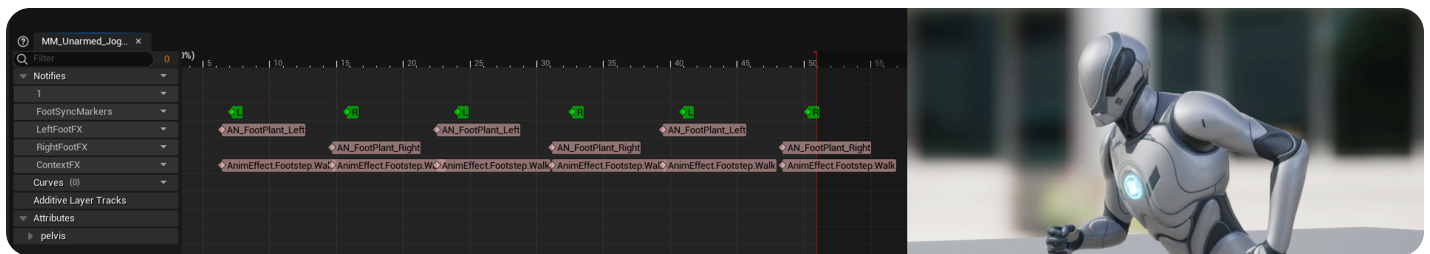


Developer  
/ Documentation  
/ Unreal Engine ▾  
/ Unreal Engine 5.4 Documentation  
/ Animating Characters and Objects  
/ Skeletal Mesh Animation System  
/ Animation Assets and Features  
/ Animation Sequences  
/ Animation Notifies

# Animation Notifies

Send and receive events synchronized to Animation Sequences using Animation Notifies.



**Animation Notifications** (Animation Notifies or just Notifies) provide a way for you to create repeatable events synchronized to **Animation Sequences**. These events can be sounds (such as footsteps for walk or run animations), spawning particles, and other types. Animation Notifies have any number of different uses, and the system can be extended with custom types.

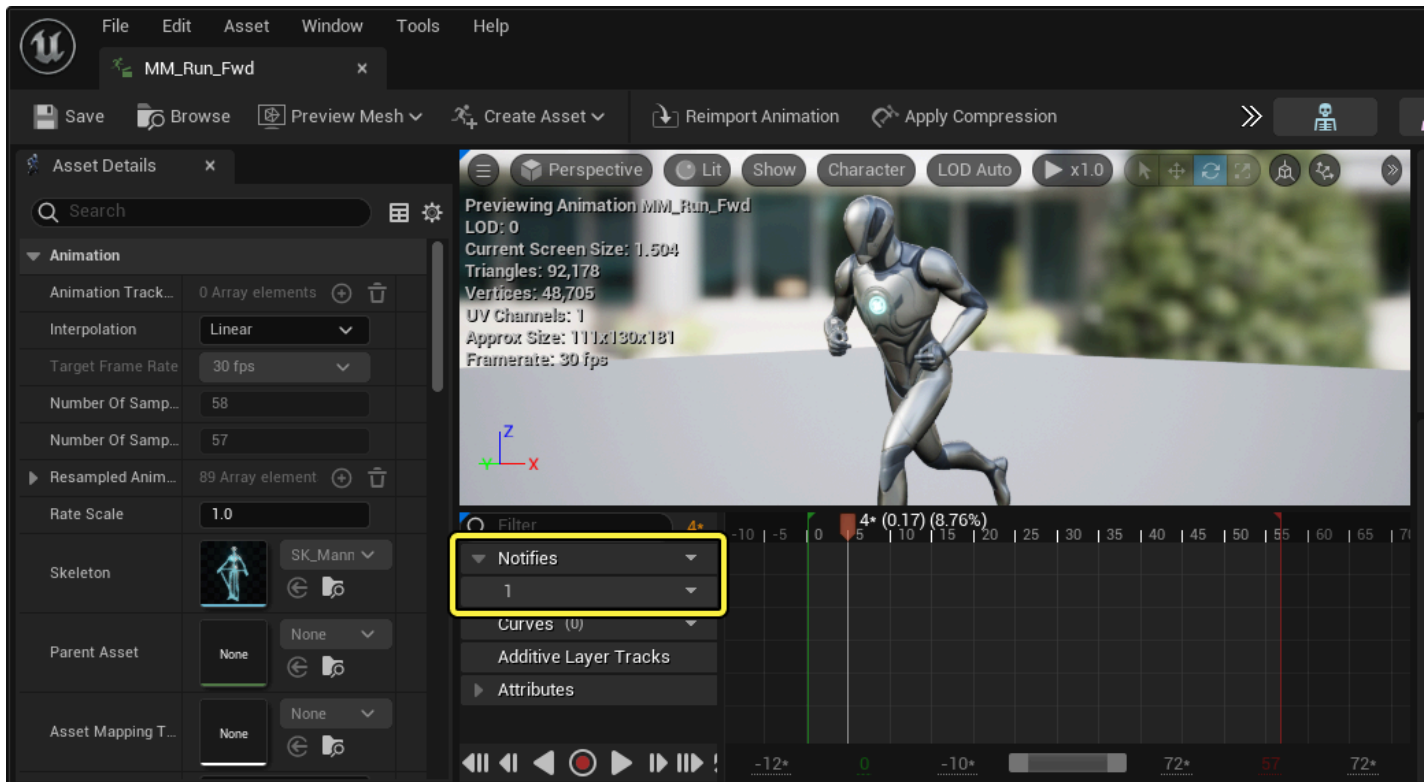
This document provides an overview of the different types of Animation Notifies, how to create them, and how to use them in a variety of ways.

## Prerequisites

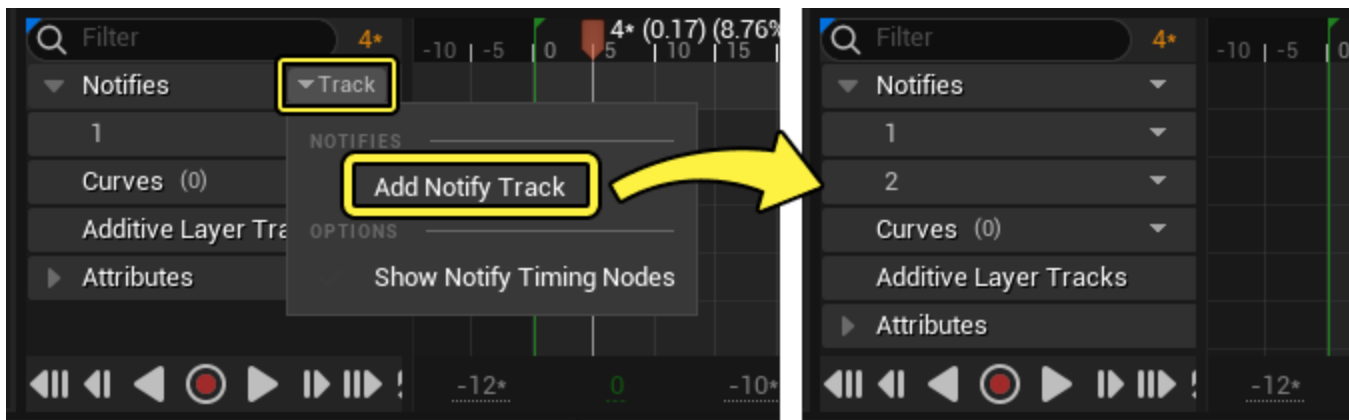
- Animation Notifies are created within [Animation Sequences](#), therefore your project needs a **Skeletal Mesh** and an **Animation Sequence**.

## Getting Started

Animation Notifications are commonly accessed and created within Animation Sequences. To get started, open an **Animation Sequence Asset**, and locate the **Notifications** track in the timeline.

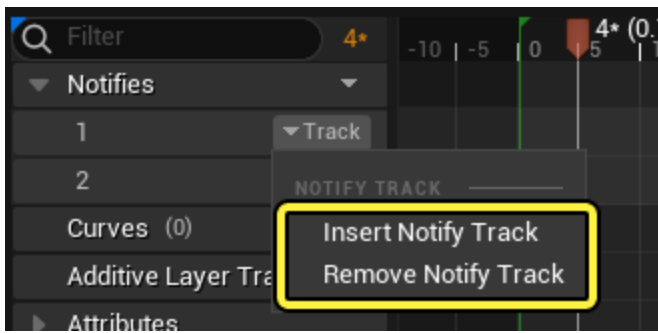


The Notifications track itself is a parent group for individual child tracks below, which contain the actual notify keyframes. By default, a single child track should exist (named 1). If no child tracks exist, or if you want to add extra notify tracks, click **Add Track (+) > Add Notify Track** on the Notifications track.

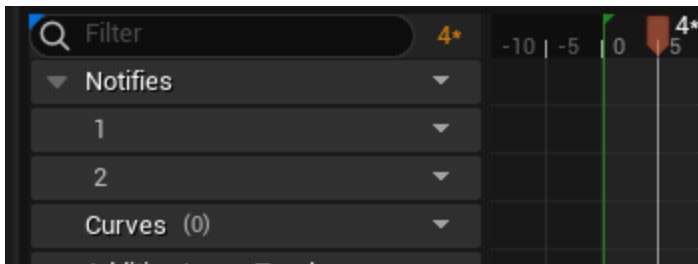


You can also insert or remove notify tracks by clicking the **Add Track (+)** dropdown menu on the child track itself and selecting either **Insert Notify Track** or **Remove Notify Track**.



