AWS re:Invent

ARV206-R

Create a 3D web app in 60 minutes

Stewart Smith

Senior Consultant, Amazon Sumerian

Amazon Web Services

AWS re:Invent

aws

Agenda

Welcome to AVR206-R. In this lab we will accomplish

- Creating a room using primitive boxes and adding lights
- Importing objects (furniture) and customizing textures
- Adding a light switch and adjusting lights
- State machine behaviors to switch on/off lights

In the scene we will create, a user can click a light switch up and down to turn on and off a light.

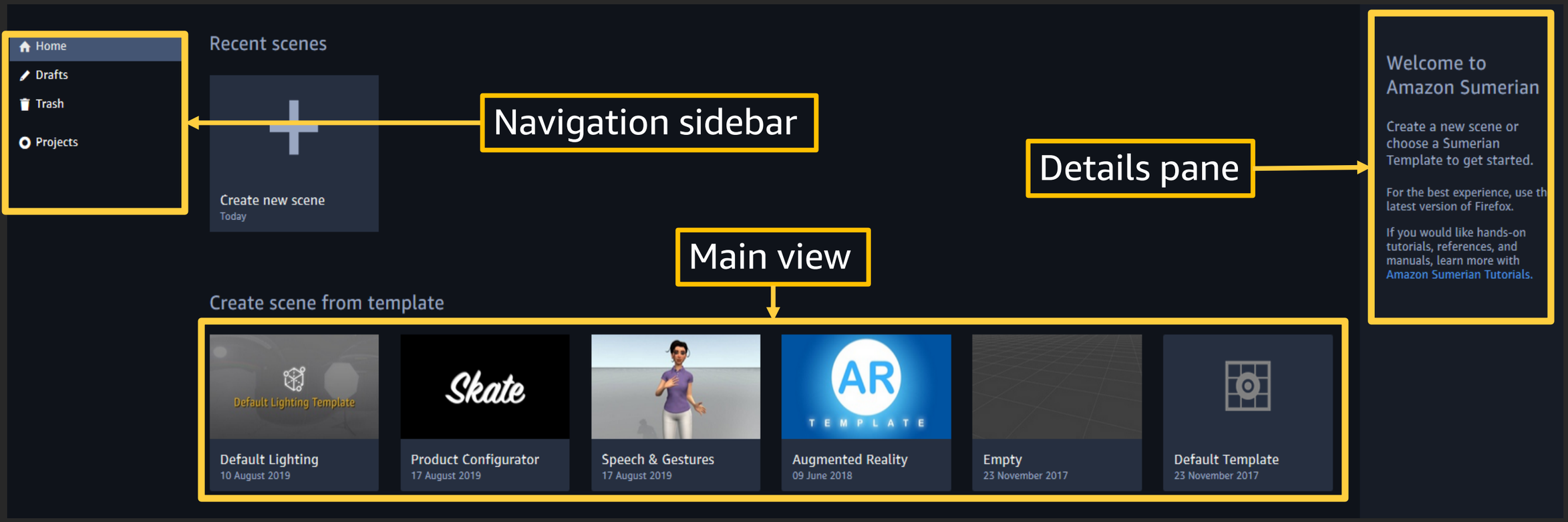
Published scene for the final product: https://bit.ly/35OtgRj

Note: This lab packet will not include the tour of the Amazon Sumerian editor. For documentation on UI, see our tutorials at https://bit.ly/2oZoFv3

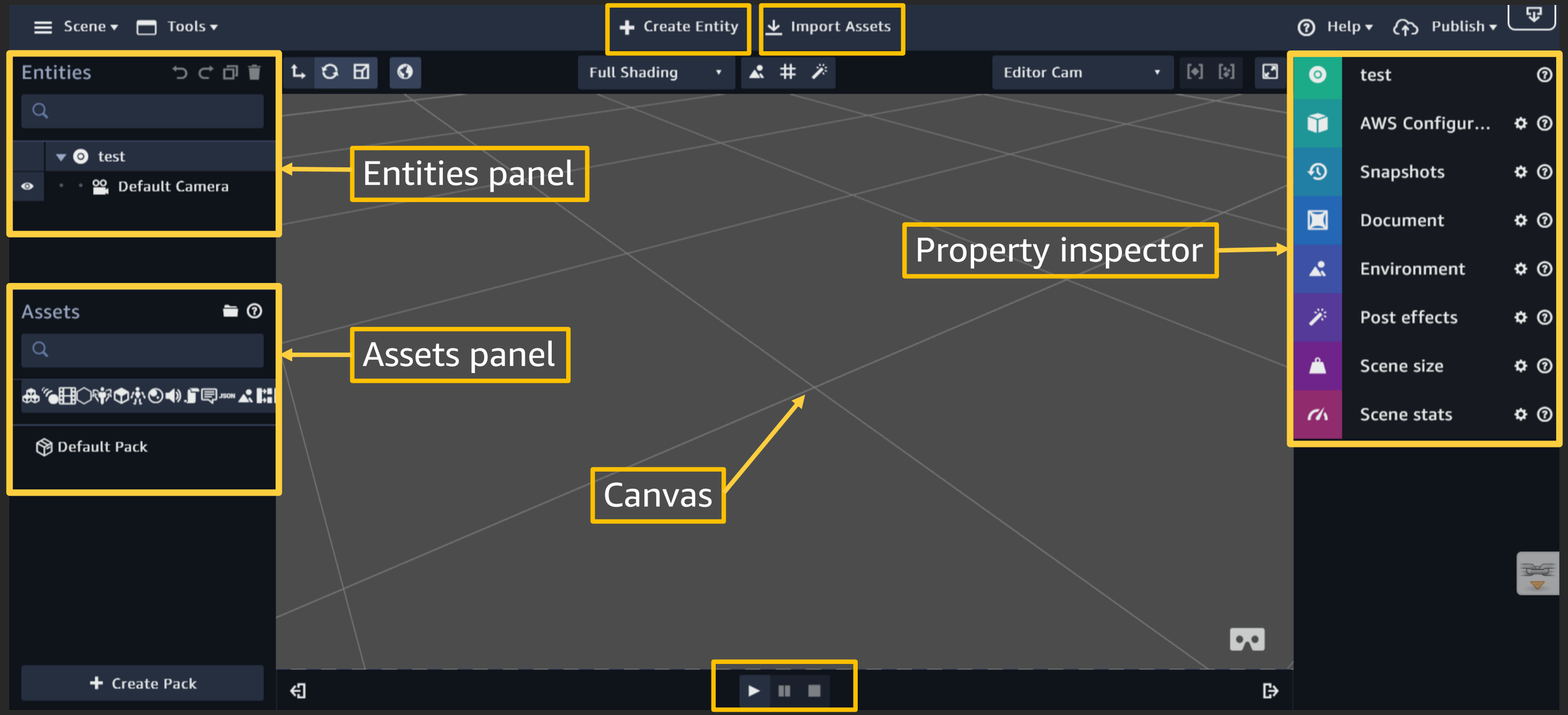
Create a new scene

Getting started: Creating a new scene

From the AWS console, find Amazon Sumerian (or go directly to the dashboard). From the dashboard, create a new *empty* scene under **Create scene** from the template section. Provide a name and click **Create**.



