A, S, D, Q** and **E** will move the view. Scroll to make the camera move faster or slower.

Clicking the question mark in the upper left menu brings up a map with many useful hotkeys.

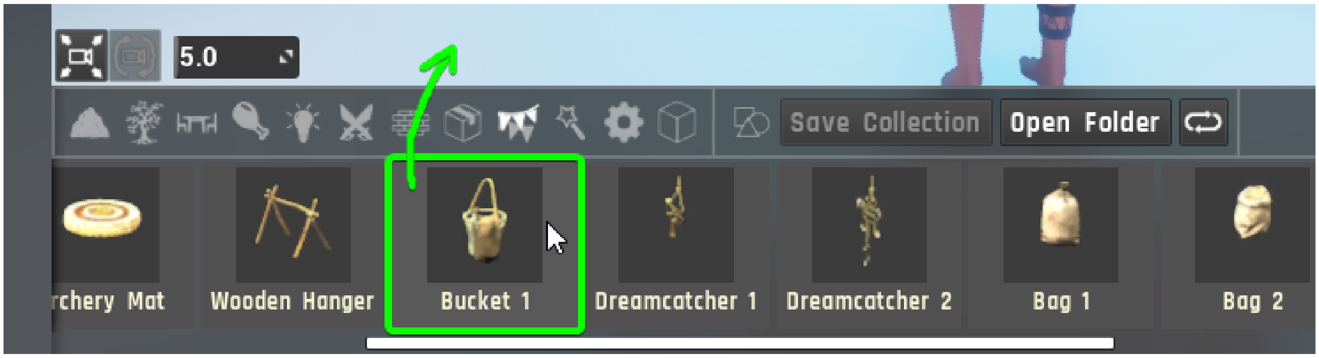




Key map

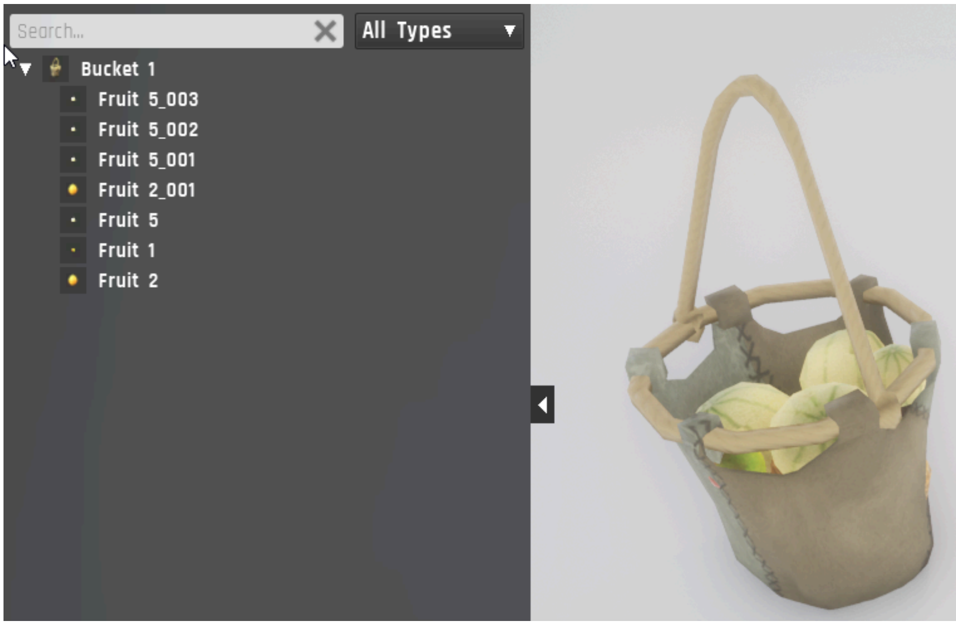
Place / Select Object - Left Mouse Button	Select All - Ctrl + A
Move Object - W	Move Camera With Object - Shift (Hold) + Drag Object
Rotate Object - E	Place Multiple Props - Shift (Hold) + Place Prop
Scale Object - R	Undo - Ctrl + Z
Focus Object - F	Redo - Ctrl + Y
Delete Selection - Del	Create Group - Ctrl + G
Duplicate Selection - Ctrl + D	Toggles grid snapping - Ctrl (Hold)
Copy Selection - Ctrl + C	Toggles camera mode - Alt (Hold)
Paste Clipboard - Ctrl + V	Reorder selected as first child - Home
Cut Selection - Ctrl + X	Reorder selected as last child - End
Save Scene - Ctrl + S	Possess Next Camera - PgDn
Save Scene As - Ctrl + Alt + S	Possess Previous Camera - PgUp

At the bottom you see the **prop browser**. Simply click on a prop then place it in the level or drag it into the scene.

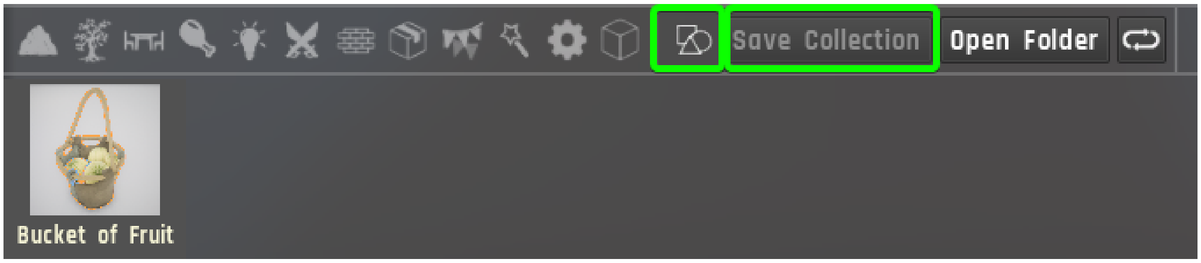


It will then also appear in the **scene hierarchy** on the left.

Here you can rename props, change their attributes (if they have them) and group them to order your scene. Drag one props name onto a group or other prop to parent it under there. It will then move along with the parent.

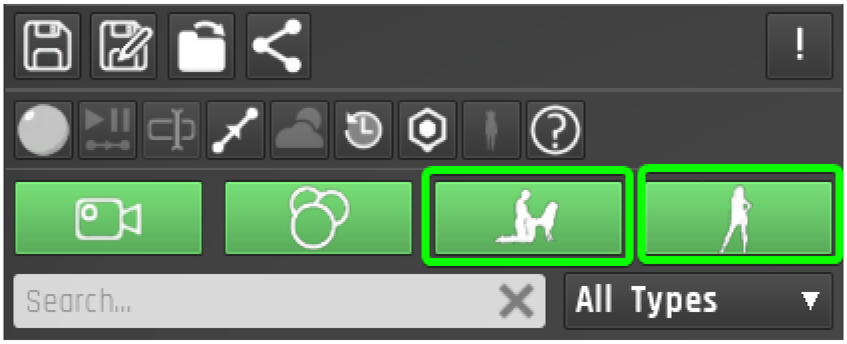


The best way to reuse your favorite prop combinations in other scenes is to save them as a collection.

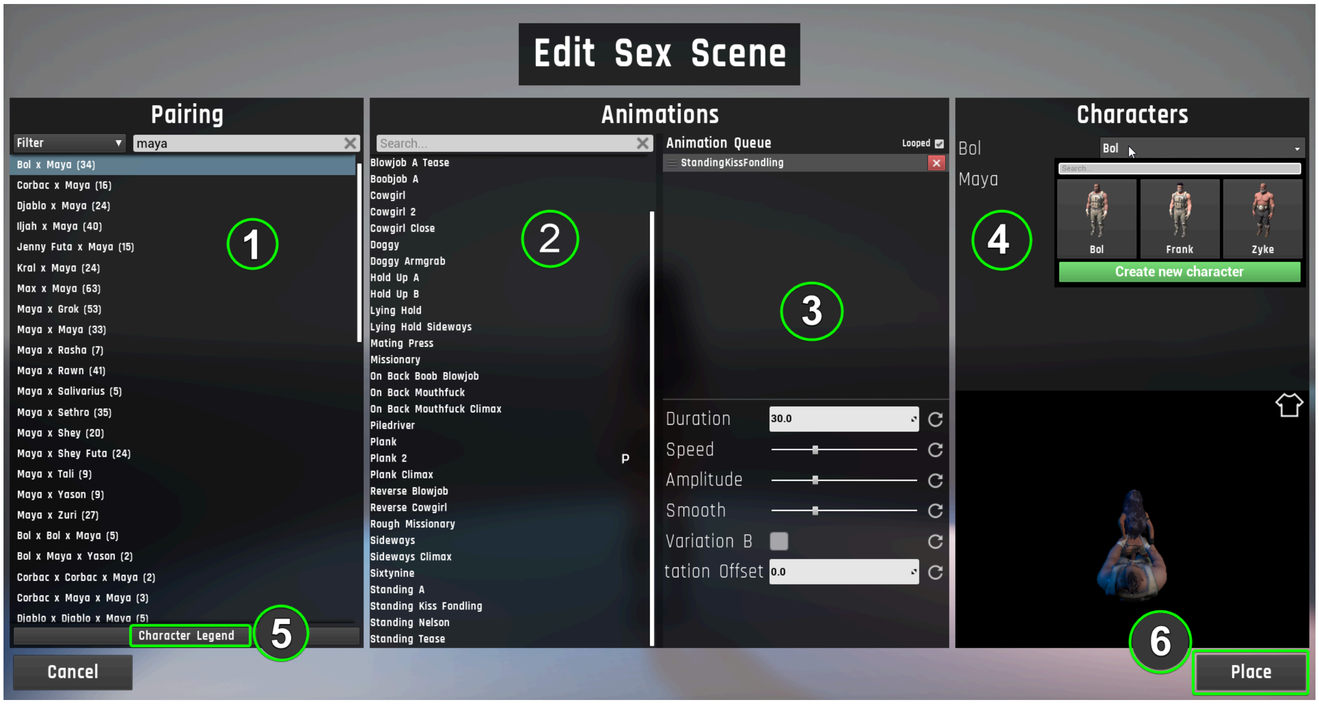


Simply select the uppermost object/group (in this case the bucket) and hit Save Collection. You will then always find it as a placeable object in the collections tab right next to the button.

Sex Scenes and Posed Characters can also be placed as decorative objects. You won’t find them in the prop browser, but instead under one of the green buttons above the hierarchy view:



Clicking the Sex Scene button opens a window where you can customize a scene:



Filter: maya

Duration 30.0
Speed
Amplitude
Smooth
Variation B □
tation Offset 0.0

Work your way through from left to right.
First, find a couple you like **(1)**. The character legend below **(5)** shows you which kind of character is behind each name. You can also apply filters, for example to exclude humans or kerpali from the list.

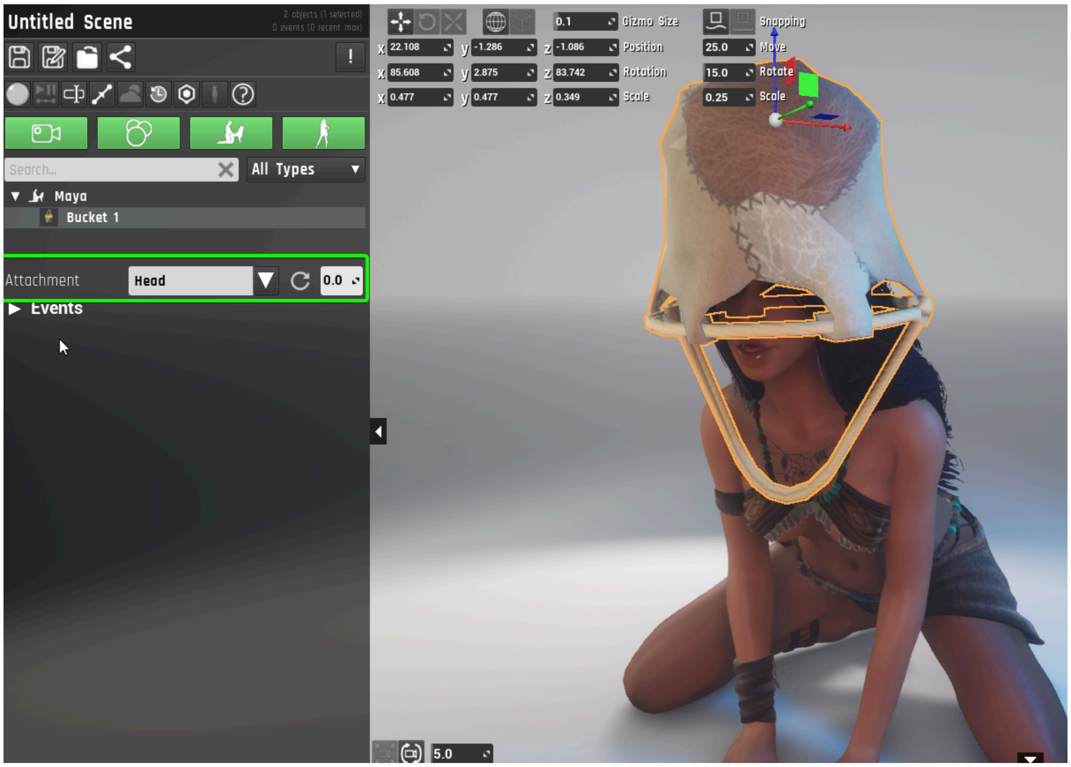
Next, choose your animations **(2)**. You can reorder them and tweak settings for them in the next section **(3).**

You can also add pairings **(1)** and Animations **(2)** to your favorites by clicking the heart icon next to them. This will move them to the top of the list for easy access.

Now, choose which sub-characters you want to use. **(4)** Each archetype comes in multiple variations - for example, there are three official characters of the “Bol” body type, as you can see. Your own saved customizations also appear here.

Lastly, click Place **(6)** to place your sex scene in the level.

Once your scene is in there, you can move, rotate and scale it like any other object.



Attachment Head 0.0

x 22.108 y -1.286 z -1.086 Position

x 85.608 y 2.875 z 83.742 Rotation

x 0.477 y 0.477 z 0.349 Scale

Gizmo Size 0.1

Move 25.0

Rotate 15.0

Scale 0.25

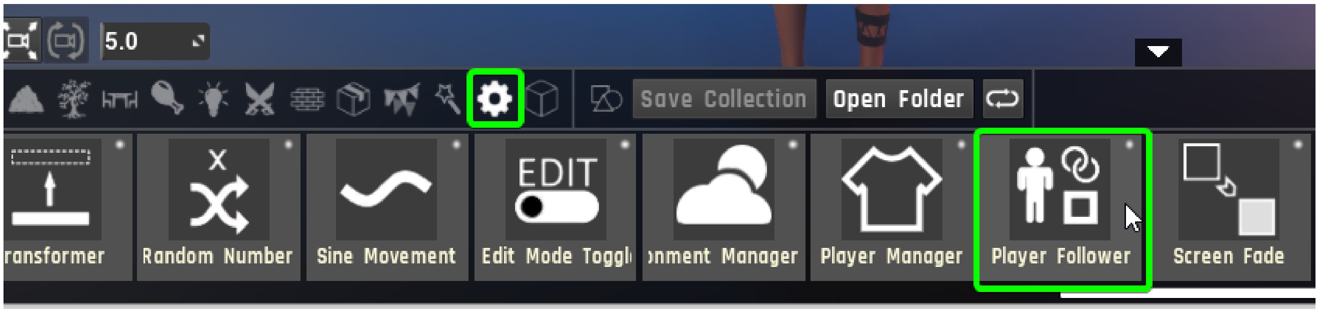
You can also parent things under it. This will make a special “Attachment” attribute appear, where you can select bones to attach your objects to. This way you can, for example, attach custom jewelry to characters.

The number to the right is the character index - if the sex scene has multiple characters, this decides which character to attach to.
-> only in older versions from 2022. Newer versions have both characters as children of the sex scene object, so you can just parent things under them directly!

You can also attach things to the player character, no matter which one that currently is. This can be useful to have, for example, a light always follow the currently possessed character:

Go to the **Automation** tab in the prop browser and find the **Player Follower** prop. Add it to your scene and it will stick to the playable character. Anything parented under it will therefore stick to the player as well.

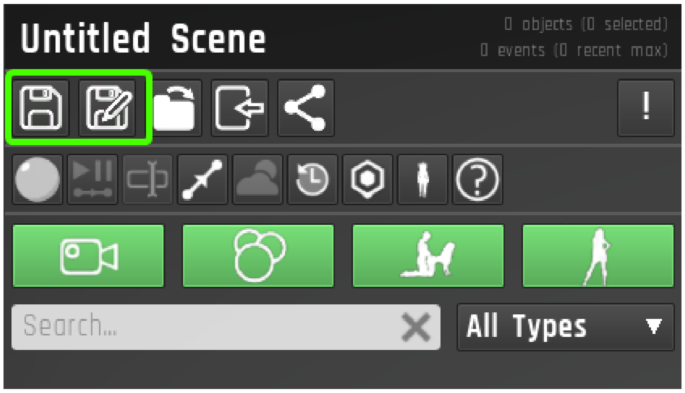
Remember to turn off Collision for those props, or they will interfere with you as you try to walk around!
(On some premade props this option might not be available yet.)



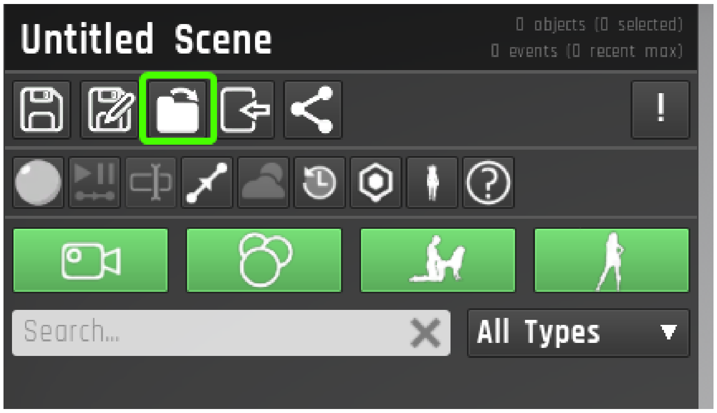
Saving, Loading and Sharing Scenes

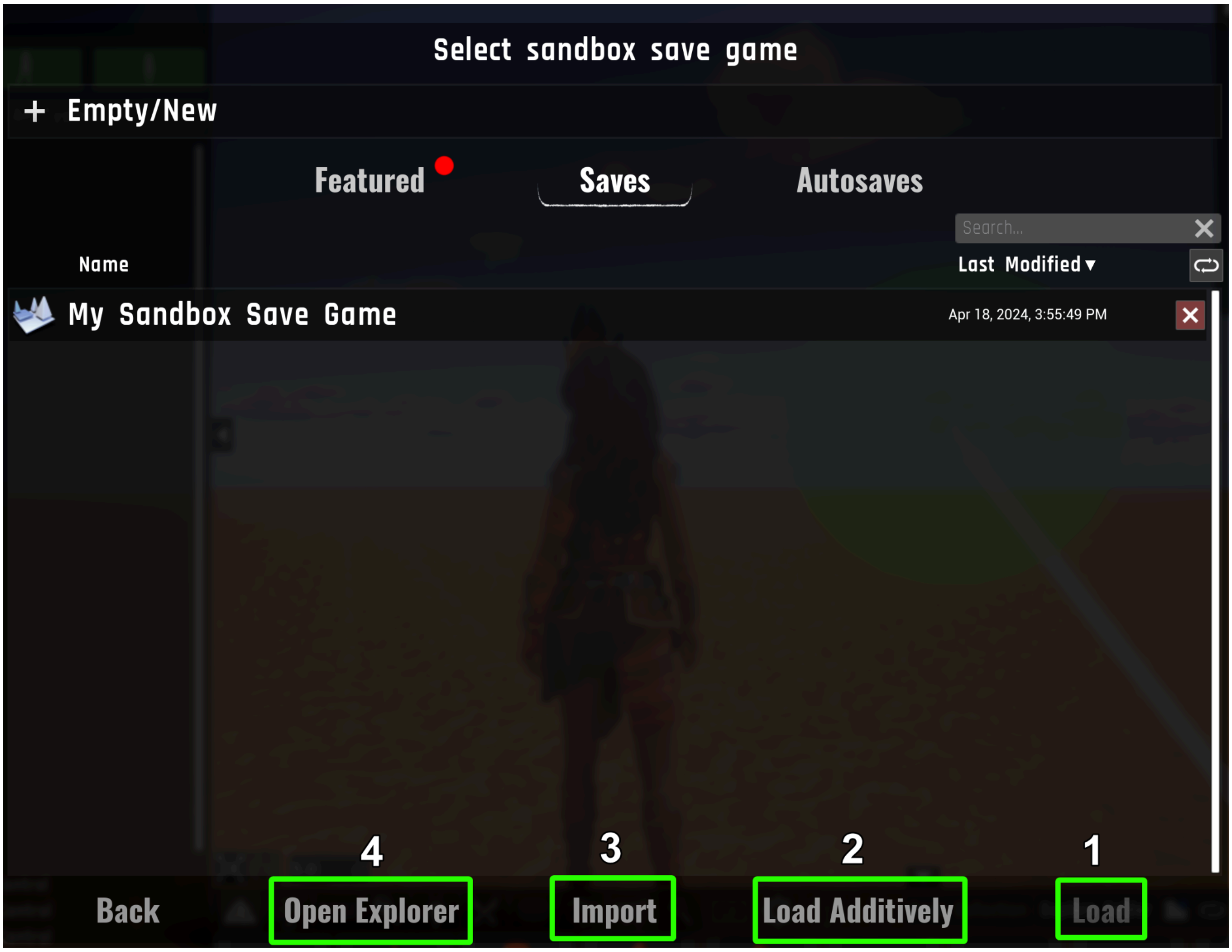
Save/Load

After creating your sandbox scene, it's time to save it. Simply click one of these buttons, choose a name and you're done:



To load a scene, click the button right next to them:





Here you can see all available saves for your sandbox map. You can load them normally **(1)** or load them additively **(2)**.