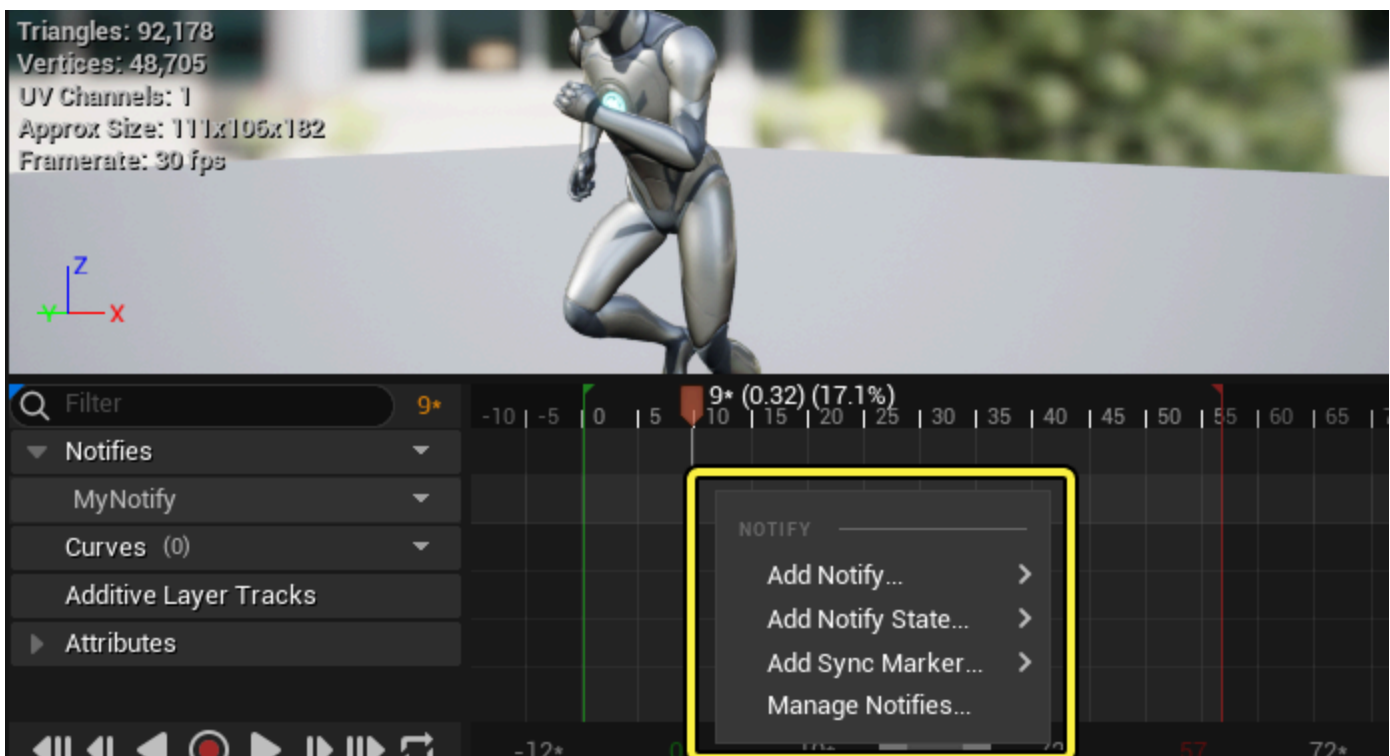


To rename a Notify track, triple-click on the track text to enable text editing.



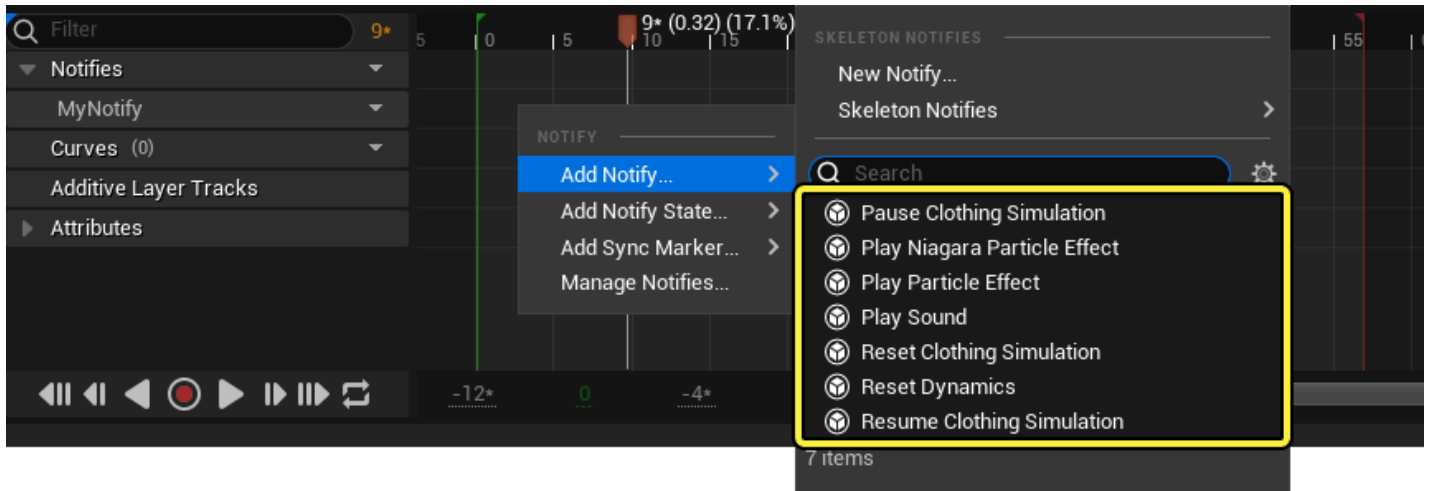
## Animation Notify Types

There are several types of Animation Notices you can create, which are viewed when creating Notices by right-clicking in the Notify track's timeline region.

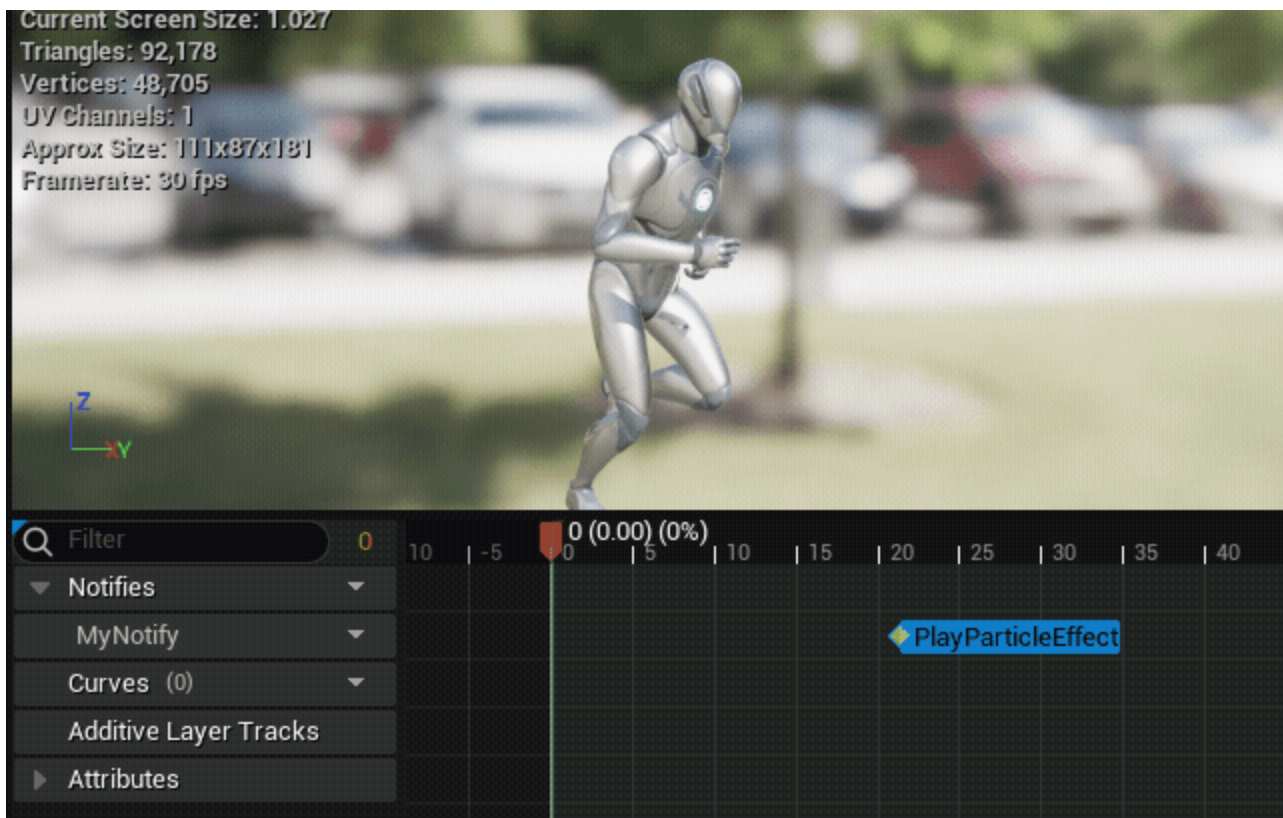


# Notify

The most basic kind of Animation Notify you can create is simply called a **Notify**, which causes different pre-made events to be triggered at a specified time. The following Notifies can be found when viewing the **Add Notify...** menu. Selecting one will create the Notify keyframe at your cursor position.



Notify keyframes can be edited by dragging them in the timeline. You can also hold **Shift**, which will move the **Playhead** in sync with the Notify, making it easier to align the Notify with a specific time in the animation.