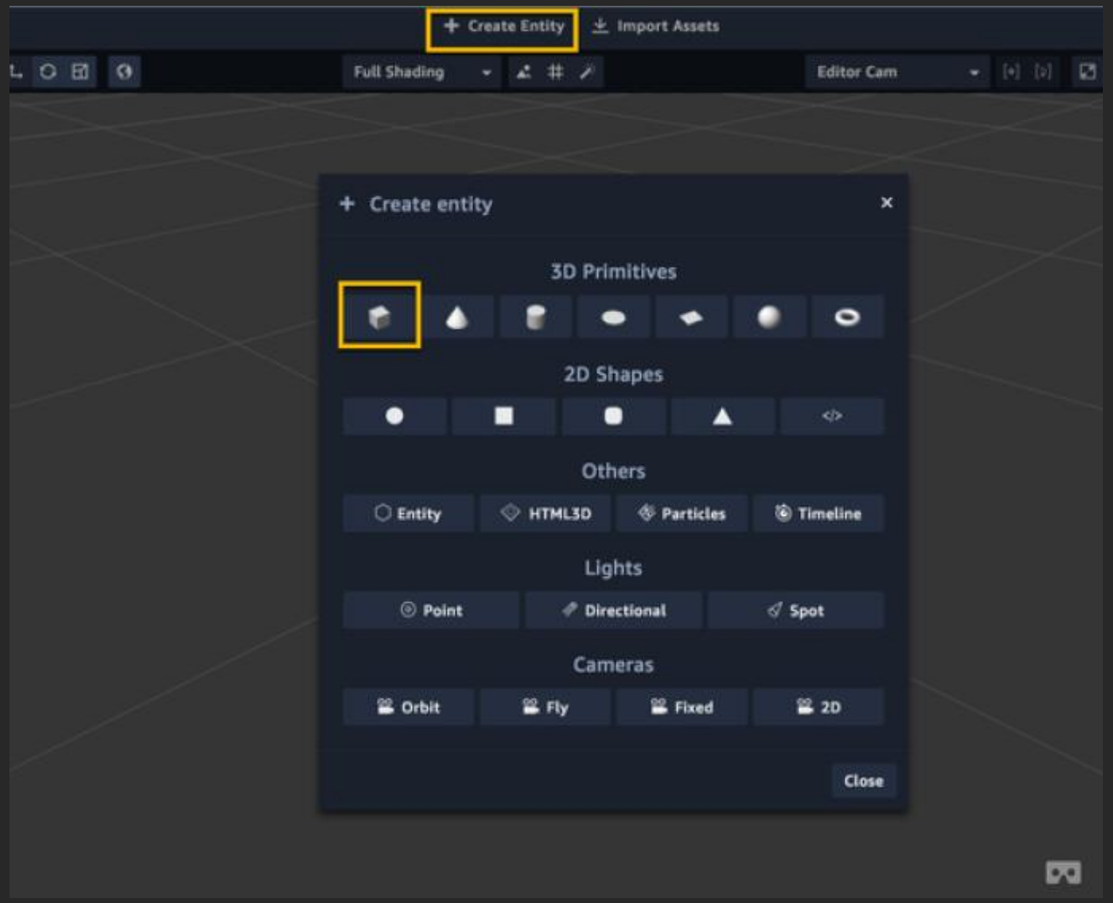
The Amazon Sumerian interface



Add a box and lights

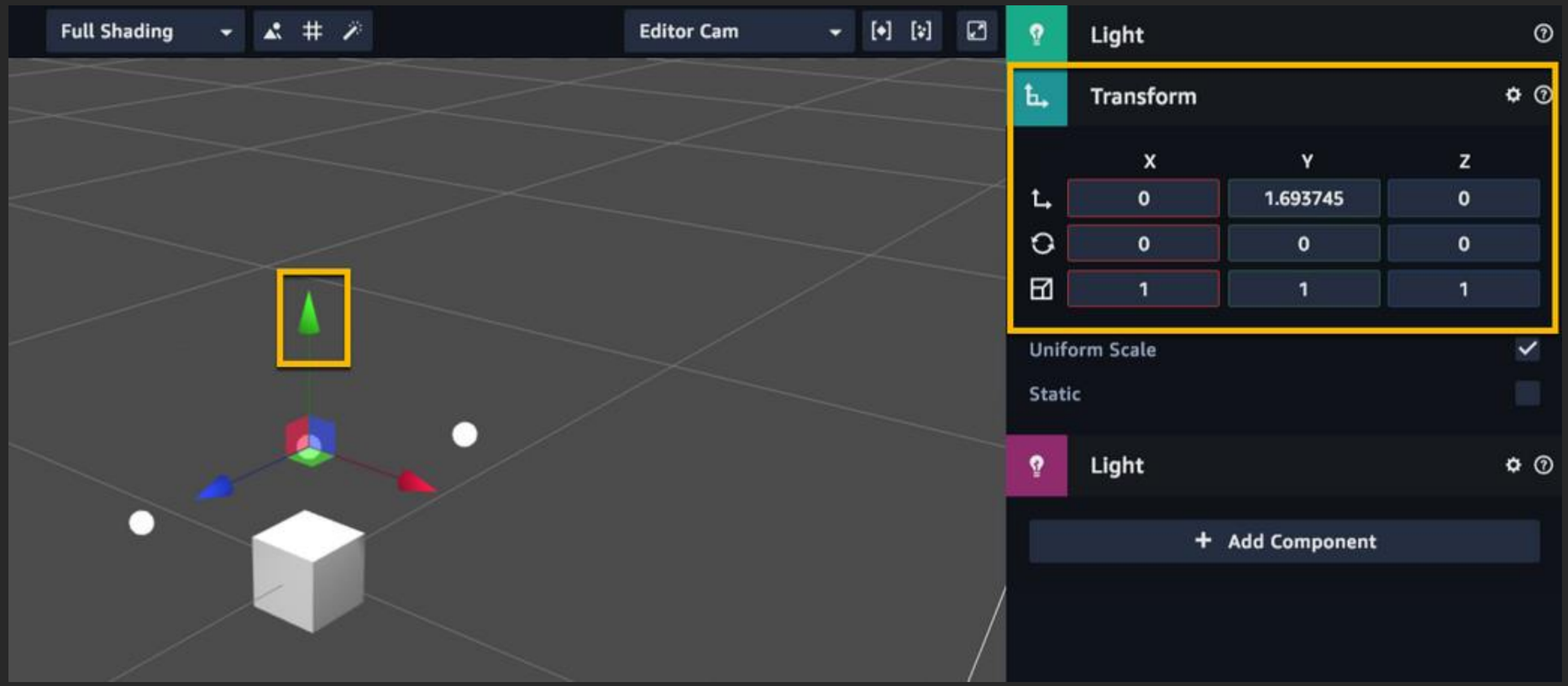
Add a box and lights

In this step, we will add a simple box and 3 lights (Point, Directional, Spot), and then move the lights around. Click **Create entity** above the canvas, and select a **Box** to add to your scene.



