

Developer

/ Documentation

/ Unreal Engine ▾

/ Unreal Engine 5.4 Documentation

/ Animating Characters and Objects

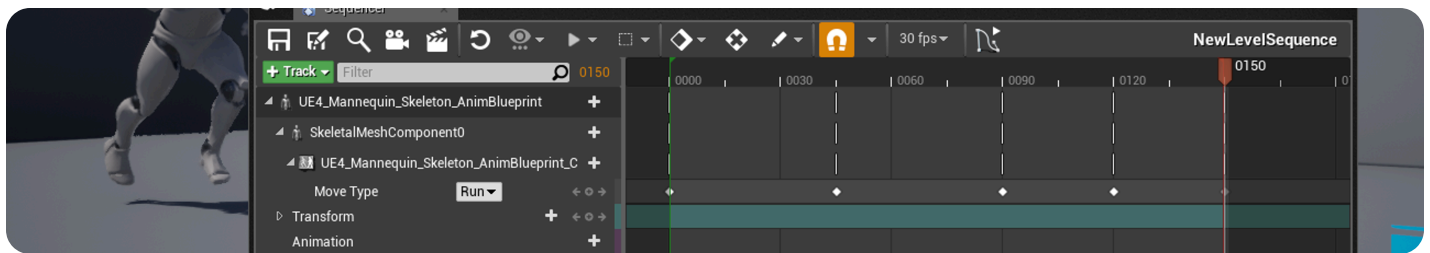
/ Cinematics and Sequencer

/ Cinematic Workflow Guides and Examples

/ Controlling Anim Instances with Sequencer

Controlling Anim Instances with Sequencer

Shows how to animate variables on anim instances through possessables



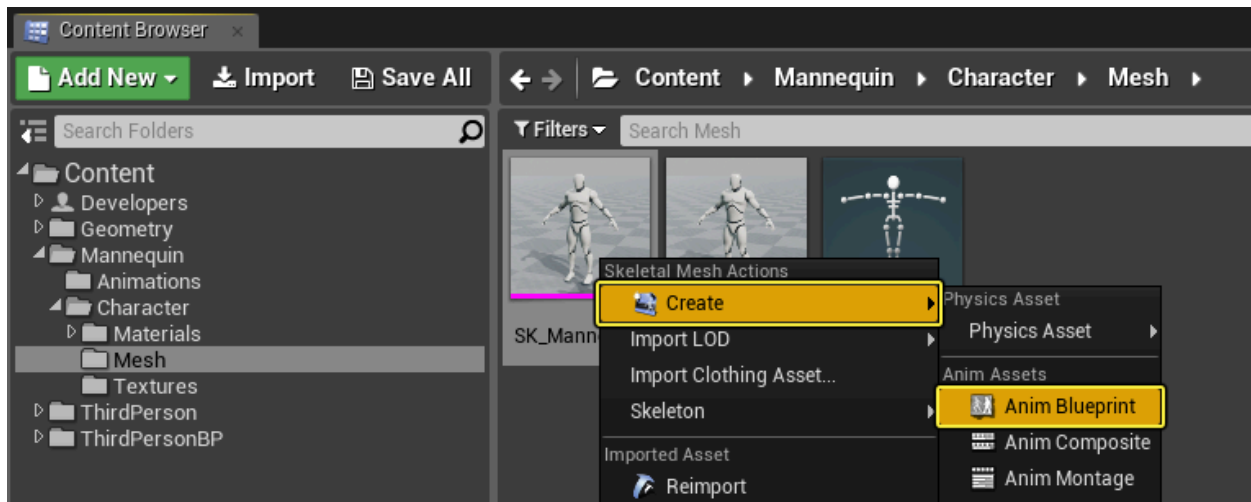
Within Sequencer, it is possible to animate variables on Anim Instances through Possessables, enabling you direct control of Anim Blueprint variables, functions and other content. By adding a Skeletal Mesh Component Track, you can gain access to the Anim Instance Track where any variables that are exposed to cinematics will be shown and available to keyframe.

In this how-to, we blend between multiple animation poses that are defined within an Animation Blueprint by keyframing those changes inside Sequencer.