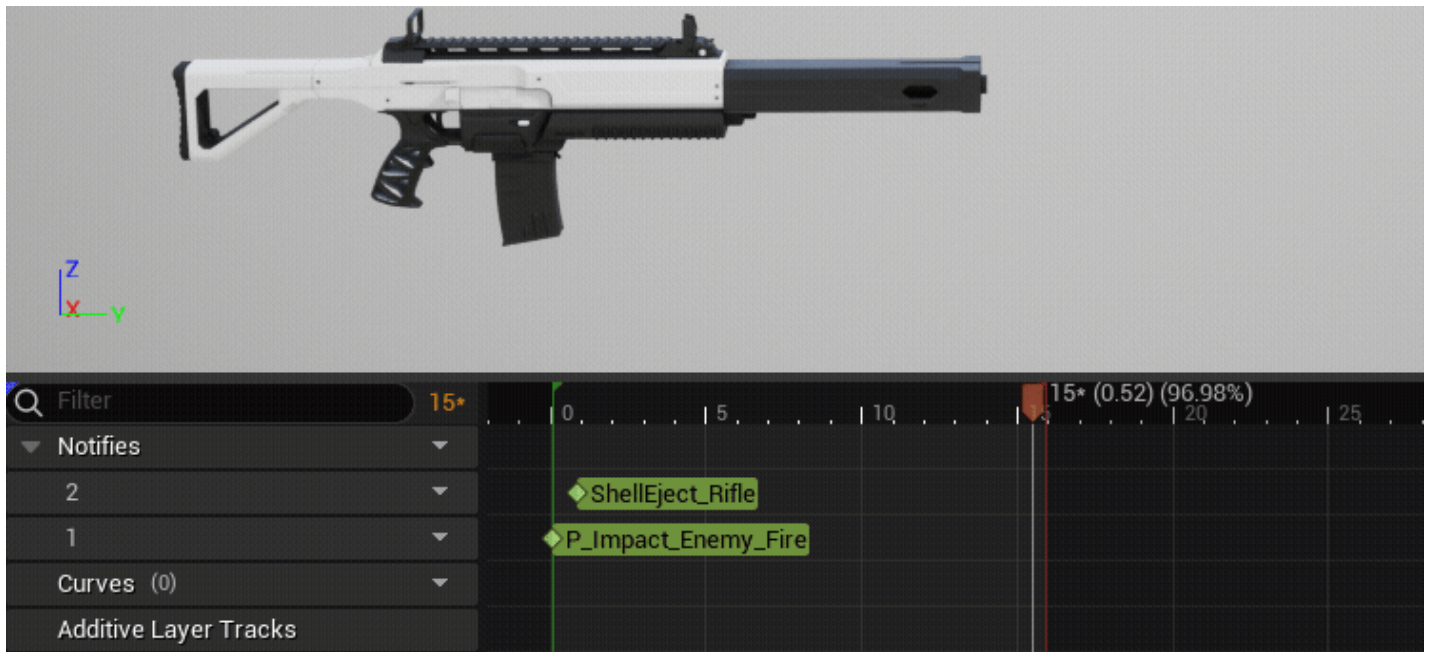


# Particle Effect

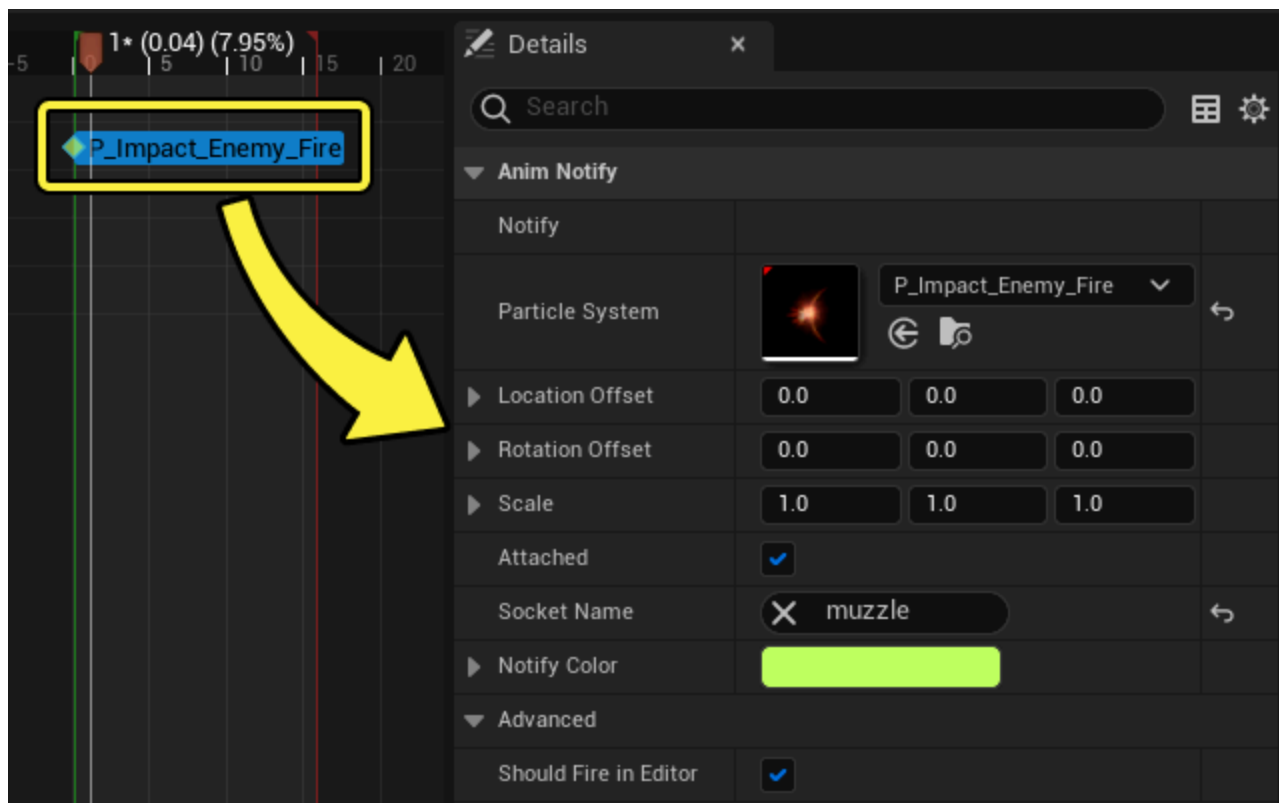
**Particle Effect** Notifies are used to spawn and play non-looping particle systems. You can select either:

- **Play Niagara Particle Effect**, which spawns [Niagara particles](#).
- **Play Particle Effect**, which spawns [legacy Cascade particles](#).

This Notify is useful for creating repeatable particle effects, such as an effect when a character lands from jumping, shooting effects for a Skeletal Mesh weapon, or other similar effects.



When selecting the Particle Notify keyframe, the following relevant properties are located in the **Details** panel:



Name	Description
<b>Particle System / Niagara System</b>	The Cascade or Niagara system to spawn for this Notify.
<b>Location / Rotation / Scale Offset</b>	An array of location, rotation, and scale transform properties to offset the system from its spawn point.
<b>Attached</b>	When spawned on a <a href="#">Socket</a> , enabling this causes the particle system to follow the Socket for its full duration. Disabling this still spawns the particle system at the Socket location, but the system does not continue to follow the Socket.
<b>Socket Name</b>	You can specify a bone or <a href="#">Socket</a> here, which spawns the particle system at its location. If nothing is specified here, then the system spawns at the root of the object.

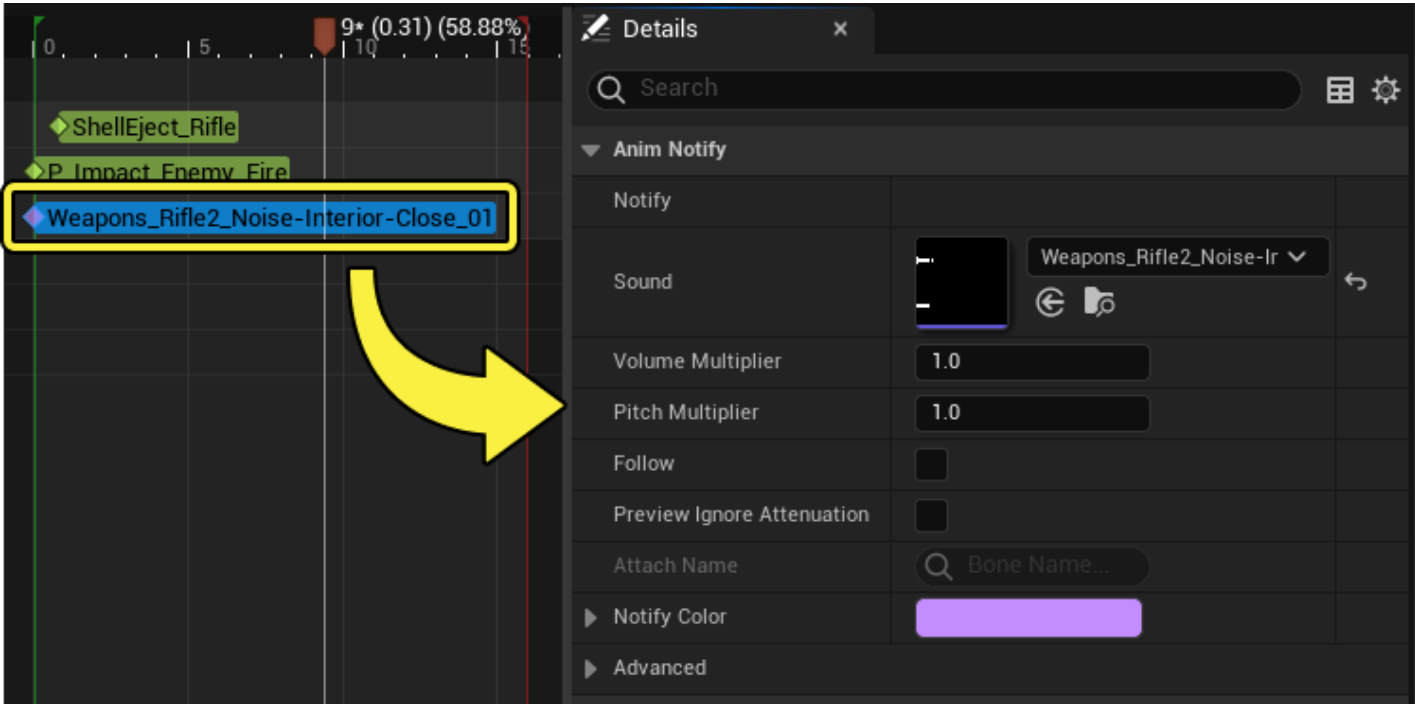
## Sound

**Sound** Notifies are used to play sound effects from imported Sound Waves, [Sound Cues](#), and [MetaSounds](#). These Notifies are useful for synchronizing foley-type sounds to animations,



such as footsteps, cloth movements, and similar repeatable sounds.

When selecting the Sound Notify keyframe, the following relevant properties are located in the **Details** panel:



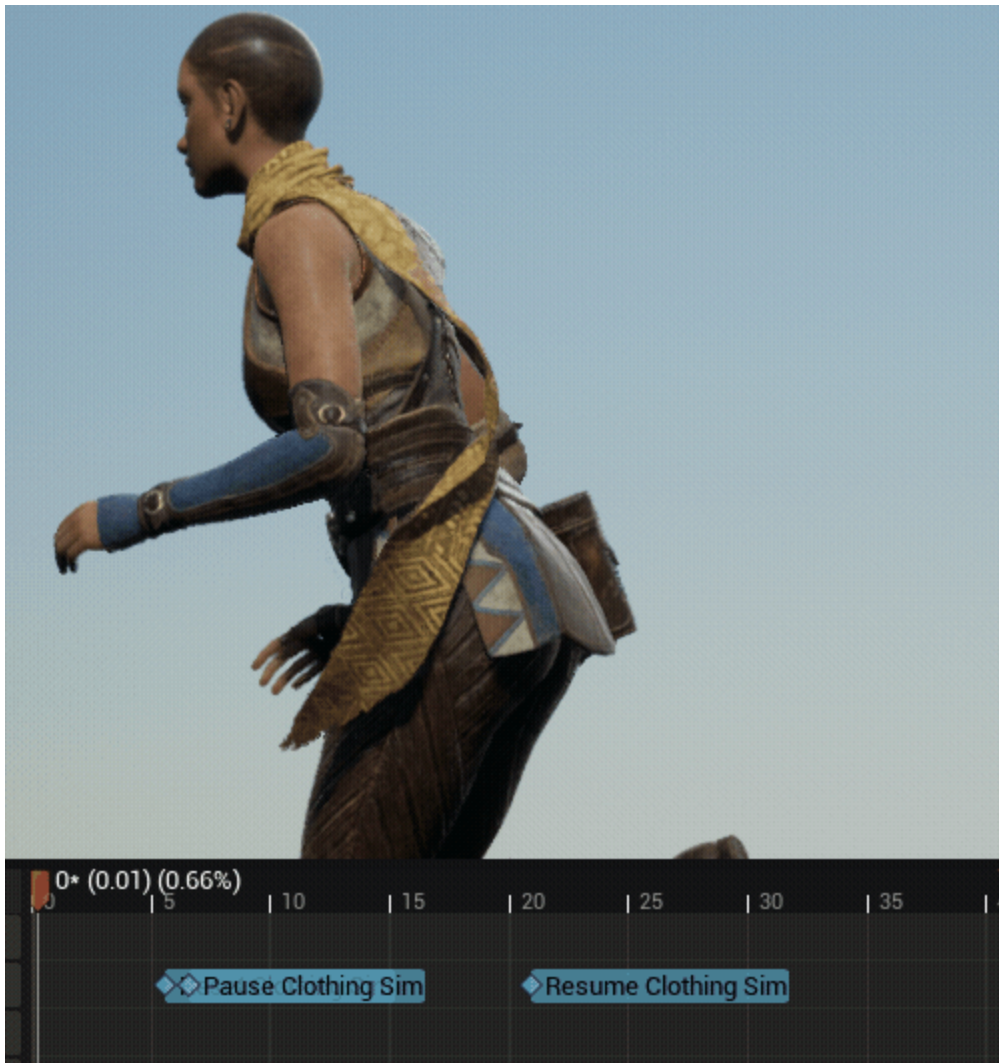
Name	Description
Sound	The sound asset to play for this Notify. You can select the following assets: <ul style="list-style-type: none"><li>• Sound Wave</li><li>• Sound Cue</li><li>• MetaSound</li><li>• Sound Bus</li></ul>
Volume Multiplier	A multiplier to increase or decrease the volume of the playing sound.
Pitch Multiplier	A multiplier to increase or decrease the pitch of the playing sound.
Follow	If enabled, the sound effect source follows the animated mesh as it moves. If disabled, the sound remains behind at the location it spawned.

