

VERSION 1.0

DECEMBER 21, 2024



FOOTSTEP SFX COMPONENT

JSX STUDIOS



SCAN TO ACCESS OUR LINKS

PRESENTED BY: JAMIE BEKYAROVICH
EMAIL: CONTACT@JSXGAMEDEV.COM
WEBSITE: WWW.JSXSTUDIOS.COM
PORTFOLIO: WWW.JSX-GAMEDEV.COM

TABLE OF CONTENTS

About (System Information).....	2
1.0 Initial Setup	3
1.1 Physical Surface Settings	3
1.2 Adding Component To Character	4
2.0 Custom Settings.....	5
2.1 Automatic vs. Standard Mode.....	6
2.2 Walk/Run Retriggering Rates & Max Running Speed.....	7
2.3 Animation Notifies	9
2.4 Functionality Settings.....	11
2.5 Audio Spawn Modes	12
2.6 Volume Multipliers.....	13
2.7 Utilizing Custom SFX.....	14
2.8 Customizing Float Curves	16
3.0 Creating New Sounds	17
3.1 Defining And Adding New Physical Surfaces.....	18
3.2 Using And Customizing Metasound Templates.....	19
3.3 Assigning Additional Sound Systems For Custom Surfaces.....	22
4.0 HUD Controls.....	23
5.0 FAQ – Common Issues And Easy Fixes	24
6.0 Patch Notes & Updates	26

ABOUT (SYSTEM INFORMATION)

System Overview:

The Footstep SFX System is a versatile, all-in-one master component designed to manage footstep sound effects for a variety of use cases. It is compatible with both third person and first-person perspectives and can seamlessly be attached to AI-controlled NPC characters.

This system offers multiple operational modes:

- **Automatic Mode:** Footstep audio is automatically triggered, eliminating the need for animation notifies.
- **Standard Mode:** Uses animation notifies to trigger audio, providing precise control over footstep timing.

Additionally, it includes three functionality settings for added flexibility:

1. **Default Functionality:** Utilizes physical materials to play specific audio files based on the surface material.
2. **Custom Functionality:** Plays unique footstep sound effects tailored to a specific character or enemy.
3. **Dual Functionality:** Combines custom character-specific footstep sounds with physical material-based sounds for a richer auditory experience.

Purpose

The Footstep SFX System was created to provide a simple, plug-and-play solution for integrating footstep sound effects into any project. After working on several projects and spending significant time setting up various footstep SFX systems, I designed this component to streamline the process.

With this system, you can quickly import and implement high-quality footstep sound effects for player characters and AI characters in both first- and third-person perspectives. This makes it ideal for rapid prototyping and production-ready projects alike.

Features and Benefits

- **Ease of Use:** Simply attach the component to your character, and it will manage footstep SFX without any additional setup. It works right out of the box, making it perfect for quick implementation and prototyping.
- **Professional Results:** While the system functions seamlessly with no setup, you can achieve even better, professional-quality audio by going the extra mile. Utilize the included Animation Notifies and configure physical materials in your project settings to unlock precise control and enhanced sound fidelity.
- **Modular and Scalable:** Its modular design makes it simple to add footstep SFX to new characters or even new projects, ensuring flexibility and scalability for any development workflow.

This system eliminates the repetitive task of setting up footstep logic, allowing you to focus on creating your game. Whether you're starting a new project or adding polish to an existing one, the Footstep SFX System is a reliable and efficient solution for all your footstep audio needs.

1.0 INITIAL SETUP

Section Overview:

This section provides an introduction to the setup process for the Footstep SFX System ensuring it is ready to use in your project. The initial setup involves preparing your project for the system by configuring necessary settings and integrating the component with your characters. The system is designed to be flexible and user-friendly, requiring minimal setup to function right out of the box. Whether you are working on a prototype or a polished project, this guide will walk you through the steps to get the system up and running. For users seeking to unlock its full potential, later sections will provide detailed instructions on setting up Animation Notifies, and fine-tuning the system for immersive, high-quality footstep sound effects. By the end of this section, you'll have a clear understanding of the setup process and how the Footstep SFX System can seamlessly integrate into your project, saving you time and effort while delivering outstanding results.

1.1 PHYSICAL SURFACE SETTINGS

Section Overview

If you are starting a new project using this system as a template, the DefaultEngine.ini file will already include the correct physical surface settings, and no further action is required unless you plan to add additional physical surfaces. However, if you are migrating this system into an existing project, you will need to manually add these settings to avoid blueprint compilation errors in the AC_FootstepSFX component. This section explains how to implement the correct physical surface settings.