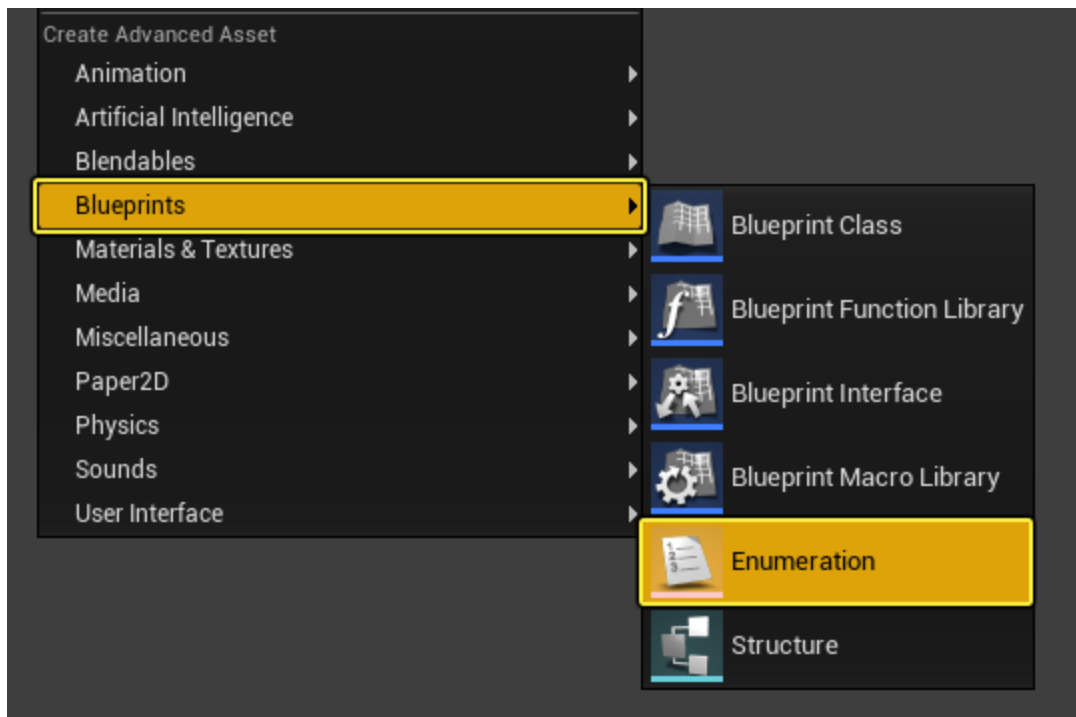
Steps

 For this guide we are using a new **Blueprint Third Person** template project.

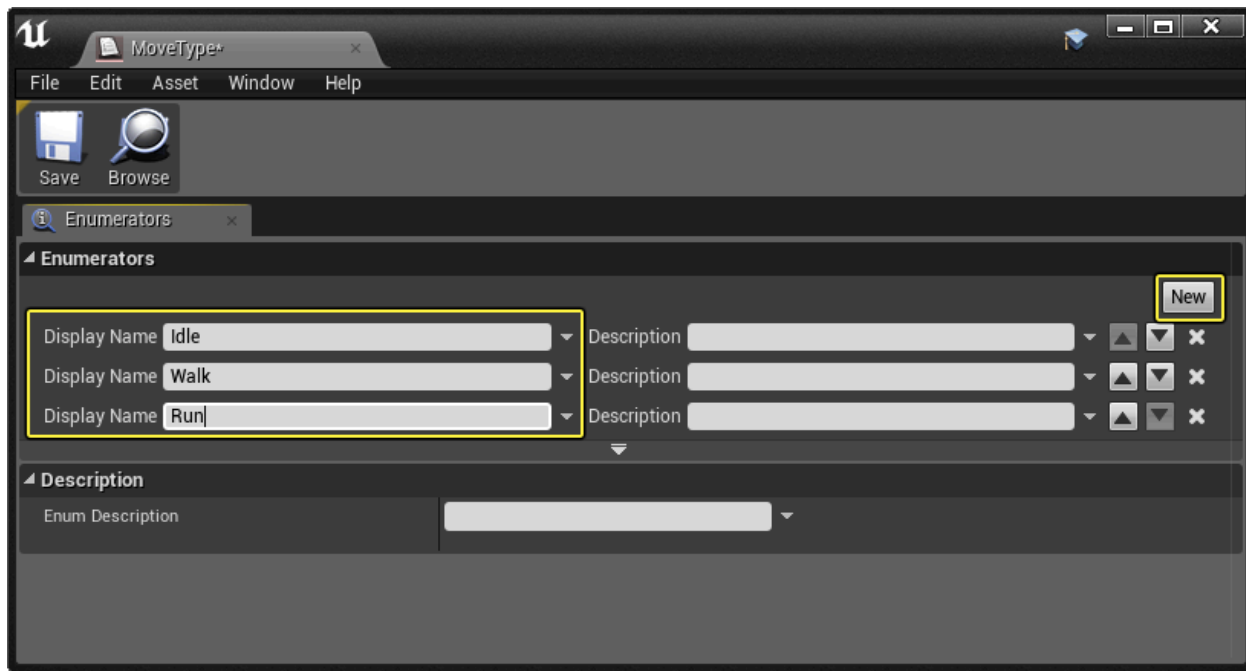
1. In the **Content/Mannequin/Character/Mesh** folder, right-click the **SK_Mannequin** then select **Anim Blueprint** under **Create** and give it any name.