Additively means that the save files contents will be added to your open scene, instead of replacing it. This can be useful, if you want to merge two scenes together.

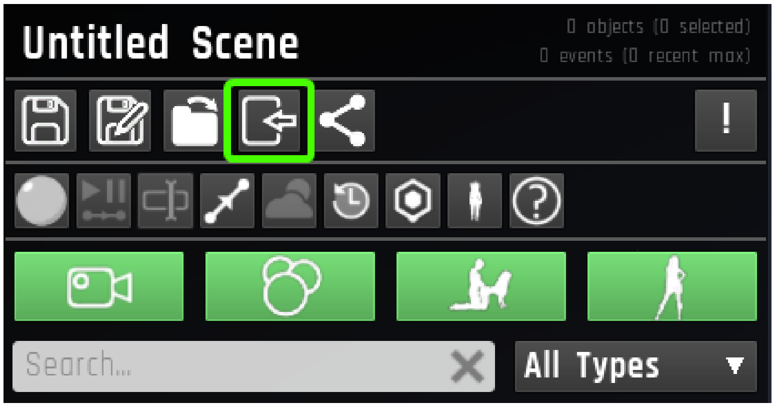
Import (3) lets you import maps you've downloaded from elsewhere. (More on that in the sharing section below!)

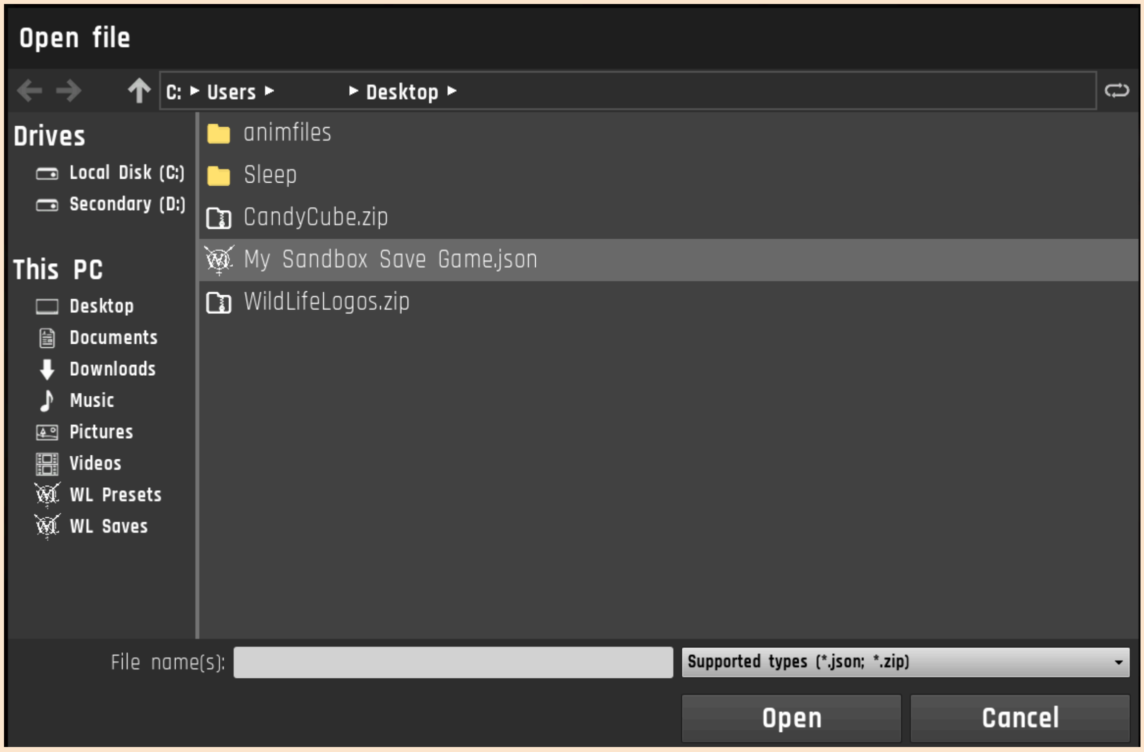
Open Folder (4) opens the save location under %localappdata%\WildLifeC\Saved\SandboxSaveGames\ on your PC. Here you can find your save .json which you can share with other people.

(You should only share the .json directly when you do not use any externally loaded local assets, otherwise other people will not see them)

Sharing

To import such a save from someone else, you can click **Import** in the Load popup (see above) or the button next to the load button, which will give you a file popup to search for the file to import.





File name(s):

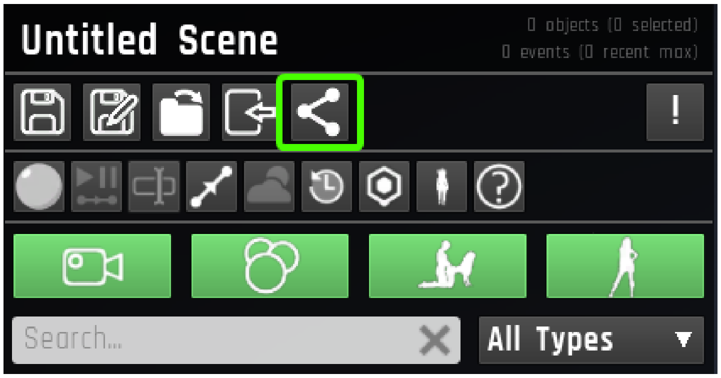
Supported types (*.json; *.zip)

As you can see, the popup accepts both **.json** files and **.zip** files.

*In the latest versions of the game, .zip has been replaced by the **.wlsave** format. It is the same thing, just renamed!*

.zip (old) or **.wlsave** (new) files are needed for when you have custom textures, models, videos or audio in the game that the other person does not have on their PC. (See the “Importing Your Own Models” section below for more information!)

To export such a scene properly, you need to click the Share button:


Untitled Scene

0 objects (0 selected)
0 events (0 recent max)

Search...

All Types

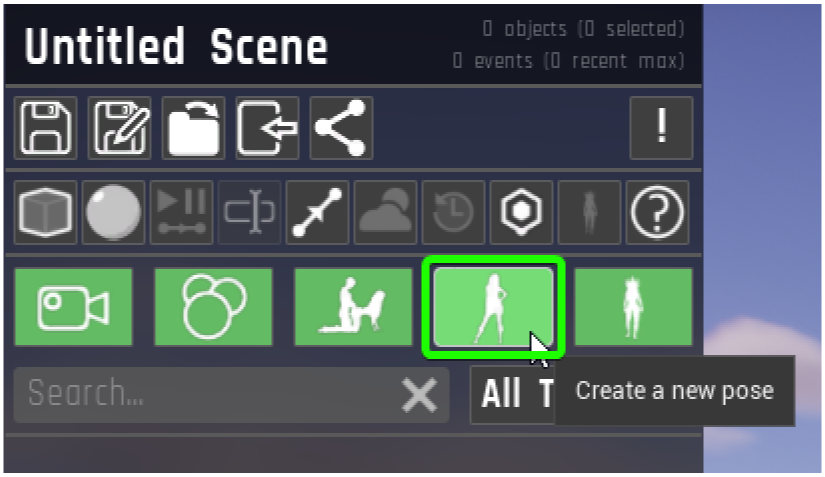
It will open a file dialogue again, where you can choose a place to save your zip archive. The game will bundle all custom assets and textures in there, so someone else can import them, following the Import instructions above.

The Custom Pose Tool

Making your own poses

(Available since February 2023 version)

The custom pose tool lets you pose a character with interactive controls. In order to do this, you first need to create a pose and place it in the scene.


Untitled Scene

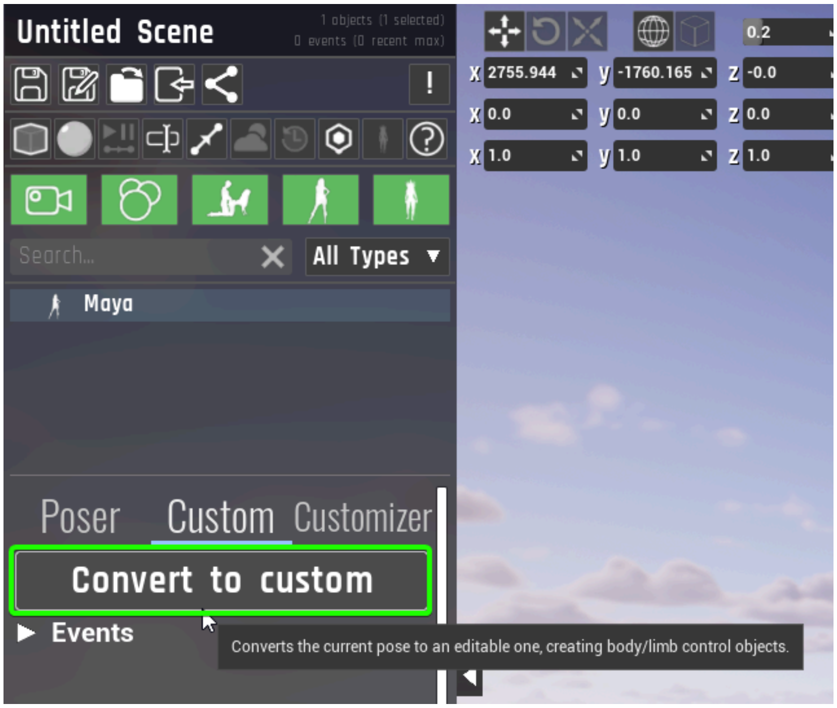
0 objects (0 selected)
0 events (0 recent max)

Search...

All T

Create a new pose

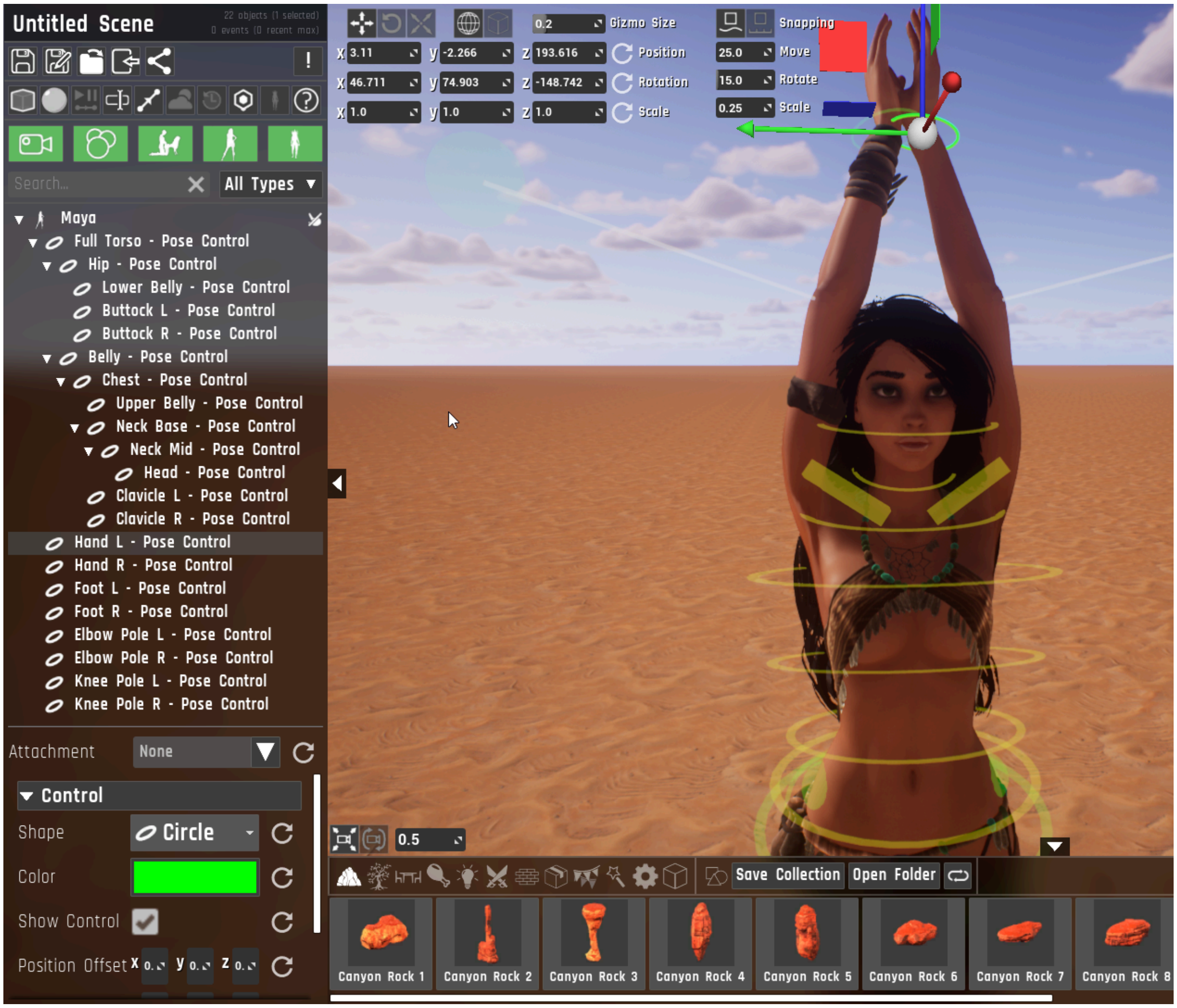
You can choose a base pose, if you want, by using the already familiar blend sliders. Then, convert it to an editable pose via this button:



Converts the current pose to an editable one, creating body/limb control objects.

This creates a number of controller objects in your scene hierarchy. Each of these can be grabbed and moved or rotated to pose your character. They can also be affected by automation props or attached to things in the scene, just like any other object.

If you have trouble selecting them in the viewport, set the character itself (topmost object) to non-clickable via the little finger icon next to the name!



Gizmo Size 0.2

Position x 3.11 y -2.266 z 193.616

Rotation x 46.711 y 74.903 z -148.742

Scale x 1.0 y 1.0 z 1.0

Snapping: Move 25.0, Rotate 15.0, Scale 0.25

Attachment: None

Control

Shape: Circle

Color: 🟩

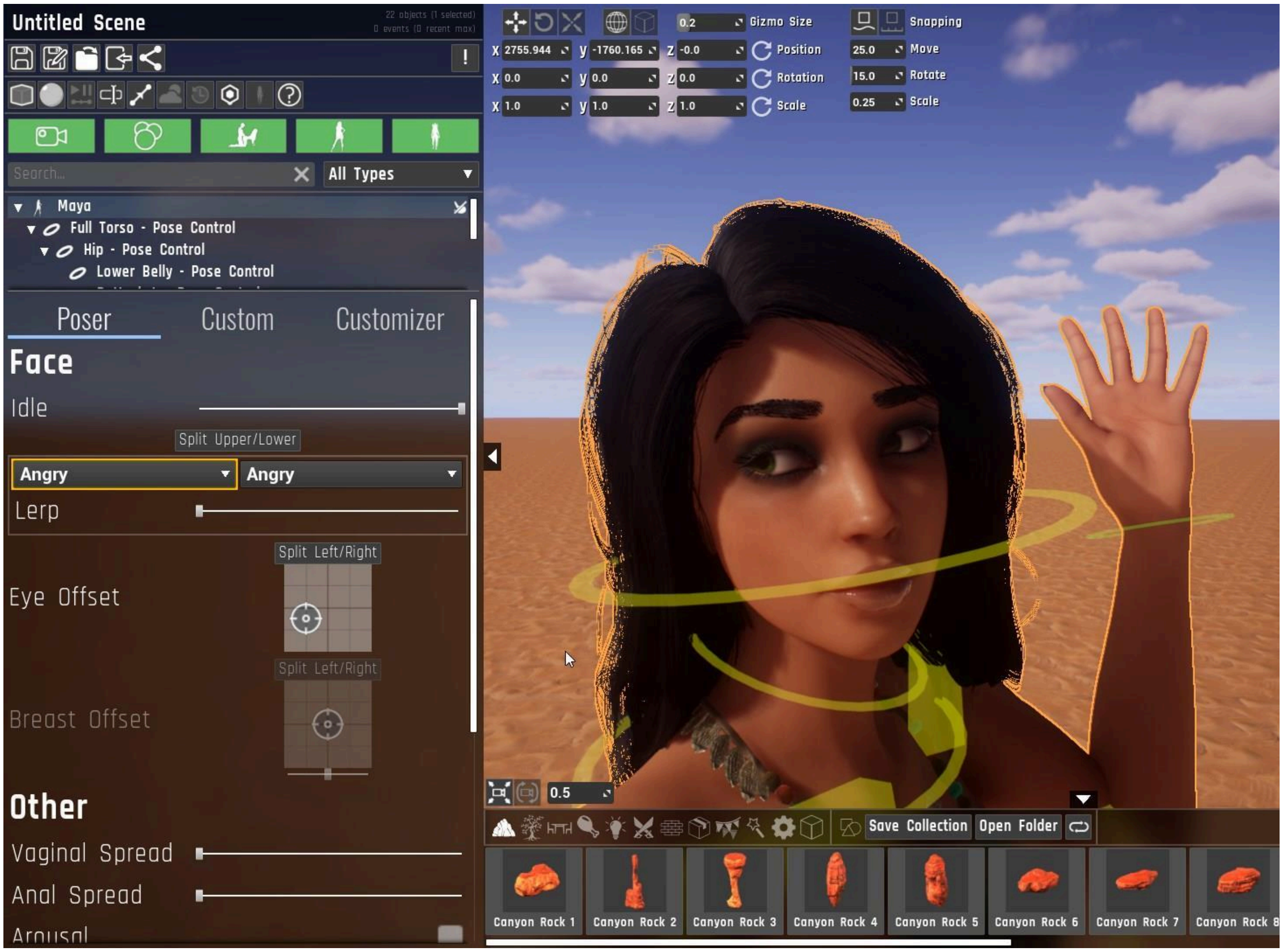
Show Control: ☑

Position Offset x 0.. y 0.. z 0..