Name	Description
<b>Preview Ignore Attenuation</b>	If enabled, <a href="#">sound attenuation</a> is disabled when playing the sound. This setting is editor-only and does not impact normal sound attenuation operation when playing or simulating the game.
<b>Attach Name</b>	If <b>Follow</b> is enabled, you can specify a bone or Socket to attach to. If nothing is specified here, then the sound spawns at the root of the object.

## Clothing Simulation

**Clothing Simulation** Notifies are used to pause, resume, and reset cloth simulations on a character. These Notifies are useful if you are teleporting the character during the animation and need to pause or reset cloth evaluation. They can also be useful if your animations are causing the cloth simulation to overreact.





You can select the following Notifies:

- **Pause Clothing Simulation**, which pauses the simulation.
- **Resume Clothing Simulation**, which restores the simulation after having been paused.
- **Reset Clothing Simulation**, which initializes the simulation back to its default reference pose.

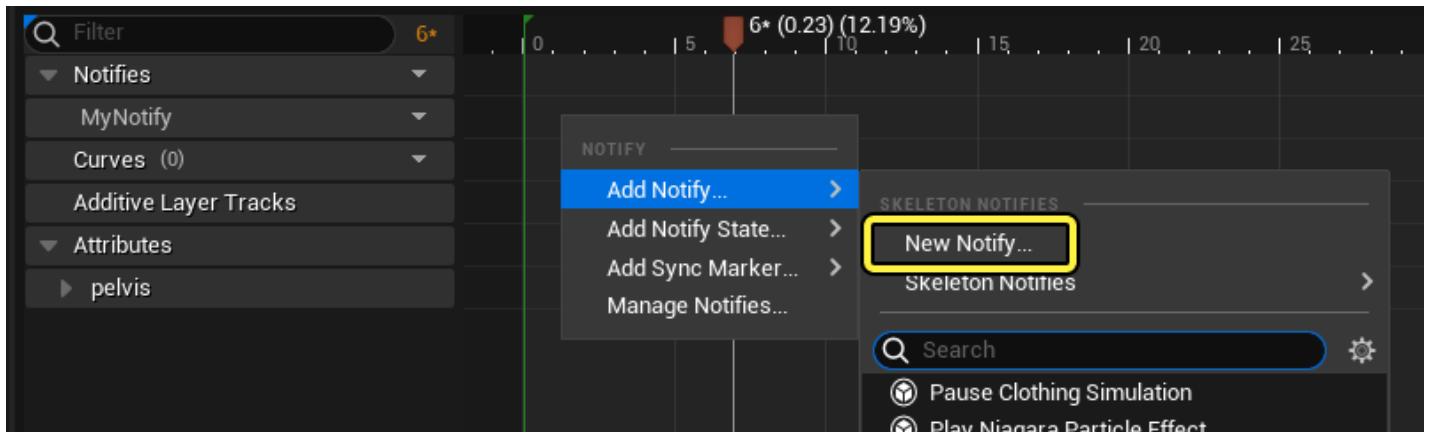
## Reset Dynamics

The **Reset Dynamics** Notify can be used to restore any [AnimDynamics](#) used on this Skeletal Mesh back to their original state. This Notify is useful if your animations are causing the AnimDynamic simulation to overreact or enter broken-looking states.

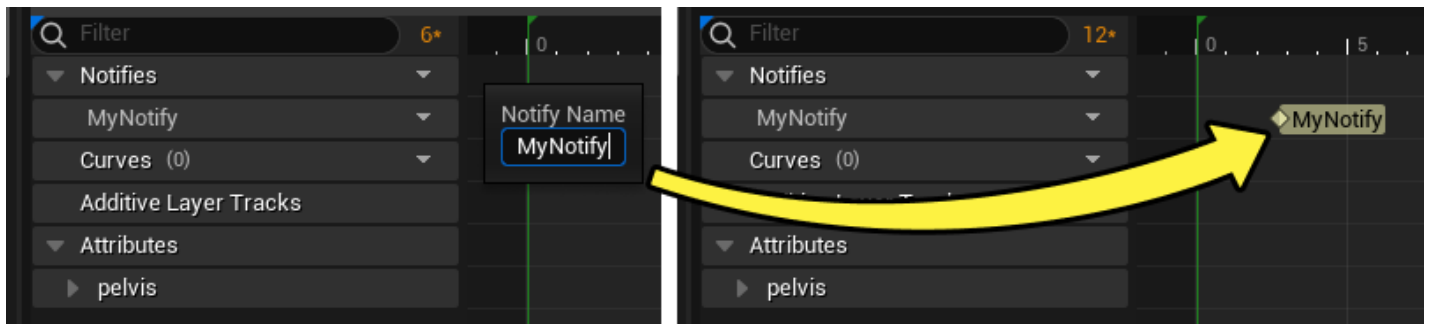
## Skeleton Notify

**Skeleton** Notifies are custom Notifies saved on the [Skeleton Asset](#), and then used as events within your [Animation Blueprint](#). These Notifies are useful for creating any arbitrary Blueprint logic you want a Notify to execute.

To create a Skeleton Notify, right-click on a Notify track, then select **Add Notify > New Notify...**

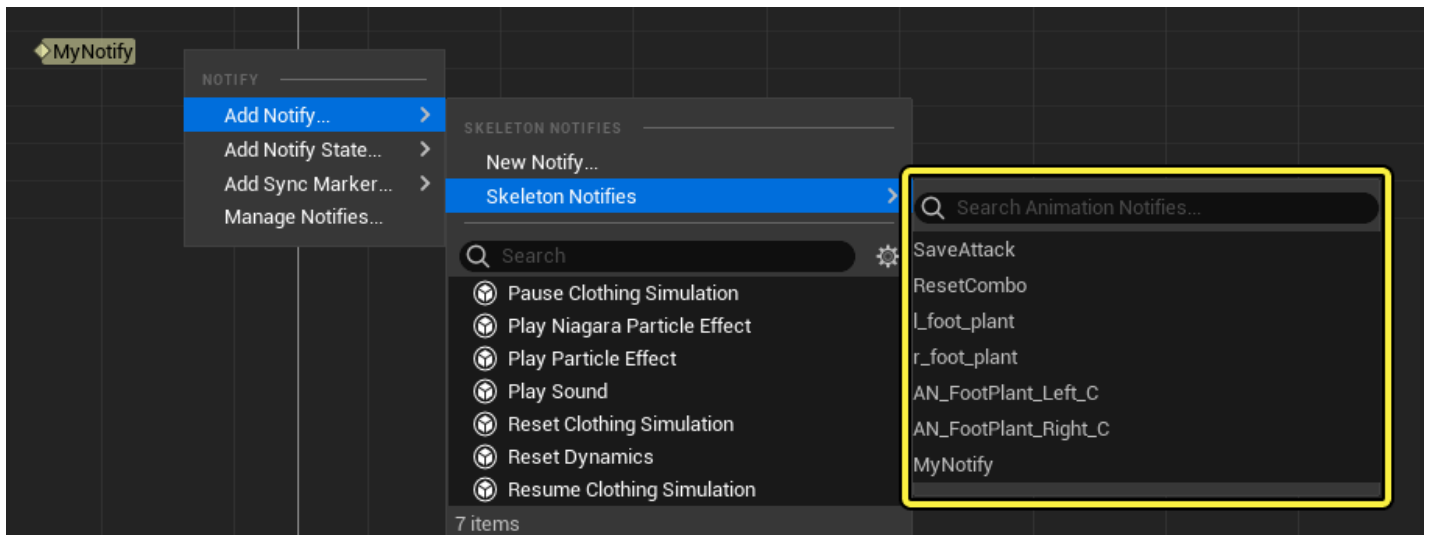


You will be prompted to enter a name for the newly created Notify. Name it and press **Enter** to create the Skeleton Notify.

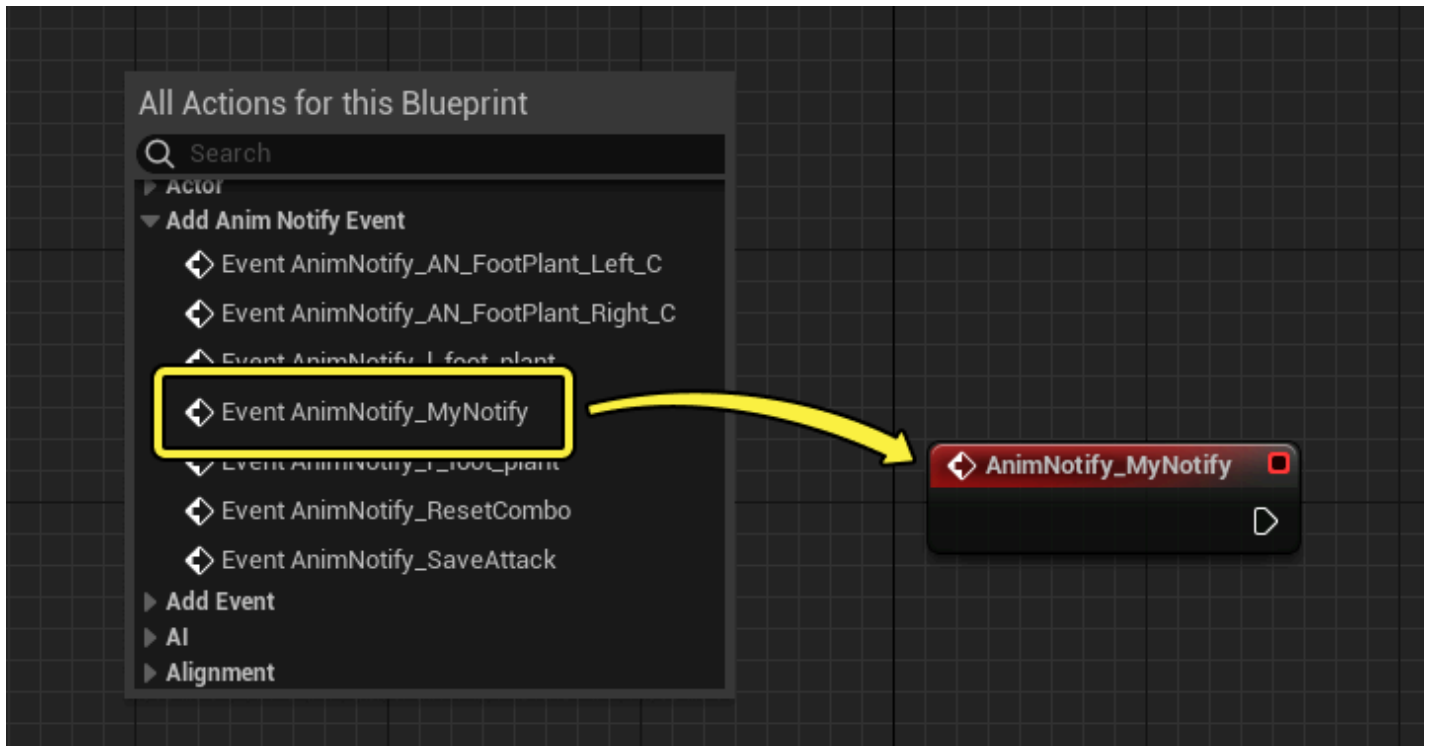


Skeleton Notifies are stored on the **Skeleton Asset**. Therefore, when creating them, you are also editing the [Skeleton](#), which requires you to save it.

