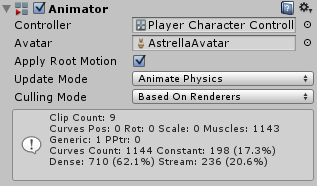
ANIMATOR AND ANIMATION

Use the Animator component to assign animation to a GameObject in your Scene. The Animator component requires a reference to an Animator Controller which defines which animation clips

to use, and controls when and how to blend and transition between them.

If the GameObject is a humanoid character with an Avatar definition, the Avatar should also be assigned in this component, as seen here:



| **Controller** | | The animator controller attached to this character. |
| --- | --- | --- |
| **Avatar** | | The avatar for this character. (If the Animator is being used to animate a humanoid character) |
| **Apply Root Motion** | | Select whether to control the character’s position and rotation from the animation itself or from script. |
| **Update Mode** | | This allows you to select when the Animator updates, and which timescale it should use. |
|  | **Normal** | The Animator is updated in-sync with the Update call, and the animator’s speed matches the current timescale. If the timescale is slowed, animations will slow down to match. |
|  | **Animate Physics** | The animator is updated in-sync with the FixedUpdate call (i.e. in lock-step with the physics system). You should use this mode if you are animating the motion of objects with physics interactions, such as characters which can push **rigidbody**  objects around. |
|  | **Unscaled Time** | The animator is updated in-sync with the Update call, but the animator’s speed ignores the current timescale and animates at 100% speed regardless. This is useful for animating a GUI system at normal speed while using modified timescales for special effects or to pause gameplay. |
| **Culling Mode** | | Culling mode you can choose for animations. |
|  | **Always Animate** | Always animate, don’t do culling even when offscreen. |
|  | **Cull Update Transforms** | Retarget, IK and write of Transforms are disabled when renderers are not visible. |
|  | **Cull Completely** | Animation is completely disabled when renderers are not visible. |

# **Animator Controller:**

An Animator Controller allows you to arrange and maintain a set of Animation Clips and associated Animation Transitions for a character or object. In most cases it is normal to have multiple animations and switch between them when certain game conditions occur. For example, you could switch from a walk Animation Clip to a jump Animation Clip whenever the spacebar is pressed. However even if you only have a single Animation Clip you still need to place it into an Animator Controller to use it on a GameObject

The Animator Controller has references to the Animation clips used within it, and manages the various Animation Clips and the Transitions between them using a State Machine, which could be thought of as a flow-chart of Animation Clips and Transitions, or a simple program written in a visual programming language within Unity.