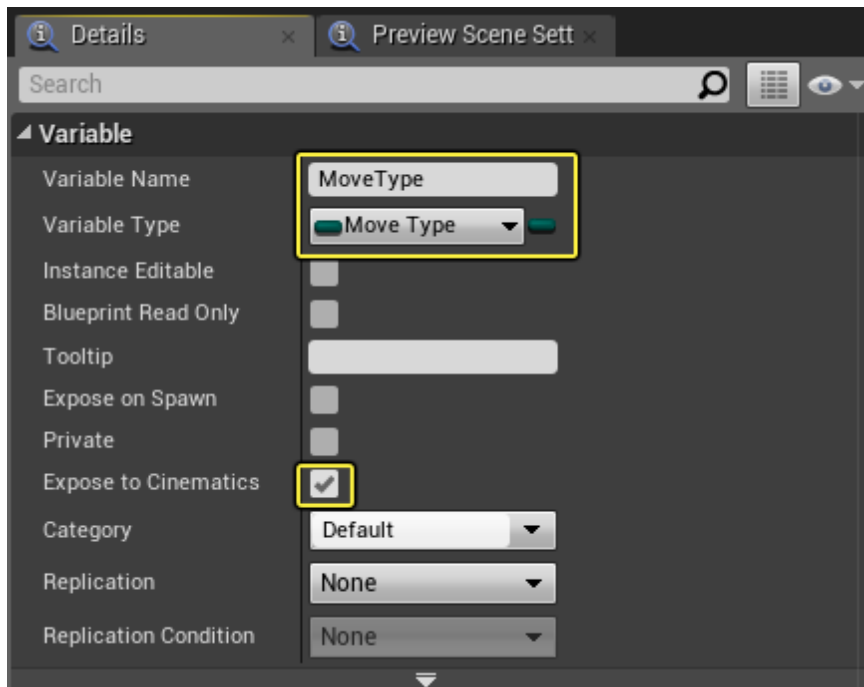
2. Right-click in the **Content Browser** and under **Blueprints**, select **Enumeration** and call it **MoveType**.