How to Implement Correct Physical Surface Settings

You can set up the physical surface settings using one of two methods:

1. Editing the DefaultEngine.ini File

- Open your project's DefaultEngine.ini file.
- Locate the section with the heading [/Script/Engine.PhysicsSettings]
- Add the following lines under this heading:
+PhysicalSurfaces=(Type=SurfaceType55,Name="Rocks")
+PhysicalSurfaces=(Type=SurfaceType56,Name="Wood")
+PhysicalSurfaces=(Type=SurfaceType57,Name="Snow")
+PhysicalSurfaces=(Type=SurfaceType58,Name="Sand")
+PhysicalSurfaces=(Type=SurfaceType59,Name="Water")
+PhysicalSurfaces=(Type=SurfaceType60,Name="Metal")
+PhysicalSurfaces=(Type=SurfaceType61,Name="Grass")
+PhysicalSurfaces=(Type=SurfaceType62,Name="Concrete")

```
[/Script/Engine.PhysicsSettings]
+PhysicalSurfaces=(Type=SurfaceType55,Name="Rocks")
+PhysicalSurfaces=(Type=SurfaceType56,Name="Wood")
+PhysicalSurfaces=(Type=SurfaceType57,Name="Snow")
+PhysicalSurfaces=(Type=SurfaceType58,Name="Sand")
+PhysicalSurfaces=(Type=SurfaceType59,Name="Water")
+PhysicalSurfaces=(Type=SurfaceType60,Name="Metal")
+PhysicalSurfaces=(Type=SurfaceType61,Name="Grass")
+PhysicalSurfaces=(Type=SurfaceType62,Name="Concrete")
```

(Physical Surface Settings Inside DefaultEngine.ini)

2. Using the Unreal Engine Interface

- Navigate to **Edit -> Project Settings -> Engine -> Physics -> Physical Surfaces** in the Unreal Engine editor.
- Add the following surface types to the list: Rocks, Wood, Snow, Sand, Water, Metal, Grass, and Concrete. Assign these to **Surface Type Numbers 55-62**.

SurfaceType55	Rocks
SurfaceType56	Wood
SurfaceType57	Snow
SurfaceType58	Sand
SurfaceType59	Water
SurfaceType60	Metal
SurfaceType61	Grass
SurfaceType62	Concrete

(Physical Surface Settings Inside Unreal Engine)

Best Practices

To avoid potential conflicts, this system uses the higher surface type numbers (55–62) rather than the beginning of the physical surface list. This ensures compatibility with existing physical material setups in your project without requiring changes to the **AC_FootstepSFX** blueprint component.

By following these steps, you'll ensure the system is correctly configured and ready to handle physical material-specific footstep sound effects.

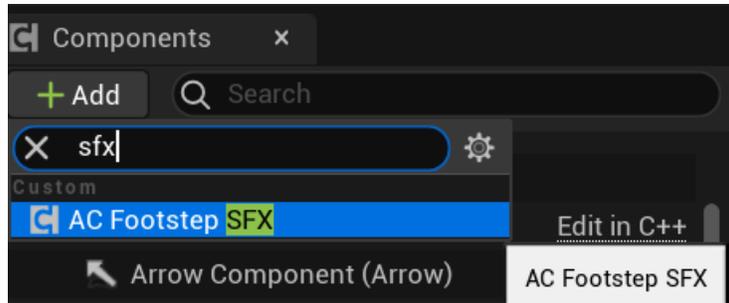
1.2 ADDING COMPONENT TO CHARACTER

Section Overview

This section explains how to add the AC_FootstepSFX blueprint component to your character blueprint, enabling the Footstep SFX System for your character.

Steps to Add the Component

1. Open your Character Blueprint in the Unreal Engine editor.
2. Navigate to the Components Tab.
3. Add a new component by clicking the + Add Component button.
4. In the search bar, type "SFX". From the dropdown menu, select AC_FootstepSFX.
5. Verify that the component has been successfully added to your blueprint.



(Adding AC_FootstepSFX To Character Blueprint)

Test the System

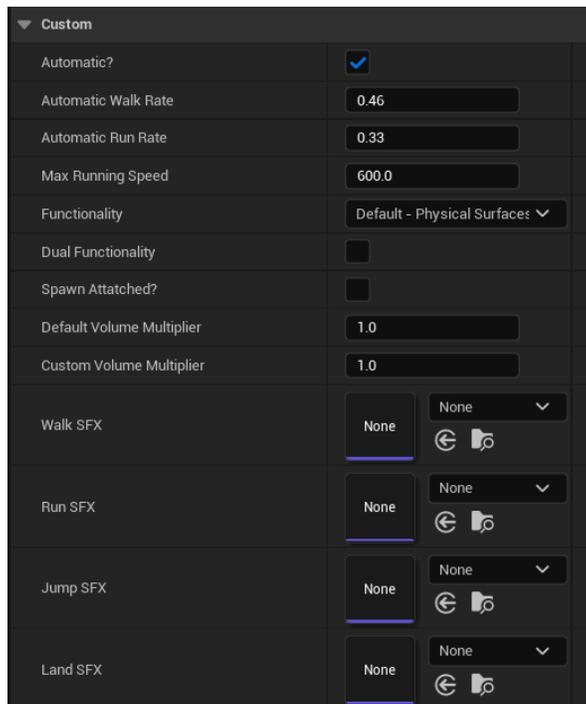
Once the component is added, your character is ready to use the Footstep SFX System. Play your level and walk around to ensure the footstep sound effects are working.

You should now hear footstep audio triggered by your character's movement. If no sound is heard, ensure the component has been added correctly and double-check that your project includes the necessary physical material settings outlined in [Section 1.1](#).

2.0 CUSTOM SETTINGS

Section Overview

The Footstep SFX Component is highly customizable, allowing you to adapt its functionality to suit the unique requirements of your project. This section provides an introduction to the range of settings and features you can adjust to optimize the system. These settings can be found in the “Custom” category on the Actor Component.