You can also add currently-existing Skeleton Notifies to the timeline by right-clicking a Notify track, then selecting one from the **Add Notify > Skeleton Notifies** menu.



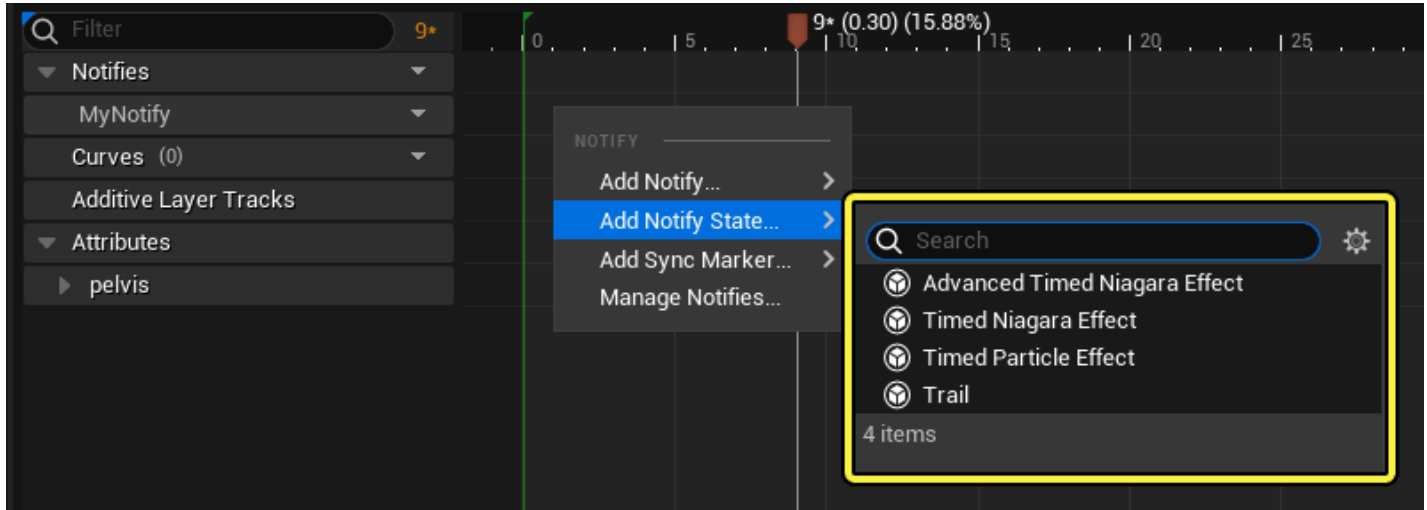
Skeleton Notices are used as **Notify Events** within Animation Blueprints, either in the [Event Graph](#) or [Transition Graph](#). To add a **Skeleton Notify Event**, right-click in the **Event** or **Transition Graph** of your **Animation Blueprint** and select the **Notify** from the **Add Anim Notify Event** menu. This adds the event node to the Graph which is executed when the Notify is called from the animation it resides in.



## Notify State

**Notify States** work similar to standard Notifies, however they operate over a duration, rather than a single event. Because of this, they provide three distinct events: a **start**, an **update**, and an **end**. These events can be accessed when creating [Notify State child classes](#).

Also like standard Notifies, they provide several different pre-made classes to choose from. You can find the following notifies when viewing the **Add Notify State...** menu. Selecting one creates the Notify keyframes at your cursor position.



Notify State keyframes are edited similar to normal Notifies by dragging them in the timeline. You can either drag the start or end to edit the range, or drag the whole Notify.

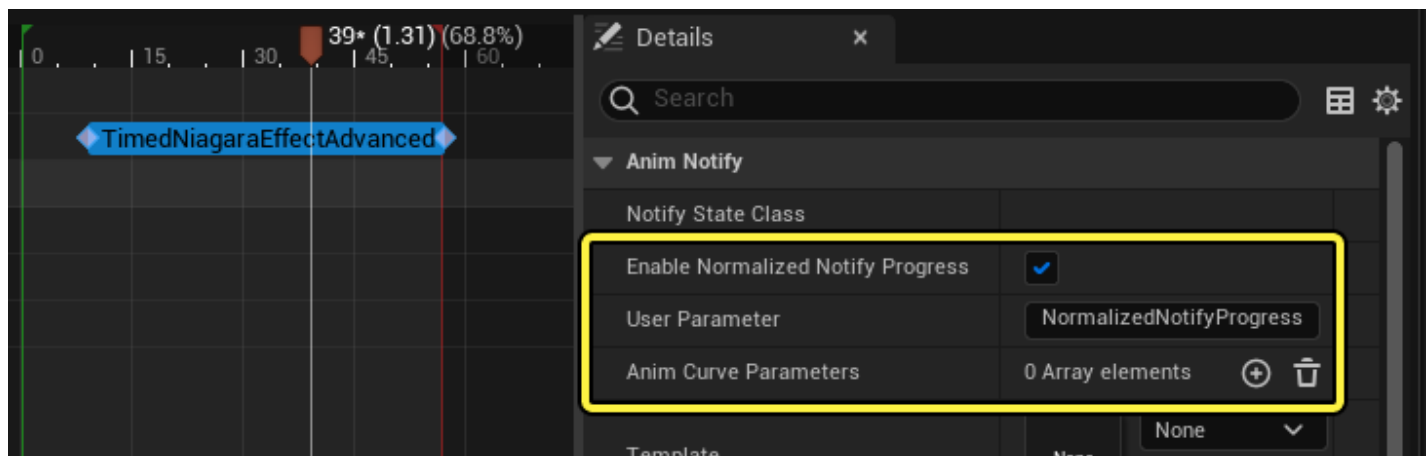


## Timed Particle Effect

Similar to [Particle Effect](#) Notifies, **Timed Particle / Niagara Effects** Notifies are used to play particles during an animation. The main difference is that you can also define the duration of the particle, if the particle is set to loop. You can select the following types of Particle Notifies:

- **Timed Particle Effect**, which spawns [legacy Cascade particles](#).
- **Timed Niagara Effect**, which spawns [Niagara particles](#).
- **Advanced Timed Niagara Effect**, which spawns Niagara particles with extra options for driving parameters within them.

Timed Particle Notifies share the same details with normal Particle Effect Notifies, however Advanced Timed Niagara Notifies contain the following additional properties:



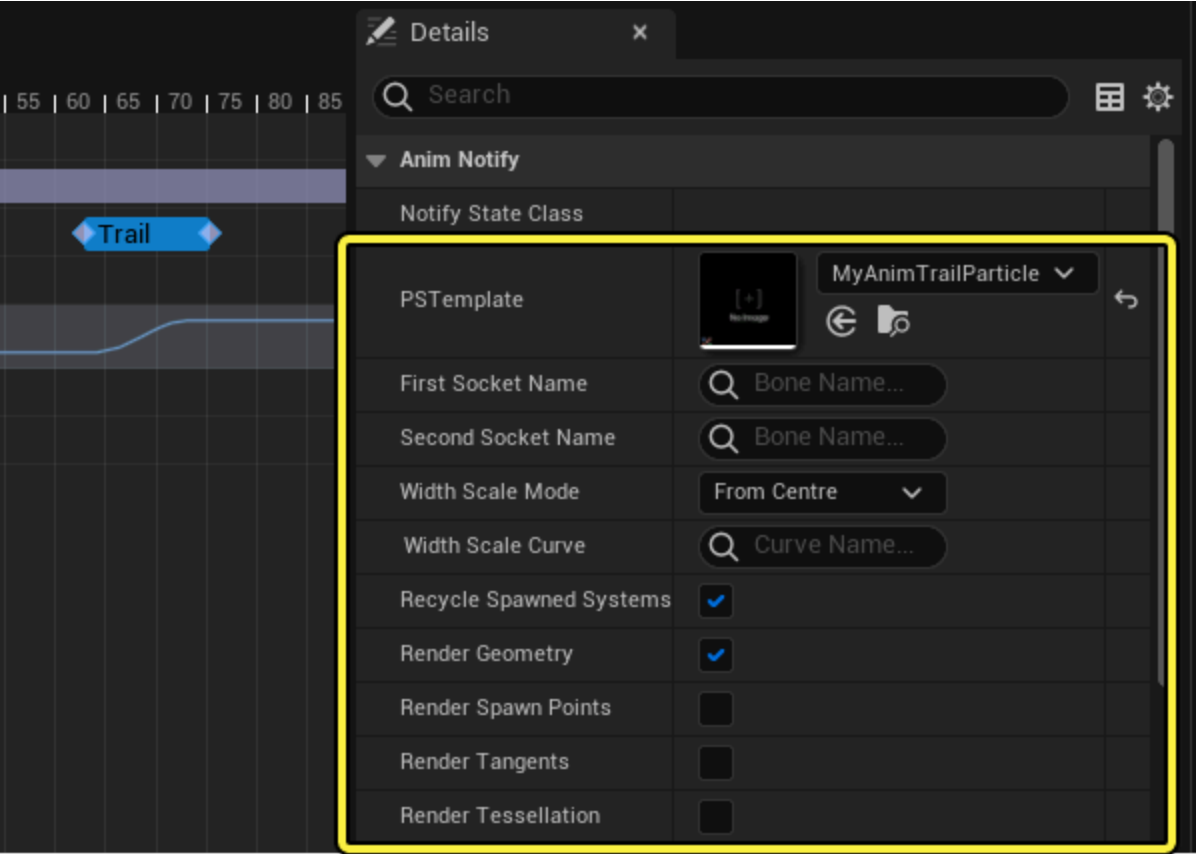
Name	Description
<b>Enable Normalized Notify Progress</b>	When enabled, causes the Notify state to output a normalized (0-1) value over the duration of the Notify to the Niagara <a href="#">User Parameter</a> specified in <b>User Parameter</b> . The parameter starts at <b>0</b> and interpolates to <b>1</b> upon the notify finishing. This property is useful if you want to link certain particle parameters to the lifetime of the notify.
<b>User Parameter</b>	The name of the Niagara User Parameter to drive if <b>Enable Normalized Notify Progress</b> is enabled.
<b>Anim Curve Parameters</b>	<p>An array where you can link <a href="#">Animation Curve</a> values to Niagara User Parameters. You can add items to the array by clicking <b>Add (+)</b>, then specify the following properties:</p> <ul style="list-style-type: none"> <li>• <b>Anim Curve Name</b>, which is the Animation Curve that drives the Niagara parameter.</li> <li>• <b>Niagara User Float</b>, which is the Niagara User Parameter controlled by the Animation Curve.</li> </ul>

## Trail

**Trail** Notifies are similar to Timed Particle Effects in that they spawn a particle system over the duration of the Notify. It differs because it is primarily used for [AnimTrail Cascade Particles](#), and contains additional properties for controlling trail attachment and properties.