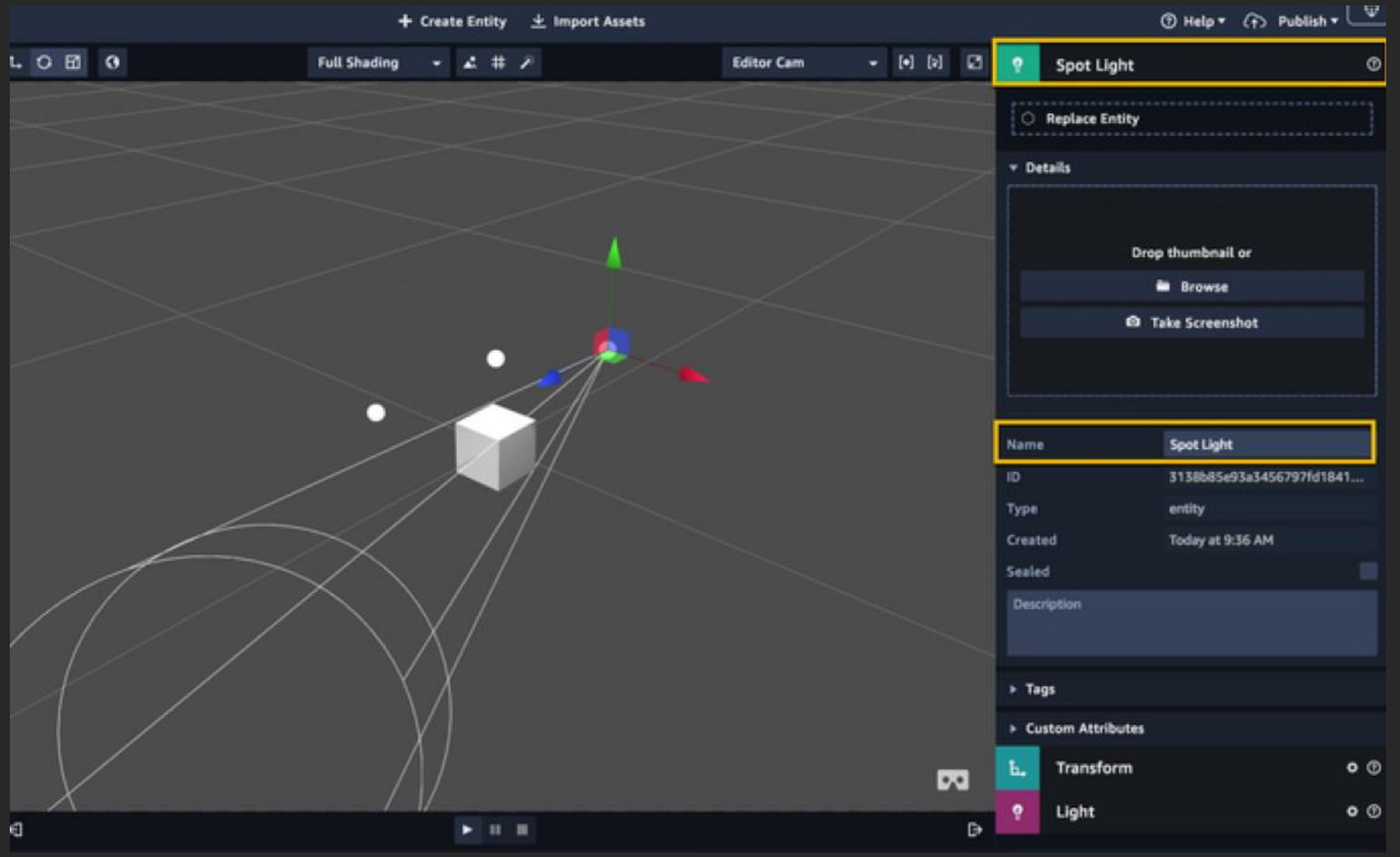
Add a box and lights

Next, we will add 3 lights by clicking **Create Entity** three times. Add **Point**, **Directional**, and **Spot** lights. With each light change the **Transform** values so they are above, and shining down on, the box. (These values do not need to be specific.) You can change transform values by moving the handles attached to the entity (i.e., the colored arrows) or by manually changing the values in the component.



Add a box and lights

You can rename the light entities by selecting the light in the entities panel, and then selecting the top tab in the inspector panel on the right. You will see a **Name** text box.