(Custom Settings Available On AC_FootstepSFX Component)

The system is designed with flexibility in mind, catering to both rapid prototyping and polished production needs. You can choose between two primary modes—Automatic Mode for ease of use and Standard Mode for precise control using Animation Notifies. Additionally, you can fine-tune the retriggering rates for footstep sounds in Automatic Mode to match the movement style of your characters, ensuring a seamless auditory experience.

The component also offers a variety of functionality settings, including options for physical material-based sounds, custom character-specific sounds, and combinations of both. Audio spawning modes, volume multipliers, and the ability to integrate custom footstep SFX further enhance the system’s versatility.

This section will break down these customizable options into detailed subsections:

- [2.1 Automatic vs. Standard Mode](#): Understanding the differences and when to use each mode.
- [2.2 Walk/Run Retriggering Rates & Max Running Speed](#): Adjusting retrigger rates to suit your character’s movement speed.
- [2.3 Animation Notifies](#): Using Animation Notifies to precisely synchronize footstep sounds with character animations.
- [2.4 Functionality Settings](#): Exploring Default, Custom, and Dual Functionality modes.
- [2.5 Audio Spawn Modes](#): Choosing how and where audio is spawned in your project.
- [2.6 Volume Multipliers](#): Controlling the volume of footstep sounds for dynamic effects.
- [2.7 Utilizing Custom SFX](#): Replacing or adding your own unique footstep sounds.
- [2.8 Customizing Curve Floats](#): Adjusting The Crossfade Values Between Walking/Running States

By following the upcoming sections, you’ll gain a deeper understanding of each feature and how to use them to create a fully tailored footstep audio experience for your project.

2.1 AUTOMATIC VS. STANDARD MODE

Section Overview

This section explains the differences between Automatic Mode and Standard Mode in the Footstep SFX Component, helping you decide which mode best fits your project’s needs. Each mode has unique strengths, making the system adaptable for both quick setups and highly polished implementations.



(Selecting This Bool Value Enables Automatic Mode – Deselecting It Enables Standard Mode)

Automatic Mode

Automatic Mode is designed for simplicity and efficiency, automatically triggering footstep sounds based on your character's movement without requiring Animation Notifies. This mode is ideal for:

- **Prototyping**: Quickly setting up footstep audio without investing time in precise configurations.
- **First Person Characters**: Suitable for basic character setups, projects where animation timing is less critical, or those without animations.
- **Basic AI Characters**: Guarantees footstep audio for NPCs, even without dedicated animations or notifies.

Automatic Mode uses retriggering logic based on movement states (walking and running) to generate footstep sounds at appropriate intervals, offering a quick and effective solution for most use cases. While it does not require animation notify setup, it relies on **Automatic Walk/Run Rates** and **Max Running Speed** settings to ensure realistic audio timing. These settings will be explained in detail in [Section 2.2](#).

Standard Mode

Standard Mode provides precise control over footstep audio by utilizing Animation Notifies. This mode is ideal for:

- **Polished Projects:** Ensuring footstep sounds align perfectly with character animations.
- **Custom Animation Sequences:** Useful when working with complex animations or bespoke footstep timing requirements.
- **Advanced Audio Design:** Allows you to fine-tune audio playback for immersive, high-quality results.

By placing Animation Notifies directly within your character's animation sequences, Standard Mode ensures that footstep sounds are triggered exactly when the foot contacts the ground, creating a more natural and immersive experience.

Choosing the Right Mode

Your choice between Automatic and Standard Mode will depend on your project's requirements:

- Use **Automatic Mode** for quick setups, prototyping, first-person characters, or AI characters where precision is less critical.
- Use **Standard Mode** for detailed, polished results that require synchronization with character animations.

Both modes can be further customized to suit your project's specific needs, and switching between them is straightforward, making the system highly flexible and adaptable.

By default, **Automatic Mode** is selected to facilitate quick prototyping, allowing footstep SFX to function immediately after adding the component to your character without additional setup.

For **first-person characters**, we recommend using **Automatic Mode** as it provides a simple and effective solution without requiring precise animation timing. This makes it ideal for setups where animations are less critical or not used at all.

However, for **third-person characters** or more advanced **AI NPCs**, we recommend using **Standard Mode** with Animation Notifies to ensure footstep sounds align perfectly with animations, delivering a more polished and immersive experience.

2.2 WALK/RUN RETRIGGERING RATES & MAX RUNNING SPEED

Section Overview

This section explains how to configure **Walk/Run Retriggering Rates** and **Max Running Speed** settings in **Automatic Mode**. These settings determine the timing of footstep sounds, ensuring they align with the character's movement speed for a natural, immersive experience. While Retriggering configurations are exclusive to Automatic Mode and ignored in Standard Mode, the **Max Running Speed** setting is used in both modes.

Automatic Walk Rate	0.46
Automatic Run Rate	0.33
Max Running Speed	600.0

(Default Settings For Automatic Walk/Run Rates And Max Running Speed)

What Are Walk/Run Retriggering Rates?

Walk/Run Retriggering Rates determine how often footstep sounds are triggered based on your character's movement state:

- **Walk Rate:** Defines the interval between footsteps when the character is walking.
- **Run Rate:** Defines the interval between footsteps when the character is running.

These rates are critical to achieving realistic audio timing, especially for characters with variable movement speeds.

What Is Max Running Speed?

Max Running Speed is used to distinguish between walking and running states. It sets the threshold at which the system switches from the **Walk Rate** to the **Run Rate** for triggering footstep sounds and determines whether to play the walking or running sound systems. Ensure this value matches your character's maximum running speed, regardless of the mode you're using.