Selecting the Trail Notify reveals the following properties in the **Details** panel.



Name

Description

<b>PSTemplate</b>	The AnimTrail Cascade Particle system to use.
-------------------	---

Name

Description

**First / Second Socket Name**

You can specify a separate bone or Socket here for each property, which are used to define the attach points for the AnimTrail. This also defines the trail's default width, based on the distance between these two attach points.

**Width Scale Mode**

If **Width Scale Curve** is used, this defines the relative scale point of the trail's width. You can select the following options:

- From Centre, which scales the trail relative to the **center point** between the two sockets.

1 



2 

- From First Socket, which scales the trail relative to the location of the **first Socket**.

1 

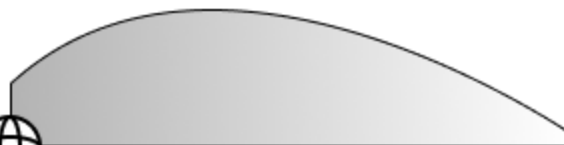


2 

- From Second Socket, which scales the trail relative to the location of the **second Socket**.

1 

2 

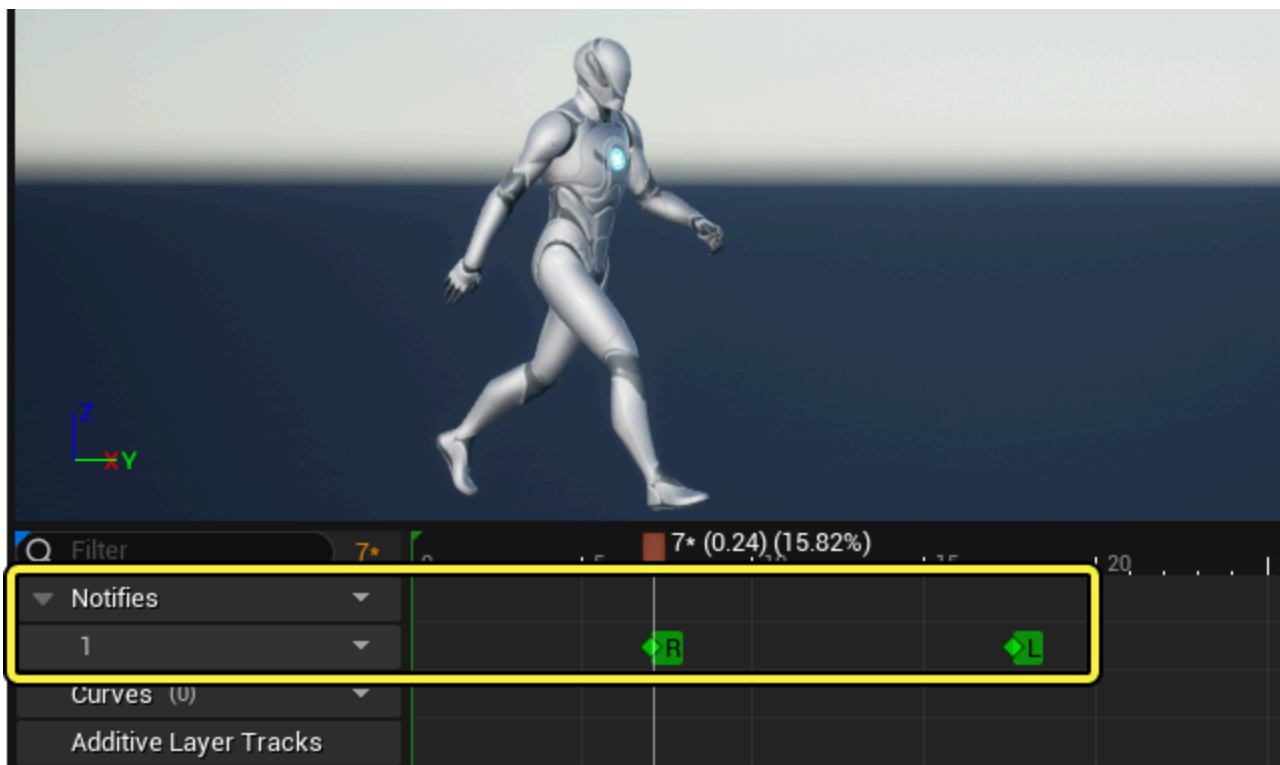




Name	Description
<b>Width Scale Curve</b>	You can optionally specify an <a href="#">Animation Curve</a> here, which can be used to animate the scale width of the trail during the animation. A value of <b>1.0</b> maintains the default width, higher numbers increase the width, and lower numbers decrease the width.
<b>Render Geometry</b>	Enables rendering of the main trail geometry, leave this enabled.
<b>Render Spawn Points</b>	Enables debug rendering of the primary spawn points for the trail.
<b>Render Tangents</b>	Enables debug rendering of the curve tangents.
<b>Render Tessellation</b>	Enables debug rendering of all vertices on the curve, showing how the curve is tessellated between the spawn points.

## Sync Marker

Sync Markers are Notifies used to inform [marker-based animation syncing](#), which is useful for blending animations together and keeping overall motion synchronized. Similar to [Skeleton Notifies](#), Sync Markers save to the Skeleton Asset, which requires you to save the Skeleton when adding them. You can then use the same Sync Markers repeatedly in any animation you want to synchronize while blending.



To learn more about using Sync Markers, refer to the [Sync Groups](#) page.

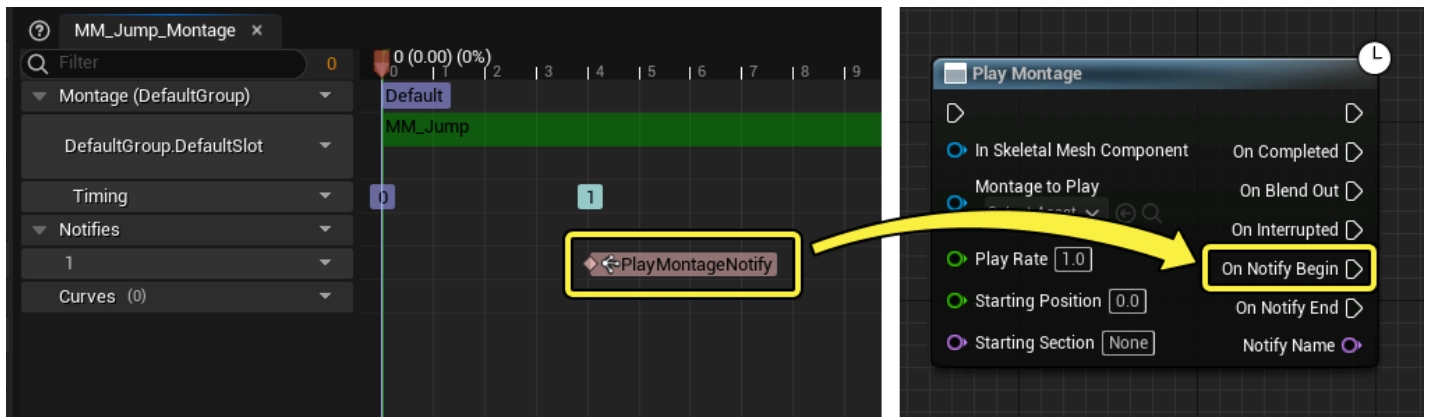
## Montage Notifies

When creating [Animation Montages](#), additional Animation Notify types are available to you to support montage-specific workflows. The following Notifies can be added in montages:

- **Montage Notify**, which can be added by right-clicking on the Notify timeline and selecting **Add Notify > Montage Notify**.
- **Montage Notify Window**, which can be added by right-clicking on the Notify timeline and selecting **Add Notify State > Montage Notify Window**.
- **Disable Root Motion**, which can be added by right-clicking on the Notify timeline and selecting **Add Notify State > Disable Root Motion**.

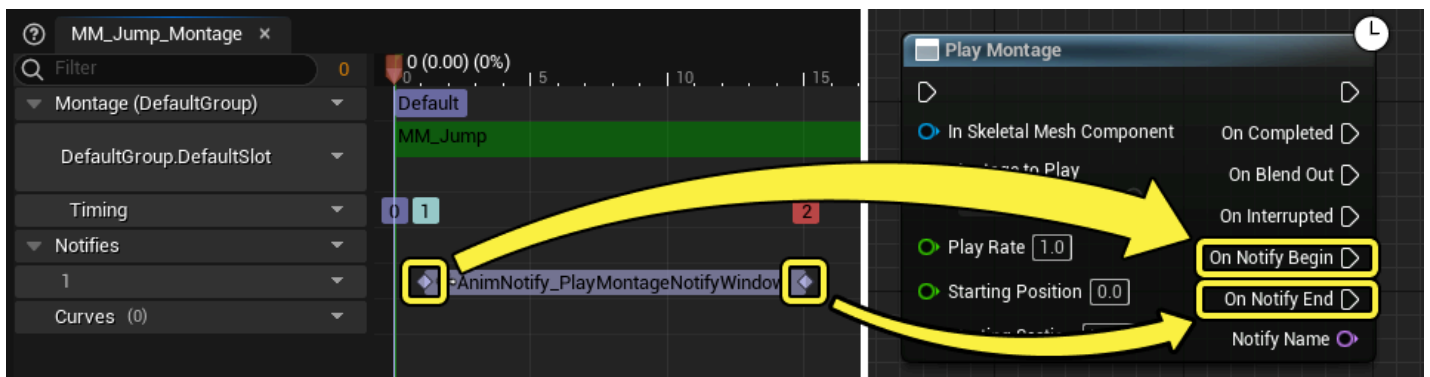
## Montage Notify

A **Montage Notify** operates similarly to standard Notifies in that it executes an event upon reaching the Notify keyframe. When reached, the **On Notify Begin** event is executed on the [Play Montage](#) Blueprint function. This requires playing the montage using the **Play Montage** node, instead of the **Play Anim Montage** node.