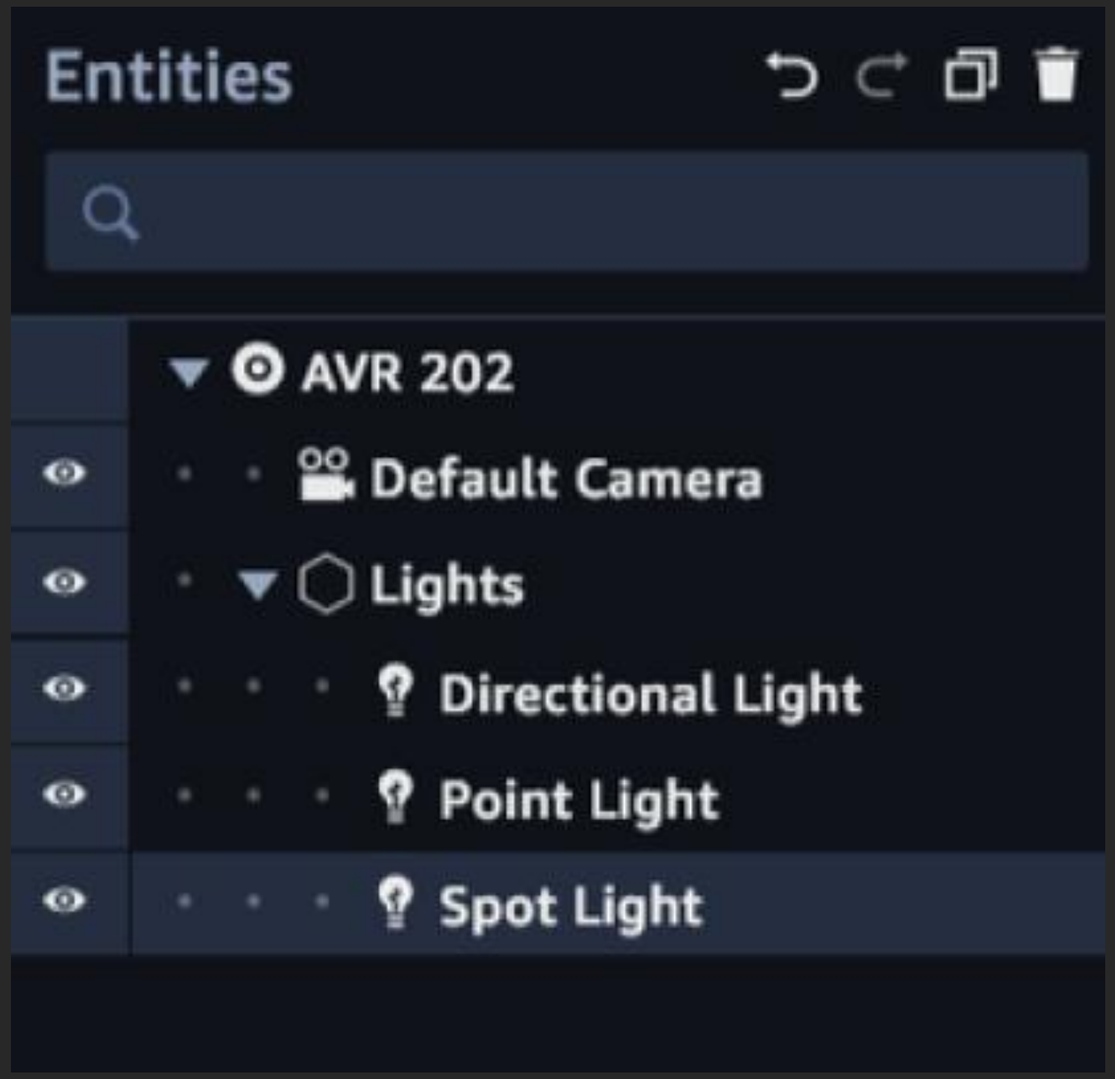


Add a box and lights

1. I will rename my lights to **Directional Light**, **Point Light**, and **Spot Light**, respectively, to match their type.
2. I positioned my lights similar to the following.

Add a box and lights

1. Finally, from the **Create entity** menu, add an *empty* entity.
2. Rename the empty entity “**Lights**.” Then, in the entities panel, drag and drop each light entity onto your new empty lights entity. (This will group all three light entities as children of the parent **Lights entity**)



Creating a room using primitive boxes

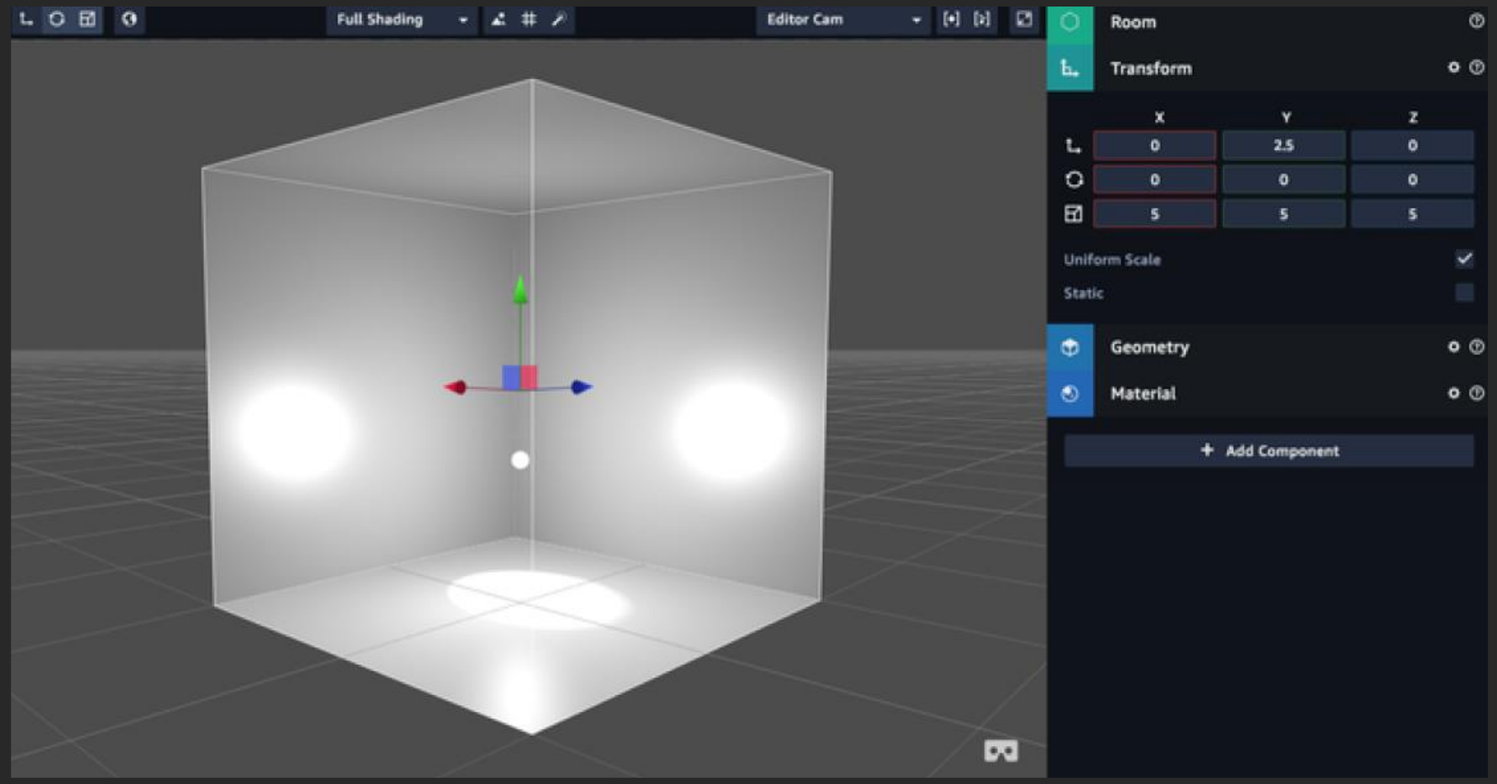
Creating a room using primitive boxes

In this step, we will create a room by using a **Box**. We will change the **Transform** values and change the **Culling** so we will only see the inside of the box (by only rendering the inner faces of the box).

Create a **Box** entity and rename it “**Room**.” With the **Room** entity selected, expand the **Material** component and then open the **Culling** property. Change the **Cull Face** drop down option to **Front.**