For example:

- If the character's movement speed exceeds the **Max Running Speed**, the system will use the Run Rate for footstep retriggering and play running sounds.
- If the movement speed is below this threshold, the system will use the Walk Rate and walking sounds.

How to Configure Walk/Run Rates and Max Running Speed

1. Open the **AC_FootstepSFX Component** in your character's blueprint.
2. Locate the settings for **Walk Retrigger Rate**, **Run Retrigger Rate**, and **Max Running Speed** in the component's properties.
3. Adjust the values to match your character's movement speed and desired footstep timing:
 - Set a **higher** retrigger rate for **walking** to create slower, more deliberate footstep sounds.
 - Set a **lower** retrigger rate for running to reflect faster, more frequent footsteps.
 - Adjust the Max Running Speed to match your character's speed threshold for transitioning between walking and running.

By default, the component is configured to work with a walk speed of **200.0 cm/s** and a run speed of **600.0 cm/s**. If your character uses these movement speeds, no changes to the component are necessary. However, if your character uses different speeds, you will need to adjust the **Walk/Run Rates** and the **Max Running Speed Threshold** to ensure the footstep sounds remain in sync with the character's movement.

Best Practices

- Test your settings in-game to ensure footstep sounds feel natural and match your character's movement animations.
- For characters with highly variable speeds, consider fine-tuning these values to provide smooth transitions between walking and running sounds.

By configuring **Walk/Run Retriggering Rates** and **Max Running Speed**, you can ensure that Automatic Mode provides realistic and responsive footstep audio tailored to your project's needs. However, if you continue to experience issues with footstep timing after careful configuration, consider using **Standard Mode**. Automatic Mode is primarily designed for quick prototyping or characters without walking/running animations.

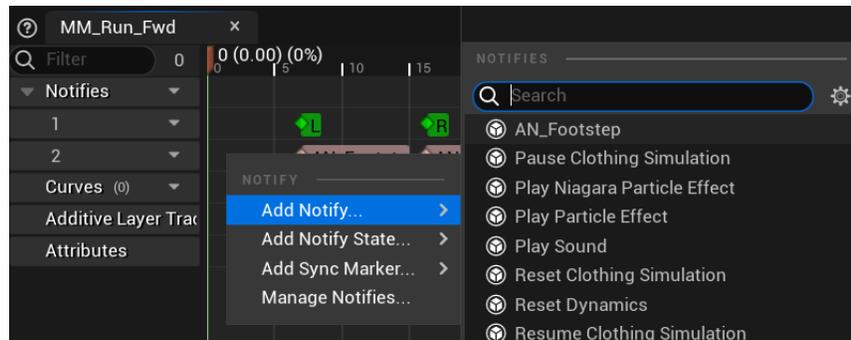
2.3 ANIMATION NOTIFIES

Section Overview

This section provides a detailed guide on setting up and using the included **Animation Notify Blueprint** in your character animations. Animation Notifies allow you to precisely trigger footstep sounds in sync with your character's animations, ensuring a natural and immersive audio experience. Additionally, this section will cover the customizable settings available within the Animation Notify Blueprint, allowing you to tailor its behavior to meet the specific needs of your project. By integrating these notifies, you can achieve precise footstep timing and enhance the overall polish of your character animation.

Setting Up Animation Notifies

Begin by opening the animation you want to set up footstep notifies for. It is recommended to create a new notify track specifically for footsteps. Once the track is created, right-click on it, hover over the **"Add Notify"** option, and select the notify blueprint named **AN_Footstep** from the dropdown menu.



(Selecting The AN_Footstep Blueprint Notify In The Animation Editor)

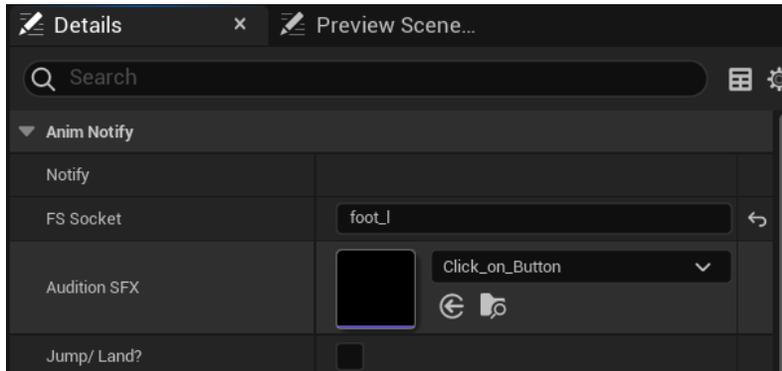
Align the notify with the point in the animation where your character’s foot makes contact with the ground. For additional footsteps, you can copy and paste the notify along the track as needed to match the animation sequence. This process ensures precise footstep timing, enhancing the realism and immersion of your character’s movements.



(Footstep Notifies Placed Along A Characters Running Animation)

Animation Notify Settings

When you select an Animation Notify in the details panel, you will see three instance-editable variables that control its behavior: **FS Socket**, **Audition SFX**, and **Jump/Land?**



(AN_Footstep Instance Edible Variables)

1. FS Socket

The **FS Socket** variable is a string that specifies the name of a skeleton bone or socket where the footstep line trace begins. If the **Spawn Attached** setting in the **AC_FootstepSFX Component** is enabled, this will also be the bone where the audio is spawned attached to.