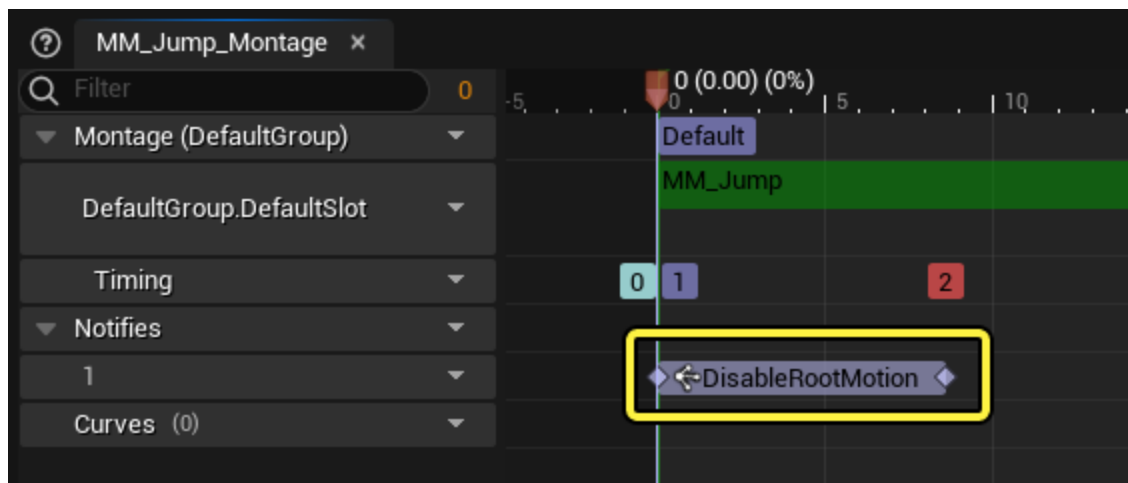
## Montage Notify Window

A **Montage Notify Window** operates similarly to Notify States, in that the Notify operates over a duration you define. This causes both **On Notify Begin** and **On Notify End** to execute when the start and end of the Notify are reached. This requires playing the montage using the **Play Montage** node, instead of the **Play Anim Montage** node.



## Disable Root Motion

A **Disable Root Motion** Notify operates similarly to Notify States, in that the Notify operates over a duration you define. This Notify suppresses root motion, allowing for user-input to control a character for its duration. This can be useful if you want to allow input control during parts of a montage, which locks out input by default.

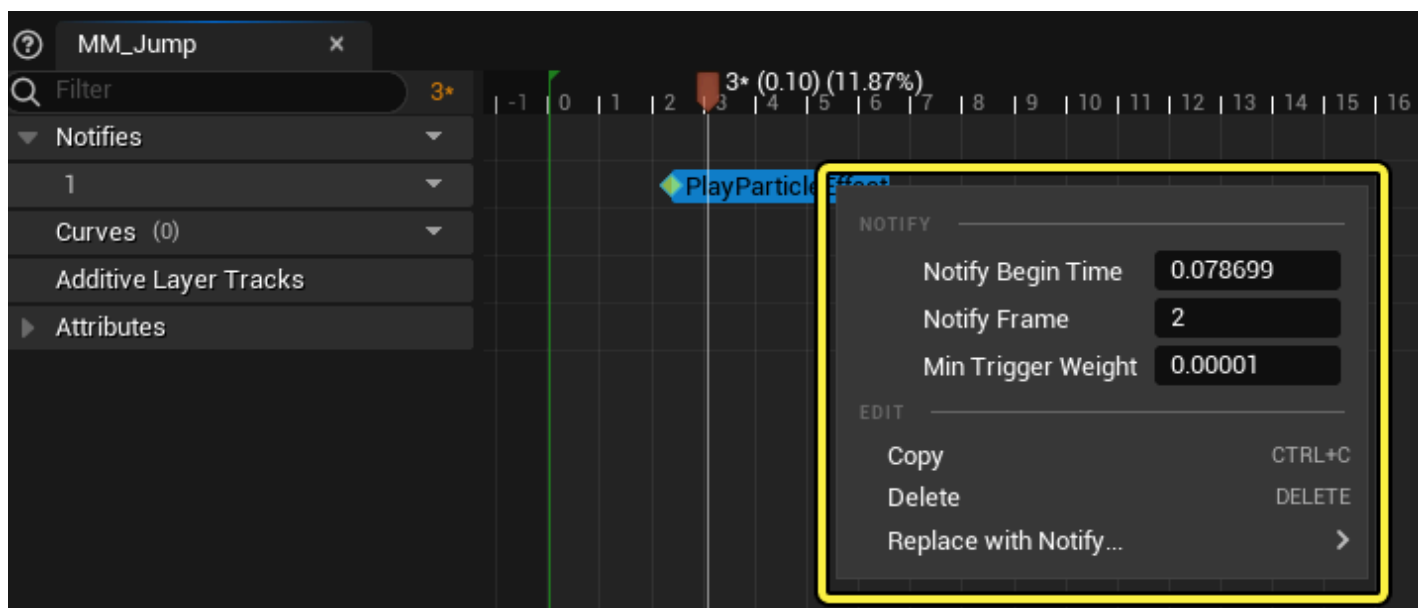


## Managing Your Notifies

As you create and use Animation Notifies, there are several features and properties that you can use to customize your workflow and Notify behavior.

## Keyframe Editing

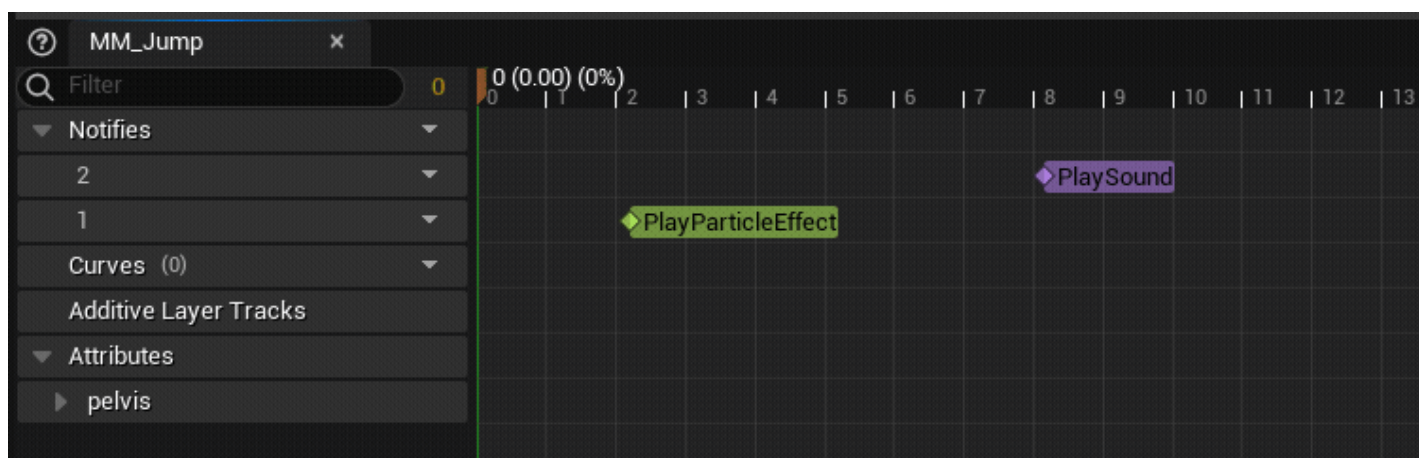
You can view information and commands for your Notify keyframes by right-clicking on them. The context menu shows the following:



Name	Description
<b>Notify Begin Time</b>	The time (in seconds) that this Notify is placed in the animation.

Name	Description
<b>Notify Frame</b>	The time (in frames) that this Notify is placed in the animation.
<b>Min Trigger Weight</b>	Determines how much weight (between <b>0</b> and <b>1</b> ) the animation needs to have to trigger the Notify. This is only relevant if the animation is being blended with another. The default value of <b>0.00001</b> means that even if this animation is barely blended in, the Notify will still execute. Setting it to higher numbers means that the animation must be blended in more fully in order for the Notify to successfully execute.
<b>Anim Notify State Duration</b>	The time (in seconds) of this Notify State's duration.
<b>Anim Notify State Frames</b>	The time (in frames) of this Notify State's duration.
<b>Copy</b>	Copies this Notify.
<b>Delete</b>	Deletes this Notify
<b>Replace with Notify...</b>	Replaces this Notify with another of the same type.

To assist with keyframe alignment, you can enable snapping options by right-clicking the top of the timeline and enabling either **Frames** or **Notifies**.



## Common Notify Properties

In addition to the properties that can be modified for each type of Notify, Notifies also contain the following common properties:

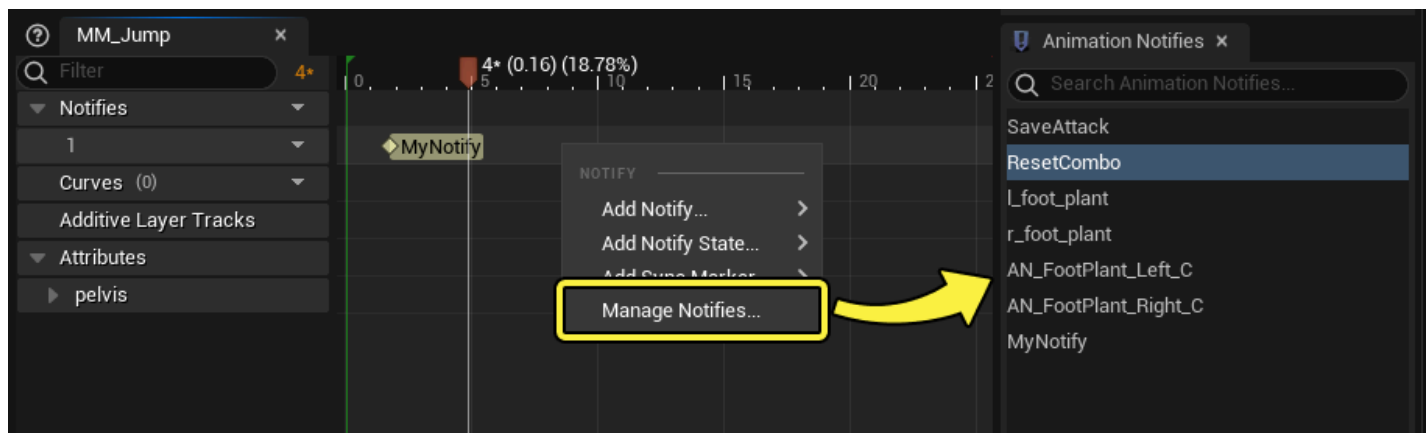
Name	Description
<b>Notify Color</b>	Sets the color of the Notify keyframe.
<b>Should Fire in Editor</b>	Enables the Notify executing when previewing it in the Editor. This does not impact normal gameplay or simulation.
<b>Notify Trigger Chance</b>	Defines the chance of the Notify executing successfully. A value of <b>0</b> means the Notify never executes, and <b>1</b> means the Notify always executes.
<b>Trigger on Dedicated Server</b>	Enabling this causes the Notify to execute on dedicated servers.
<b>Trigger on Follower</b>	If enabled, this Notify also triggers when the animation is a follower, when using <a href="#">Sync Groups</a> . By default, only Sync Group leader animations execute their Notifies.
<b>Notify Filter Type</b>	Defines a method for filtering Notifies to control if they should execute or not. You can filter by <a href="#">LOD</a> and specify the maximum LOD to filter the Notify from.
<b>Trigger Weight Threshold</b>	Determines how much weight (between <b>0</b> and <b>1</b> ) the animation needs to trigger the Notify. This is only relevant if the animation is blended with another. The default value of <b>0.00001</b> means that even if this animation is barely blended in, the Notify still executes. Setting it to higher values means the animation must be blended in more fully for the Notify to successfully execute.
<b>Montage Tick Type</b>	<p>When using Montage Notifies, this property determines how the Notify is ticked and executed in regards to its accuracy and sequential ordering. You can select either:</p> <ul style="list-style-type: none"><li>• <b>Queued</b>, which is a less precise, asynchronous, but more performant ticking method. Select this if you are okay with minor frame inaccuracy when your Notify executes.</li></ul>