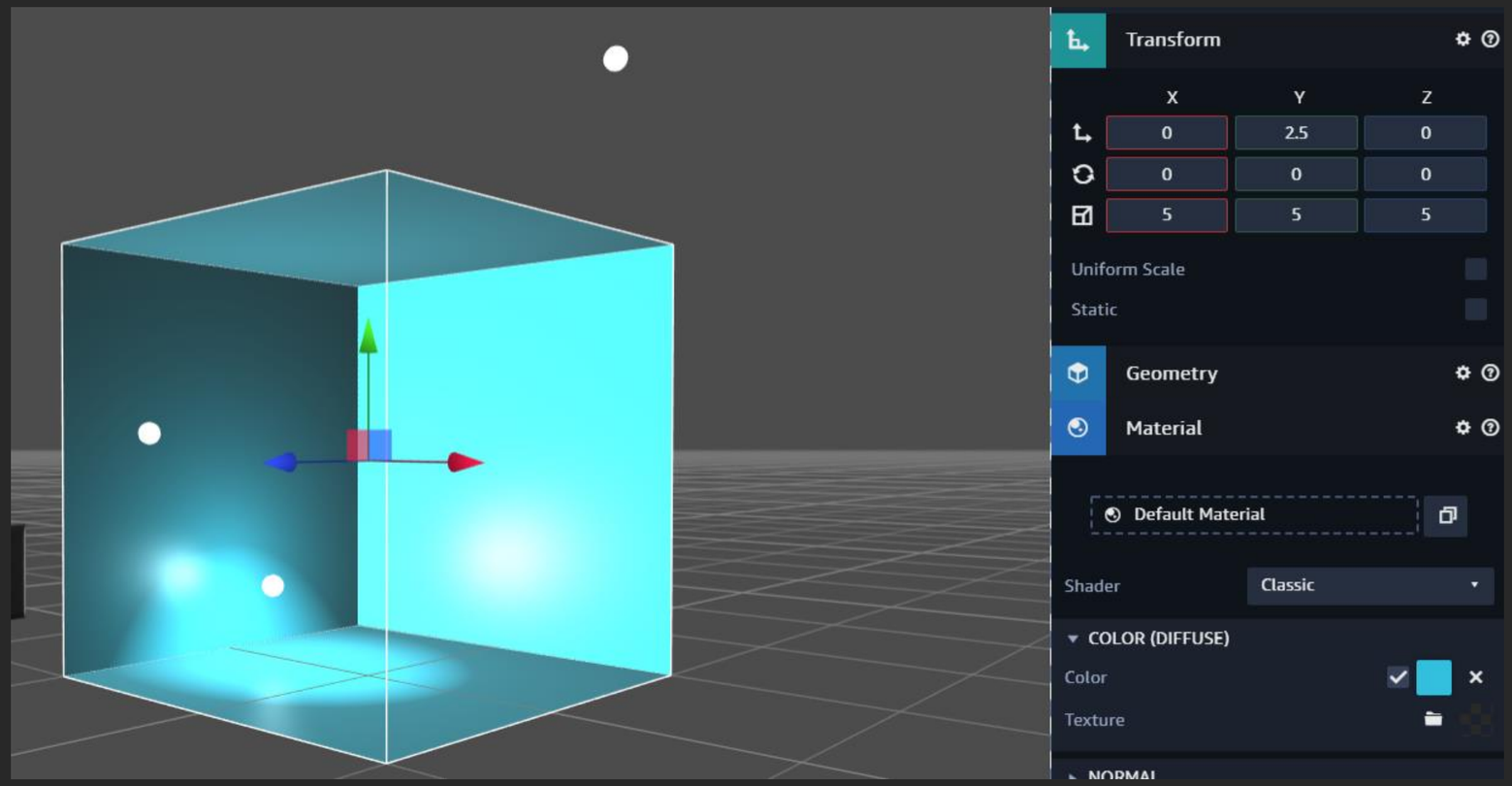
Creating a room using primitive boxes

*Note: This makes it so we will only see the inside of the **Box** by only rendering the inner faces of the **Room**.* Change the **Transform** values to the following. ***Note: We recommend making your Room Transform values identical to ours.***



Creating a room using primitive boxes

Optional: Change the color of the floor/walls by selecting one of those entities. Expand the **Material** component, and then open the **Color (Diffuse)** property. Change the color using the color picker. *Note: Changing the color of one entity will change the color of all associated duplicates.*



Importing objects (furniture)

Importing objects (furniture)

In this step, we will import several objects to furnish our room. We will add a chair, table, rug, and lamp. Feel free to import any objects you like.

1. Click **Import Assets** above the canvas. This will open the **Asset Library**.
2. You can scroll down to find the furniture assets or type ASIN in the search box to see furniture assets.
3. We will select and add a chair.
4. Once the object/asset populates in the **Asset Panel**, expand the asset's contents and find the entity asset type, which has a hexagon icon next to it. (Note: these furniture objects are named after the ASIN number, a retail ID associated with objects available for purchase on Amazon.com). Drag the entity onto the canvas.
5. Rename your entity “**Chair**”. Note: You can rename the entity “Chair” either from the Asset Panel or the Entities Panel. Position your chair somewhere in your room.

Importing objects (furniture)

We will follow this process for the following assets:

- Table
- Lamp
- Rug

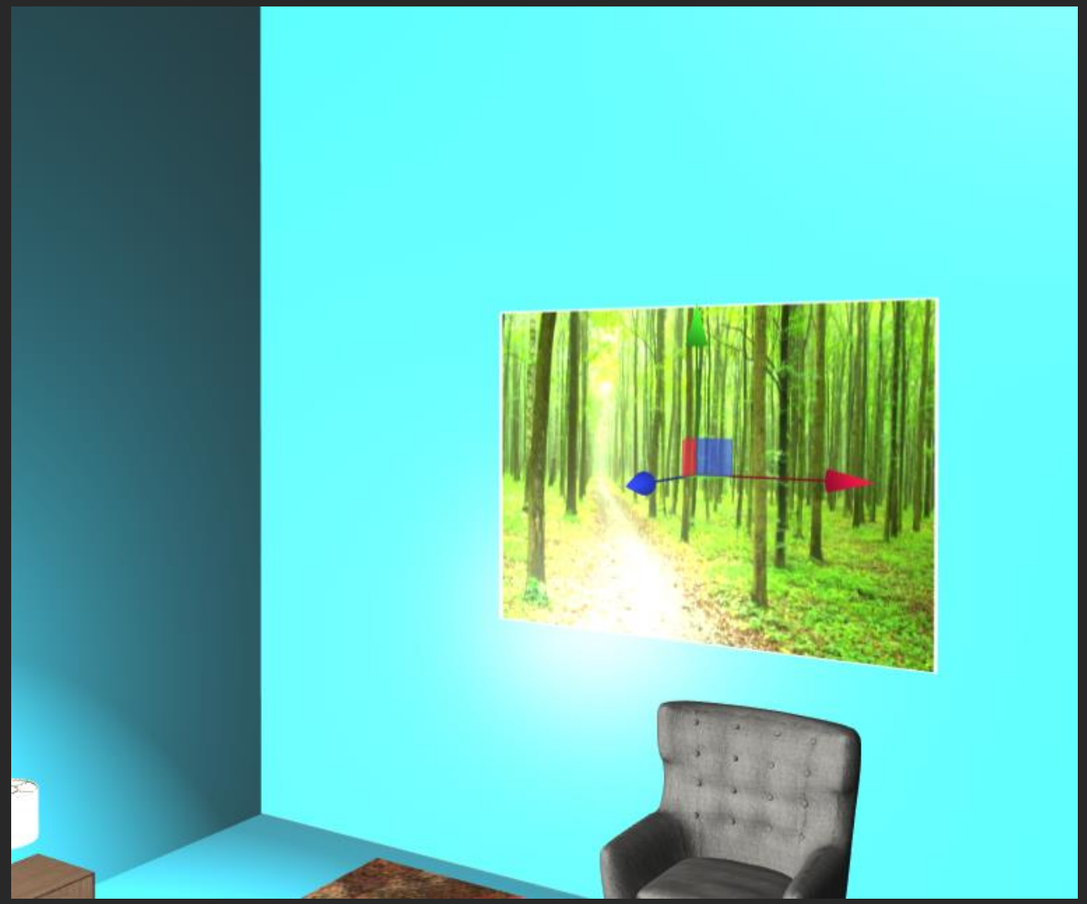
Our scene will look something like this.