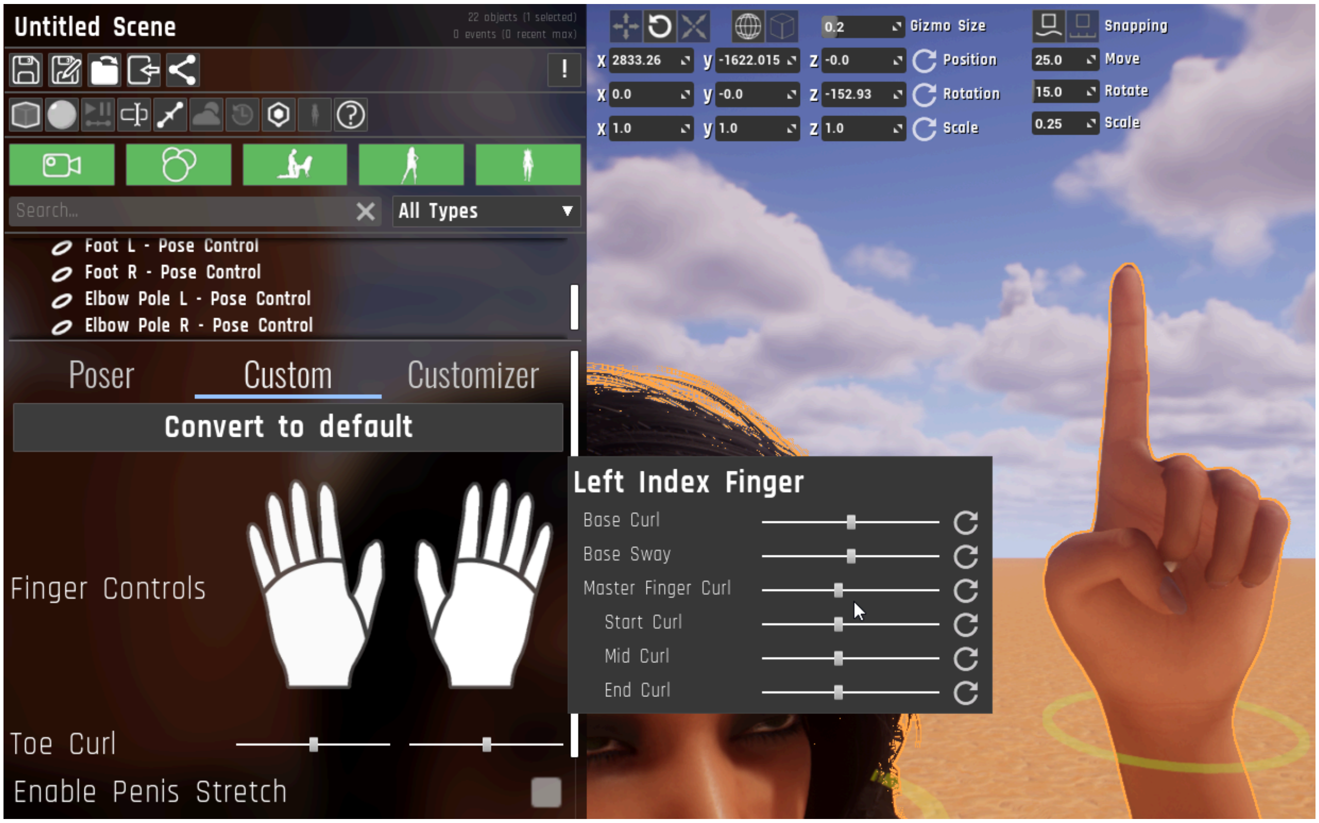
0.5

Face, boob and vaginal/anal controls are found on the character itself, in the basic **Poser** Tab. They are always available, whether custom pose is active or not:

Don’t forget to switch off breast physics to enable manual boob control!



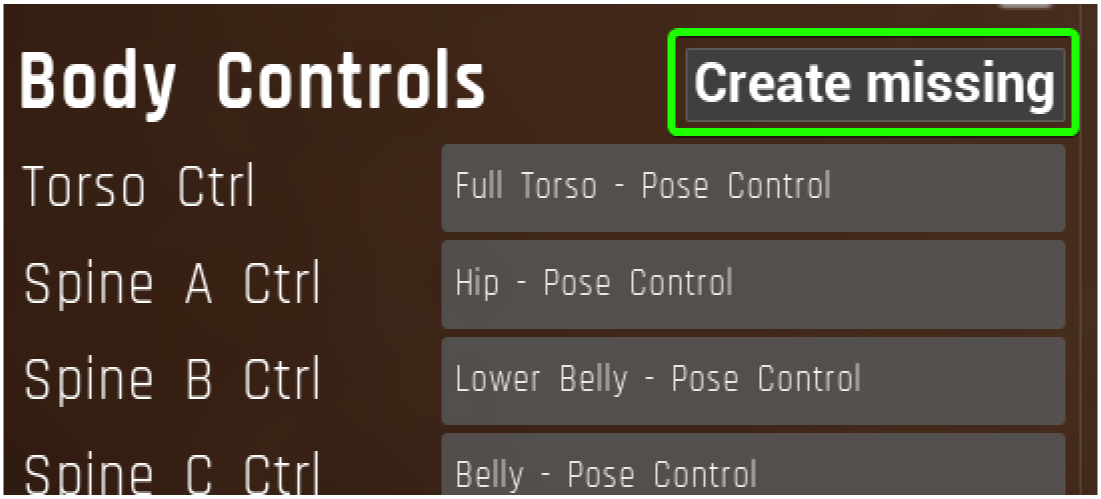
The controls for Fingers and Toes can also be found on the character itself, in the **Custom** Tab.



The custom Tab also contains an overview of which controller object is currently controlling which limb.

You can even replace them with objects of your own here, by pasting another object name into the field!

Deleting a controller object will reset that limb's position to default. This can be useful to quickly reset a pose and start anew. To re-create any deleted controls, simply click "Create Missing".



Torso Ctrl	Full Torso - Pose Control
Spine A Ctrl	Hip - Pose Control
Spine B Ctrl	Lower Belly - Pose Control
Spine C Ctrl	Belly - Pose Control

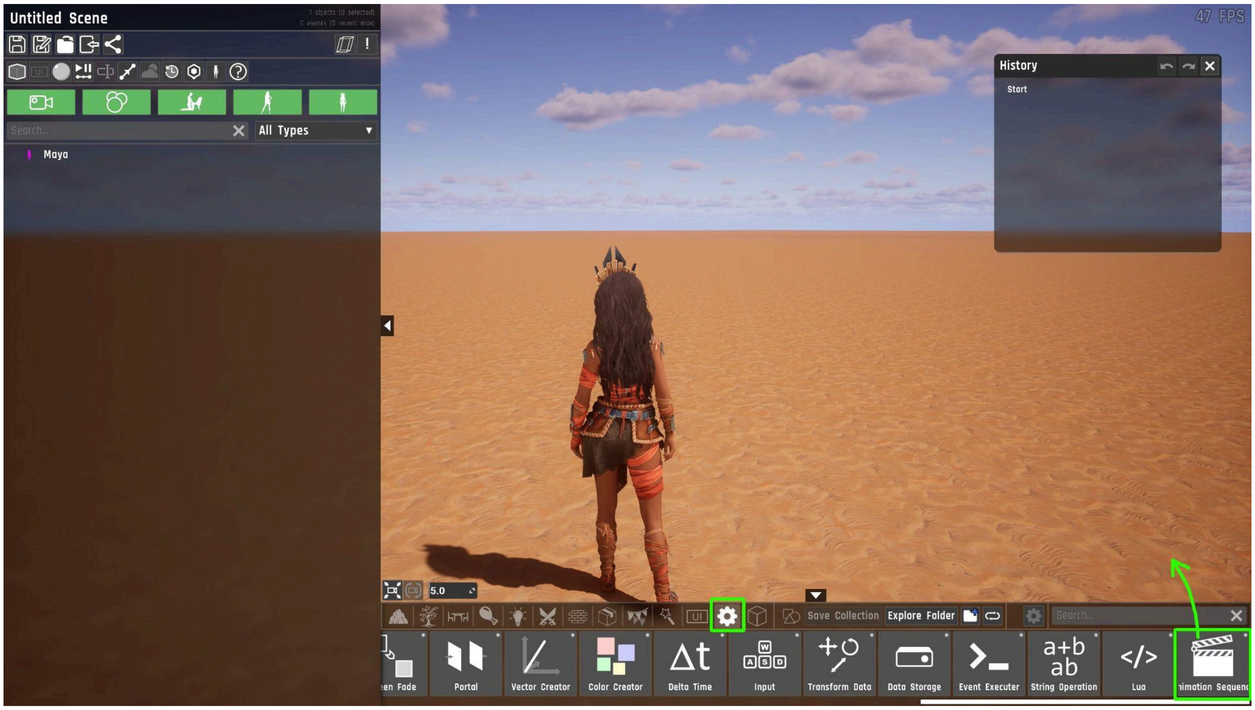
The Animation Sequencer

Making your own animations

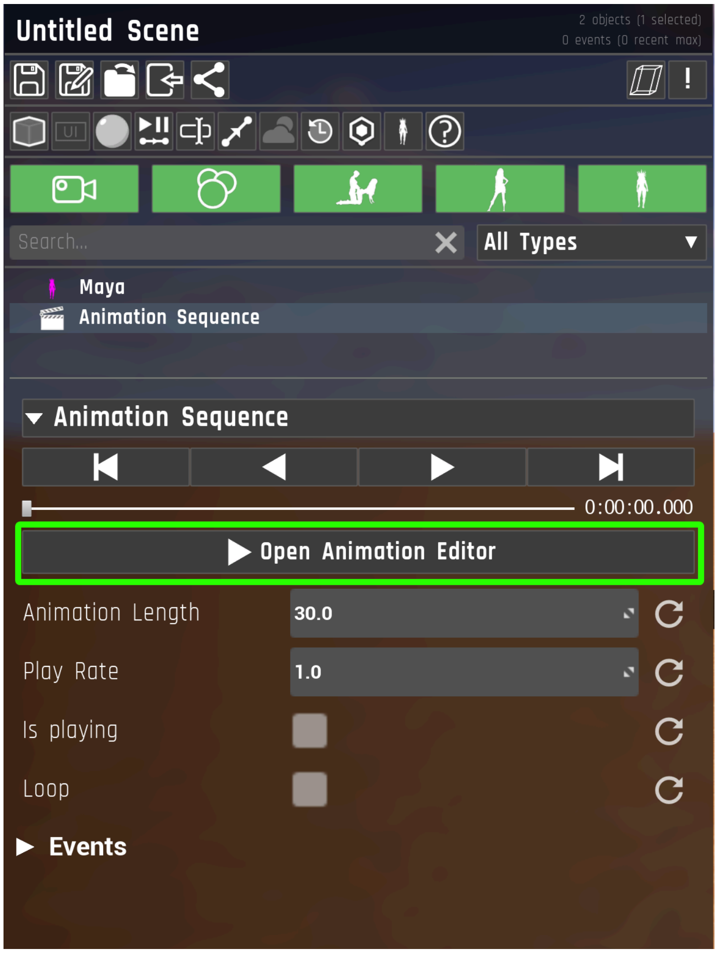
(Available since April 2024 version)

The animation sequence prop allows you to animate just about anything in your scene. Not only position or rotation - whatever you can move or change in the object properties on the left can be animated!

To get started, find the **Animation Sequence** prop in the Automation section at the bottom of the level editor and place it in your scene.


Untitled Scene
1 objects (0 selected)
0 events (0 recent max)
Search...
All Types
Maya
47 FPS
History
Start
5.0
Save Collection
Explore Folder
Search...
een Fade
Portal
Vector Creator
Color Creator
Delta Time
Input
Transform Data
Data Storage
Event Executer
String Operation
Lua
nimation Sequenc

After clicking on it in your hierarchy view, you can see a button for **Open Animation Editor.** Click it to open the animation timeline.

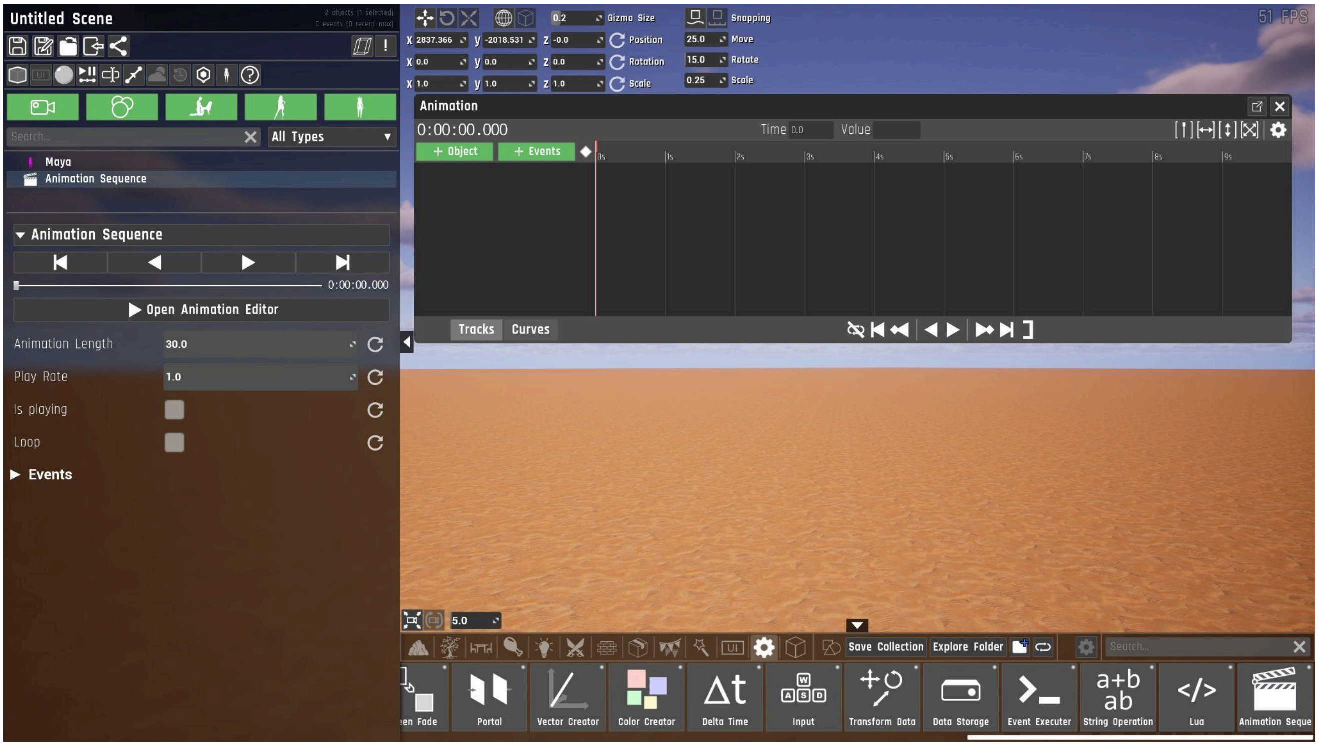


Animation Length 30.0

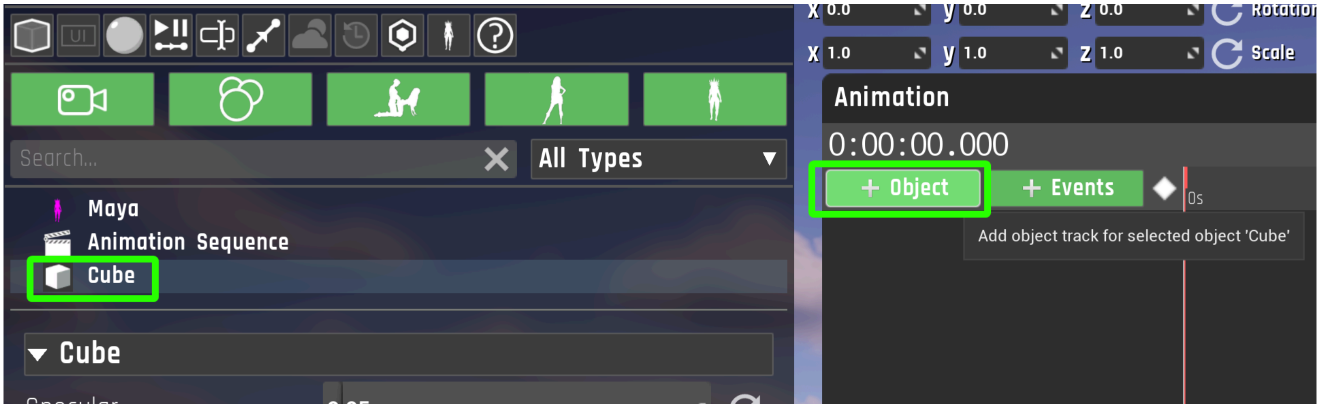
Play Rate 1.0

Is playing

Loop

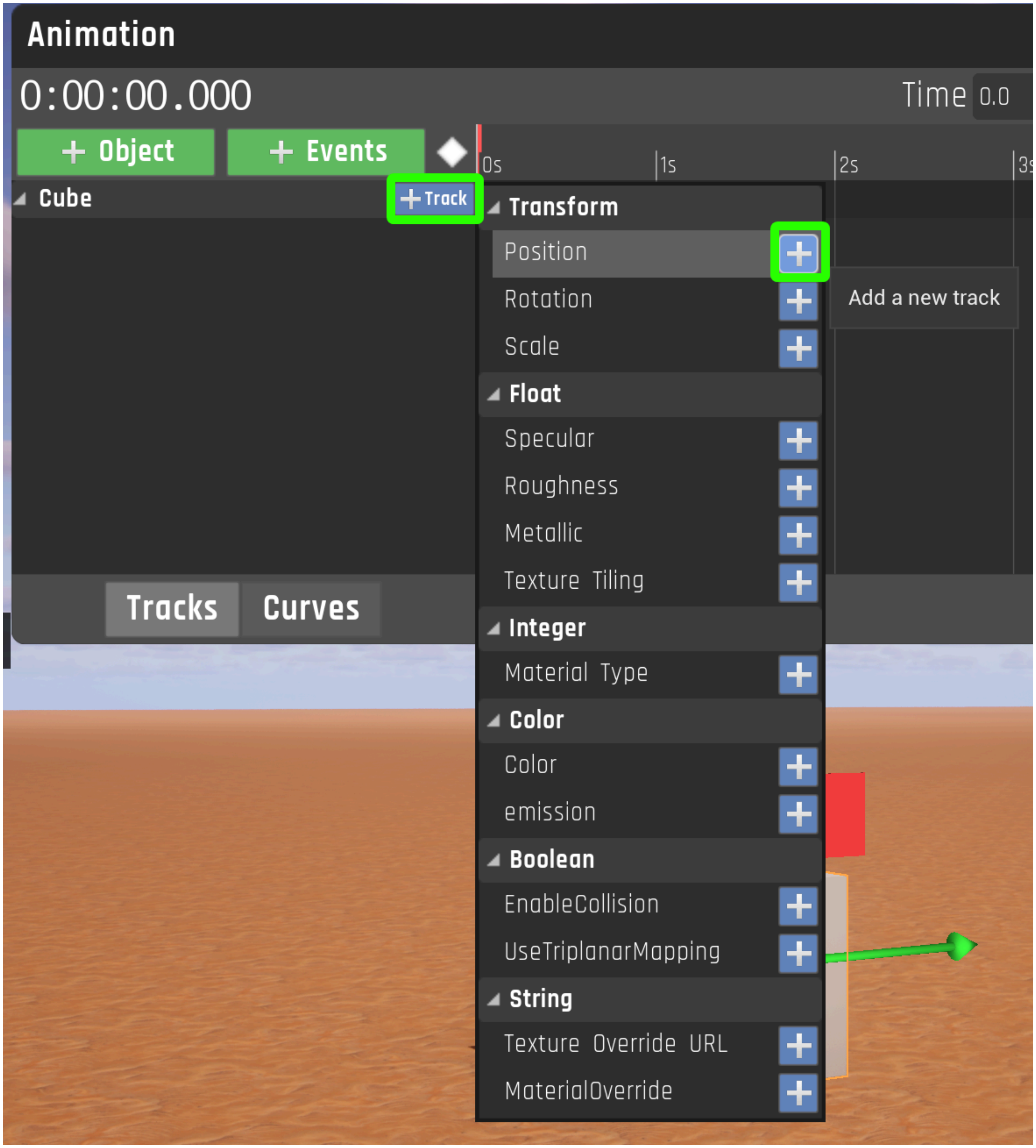


We now have a timeline, but it is empty. Next, we need something to animate. Select any object you like, then click the **+Object** button to add it to the timeline. I am adding a cube here for demonstration:



Add object track for selected object 'Cube'

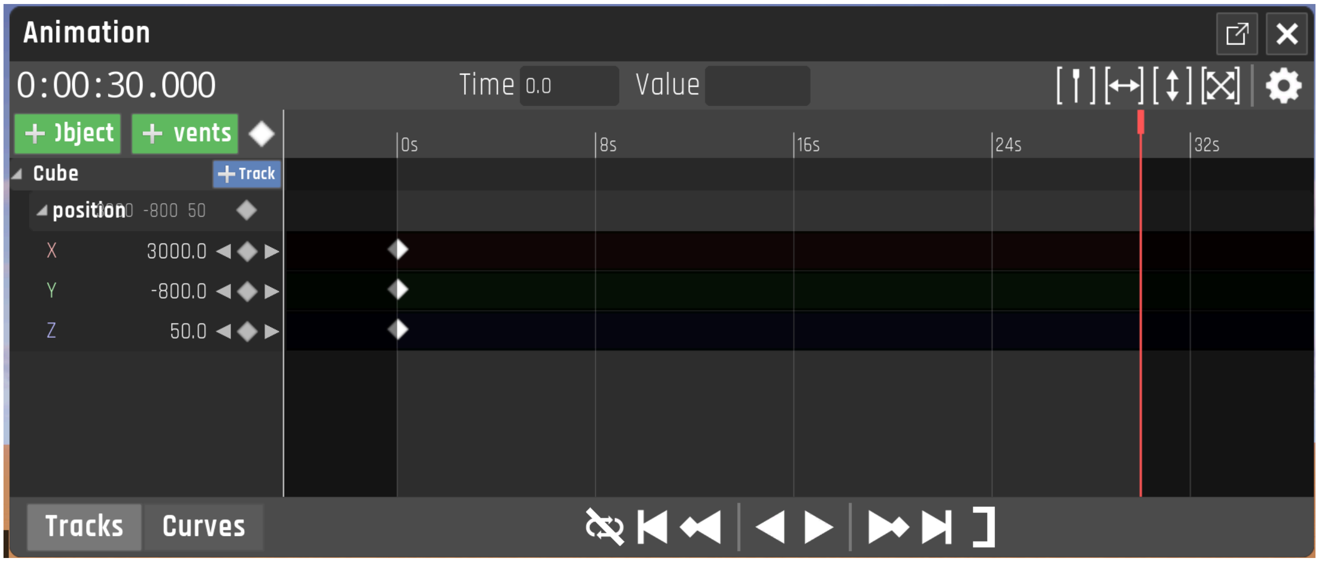
Next, we add **tracks** for the attributes we want to animate on our cube, by clicking the **+Track** button. As you can see, any attribute is possible. Let's start with something simple and animate the **position.**



We now see three tracks for X,Y and Z positions. The Editor has also already added the first **key,** at a time of 0. A key or keyframe is a point in time, where we define a value.

Once we have at least two keys, the sequencer will automatically interpolate between them, thus creating a fluid motion.

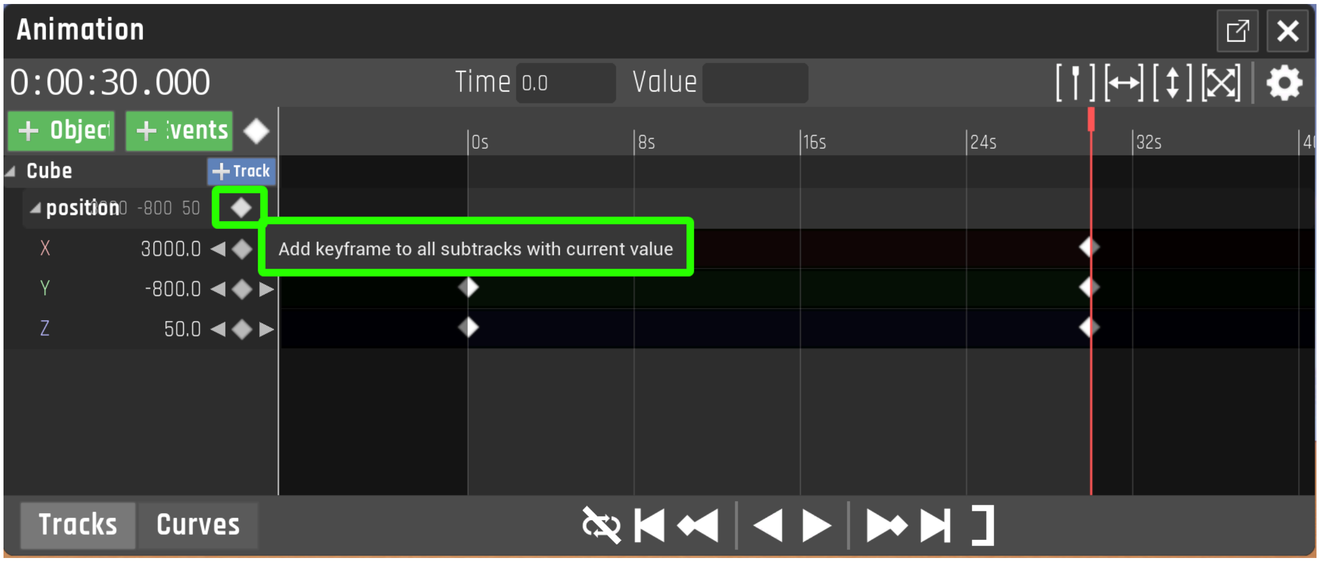
Tip: You can move along the timeline by pressing and dragging RMB and zoom in/out by using Ctl + Scroll wheel.



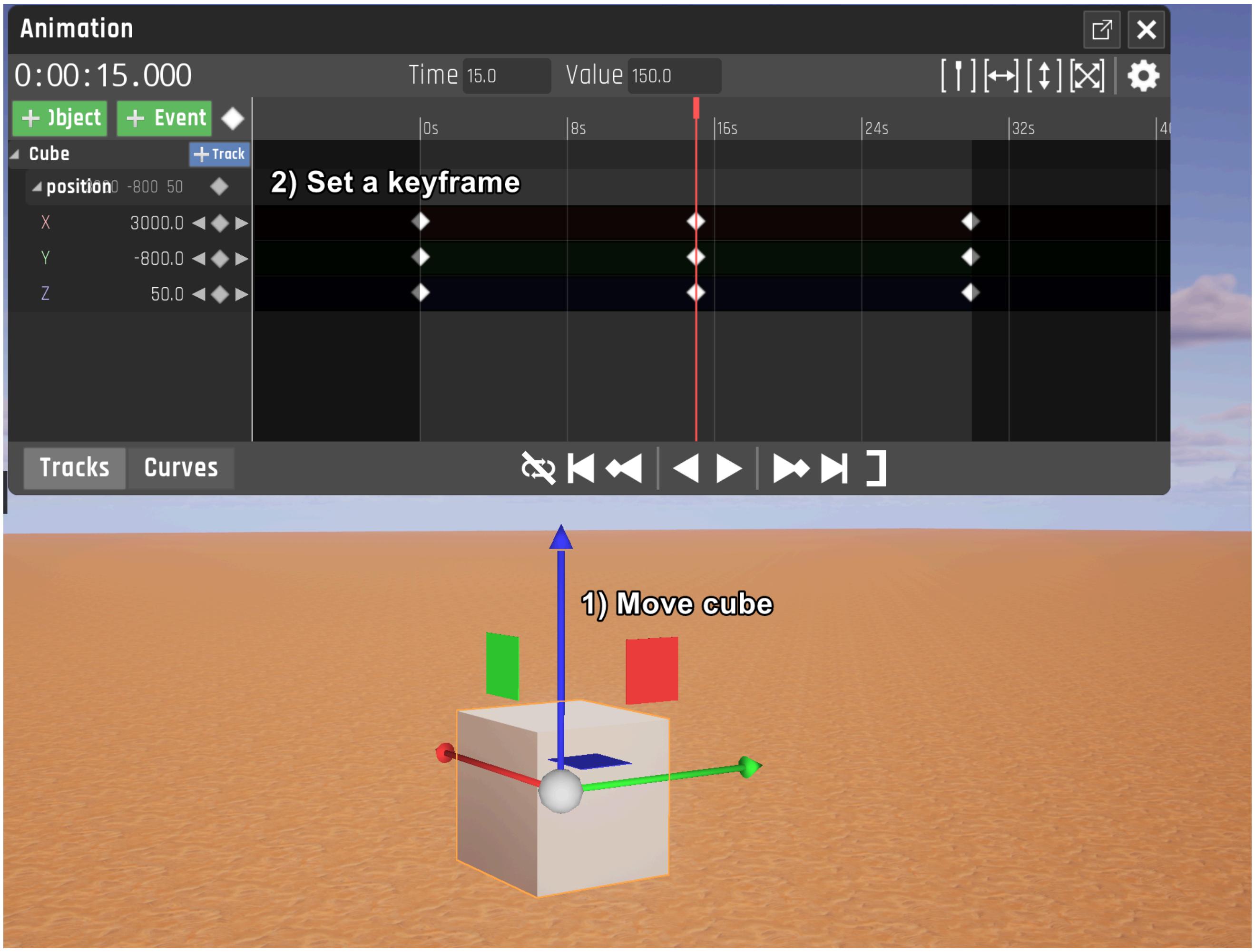
Time to set some more keyframes.
We'll want a smooth loop, so first we drag the red timeline marker to the end of our timeline and set a key there. That way our cube will start and end in the same position.

Tip: Click the cogwheel in the top right corner and set the snapping sample rate to something lower, to more easily snap to the timesteps you need. (Sample rate of 1 = snap to full seconds, 2 = snap to 0.5 seconds, 10 = snap to 0.1 seconds, etc.)

We click the keyframe button next to our tracks to set a new keyframe:



Next, we move the timeline marker to the middle. But this time, before(!) we set our keyframe, we move the cube up a little in the viewport.

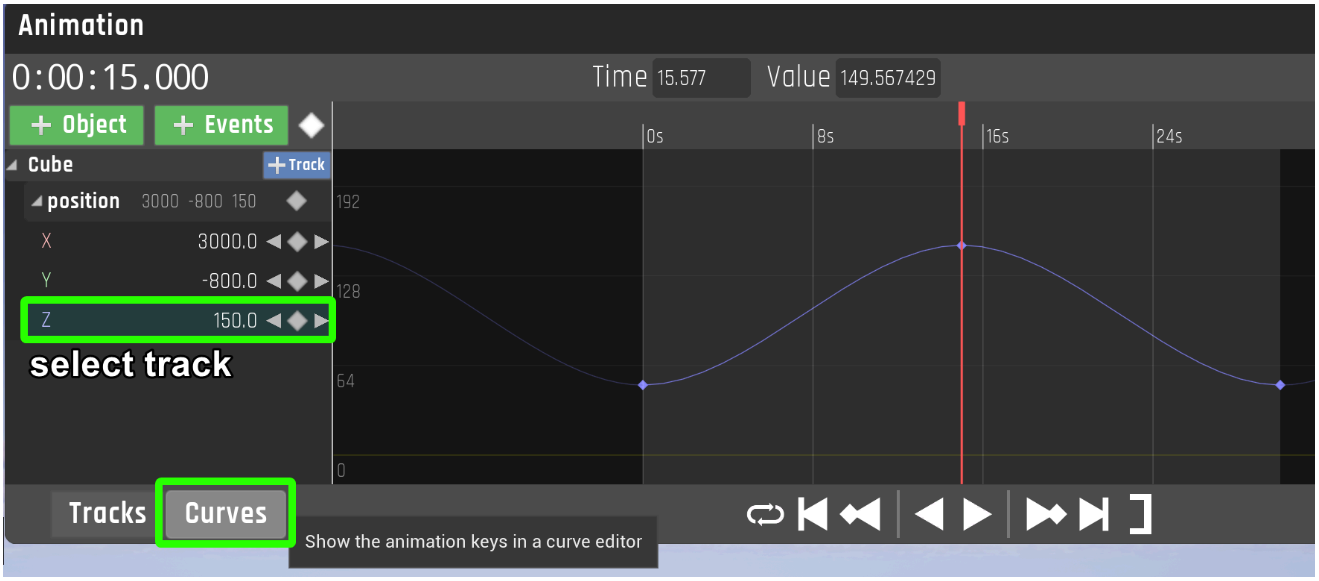


2) Set a keyframe

1) Move cube

And that's it! Now when we drag the red timeline marker, we can see the cube move up and down smoothly.

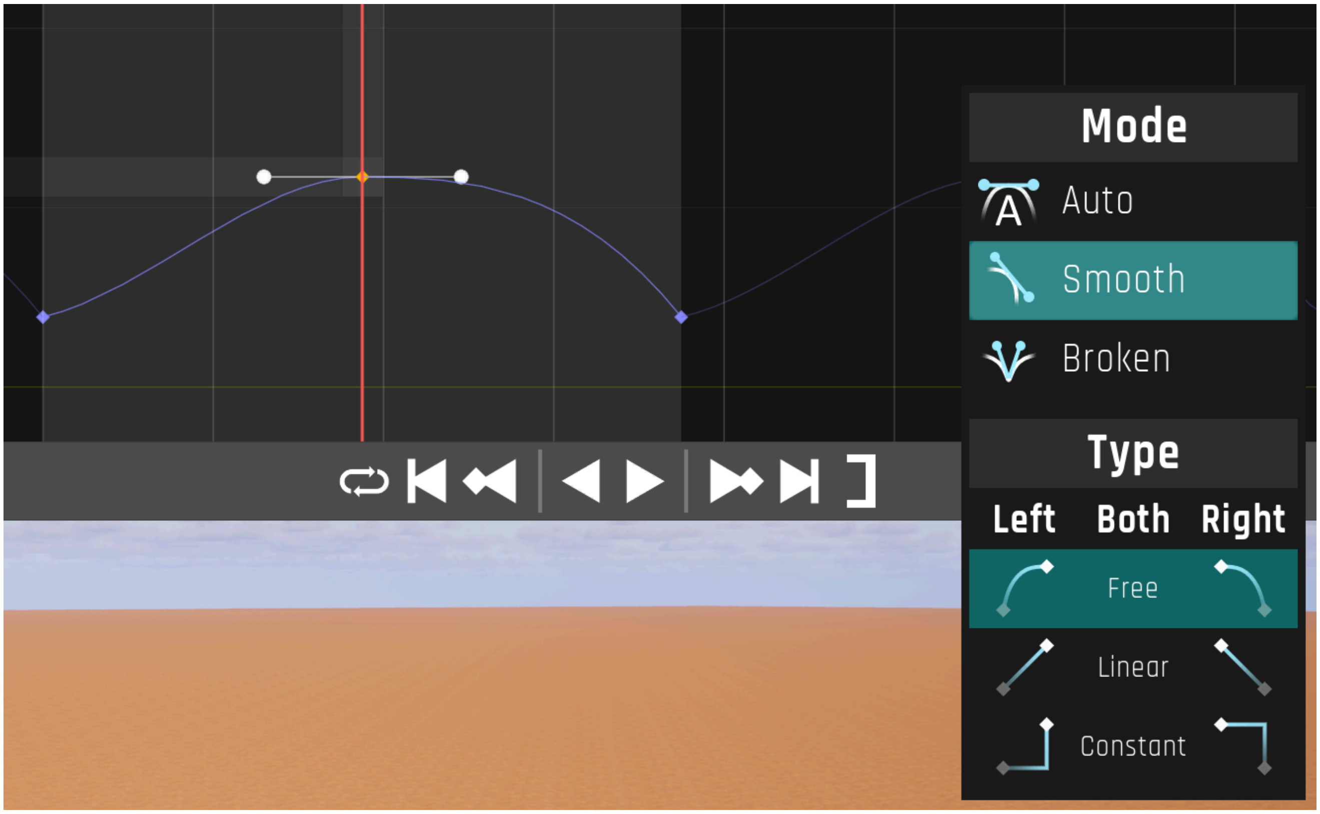
Advanced: In order to change the way the keyframes are interpolated, we can also have a look at the **Curve View.** Right now, our Z position is a smooth curve: the cube starts rising slowly, gets faster, then slows down again towards the top.



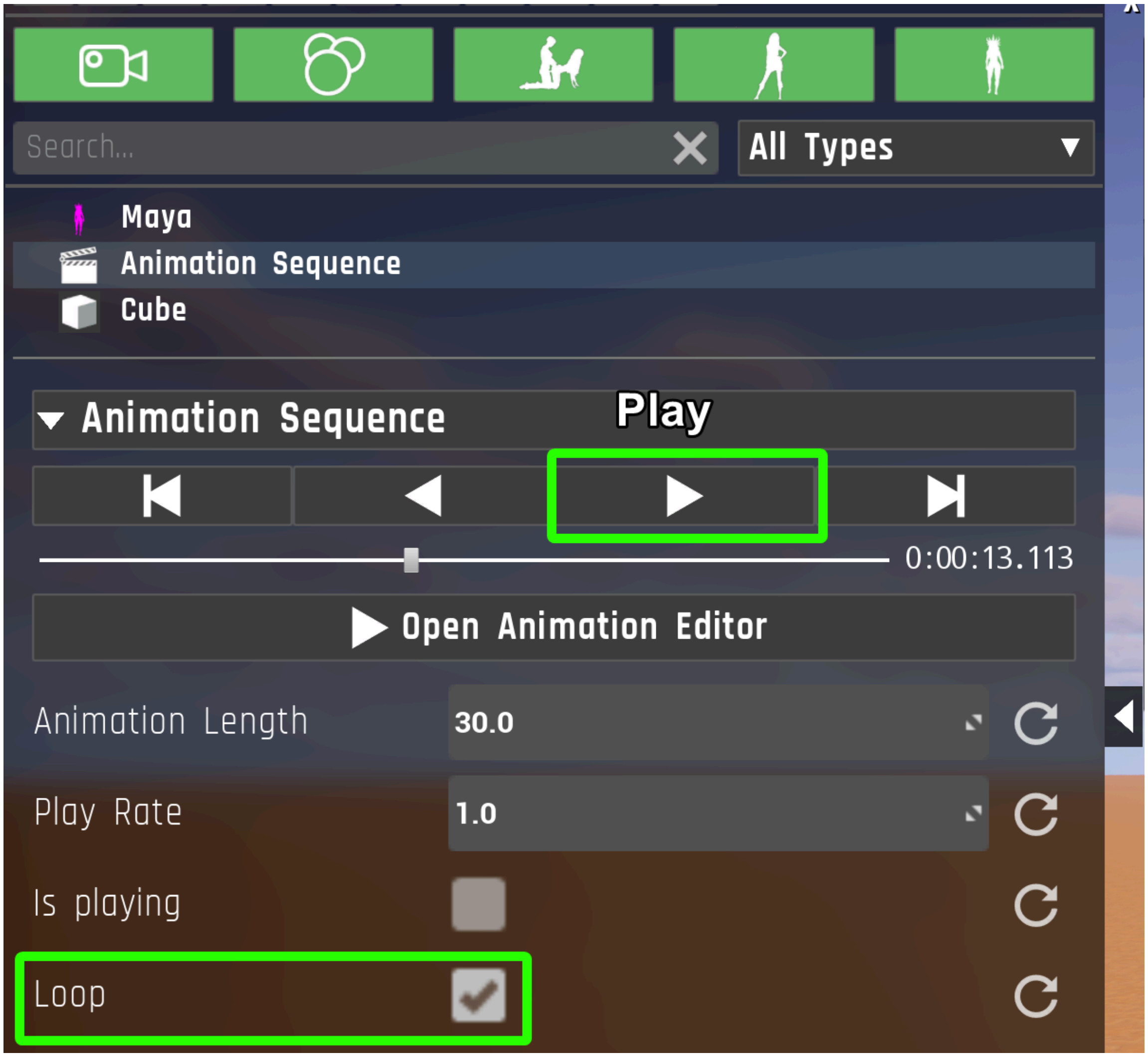
select track

Show the animation keys in a curve editor

By selecting keys on the curve and pressing RMB, we can get different interpolation options. We can also grab the tangents and edit them ourselves.