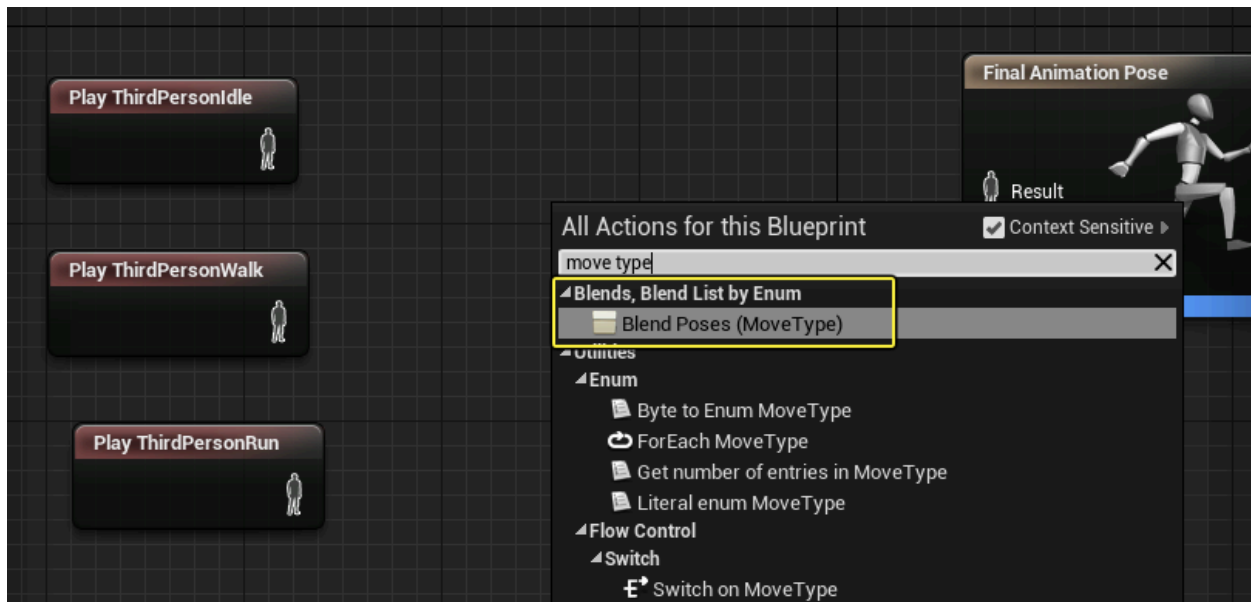
3. Create three enumerators called, **Idle**, **Walk** and **Run** by clicking the **New** button.



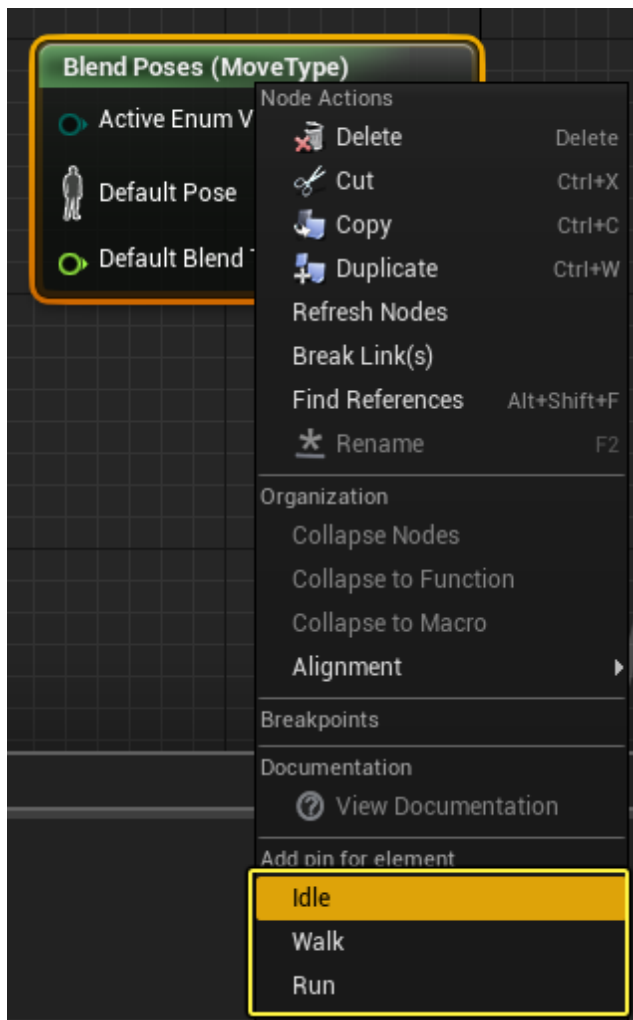
4. Open the **Animation Blueprint** created in step 1 and create a variable of the **MoveType** type called **MoveType** and enable **Expose to Cinematics**.