- For projects using the provided Unreal Engine skeletons, set this to `foot_l` or `foot_r`.
- If you're using a custom skeleton with different bone names, update this variable with the appropriate names for the feet.

Additional details about the **Spawn Attached** setting can be found in [section 2.5](#).

2. Audition SFX

The **Audition SFX** variable is used to test the timing of footstep sounds directly in the animation editor. The notify blueprint requires data such as surface material and player speed to determine which sound to play during runtime, which makes it impossible to hear the correct sound in the editor. To address this, the **Audition SFX** variable allows you to select an audio file to audition the placement of your footstep notifies in the animation editor.

- This selected sound will only play in the animation editor and **will not be used during runtime**.
- During runtime, the system will ignore the Audition SFX and play the appropriate sound based on surface material or the custom footstep SFX selected in the blueprint component.

3. Jump/Land?

The **Jump/Land** Boolean variable is used for animations involving a jump or landing after a fall.

- **Jump** refers to smaller jumps, such as jumping in place.
- **Land** refers to larger landings, such as falling off a high surface.

The system automatically differentiates between the two using player velocity, but you must set this variable to **true** for landing animations to help the system identify these specific actions.

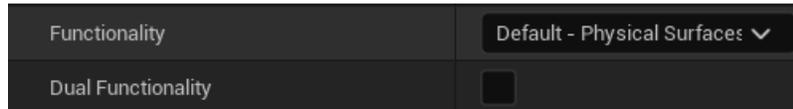
- For walk or run animations, leave this variable set to **false**.
- Only enable this variable when the animation represents a landing after a jump or fall.

By correctly setting these variables, you can fine-tune the behavior of the Animation Notify Blueprint and ensure your footstep audio is aligned with your animations for a polished and immersive experience.

2.4 FUNCTIONALITY SETTINGS

Section Overview

This section explains the different functionality settings of the **Footstep SFX Component**, including **Default Functionality**, **Custom 1-Way**, **Custom 2-Way**, **Custom 4-Way**, and **Dual Functionality**. These settings allow you to customize how footstep sounds are triggered and tailored for different characters and scenarios.



(Default Functionality Settings Inside AC_FootstepSFX Component)

Functionality Modes

1. Default Functionality

- Uses the physical surface settings (e.g., concrete, grass, water, snow) to determine which sounds to play.
- Automatically manages walking, running, jumping, and landing sounds based on the character's velocity and **Max Running Speed** settings.

2. Custom 1-Way

- Allows you to provide a single custom footstep sound for walking, running, jumping, and landing.
- Ideal when you only have one sound system available (e.g., a single walking sound without separate running, jumping, or landing sounds).

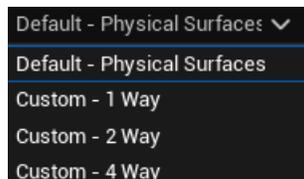
3. Custom 2-Way

- Enables separate custom sounds for walking/jumping and running/landing.
- Useful when you have two sound systems available (e.g., walking and running sounds but no jumping or landing sounds).

4. Custom 4-Way

- Provides full customization by allowing unique sounds for walking, running, jumping, and landing.
- Perfect for creating character-specific sounds, such as deep, heavy footsteps for a large monster.

Note: These custom settings are designed for character-specific sounds, not additional physical surfaces. For example, you might use them for an enemy that requires unique sounds, such as a heavy creature with distinct footsteps.



(Functionality Settings Available In Drop Down Menu)

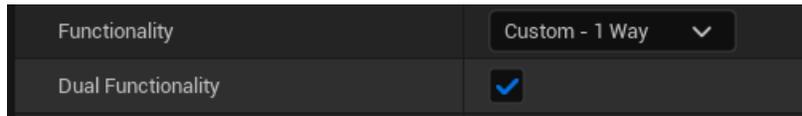
Dual Functionality

The **Dual Functionality** option allows you to combine custom character sounds with physical surface sounds. This is useful when you want character-specific sounds (e.g., heavy monster footsteps) to play alongside surface-specific sounds (e.g., snow crunching or water splashing).

For example:

- A character with custom sounds might run through shallow water or snow. Dual Functionality ensures that both the character’s specific sounds and the corresponding physical surface sounds are played simultaneously, enhancing the audio experience.

To enable **Dual Functionality**, select the desired type of custom functionality (Custom 1-Way, Custom 2-Way, or Custom 4-Way) and set the **Dual Functionality** option to **true**.



(Dual Functionality Enabled Utilizing Custom 1-Way Sounds Along Side Physical Material Sounds)

Summary

These functionality settings provide flexibility for designing footstep sounds tailored to your project’s unique needs. Whether you need default physical surface sounds, character-specific audio, or a combination of both, the **Footstep SFX Component** makes it easy to achieve immersive and dynamic audio for your characters.

2.5 AUDIO SPAWN MODES

Section Overview

This section explains how to use the **Spawn Attached** variable in the Footstep SFX Component. This setting determines how and where footstep sounds are spawned during gameplay. By adjusting this variable, you can control whether the audio is dynamically linked to the character, following their movement, or remains stationary at the location where the footstep occurred. The choice between attaching footstep sounds to the character or spawning them at the line trace location can significantly impact how the audio interacts with your environment, especially in maps with complex audio setups. Understanding this setting is essential for achieving seamless and immersive sound design tailored to your project's needs. Now, let’s explore how the **Spawn Attached** variable works in different modes and when it’s most beneficial to use.