Name	Description
	<ul style="list-style-type: none"> <li>• <b>Branching Point</b>, is a more precise, synchronous, and higher-cost ticking method. Select this if you want to ensure your Notify executes with maximum frame accuracy, which can be useful if you are using the Notify to execute other events in a controlled sequence. For example, Branching Point is useful when making gameplay-altering decisions, such as branching a decision based on this Notify.</li> </ul>
<b>Slot</b>	When using Montage Notifies, this property links the Notify to an animation within a specific Slot, as <a href="#">some montages can have more than one Slot</a> .
<b>Link Method</b>	<p>When using Montage Notifies, this property defines the method to link a Notify to an animation. Linking will cause the Notify keyframes to follow the montage animation section if they are adjusted. You can select from:</p> <ul style="list-style-type: none"> <li>• <b>Absolute</b>, which disables any automatic adjustment.</li> <li>• <b>Relative</b>, which will move the notify with the segment, but not adjust if the scale changes.</li> <li>• <b>Proportional</b>, which will both move and adjust with the segment location and scale.</li> </ul>
<b>Linked Sequence</b>	A read-only property that displays the currently-linked Animation Sequence. This changes if you align the Notify with a different sequence in a multiple-sequence montage.

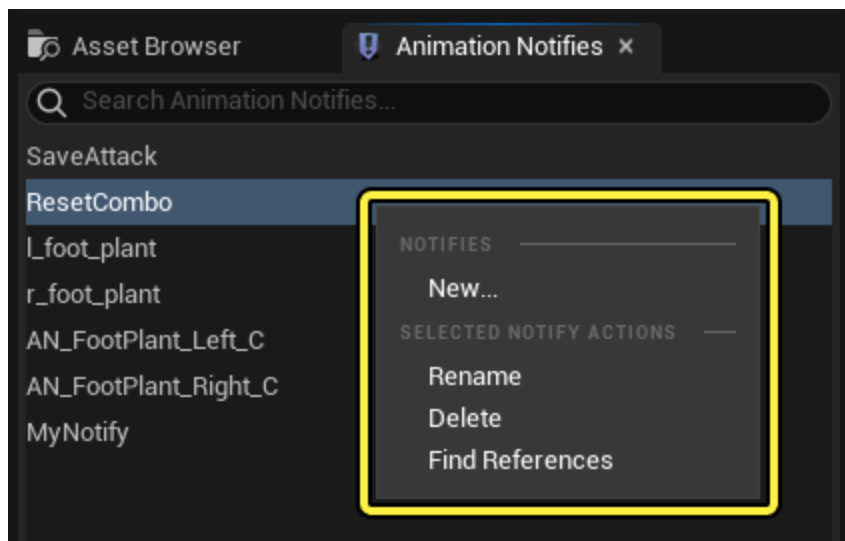
## Animation Notify Panel

The Animation Notifies panel can be used to manage your currently existing [Skeleton Notifies](#). You can access it by right-clicking in the Notify timeline area and selecting **Manage Notifies**.





You can right-click on a Notify and perform the following actions:

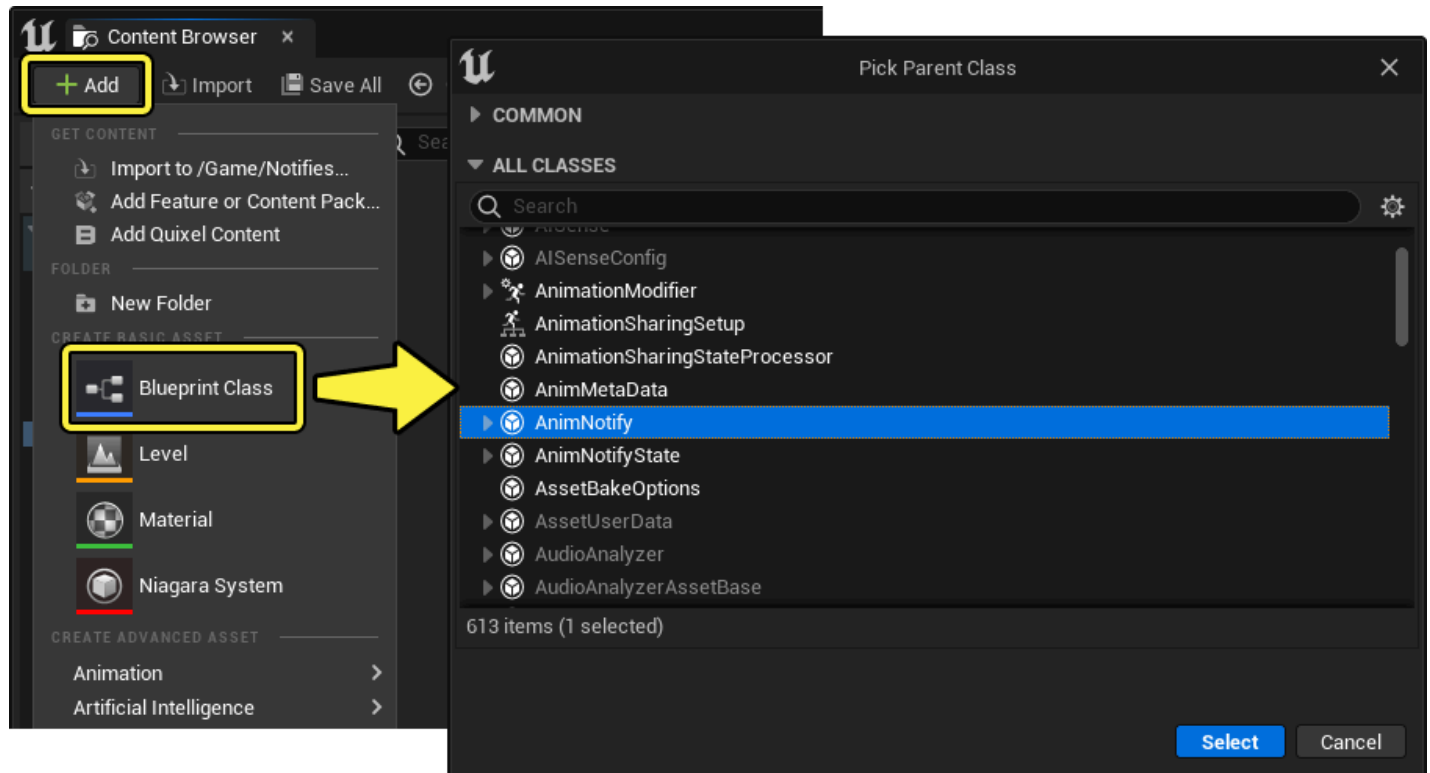


Name	Description
<b>New...</b>	Creates a new Skeleton Notify.
<b>Rename</b>	Renames the currently-selected Skeleton Notify.
<b>Delete</b>	Deletes the currently-selected Skeleton Notify.
<b>Find References</b>	Opens the <a href="#">Animation Asset Browser Panel</a> and filters the list to only show animation assets that use this Notify.

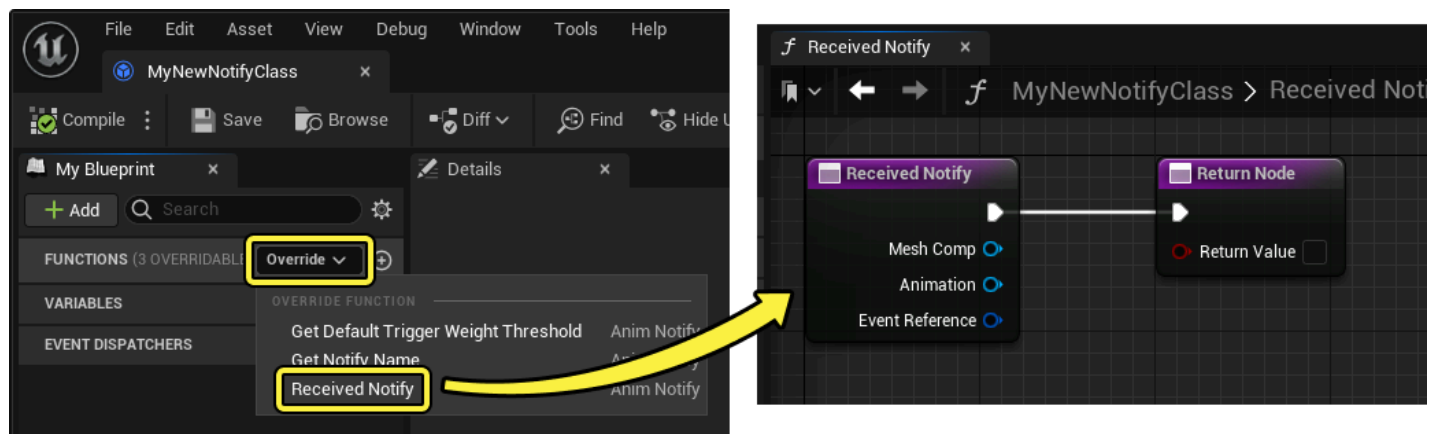
## Custom Notify Classes

If your project requires different Notify classes from those provided, you can create custom Notifies and Notify States with their own pre-defined Blueprint logic. This is a way for you to extend and create different Notify templates.

To create new Notify classes, navigate to the [Content Browser](#) and click **Add (+) > Blueprint Class**. Navigate in the **All Classes** section and select either **AnimNotify**, to create a child class of a [Notify](#), or **AnimNotifyState**, to create a child class of a [Notify State](#).



To start building unique functions for **AnimNotify** classes, click the **Override** function drop down menu in **My Blueprint** and select **Received Notify**. This opens the function graph where you can now script custom logic from the **Received Notify** entry node.



You can override and implement the following functions in either **AnimNotify** or **AnimNotifyState** classes:

Function	Description
<b>Get Default Trigger Weight Threshold</b>	This function sets the default value of <b>Trigger Weight Threshold</b> for this Notify, however it can be overridden in the <a href="#">Notify keyframe details</a> .
<b>Get Notify Name</b>	This function sets the name of the Notify or State keyframe in the timeline.
<b>Received Notify</b>	Used in standard Notifies, this function executes once when the Notify keyframe is reached.
<b>Received Notify Begin</b>	Used in Notify States, this function executes once when the Notify region begins.
<b>Received Notify End</b>	Used in Notify States, this function executes once when the Notify region ends.
<b>Received Notify Tick</b>	Used in Notify States, this function executes continuously throughout the duration of the Notify State.

Once you have created your custom Notify class, you can add it to the Notify timeline from either the **Add Notify** or **Add Notify State** menu.