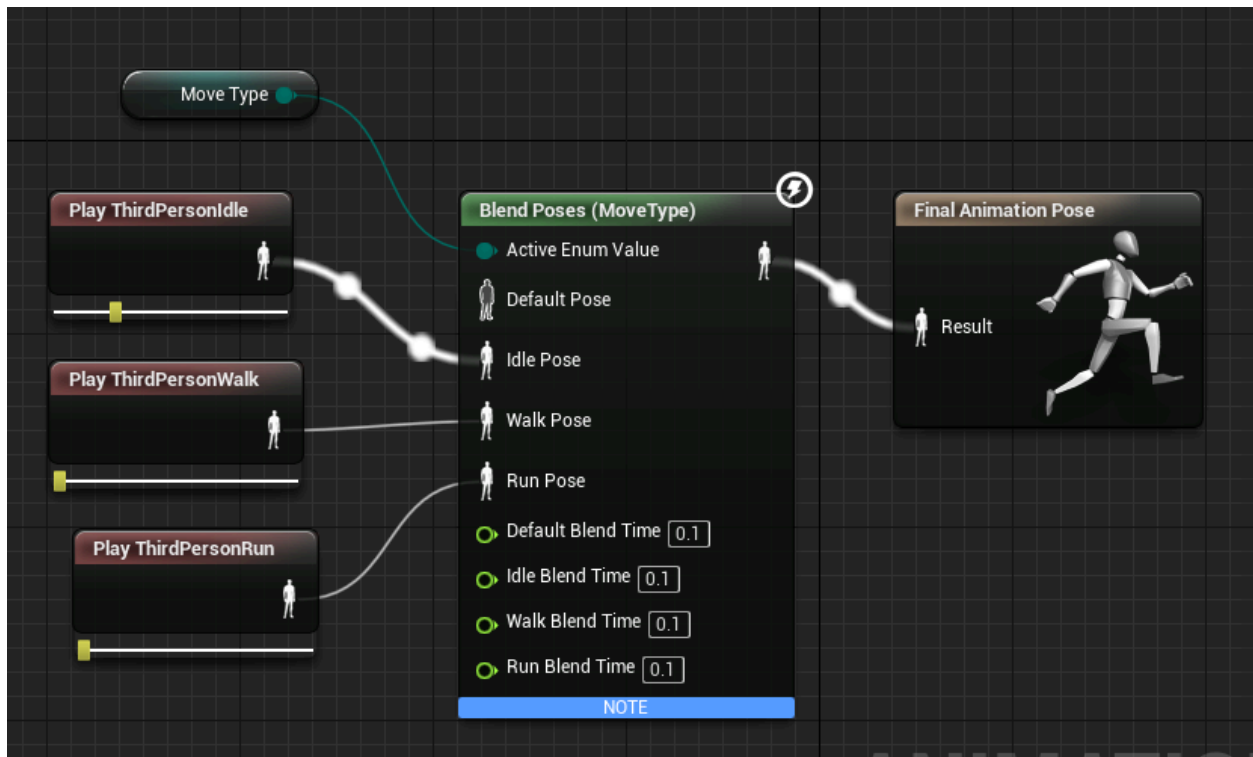
5. On the **Anim Graph**, add the **ThirdPersonIdle**, **ThirdPersonWalk** and **ThirdPersonRun** animations and a **Blend Poses by Move Type** node.



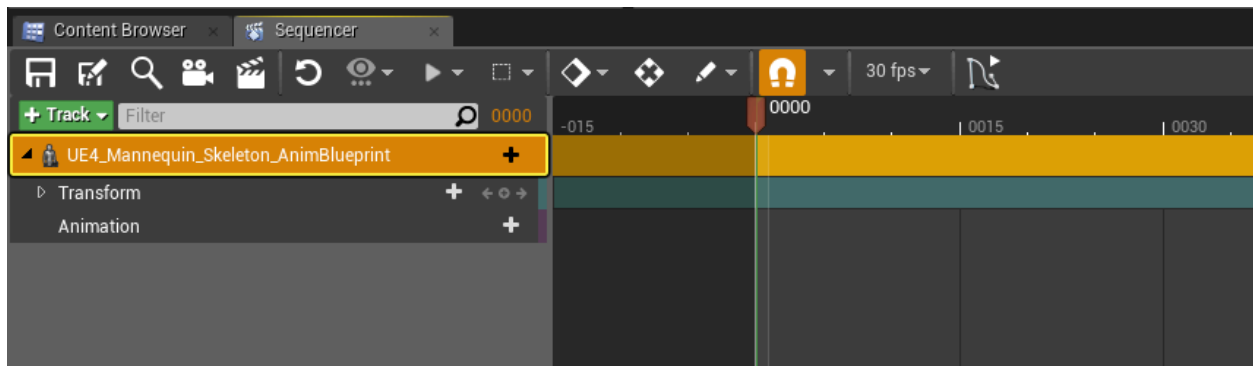
6. Right-click on the **Blend Poses** node then add a pin for **Idle**, **Walk** and **Run**.