Lastly, we can preview the animation by pressing play on the sequence prop. Checking the “Loop” option ensures it loops when reaching the end.



Animation Length 30.0

Play Rate 1.0

Is playing ☐

Loop ☑

And that's it for an introduction to animation! You can also animate characters the same way, by combining the Custom Poser with an animation sequence. Or animate anything else in your scene - cameras or lights, for example.

Bone Controllers

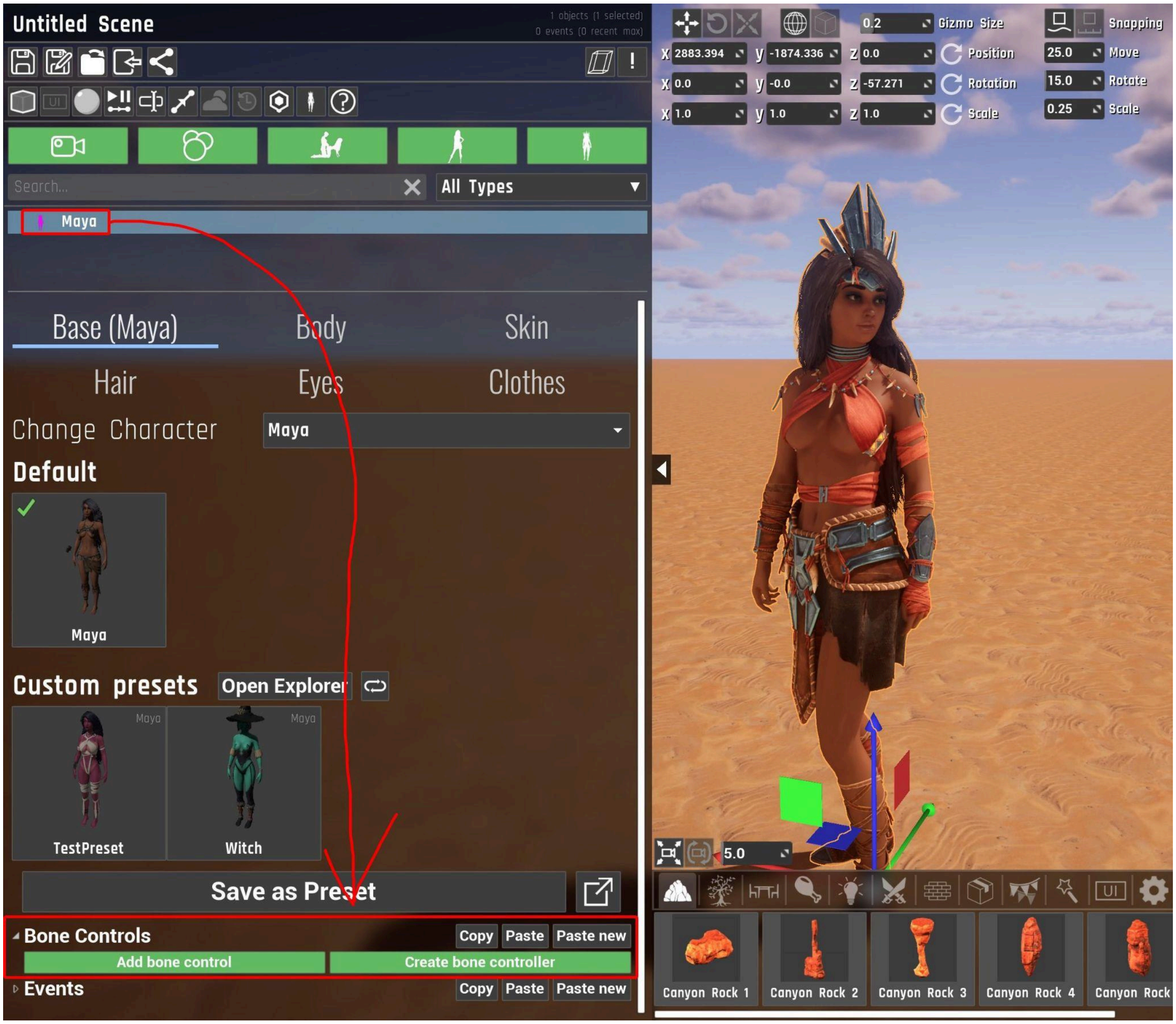
Make adjustments to built-in animations

(Available since June 2024 version)

Ok, but what if you don't want to make your own animation from scratch, but merely modify one that already exists in the game? To do this, you can add Bone Controllers to any character.

First, select a character - it can be any, whether the one you control, an NPC or one that is already being controlled by a pose or sex scene.

At the bottom, you find the Bone Controller Tab.



Untitled Scene

1 objects (1 selected)
0 events (0 recent max)

Search...

All Types

Maya

Change Character: Maya

Gizmo Size: 0.2

Position: X 2883.394, Y -1874.336, Z 0.0

Rotation: X 0.0, Y -0.0, Z -57.271

Scale: X 1.0, Y 1.0, Z 1.0

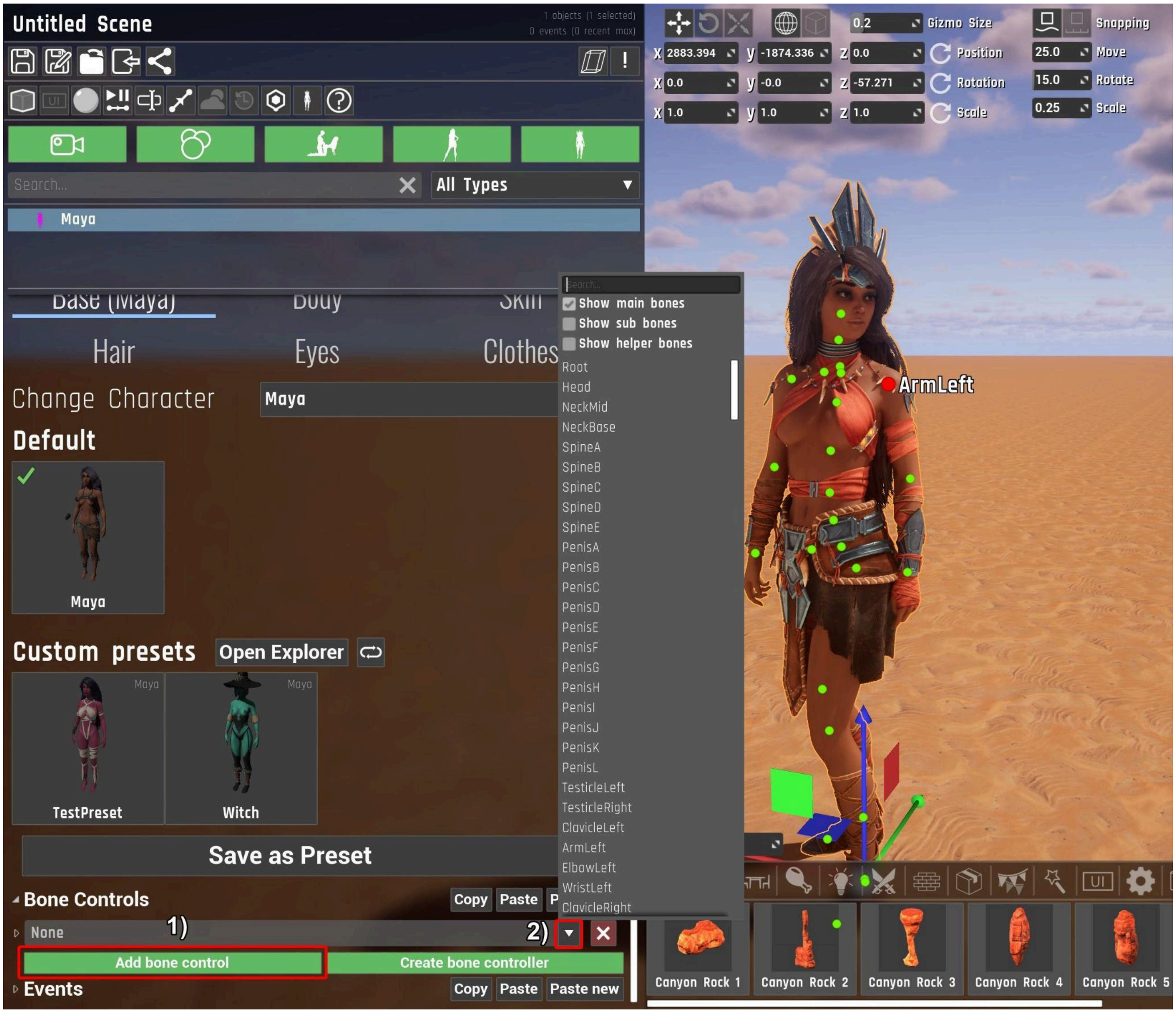
Snapping: Move 25.0, Rotate 15.0, Scale 0.25

Basic Offset:

To add a simple offset to whatever animation is already playing, click the LEFT button. This creates an “empty” bone control, meaning it does not have an object controlling it but is just a plain offset.

Then, click the dropdown and select the bone you want to control. Here I have selected ArmLeft - you can select these in the list or directly in the viewport.

Note: There is still a bug right now, where clicking on a bone in the viewport sometimes selects an invisible bone nearby. So double check that you actually got the right one!



Untitled Scene

1 objects (1 selected)
0 events (0 recent max)

Gizmo Size: 0.2

Position: X 2883.394, Y -1874.336, Z 0.0

Rotation: X 0.0, Y -0.0, Z -57.271

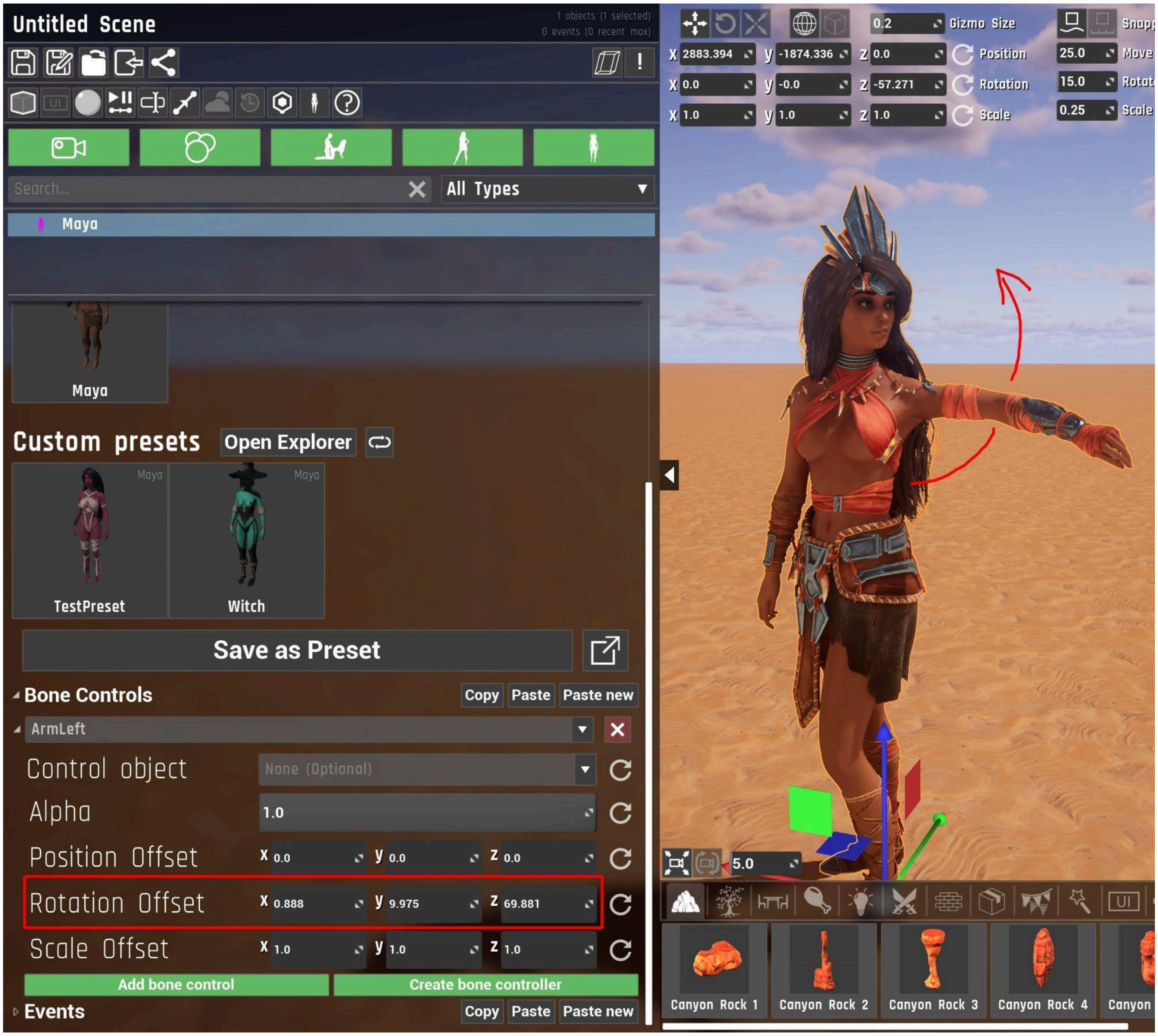
Scale: X 1.0, Y 1.0, Z 1.0

Snapping: Move 25.0, Rotate 15.0, Scale 0.25

Change Character: Maya

☑ Show main bones
□ Show sub bones
□ Show helper bones

Now, all you need to do is set the offset you want - here I have rotated the arm up.



ArmLeft

Control object: None (Optional)

Alpha: 1.0

Position Offset: X 0.0, Y 0.0, Z 0.0

Rotation Offset: X 0.888, Y 9.975, Z 69.881

Scale Offset: X 1.0, Y 1.0, Z 1.0

Gizmo Size: 0.2

Position: X 2883.394, Y -1874.336, Z 0.0

Rotation: X 0.0, Y -0.0, Z -57.271

Scale: X 1.0, Y 1.0, Z 1.0

Move: 25.0

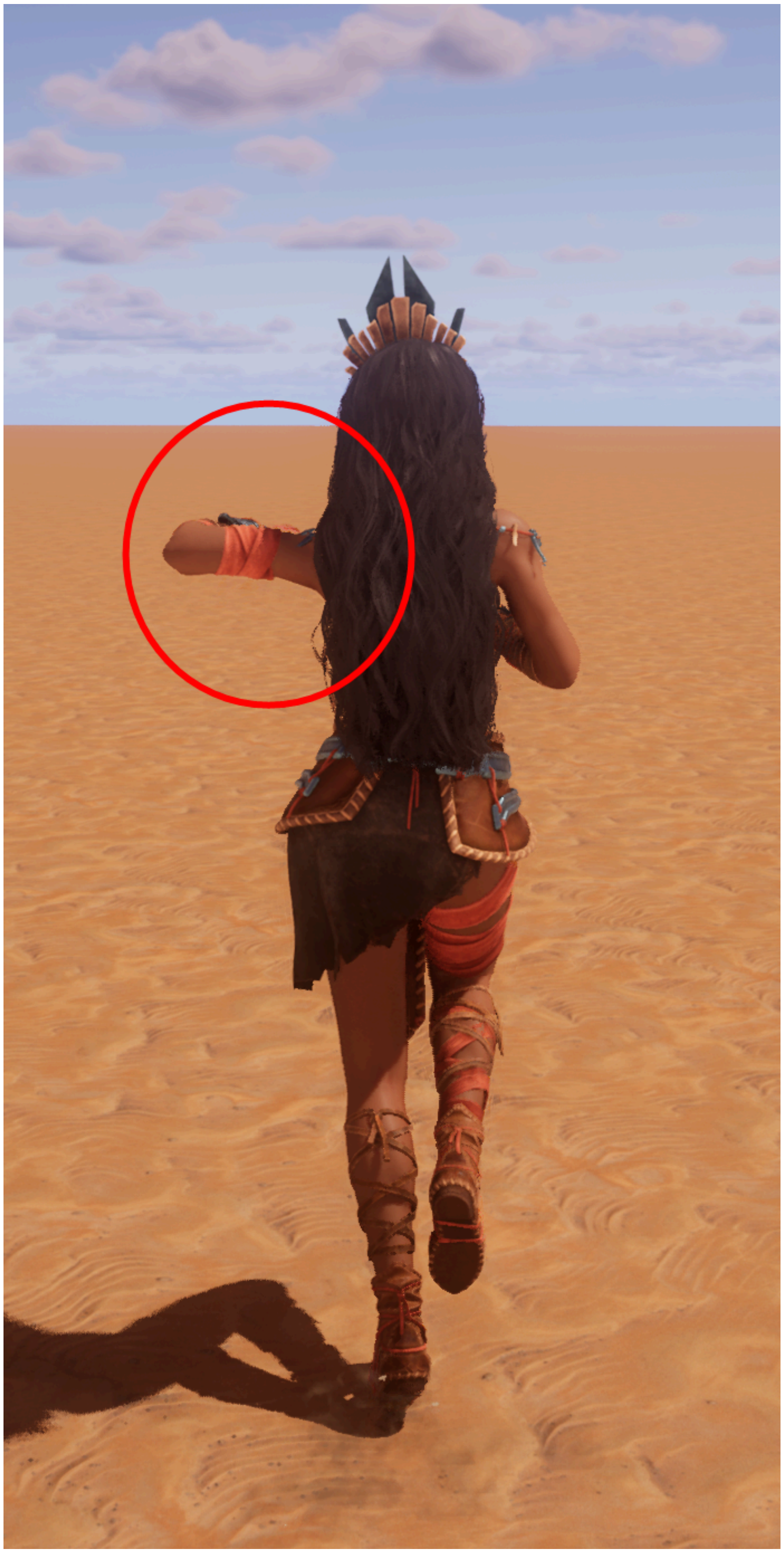
Rotat: 15.0

Scale: 0.25

5.0

This offset will now be added to whatever the character does. Whether it be running, sexing, posing - that arm will always be rotated up.

To delete it again, just click the red X next to the bone dropdown!