Customizing textures

Customizing textures

In this step, we will place a picture on the wall by using a **Quad** entity and then adding a custom image as the texture.

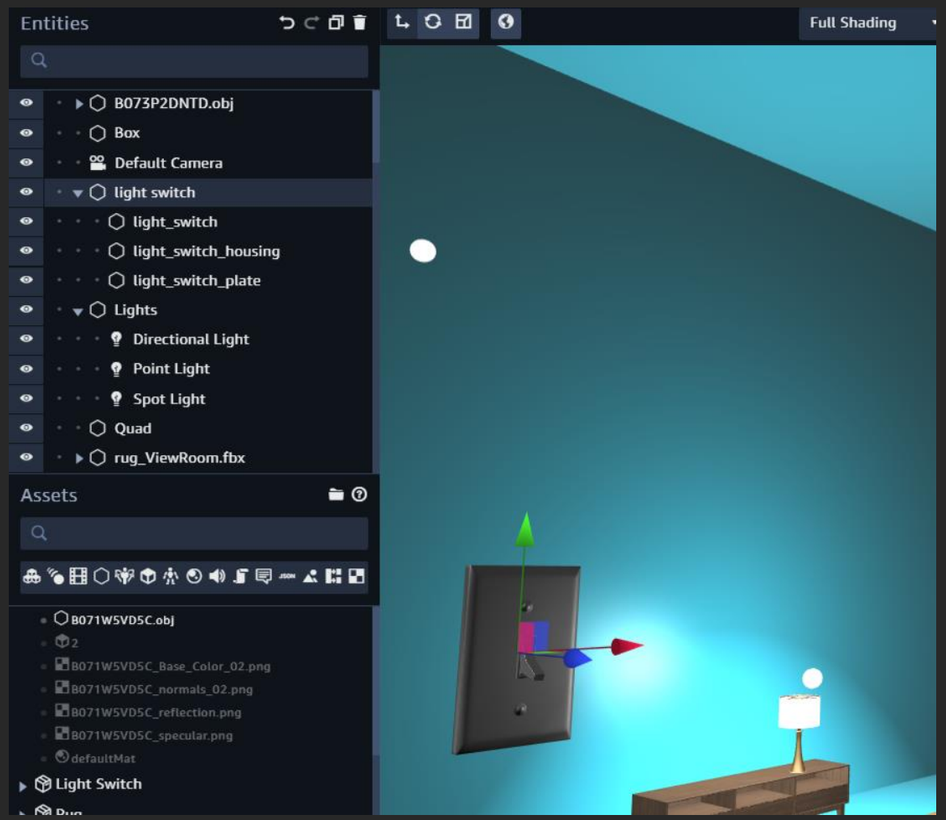
1. From the **Create Entity** menu, add a **Quad**.
2. Rename the Quad “**Picture**”.
3. Position the Picture somewhere on the wall and adjust the **Scale** values to be (1, 1.5. 1).
4. With the Picture entity selected, expand the **Material** component and then expand the **Color (Diffuse)** property. Either upload your own image or drag and drop your own image on the texture drop input. You may use any picture you want.
5. Our scene will look something like this



Adding a light switch

Adding a light switch

1. In this step, we will add the light switch. Open the **Asset Library.** Search for and add the **Light switch** asset.
2. Like you did with the furniture entities, from the Assets panel, add the Light Switch entity to your scene.
3. The light switch will appear in your scene, but it will probably be too small to see. Change the **Scale** of the light switch to (7, 7, 7). Move the light switch to be somewhere on the wall
4. Lastly, we want to move the switch to be in the down position (rather than sticking straight out like it is currently). In the **Entities** panel, expand the light switch entity until you can see the light_switch child entity. Select it.
5. Adjust the Z Rotation of the light_switch entity to 23.



Adding a light switch

Adjust Your Lights - Move your **Point** light to be inside the lamp.

Now that we have everything in our scene, we can adjust the lights. We will rotate the **Directional** and **Spot** lights to feel a little more comfortable.

State machine behaviors

State machine behaviors

A state machine is the system to build behaviors, which are changes to entities initiated by a user. Behaviors can act alone or they can act in a sequence. They can be movements, animations, sounds, camera changes, collisions, reactions, AWS service integrations, etc. Behaviors are built using a collection of states that are connected by transitions, as an entity transitions from one state to another.

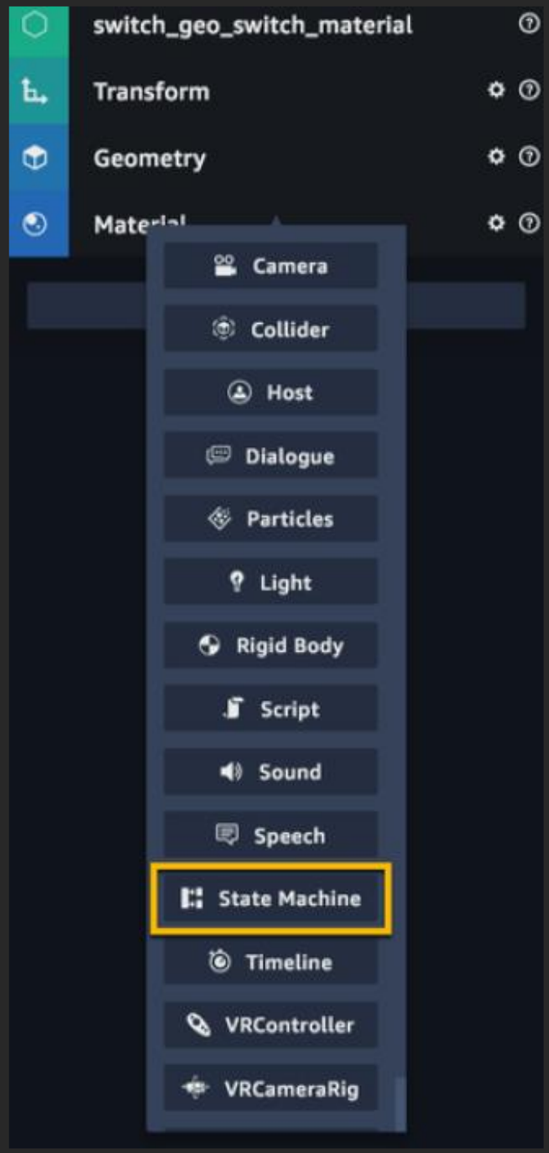
In this step, we will create two behaviors, which will each serve a different purpose. The **Switch On/Off** behavior will allow a user to click the light switch so that it flips up and down. (Technically, it will be rotating.) This behavior will also emit a message that can be used by the second behavior (**Light On/Off**) to initiate, thus turning the lights on and off.

State machine behaviors

Switch On/Off Behavior

This behavior will be attached to the **light_switch** object inside the **Light switch** entity (i.e., the switch of the light switch)

1. Select the light_switch entity, nested under the light switch
2. In the inspector panel, click **Add Component** and add a **State Machine** component
3. Click the **+** button to add a new behavior. This will open the **State Machine editor**.
4. Rename the behavior "**Switch On/Off Behavior**"
5. With the default, and only, State 1 selected, change the name of the state to "**Wait for Click**." Then click **Add Action**.