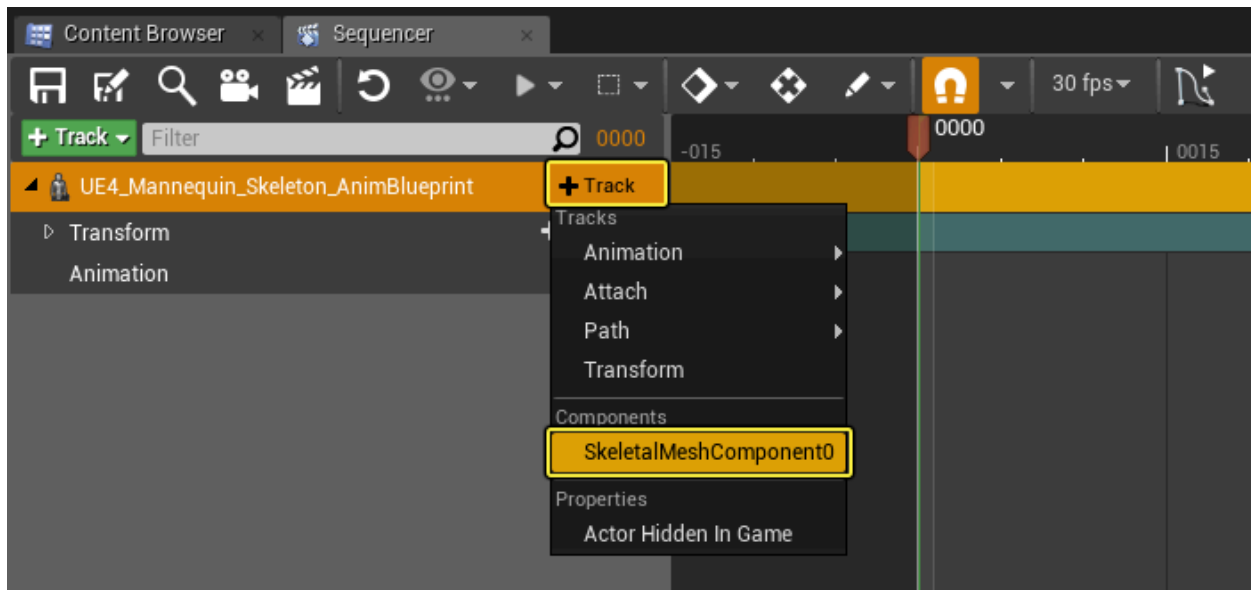
7. Add the **Move Type** variable to the graph and connect each of the nodes to the **Final Animation Pose** as shown below.



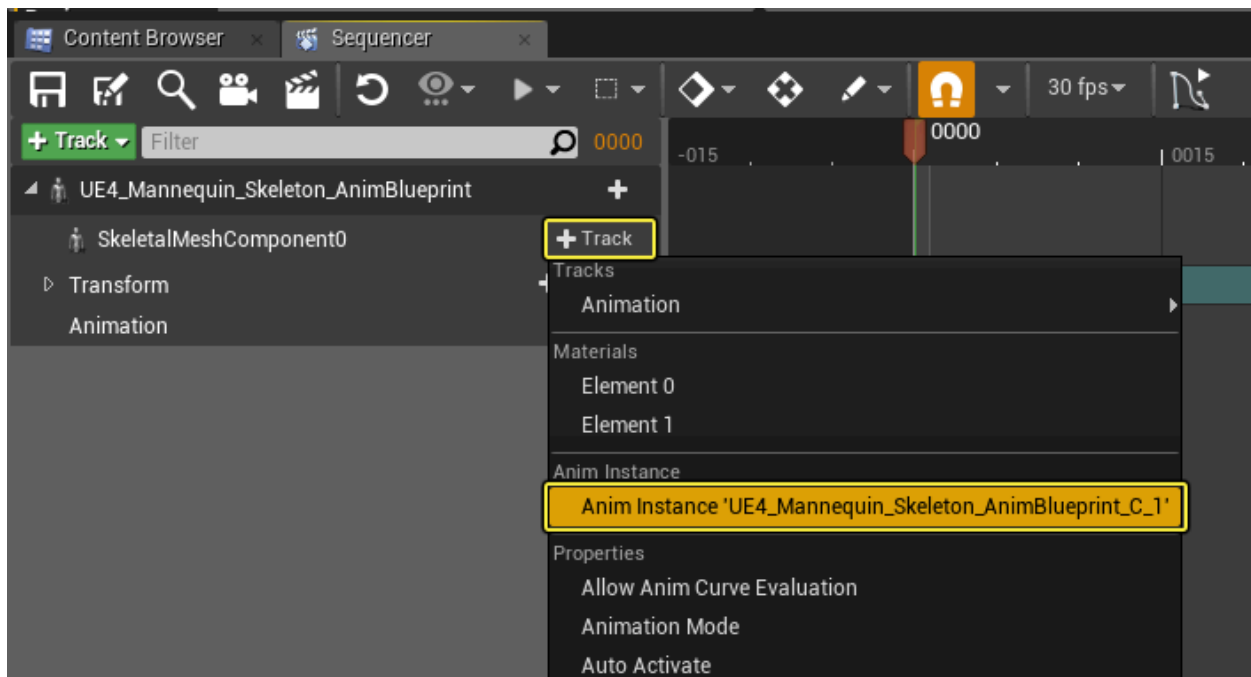
8. Drag the **Animation Blueprint** into the Level, then create a new **Level Sequence** (give it any name) and add the Anim Blueprint to the Sequence.