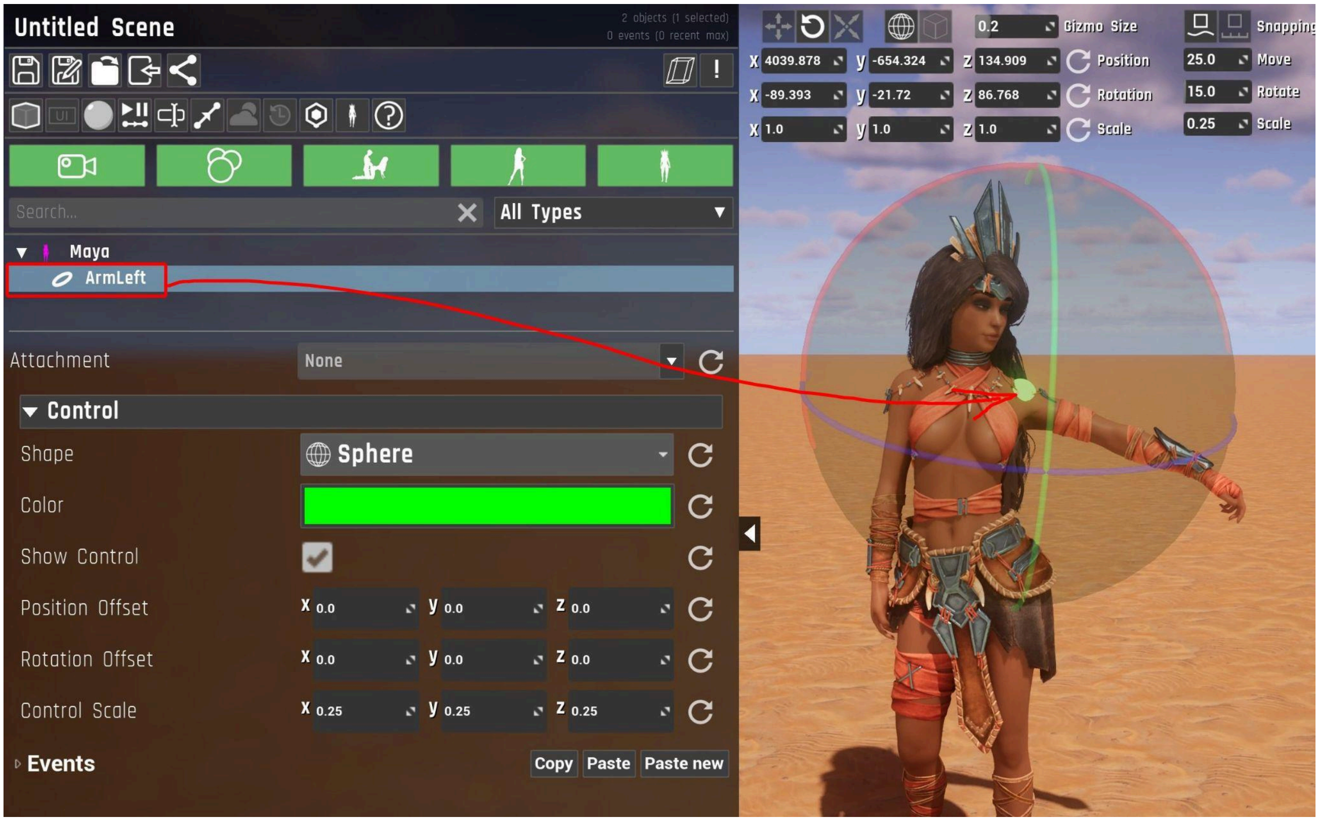


Advanced Control:

Now that is already useful, but it’s a bit basic. Next, lets create an actual object that controls the bone, by clicking the RIGHT button instead.

Note: Make sure you pause any ongoing animations before setting up your bone controllers! This will make things much easier.

Everything is the same as before, but now we see an object has been created under our character. This is our Bone Controller.
The difference is, instead of just having an offset, this object now fully controls the bone - so the animation on it will no longer play.



Untitled Scene

2 objects (1 selected)
0 events (0 recent max)

Search...

All Types

▼ Maya
ArmLeft

Attachment: None

▼ Control

Shape: Sphere

Color: 🟩

Show Control: ☑

Position Offset: X 0.0 Y 0.0 Z 0.0

Rotation Offset: X 0.0 Y 0.0 Z 0.0

Control Scale: X 0.25 Y 0.25 Z 0.25

Events

Gizmo Size: 0.2

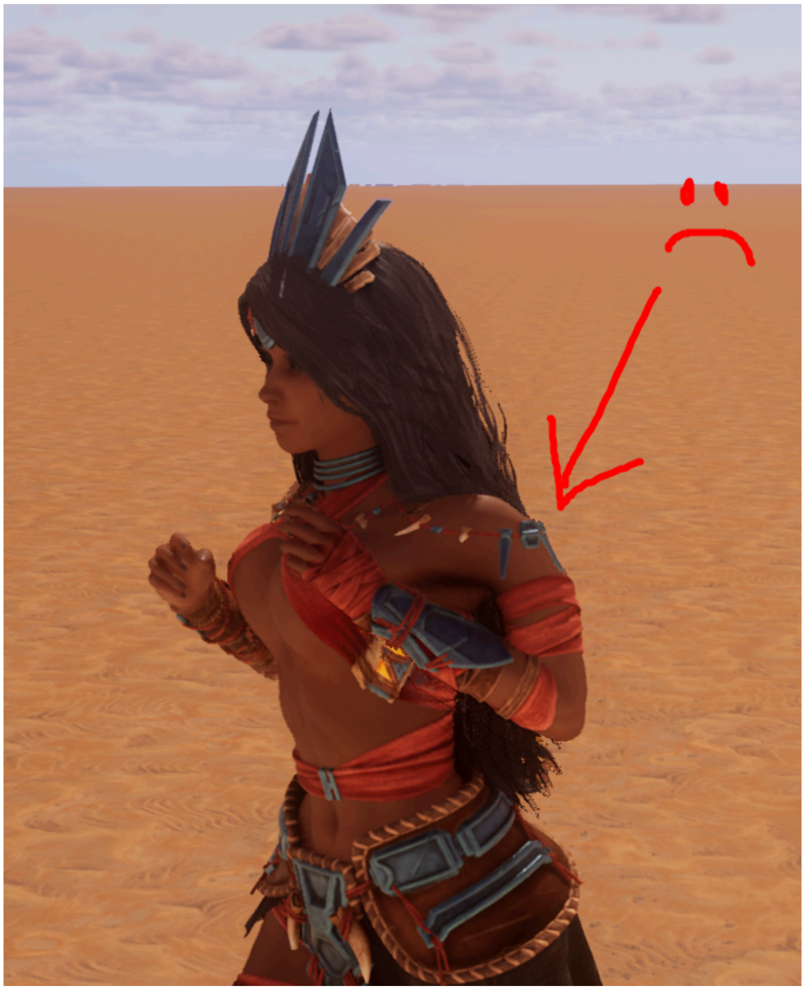
Position: X 4039.878 Y -654.324 Z 134.909

Rotation: X -89.393 Y -21.72 Z 86.768

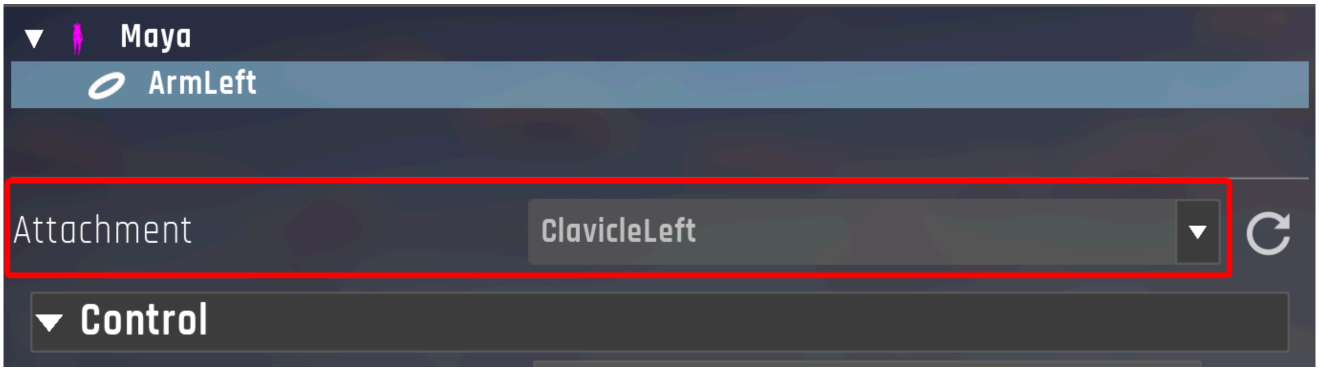
Scale: X 1.0 Y 1.0 Z 1.0

Snapping
Move: 25.0
Rotate: 15.0
Scale: 0.25

If we went into play mode now to run around though, we would notice that the arm now seems oddly “stuck in place” - this is because the controlling object follows the character, but not the correct parent bone yet.

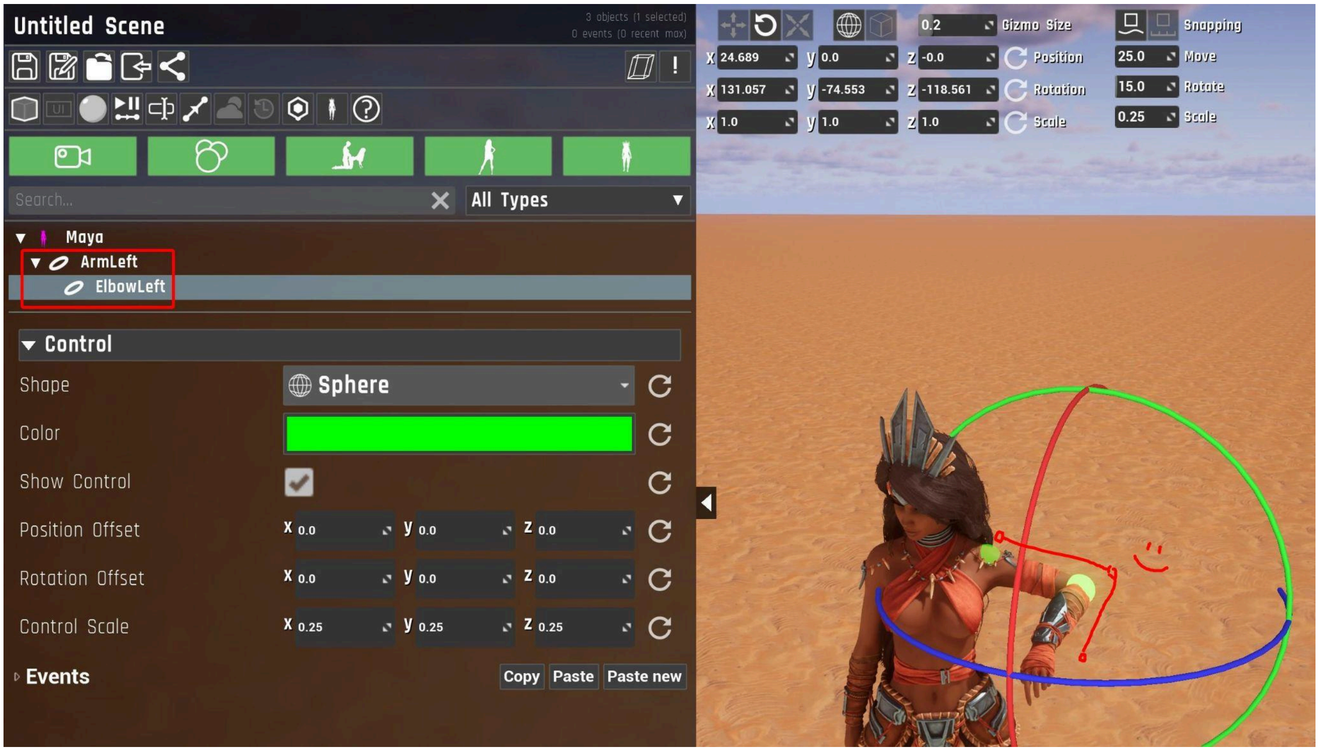


To prevent this, make sure that the attachment of the object is correct - in this case, the clavicle would be the right parent to use.



Attachment ClavicleLeft

When using multiple objects to control multiple bones, we can also parent them under each other, to form logical hierarchies - here I have made an additional control for the elbow and parented it under the arm control. Now I can bend the arm naturally, using both controls.



Control

Shape	Sphere		
Color	🟩		
Show Control	☑		
Position Offset	X 0.0	Y 0.0	Z 0.0
Rotation Offset	X 0.0	Y 0.0	Z 0.0
Control Scale	X 0.25	Y 0.25	Z 0.25

Position	X 24.689	Y 0.0	Z -0.0
Rotation	X 131.057	Y -74.553	Z -118.561
Scale	X 1.0	Y 1.0	Z 1.0

Gizmo Size 0.2

Snapping: Move 25.0, Rotate 15.0, Scale 0.25

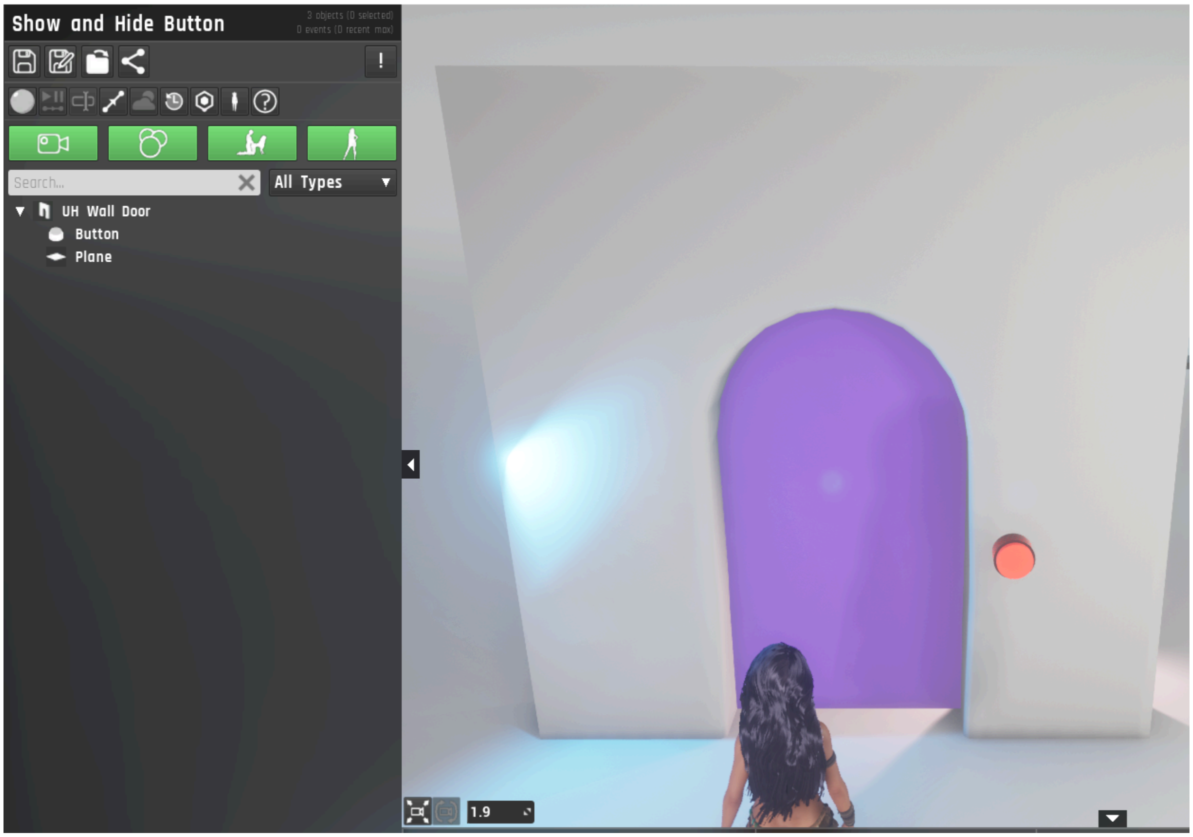
…and that's it for Bone Controls! They are not as powerful as the full custom poser, but they allow you to quickly edit existing animations - and sometimes that's all you need to make something cool.

The Event System

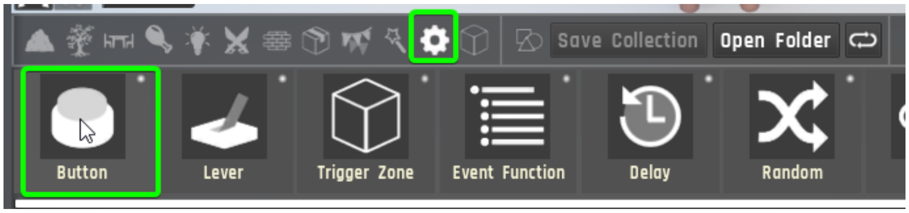
Making an Interactable Door

The event system lets objects in your scene react to each other and the player. It can get very complex, so before we get to our door, let's start with something simpler: A barrier you can make disappear by pressing a button.

–

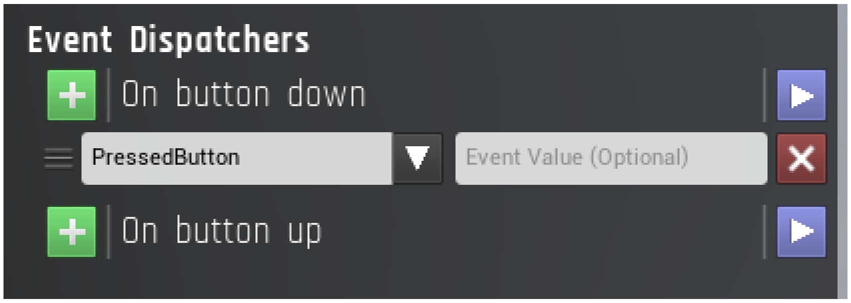


Here we have created a doorway with a plane in it that blocks the player from going through. In the automation tab, we find the button, which we place anywhere near the plane.



We can already press this button now, but nothing happens. In order for objects to react to it, it needs to **dispatch an event**.

The basic premise of an event is just a message being sent from any prop (**Dispatcher**), and being received by any other prop (**Receiver**) to execute a desired function.



Event Dispatchers

On button down

PressedButton

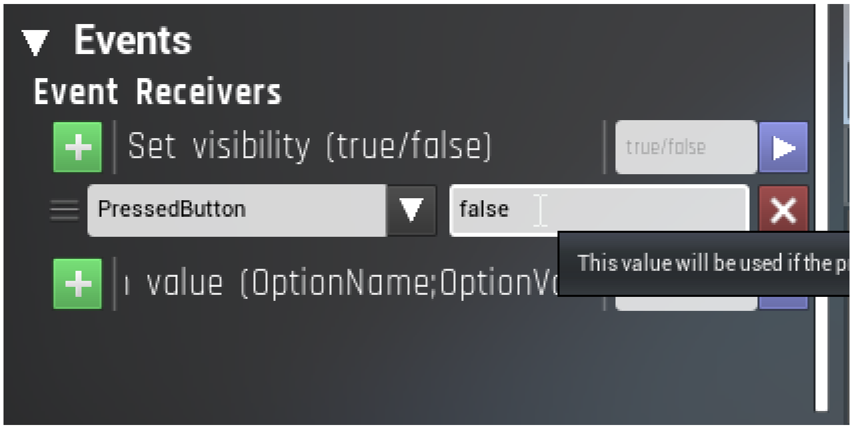
Event Value (Optional)

On button up

We click on the + under the “On Button Down” Event Dispatcher to add our event to that dispatcher. We can call it anything we like, here it’s simply “PressedButton”.

We can also choose to send a value along with the event, but lets ignore that for now. All we need to know is that the button was pressed.

Now we need to make the Plane listen to this event. To do so, we add an event receiver under “Set Visibility”:



PressedButton

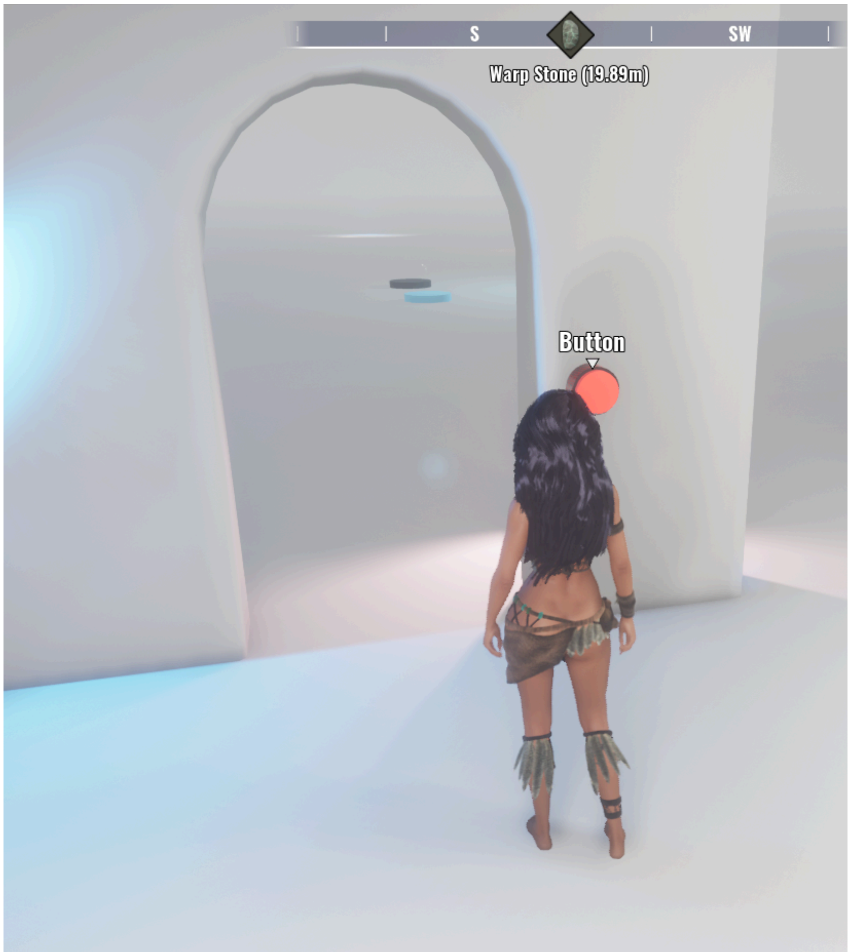
false

This value will be used if the p

We can type in the name directly, or press the arrow next to the event field to select our PressedButton event.