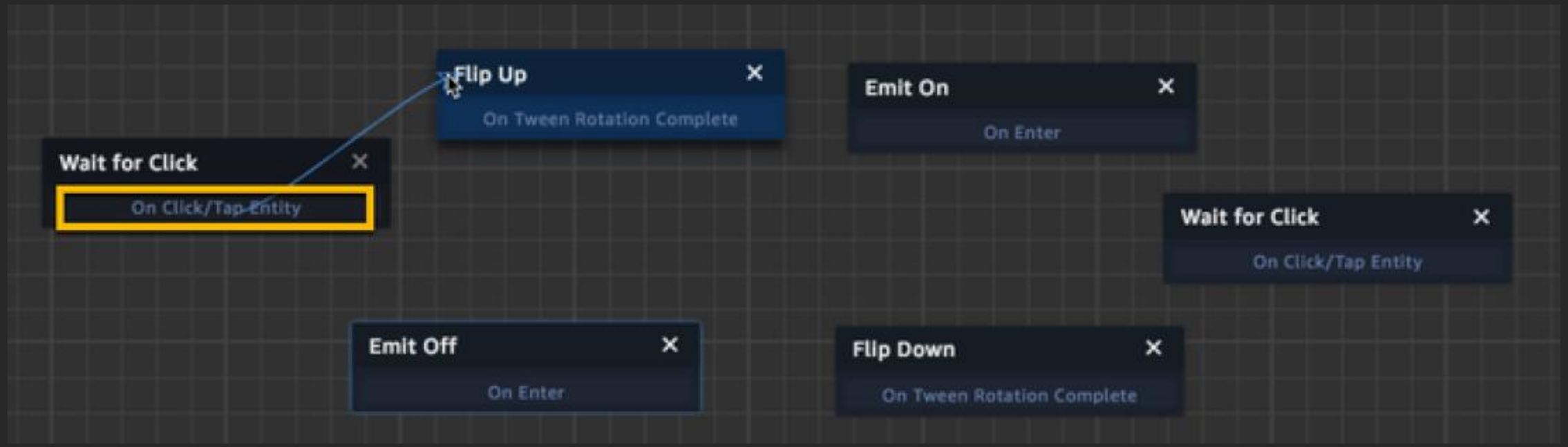
State machine behaviors

6. This will open the action library. Search for and add the **Click/Tap on entity** action
7. Add a new state
8. Change the name to “**Flip Up**”
9. Add another action. Search for and add the **Tween Rotate** action
10. Adjust the action’s properties to the following:
11. Select the **Wait for Click** state and duplicate it
12. Reposition the state’s node and then duplicate the Flip Up state. Rename to **Flip Down**. Change the properties to the following:
13. Duplicate the **Emit On** state and rename to “**Emit Off**.” Change the Channel message to “**lightOff**”

State machine behaviors

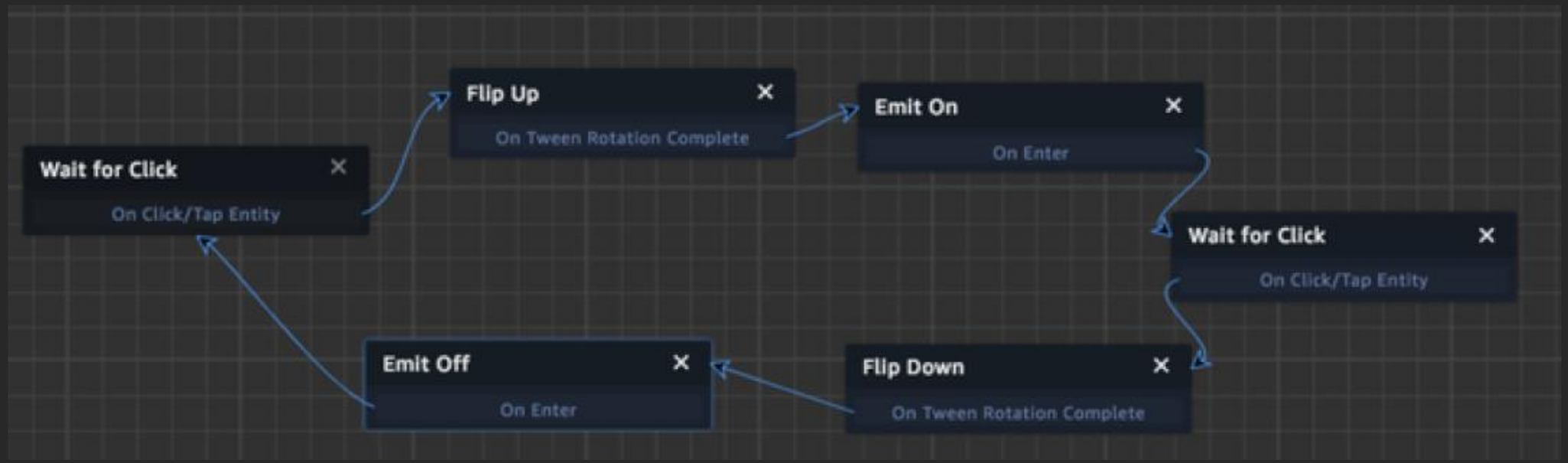
Reposition the state nodes to the following:

Now we need to add transitions between each state. Starting with the first **Wait for Click** action, click the transition label (i.e., **On Click/Tap Entity**) at the bottom of the state. Drag a transition arrow to the next state



State machine behaviors

Do that for all the states until the behavior cycle is completed



Note: Make sure the first Wait for Click state is set as the Initial State

Test your behavior. Press play and click on the switch. See it flip up and down. If you have the **State Machine** editor open, see the states transitioning.