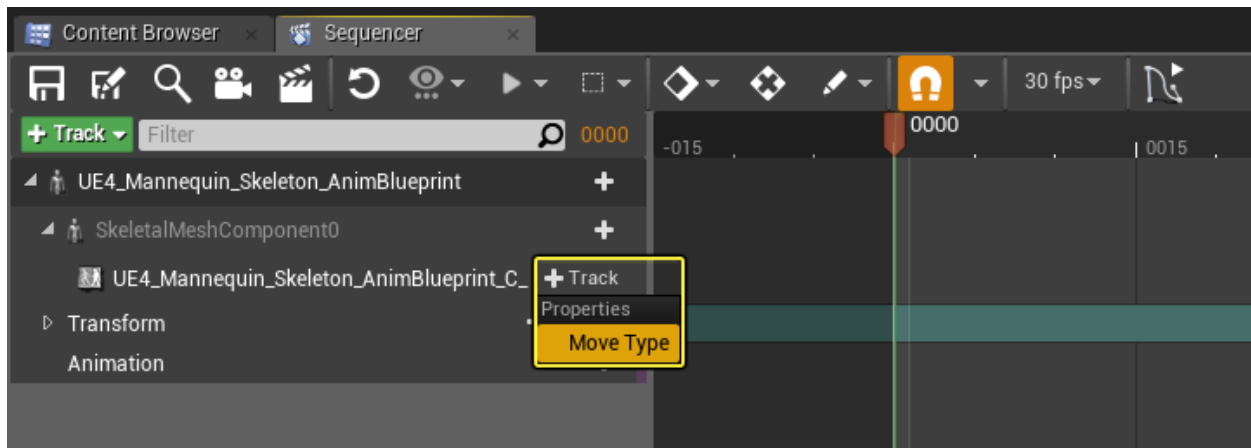
9. Click the **+ Track** button on the Anim Blueprint and add an **SkeletalMeshComponent0** track.



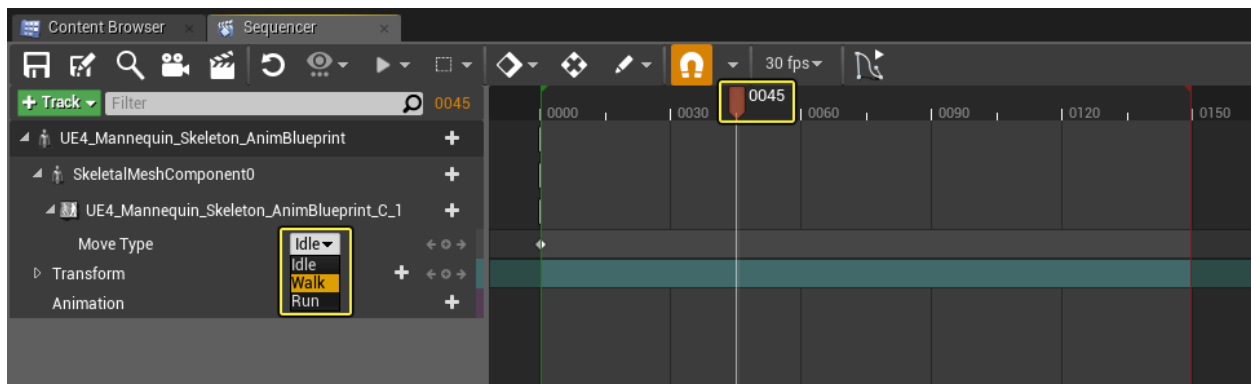
10. Click the **+ Track** button on the SkeletalMeshComponent and add the **Anim Instance** track.