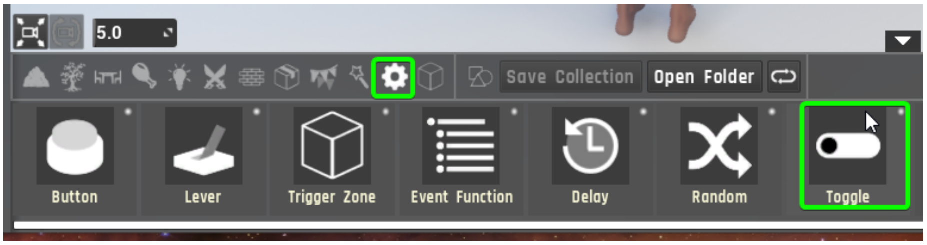
We also add “false” into the default value field. This means that when the plane receives this event, it will set its visibility to false.
*(Advanced: Alternatively, you can also send over a value from the event **dispatcher** by setting the value in its “Event value” text box (see previous image). As the tooltip suggests in the default value field of the event **receiver**, its value is only used when the event **dispatcher** did not provide a value.)*

Let’s test our setup!



If everything was set up correctly, pressing the button now makes the plane disappear. However, it stays invisible afterwards, even if we press the button again.

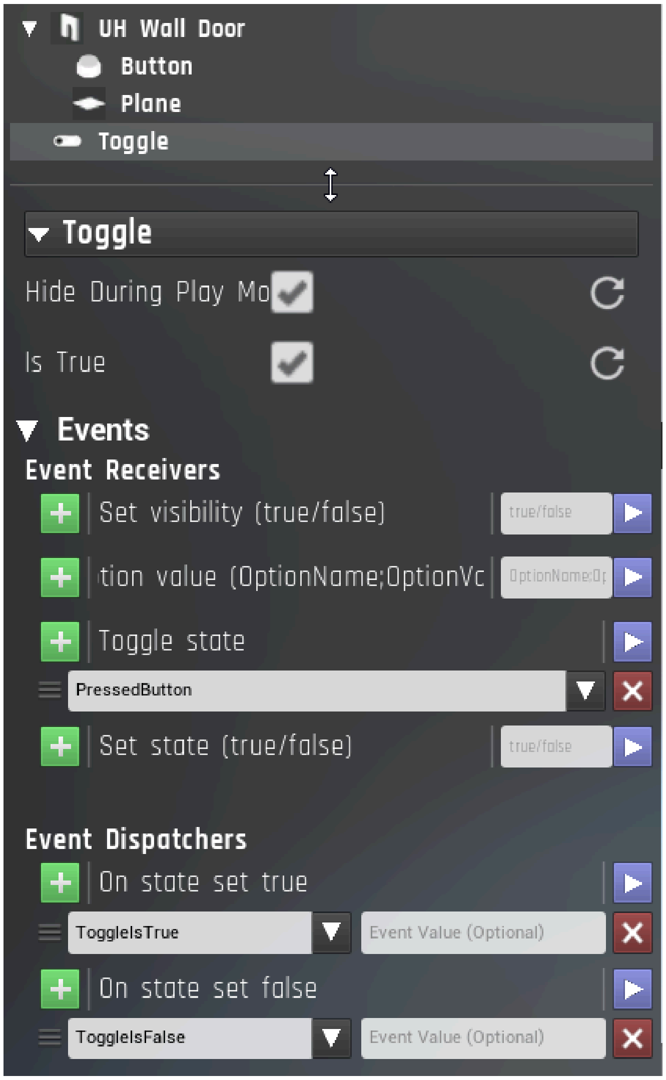
To make a proper door, we need to be able to **toggle** the open and closed state.
Time to find the Toggle prop under the automation tab and add it to our scene:



The function of the toggle prop is to switch between two states, “True” and “False”, perfect for our use case.

The first step now is to make the toggle prop listen to our button. Every time the button is pressed, it should toggle its state.

Therefore, just like with our plane before, we add the “PressedButton” event to the Toggle State event receiver.



UH Wall Door
Button
Plane
Toggle

Toggle

Hide During Play Mo ☑

Is True ☑

Events

Event Receivers

Set visibility (true/false) — true/false

tion value (OptionName;OptionVc — OptionName:Op

Toggle state

PressedButton

Set state (true/false) — true/false

Event Dispatchers

On state set true

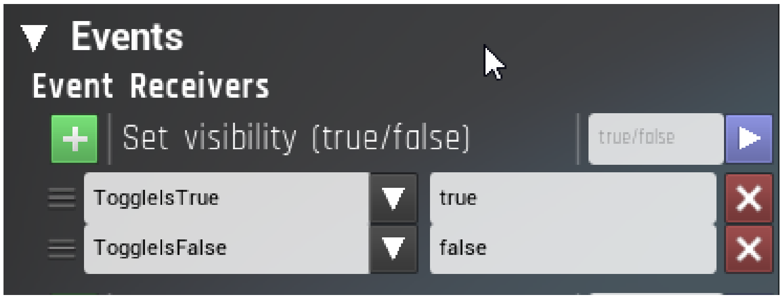
ToggleIsTrue — Event Value (Optional)

On state set false

ToggleIsFalse — Event Value (Optional)

We also need a way for other objects to know if the toggle is currently on or off. So, we make it dispatch two more events: “ToggleIsTrue” under OnStateSetTrue and “ToggleIsFalse” under OnStateSetFalse. (Again, we can name these anything we like!)

Lastly, let's update our plane to listen to these two new events. We delete the old PressedButton receiver and replace it with our two new ones. One to set the Visibility to true, one to set it to false.



Events

Event Receivers

Set visibility (true/false) — true/false

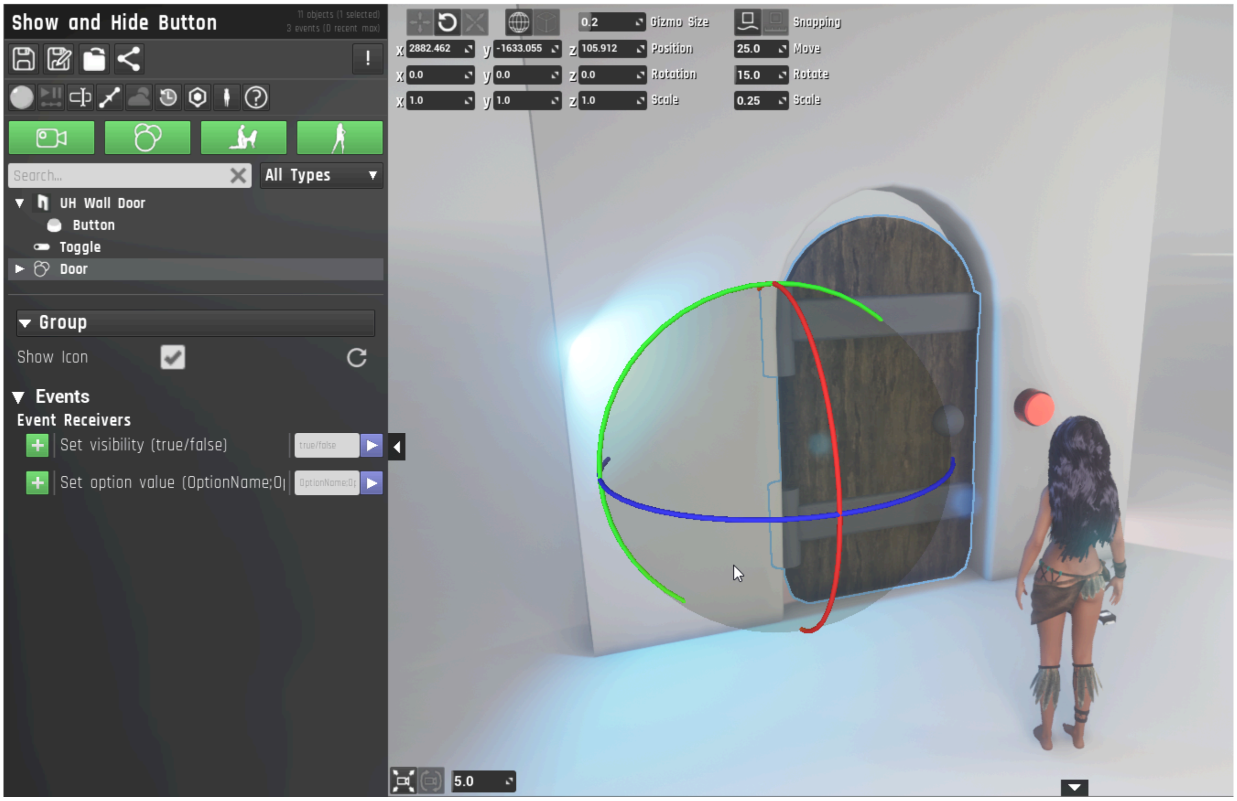
ToggleIsTrue — true

ToggleIsFalse — false

Going back to our scene, we should now be able to switch our “barrier” on and off with the button - that’s almost a working door!

If the button feels sluggish, you’ll need to set the button down duration on the button prop to a shorter value.

Now for the door proper. I have just assembled a simple one out of prototype shapes here, but you can surely come up with a better looking one! :)



Show and Hide Button

11 objects (1 selected)
3 events (0 recent max)

Search... All Types

- UH Wall Door
  - Button
  - Toggle
  - Door

Group

Show Icon ☑

Events

Event Receivers

- Set visibility (true/false) true/false
- Set option value (OptionName;O| OptionName;O

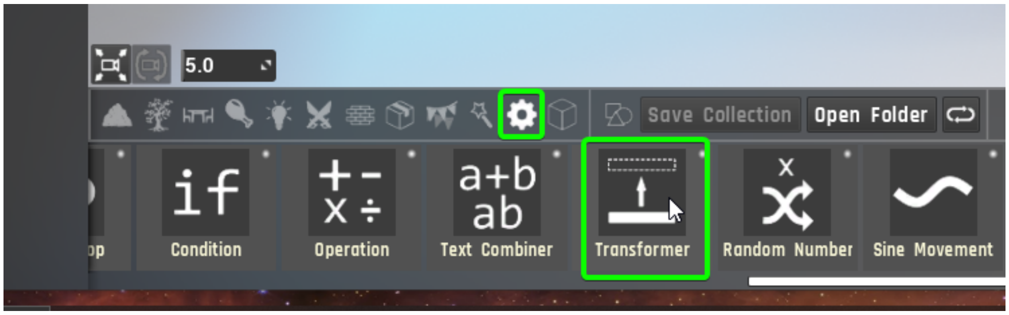
	x	y	z	
Position	2882.462	-1633.055	105.912	
Rotation	0.0	0.0	0.0	
Scale	1.0	1.0	1.0	

Gizmo Size 0.2

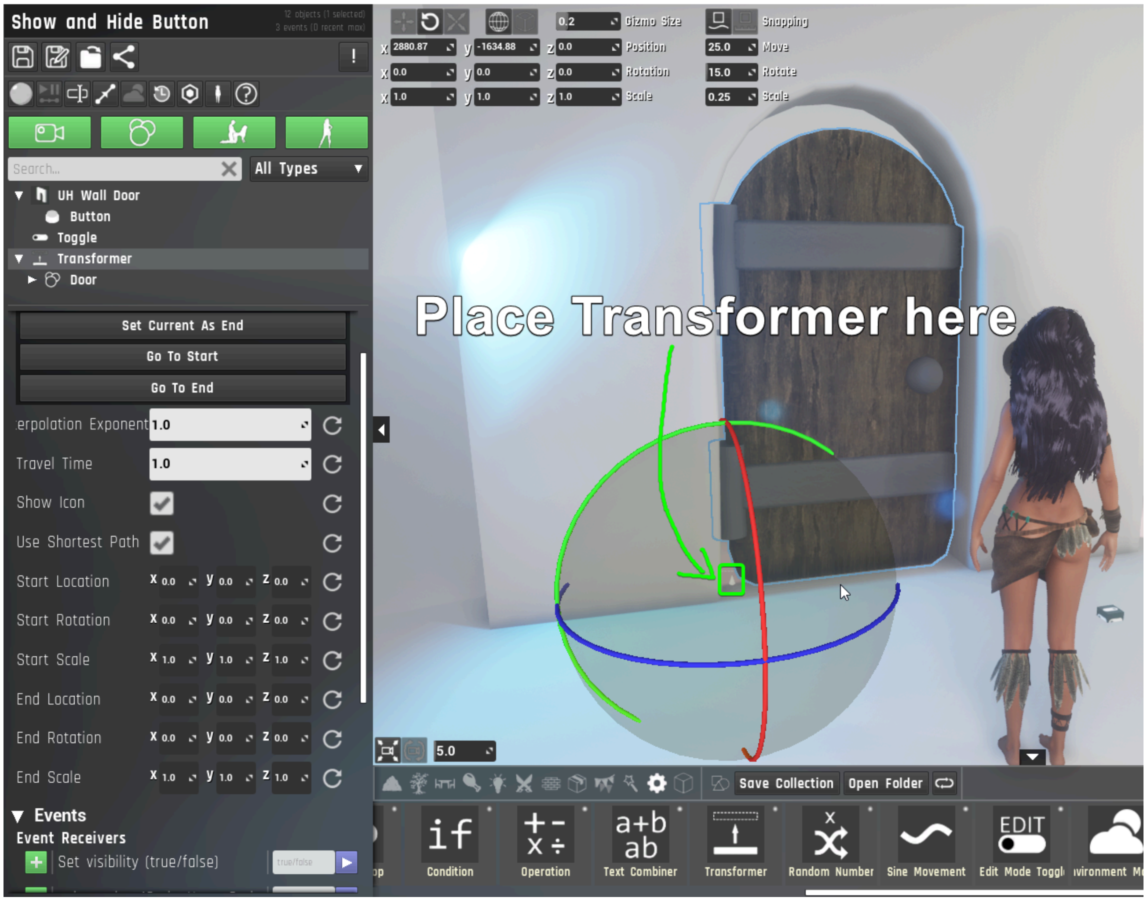
Snapping: Move 25.0, Rotate 15.0, Scale 0.25

5.0

In order to make our door swing open and closed, we'll need the **Transformer** prop. Again, we find it in the automation tab.



The Transformer is going to be the thing that rotates. So we need to place it where the door hinges are, then group our door under it, like so:

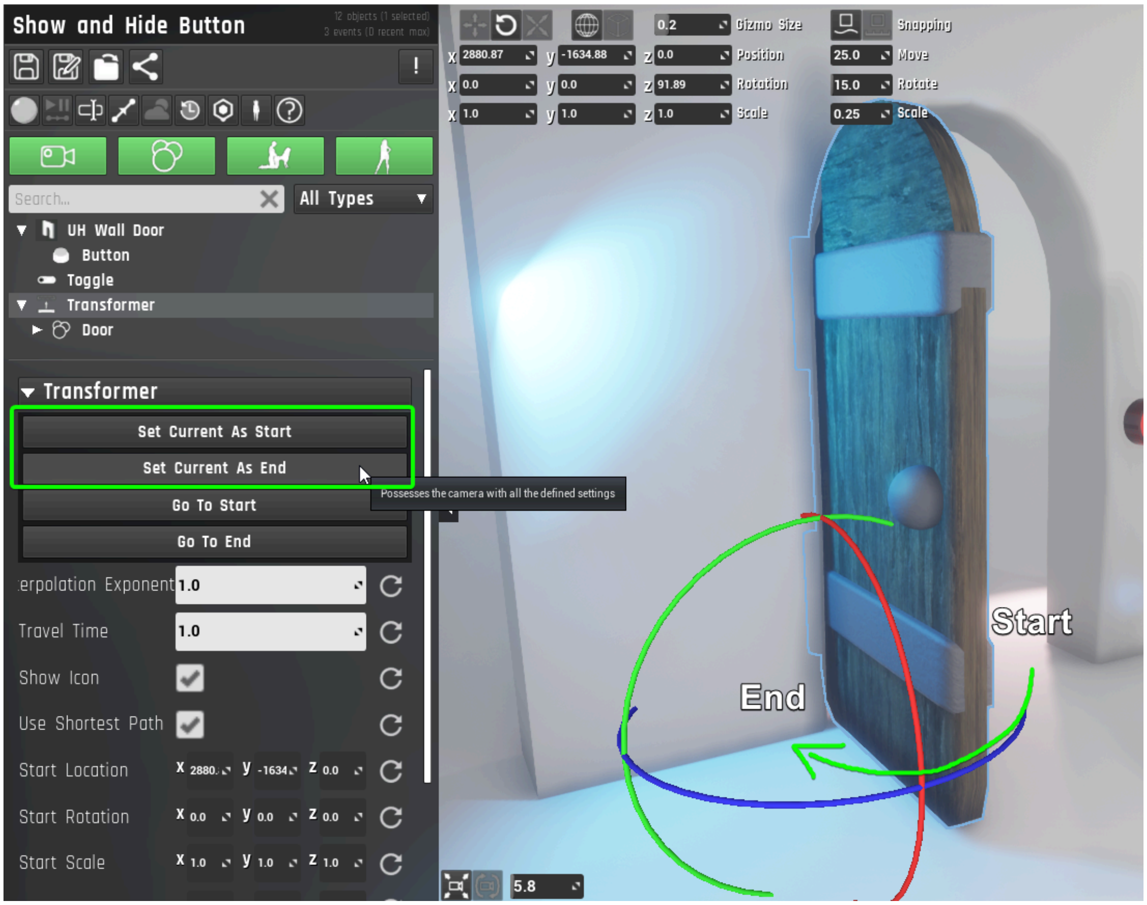


erpolation Exponent 1.0
Travel Time 1.0
Show Icon ☑
Use Shortest Path ☑
Start Location X 0.0 Y 0.0 Z 0.0
Start Rotation X 0.0 Y 0.0 Z 0.0
Start Scale X 1.0 Y 1.0 Z 1.0
End Location X 0.0 Y 0.0 Z 0.0
End Rotation X 0.0 Y 0.0 Z 0.0
End Scale X 1.0 Y 1.0 Z 1.0

Gizmo Size 0.2
Position x 2880.87 y -1634.88 z 0.0
Rotation x 0.0 y 0.0 z 0.0
Scale x 1.0 y 1.0 z 1.0
Snapping
Move 25.0
Rotate 15.0
Scale 0.25

Now for the door opening animation.
“Set Current as Start” saves the current position, rotation and scale as Start Point, so we click that first.