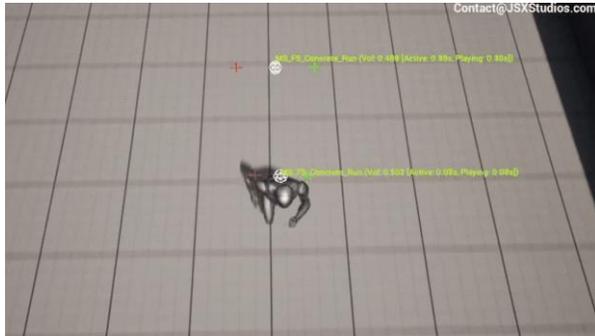
Variable Settings

- **Spawn Attached = True:** Footstep sounds are attached to the character and follow them for the duration of the audio playback.
 - In **Standard Mode**, the audio is spawned attached to the sockets specified in the Animation Notify.
 - In **Automatic Mode**, the audio is attached to the character's root component.
- **Spawn Attached = False:** Footstep sounds are spawned at the line trace location and remain stationary, not following the character.

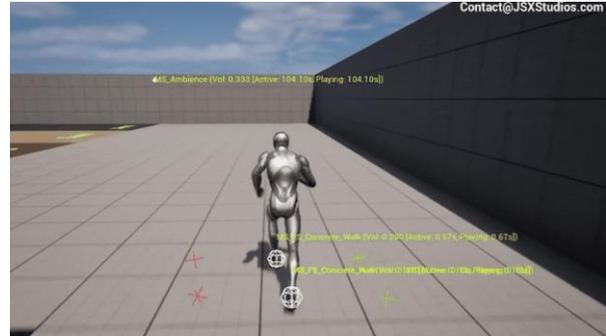
When to Use Spawn Attached

Using **Spawn Attached** is helpful in scenarios where your map contains multiple audio volumes, and you want the footstep sounds to be influenced by those volumes. For example - Imagine an audio volume that applies reverb inside a cave and reduces external sounds:

- If **Spawn Attached** is set to **false**, a footstep sound spawned just outside the cave may be attenuated or cut off when the character enters the cave, as it remains outside the audio volume.
- By setting **Spawn Attached** to **true**, the footstep sound follows the character into the cave, ensuring it is affected by the cave's audio volume settings and avoiding abrupt audio culling.



(Audio Spawning At Line Trace Location)



(Audio Spawning Attached To Character)

General Recommendations

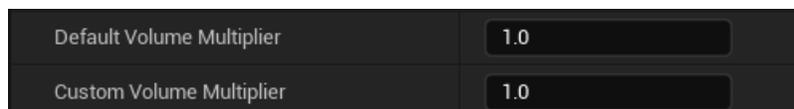
In most cases, this setting can be left **off** (Spawn Attached = False). However, if your project utilizes multiple audio volumes and your footstep sound class is set to **Apply Ambient Volumes**, consider enabling **Spawn Attached** to ensure smooth and uninterrupted audio transitions when the character moves between audio volumes.

By understanding and configuring this setting, you can ensure your footstep audio behaves naturally and integrates seamlessly with your game's environment.

2.6 VOLUME MULTIPLIERS

Section Overview

This section explains how to use the **Default Volume Multiplier** and **Custom Volume Multiplier** settings in the Footstep SFX Component. These settings allow you to control the overall footstep volume for individual characters and fine-tune the balance between physical surface sounds and custom character sounds, especially when using Dual Functionality.



- **Default Volume Multiplier:** Serves as a master volume control for physical surface footstep sounds (e.g., concrete, grass, water).
- **Custom Volume Multiplier:** Acts as a master volume control for custom footstep sounds (Custom 1-Way, 2-Way, 4-Way).

Key Features and Benefits

1. Per-Character Volume Control

These multipliers enable you to adjust the footstep volume for each character individually, directly from their Footstep SFX Component. This is especially useful in projects with multiple characters that may require different audio profiles. For example:

- A large, heavy character might need louder, more impactful footsteps.
- A stealthy character might require quieter footstep sounds.

2. Balancing Dual Functionality

In Dual Functionality mode, the volume multipliers allow you to mix custom character sounds with physical surface sounds to achieve the desired balance. For instance:

- If the physical surface sounds (e.g., snow crunching) are too prominent compared to the custom character sounds (e.g., a monster’s deep footsteps), you can reduce the **Default Volume Multiplier** or increase the **Custom Volume Multiplier** to create a better mix.