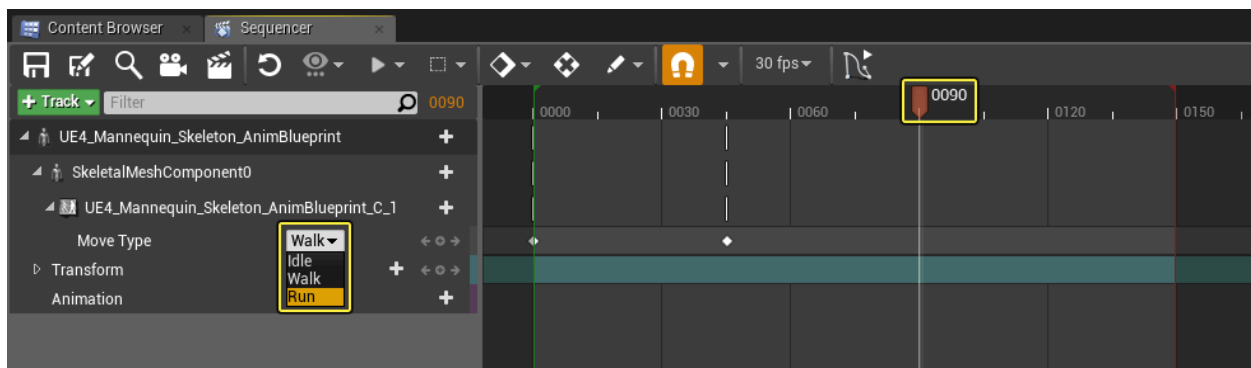
11. Click the **+ Track** button on the Anim Instance and add the **Move Type** property.



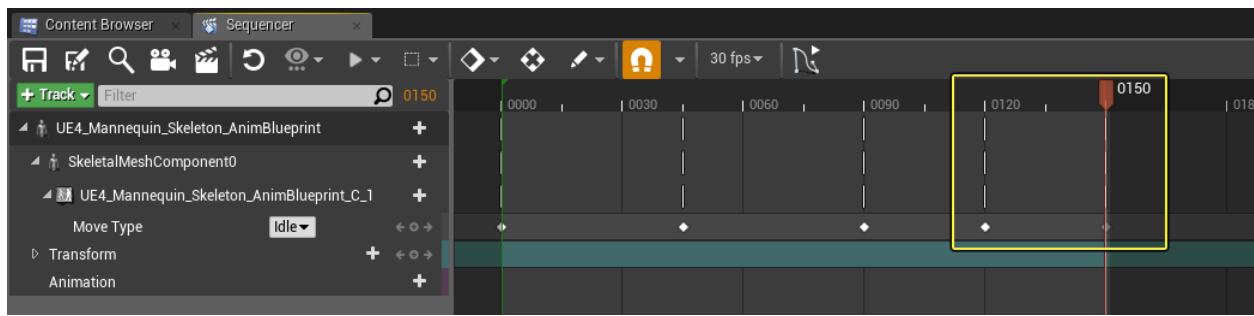
12. Scrub the Timeline to frame **45** and change the **Move Type** drop-down to **Walk**, adding a key.



13. Scrub the Timeline to frame **90** and change the **Move Type** drop-down to **Run**, adding another key.



14. Add a key for **Move Type** at frame **120** set to **Walk** and another key at frame **150** set to **Idle**.



15. Set the Level Sequence to **Auto Play** in the **Details** panel, then click the **Play** or **Simulate** button to play/simulate in the Editor.

End Result

When you Play or Simulate, the Level Sequence will play back and change the character's state based on the key framed property **Move Type** defined in the Sequence. Animating variable properties is useful in the event that you have characters that have Animation Blueprints that drive their animation logic, however you want to control what animation the character enters through a Sequence.

