Animation Debugging and Optimization

Explore documentation about Unreal Engine's tools and techniques you can use to debug and optimize your animation system.

Unreal Engine features a suite of debugging and optimizations tools and techniques you can use to streamline your project's animation system, to increase performance and reduce file sizes. In the following document you can read about tools and features you can use to polish and refine your animation system in Unreal Engine.

Debugging Tools

Unreal Engine features a few debugging tools you can use to analyze your animation system in a controlled environment to make adjustments and find solutions to problems.

Rewind Debugger

With the [Rewind Debugger](#) you can record segments of your project's **Play In Editor** (**PIE**) gameplay, then using the visual timeline-based interface, scrub through the recording in real time to observe transition behaviors, variable values, pose blends and more. Recorded gameplay provides a more stable working than traditional simulation, and can preserve incorrect animation behavior for easier collaboration and debugging.

For more information on debugging animation systems with the **Rewind Debugger**, see the following documentation:


Rewind Debugger

With the Rewind Debugger you can record real-time segments of projects and preserve the data for debugging workflows.

Animation Insights

You can use the [Animation Insights](#) [Plugin](#) to profile your project's animation system, to see a visual graph of all operations over time. You can use this graph to determine what animation processes are being evaluated, how much performance budget they use, and when, to make informed optimization choices to achieve your project's desired performance quality.

For more information about profiling animation systems with **Animation Insights**, see the following documentation:
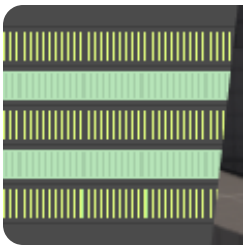

Animation Insights

Use the Animation Insights to observe and profile your project's gameplay and animation performance during runtime.

Pose Watch

You can use [Pose Watching](#) to toggle dynamic visual debug renders in the viewport during project simulation of individual animation data sources, when working with complex Animation Blueprints and layered animation systems. When rendering individual animation sources you can visually isolate each node's or layer's influence on the final output pose, to determine the source of bugs or irregular animation behavior in your animation system.

For more information about debugging animation systems with **Pose Watching**, refer to the following documentation:
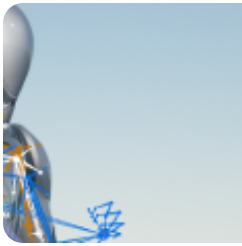
Animation Shortcuts and Tips

Workflow shortcuts and tips for animators and animation programmers in Unreal Engine.

Animation Optimization

You can use [Animation Optimization](#) techniques and features to improve your animation system's performance and quality, as well as reduce file sizes.

For more information about **Animation Optimization** in Unreal Engine, see the following documentation:
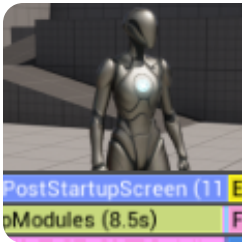
Animation Optimization

Use a variety of methods and techniques to optimize Animation Blueprint's performance and stability.

Animation Budget Allocator

The [Animation Budget Allocator](#) is a [Plugin](#) for Unreal Engine, that you can use to throttle animation evaluation and quality, on multiple characters to reduce the performance cost of your projects entire animation system.

For more information about optimizing animation systems with the **Animation Budget Allocator**, see the following documentation:
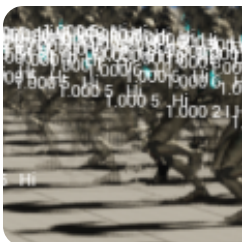
Animation Budget Allocator

System for constraining the time taken for animation data by dynamically throttling Skeletal Mesh Component ticking.