3. Efficient Sound Design

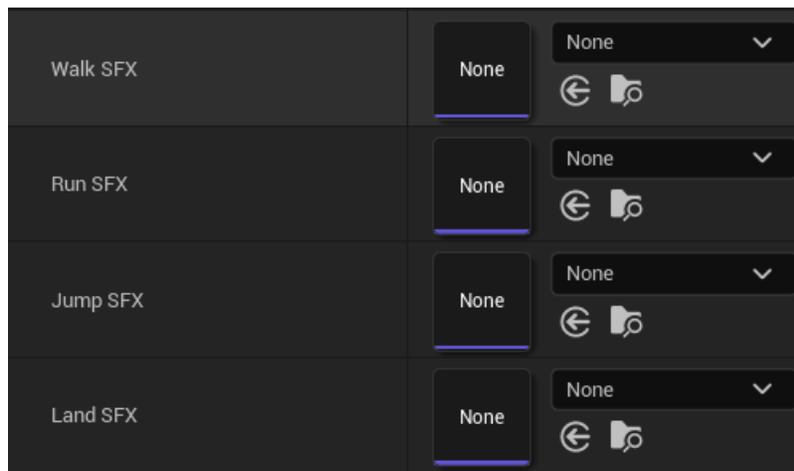
By adjusting these multipliers, you can quickly tweak audio levels without needing to modify individual sound assets or global volume settings. This flexibility ensures efficient sound design workflows and precise audio control.

By leveraging these settings, you can ensure your project delivers a tailored and immersive audio experience for every character and environment.

2.7 UTILIZING CUSTOM SFX

Section Overview

This section explains how to apply custom, character-specific footstep sounds to your character using the Footstep SFX Component. These custom sounds are intended to provide unique audio profiles for specific characters, such as a large monster with heavy footsteps, rather than additional physical surface sounds.



(Custom Footstep SFX Variables In FootstepSFX Component)

Setting Up Custom SFX

1. **Navigate to the Custom Category:** Select your character's **AC_FootstepSFX Component** and locate the custom category in the component's properties.
2. **Custom Sound Variables:** You'll find four editable variables:
 - **Custom Walk SFX:** For walking sounds.
 - **Custom Run SFX:** For running sounds.
 - **Custom Jump SFX:** For jump sounds.
 - **Custom Land SFX:** For landing sounds.
3. **Select a Functionality Mode:**
Under the **Functionality Settings**, change the mode from **Default – Physical Surfaces** to one of the custom modes:
 - **Custom 1-Way:**
 - Requires only the **Custom Walk SFX** variable to be set.
 - This sound will be used for all footstep actions (walking, running, jumping, and landing).
 - **Custom 2-Way:**
 - Requires both **Custom Walk SFX** and **Custom Run SFX** variables.
 - **Custom Walk SFX** is used for walking and jumping.
 - **Custom Run SFX** is used for running and landing.
 - **Custom 4-Way:**
 - Requires all four sound variables to be set: **Custom Walk SFX**, **Custom Run SFX**, **Custom Jump SFX**, and **Custom Land SFX**.
 - Each sound will be played based on the corresponding action.

Benefits of Custom SFX

- **Unique Character Sounds:** Custom SFX allow you to create distinctive footstep profiles for characters with unique movement styles or weights, such as heavy monsters, robotic enemies, or stealthy protagonists.
- **Flexible Audio Design:** Whether you have a single sound system or a full set of custom sounds, the different custom functionality modes (1-Way, 2-Way, 4-Way) adapt to your available audio resources.
 - **Custom 1-Way** is ideal for minimal sound resources.
 - **Custom 2-Way** and **Custom 4-Way** provide more granular control for diverse audio needs.
- **Quick Integration:** Setting up custom sounds is straightforward and requires no changes to the underlying system logic.

For added depth, consider enabling **Dual Functionality** to mix custom character sounds with physical surface sounds (see [Section 2.4](#)).

By following these steps, you can easily integrate custom footstep sounds for your characters, creating a more engaging and tailored audio experience for your players.

2.8 CUSTOMIZING FLOAT CURVES

Section Overview

This section explains how **Float Curves** are used to dynamically crossfade between walking/running and jumping/landing sound systems, ensuring smooth transitions based on character movement. By understanding and customizing these curves, you can fine-tune how footstep sounds blend together for a more immersive audio experience.

How Float Curves Work

The system uses the character's velocity to determine the appropriate volume levels for different footstep sounds. Here's how it works:

1. The character's velocity is normalized to a range of **0** to the character's **Max Running Speed**.
2. The normalized value is then clamped to a range of **0** to **1**.
3. This clamped value is fed into a **Float Curve**, which outputs a corresponding volume level.
4. The output value from the curve determines the volume of the footstep sound to be played.

Both walking/running and jumping/landing sounds are triggered simultaneously, but their volumes are adjusted based on the float curve settings, creating a natural crossfade effect. For example:

- At lower speeds, the float curve may prioritize walking or jumping sounds.
- At higher speeds, the float curve shifts focus to running or landing sounds.

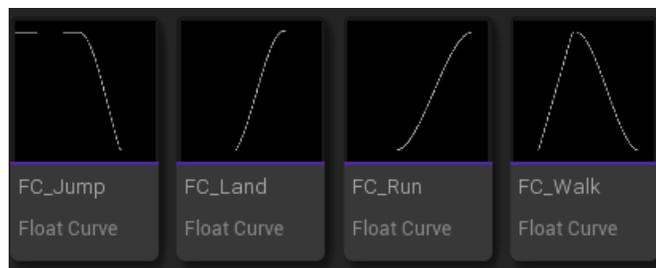
Customizing Float Curves

While the default float curves are designed to work well for most scenarios, you can customize them to achieve specific effects:

- **Adjust Sound Emphasis:** Modify the curve to increase or decrease the volume of certain sounds (e.g., more emphasis on running at higher speeds).
- **Create Unique Audio Profiles:** Design custom curves for different characters to reflect their unique movement styles.