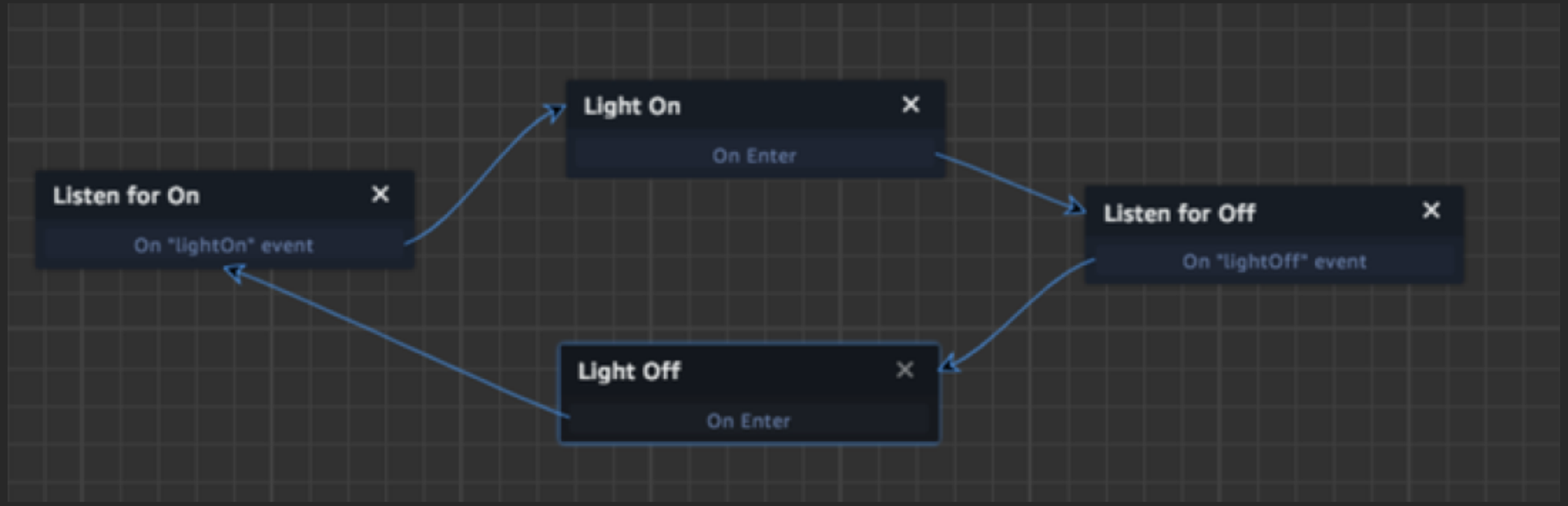
State machine behaviors

Light On/Off Behavior

1. Let's create a behavior for the *Point Light*. This behavior will turn the light on and off. To have it occur on the click of the light, we will use the messages being emitted by our first behavior ("lightOn" & "lightOff"). We will start this new behavior by using a *Listen* action.
2. Select *Point Light* and add a **State Machine**. Add a new behavior and name it "Light On/Off Behavior".
3. Change the name of *State 1* to "Listen for On". Add an action. Search for and add a *Listen* action. In the **Message channel,** add the "lightOn" message.
4. Add another state. Rename it "Light On". Add an action. Search for and add a *Set Light Properties* action. Make sure the **Intensity** is set to **0.** Finally**,** add a *Transition* action.
5. Duplicate the *Listen for On* state. Rename "*Listen for Off".* Change the **Message channel** to "lightOff".
6. Duplicate the *Light On* state. Rename "Light Off". Change the **Intensity** to **1**. Click **Set as Initial State** (because we want the light to be off when we start the scene).

State machine behaviors

Finally, starting with the **Light Off** state (our initial state), add transitions clockwise. Your behavior should look like this:



Press play and click the *Light switch* to see your Point Light turn on and off

State machine behaviors

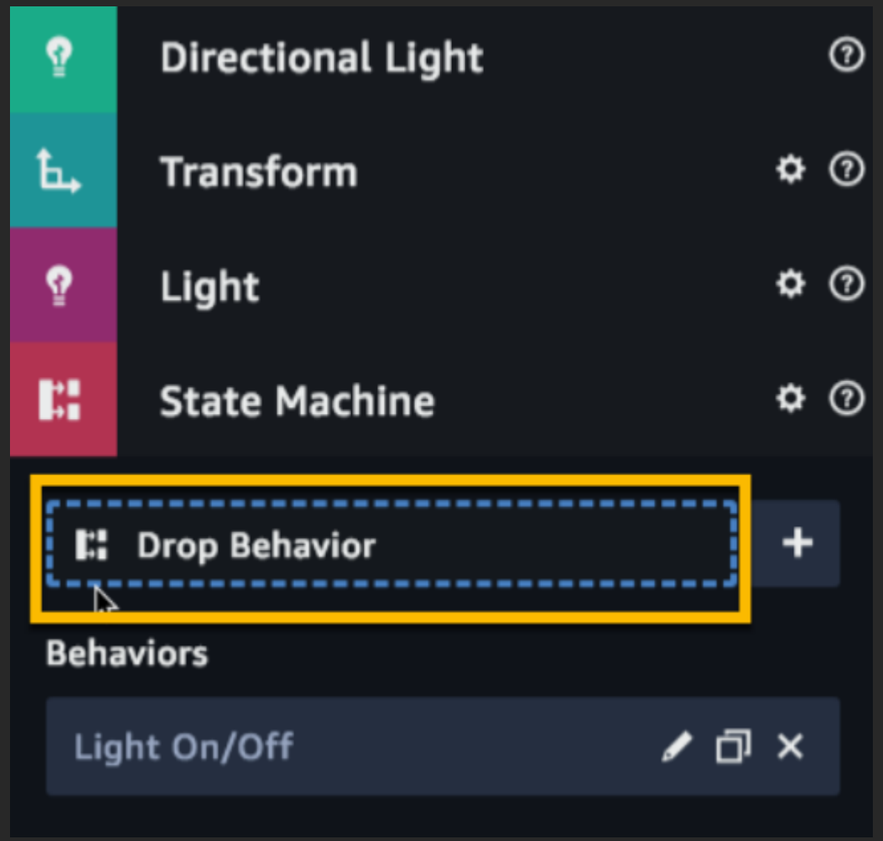
Apply Behavior to Other Entities

The Light On/Off Behavior works with the Point Light. In this final step, let's apply the same behavior to another Light entity.

1. Select the Directional Light entity
2. Add a State Machine component
3. In the Assets panel, filter to only see Behaviors
4. Drag and drop the Light On/Off Behavior onto the Directional Light drop input in the State Machine component

Test your scene again to turn on/off both the Point and Directional light

Now, publish and share your scene



Never stop learning

Additional and optional steps

We have completed the basics of this workshop. You can complete these at home in your free time

Turning on/off more lights

Apply the Light On/Off Behavior to the Directional and Spot lights by adding a State Machine component and then dragging the behavior from the Assets panel to the behavior’s Drop Input.

Adding a host for instruction

Add a Host with a custom speech that provides instructions to click the light switch. See the Speech Component tutorial at https://bit.ly/2LbLjcQ for more information.

Play/stop sound upon click

Add a Sound component to the Room entity and add a song/audio file to your scene. In the Light On/Off Behavior, add Play Sound and Pause Sound actions to play/pause your audio file.

Download Free Book

Amazon Sumerian by Tutorials

Are you ready to develop 3D immersive experiences with Amazon Sumerian? The raywenderlich.com Tutorial Team wrote this book to help you get started.

You do not need to be a programmer or a 3D whiz kid. You just need some free time and a modern web browser.

Use the QR code
to download today!

Further reading

- To get started in VR, check out how to use the newly released VR Asset Pack: https://bit.ly/2oZrdcB
- Light switch scene workshop on Twitch: https://bit.ly/31tGfof
- AWS Configuration (for creating Amazon Cognito identity pool IDs): https://bit.ly/2VZmUKI
- State machine basics: https://bit.ly/2oWZERh
- Event basics: https://bit.ly/33QuEBn
- Using the host component: https://bit.ly/2LbLjcQ
- Skyboxes and skyspheres: https://bit.ly/2N1obMW

Thank you!

Stewart Smith

stewarsm@amazon.com

AWS re:Invent

aws

Please complete the session survey in the mobile app.