Then we rotate our Transformer until the door is open and click “Set Current As End”



Show and Hide Button

12 objects (1 selected)
3 events (0 recent max)

x 2880.87 y -1634.88 z 0.0 Position
x 0.0 y 0.0 z 91.89 Rotation
x 1.0 y 1.0 z 1.0 Scale
0.2 Gizmo Size
25.0 Move
15.0 Rotate
0.25 Scale

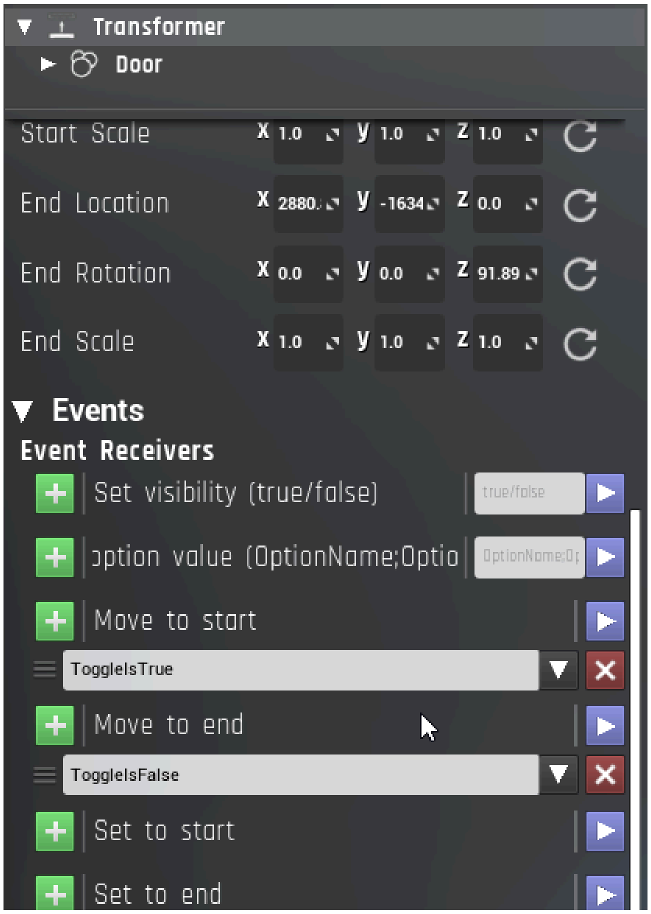
Transformer

Possesses the camera with all the defined settings

erpolation Exponent 1.0
Travel Time 1.0
Show Icon ☑
Use Shortest Path ☑
Start Location X 2880. Y -1634. Z 0.0
Start Rotation X 0.0 Y 0.0 Z 0.0
Start Scale X 1.0 Y 1.0 Z 1.0

We can test the positions with the Go To Start and End buttons below and tweak the travel time as well.

Now, this Transformer needs to listen to our Toggle from earlier. So we add the ToggleIsTrue and ToggleIsFalse events to the Move to Start and Move to End Event Receivers, respectively.



Transformer
Door

Start Scale X 1.0 Y 1.0 Z 1.0

End Location X 2880. Y -1634 Z 0.0

End Rotation X 0.0 Y 0.0 Z 91.89

End Scale X 1.0 Y 1.0 Z 1.0

Events

Event Receivers

Set visibility (true/false) true/false

ption value (OptionName;Optio OptionName;Op

Move to start
ToggleIsTrue

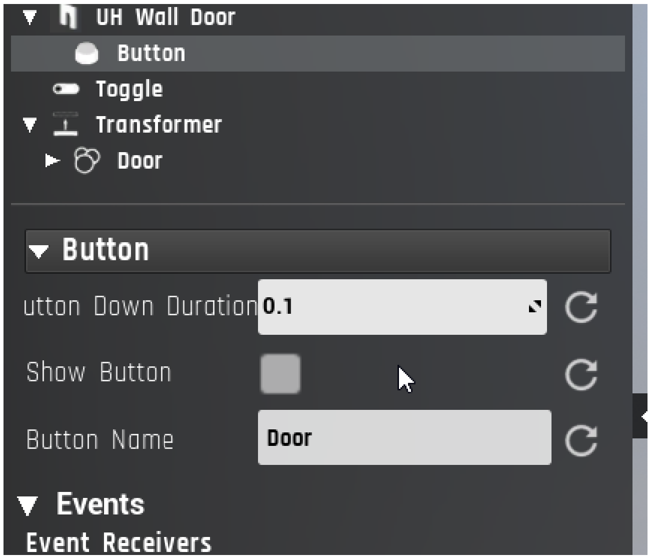
Move to end
ToggleIsFalse

Set to start

Set to end

When we test our setup now, we should see the door swing open and closed as we press the button.

As a last polishing step, we deactivate "ShowButton" on our button prop, rename it to "Door" and move it to the middle of our door.



UH Wall Door
Button
Toggle
Transformer
Door

Button

utton Down Duration 0.1

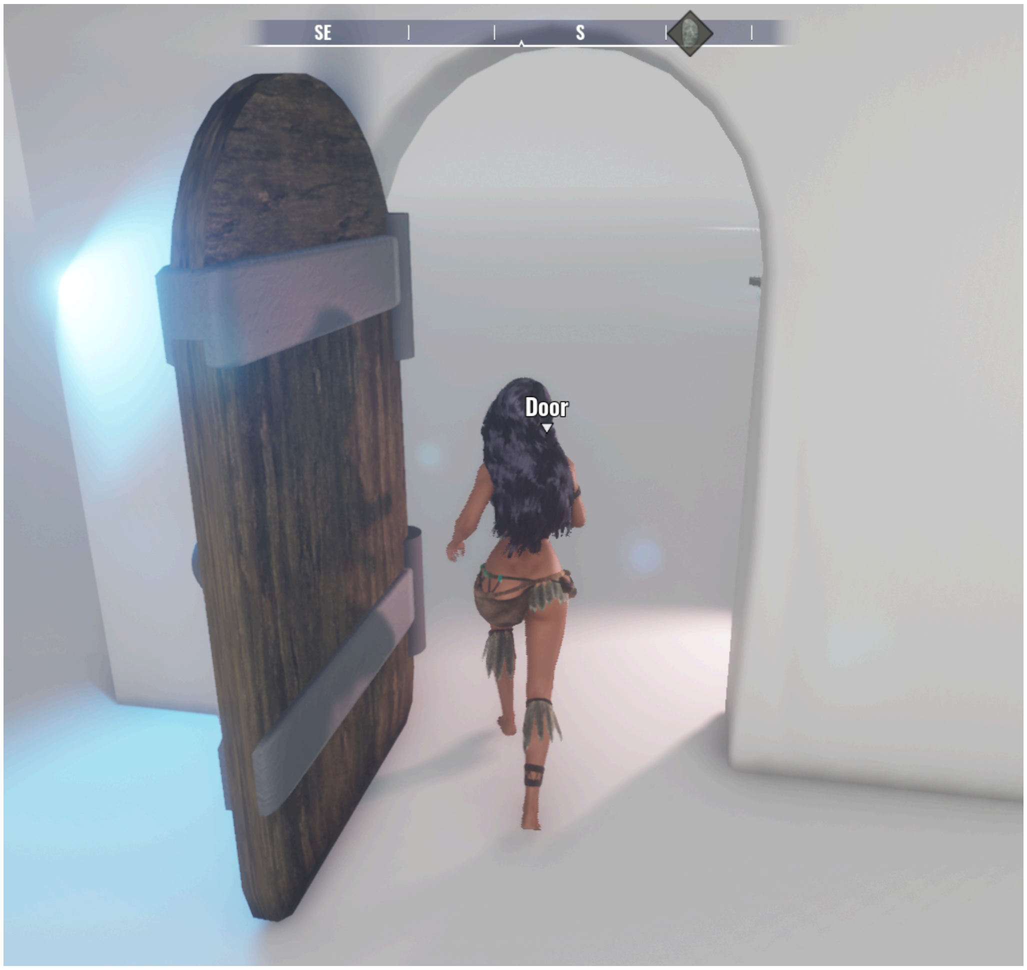
Show Button □

Button Name Door

Events

Event Receivers

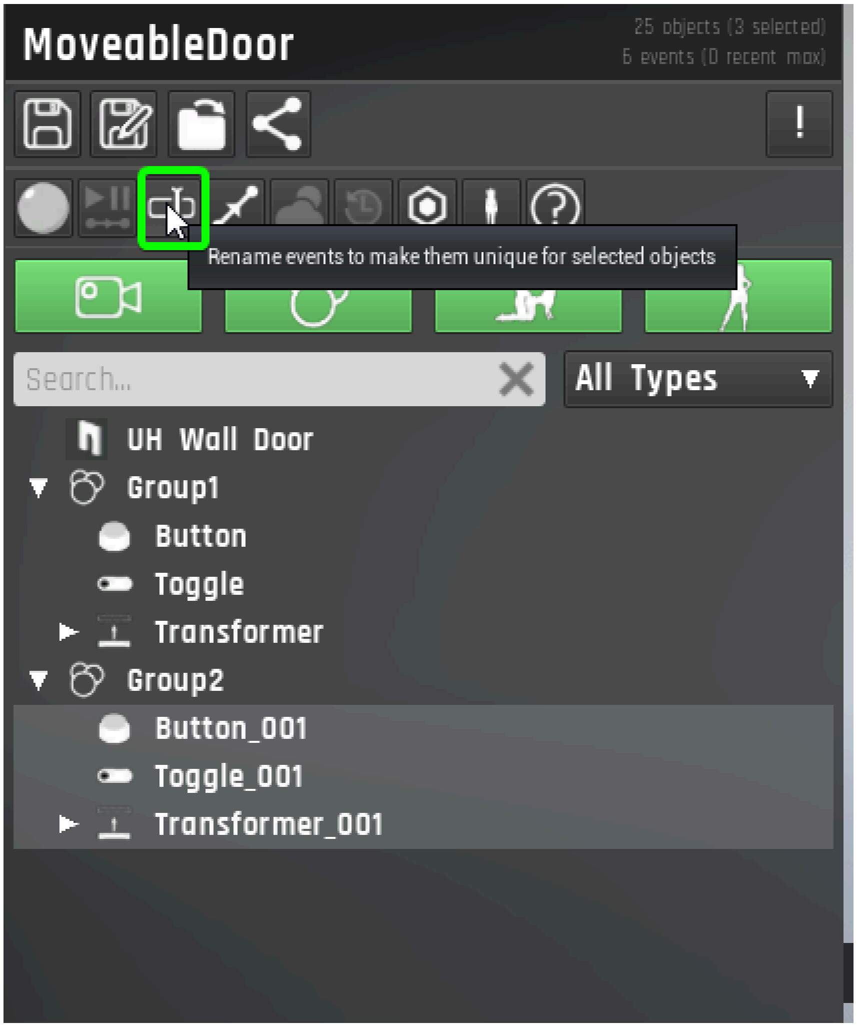
Now it looks as if we are interacting with the door itself to open it :)



Two last things to be aware of:

1) If we want to **duplicate** our door, we must remember that the Transform prop saves **transform values relative to its parent** for its start and end position. Meaning it will snap to the position of the old door when we open it if it has the same parent, even if it was placed somewhere else.
So we either need to update the Start and End on the new door, or wrap the whole setup in a group from the beginning. That way we can then use that group to move our new door to another place, while keeping the old values on the Transform prop. (Just try it, you'll see what I mean)

2) When making a duplicate door, you will notice that pressing the button on one will open the other one as well. This is because both are still using the same event names.
To quickly fix this, select all the objects on the new door and hit the Rename Events button at the top. It will auto-number the event names so they are unique again and each door button only affects its own door.



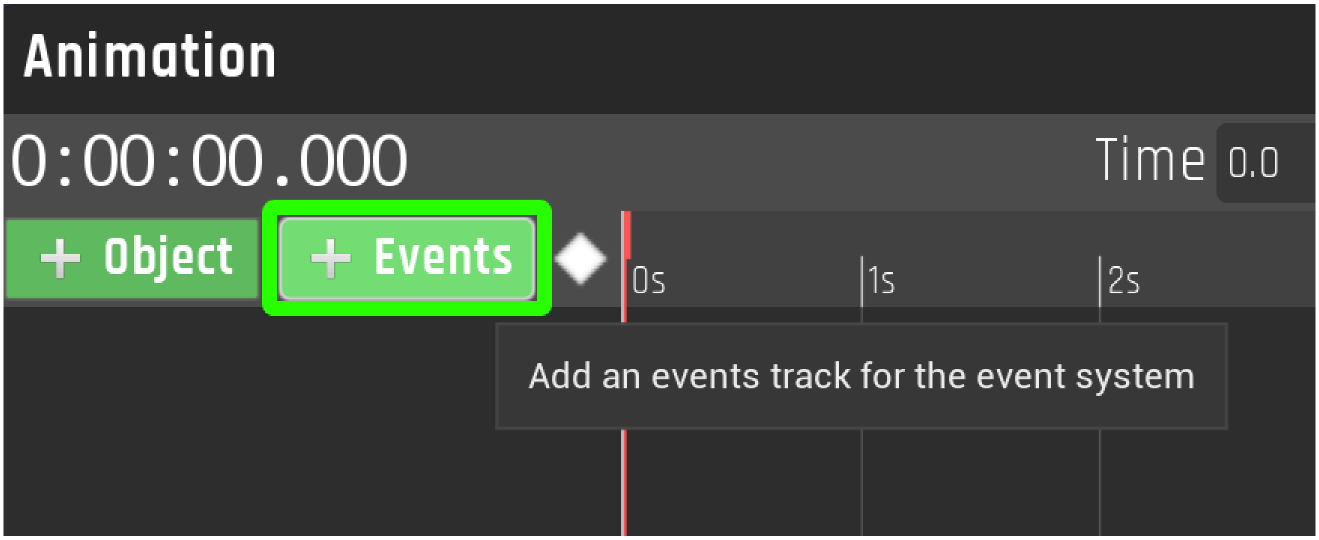
Rename events to make them unique for selected objects

That's it for a first introduction to Events! There are a lot more things you can do with the various props under the Automation tab and by sending values along with events instead of setting them on the receiver, like we did here. But the basic principle always remains the same, so I hope this could give you some help with getting started. Make sure to check out the **Event Showcase Facility** under featured Showroom maps for more examples!

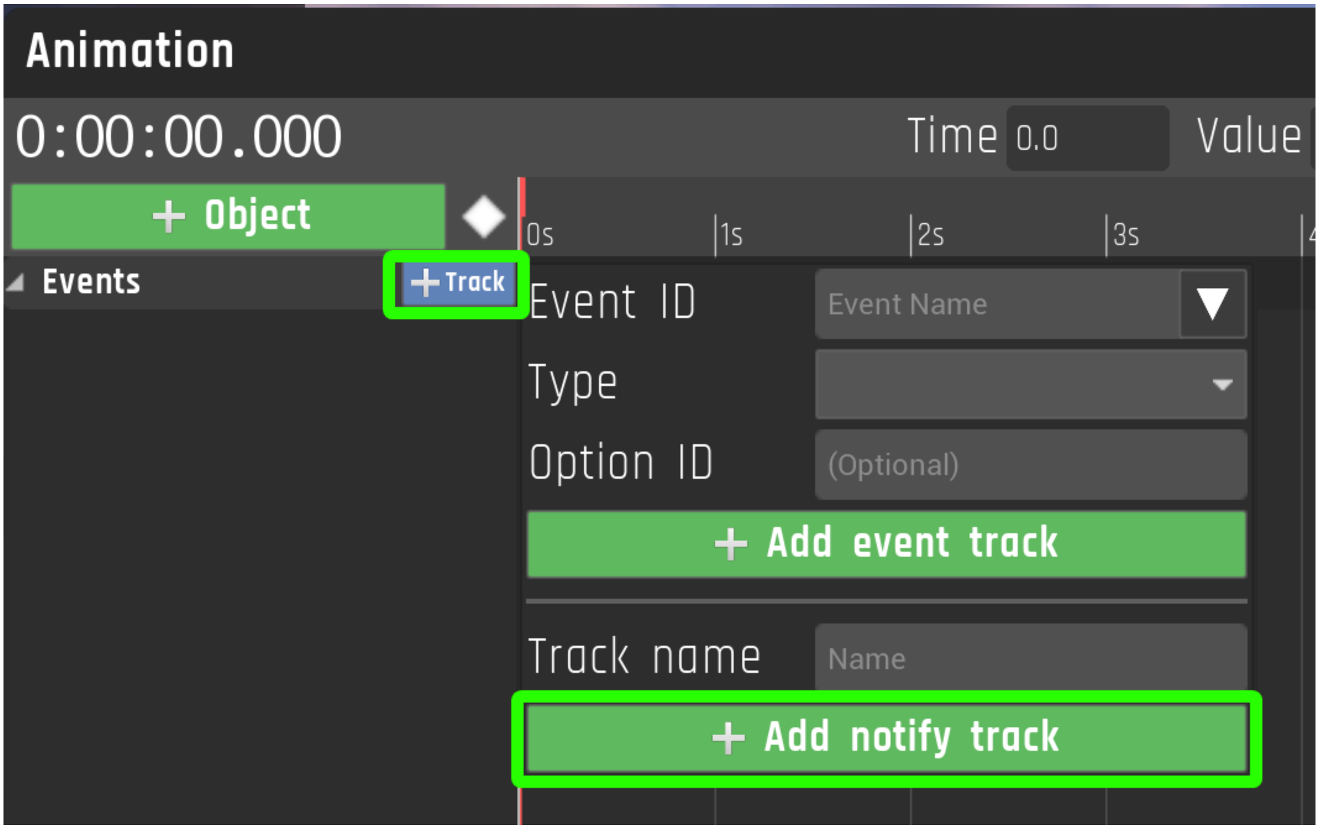
Using Events with the Animation Sequencer:

(available since April 2024)

With the addition of the Animation Sequencer, many event driven behaviors are now much easier to realize, since animation sequences can also dispatch events. To do so, simply add an Event Notify track to the sequence:



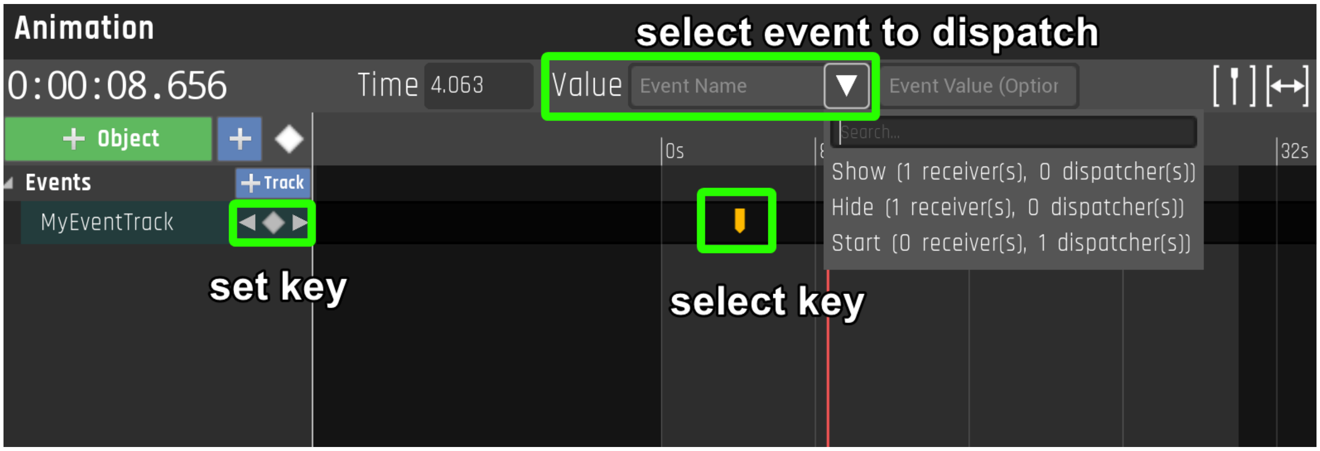
Add an events track for the event system



The **Notify Track** will simply dispatch events once, like in our door example.

Note: the name of the track is not the name of the event to dispatch! One notify track can dispatch many events.

To add an event to the notify track, set a keyframe at the desired time. Then select the keyframe and look at the top: You can select the event to fire as the “value” of the keyframe.



select event to dispatch

set key

select key

Animation
0:00:08.656
Time 4.063
Value Event Name
Event Value (Optior
+ Object
Events
+ Track
MyEventTrack
0s
32s
Search...
Show (1 receiver(s), 0 dispatcher(s))
Hide (1 receiver(s), 0 dispatcher(s))
Start (0 receiver(s), 1 dispatcher(s))

The **Event Track** is more advanced and will animate a given value over time and dispatch the event every frame. For example, it could smoothly animate a float value that then gets dispatched to all sorts of props.

Importing Your Own Models

*This bit was created by the amazing **hypermodule** and shared on our Discord. Many thanks! <3*

Click here for tutorial