To customize the curves:

1. Open the **AC_FootstepSFX Component Blueprint**.
2. Locate the float curve variables for both default and custom sounds.
3. Create and assign new float curve assets and adjust their settings to control how sound volumes transition smoothly between walking, running, jumping, and landing.



(Default Float Curve Assets)

Separate Float Curves for Characters

If you want different float curve profiles for separate characters:

- Disable the private setting for the float curve variables in the **Footstep SFX Component**.
- Assign unique float curves to each character's component to tailor their audio transitions individually.

Best Practices

- **Test Adjustments in Game:** Ensure that any changes you make to the float curves produce smooth and natural transitions during gameplay.
- **Maintain Balance:** Avoid abrupt shifts in sound by keeping the curve smooth and gradual across its range.
- **Use Default Settings as a Reference:** The default curves provide a solid baseline and can be used as a starting point for customization.

By customizing float curves, you can create seamless audio transitions that adapt to your character's movements, enhancing the overall sound design of your project.

3.0 CREATING NEW SOUNDS

Section Overview

This section provides an overview of how to expand the Footstep SFX System by creating and integrating new sounds into your project. It outlines the steps for developing additional sound systems for custom physical surfaces, adding those surfaces to your project, and leveraging the provided MetaSound templates to enhance your audio design.

You will learn how to define and configure new physical surfaces to ensure the system recognizes them and how to assign custom sound systems tailored to these surfaces. This section will also introduce the MetaSound templates included with the system, which offer a flexible and modular approach to creating dynamic and immersive soundscapes.

Whether you are adding entirely new environments with unique materials, like mud or ice, or designing sound systems that respond dynamically to gameplay scenarios, this section will guide you through the process of expanding and customizing your project's footstep audio.

The following subsections will break down each topic in detail, providing clear instructions and best practices for:

- [Defining and Adding New Physical Surfaces](#)
- [Using and Customizing MetaSound Templates](#)
- [Assigning Additional Sound Systems for Custom Surfaces](#)

By the end of this section, you will have the tools and knowledge to extend the Footstep SFX System and create a richer, more tailored audio experience for your game.

3.1 DEFINING AND ADDING NEW PHYSICAL SURFACES

Section Overview

This section explains how to define new physical surfaces, create corresponding physical material assets, and synchronize them to ensure proper functionality within the Footstep SFX System.

Adding New Physical Surfaces

You can add new physical surfaces to your project using one of two methods:

1. DefaultEngine.ini File:

- Open your **DefaultEngine.ini** file.
- Locate the [/Script/Engine.PhysicsSettings] heading.
- Add your new surface types in the following format:

```
+PhysicalSurfaces=(Type=SurfaceType01,Name="YourSurfaceName")
```

```
[/Script/Engine.PhysicsSettings]
+PhysicalSurfaces=(Type=SurfaceType55,Name="Rocks")
+PhysicalSurfaces=(Type=SurfaceType56,Name="Wood")
+PhysicalSurfaces=(Type=SurfaceType57,Name="Snow")
+PhysicalSurfaces=(Type=SurfaceType58,Name="Sand")
+PhysicalSurfaces=(Type=SurfaceType59,Name="Water")
+PhysicalSurfaces=(Type=SurfaceType60,Name="Metal")
+PhysicalSurfaces=(Type=SurfaceType61,Name="Grass")
+PhysicalSurfaces=(Type=SurfaceType62,Name="Concrete")
```

(Physical Surface Settings Inside DefaultEngine.ini)

3. Using the Unreal Engine Interface

- Navigate to **Edit -> Project Settings -> Engine -> Physics -> Physical Surfaces** in the Unreal Engine editor.
- Manually add your custom surface types to the list.

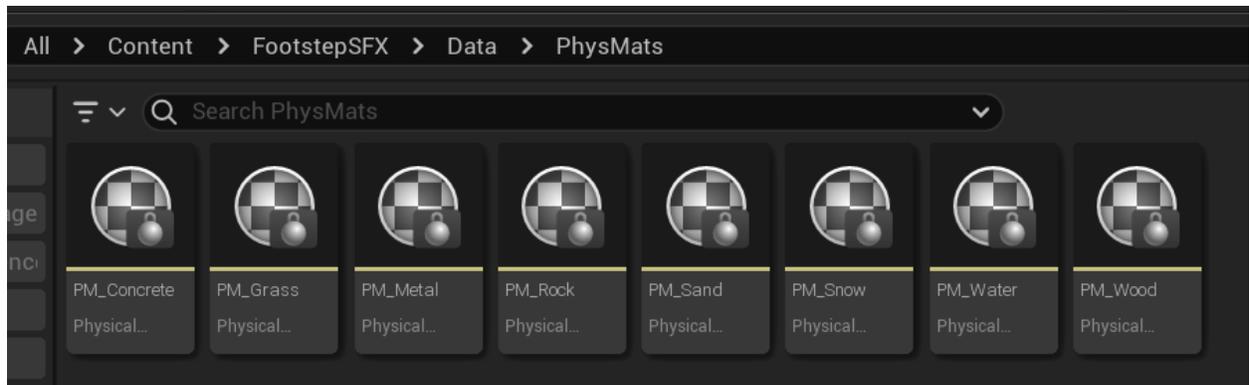
SurfaceType55	Rocks
SurfaceType56	Wood
SurfaceType57	Snow
SurfaceType58	Sand
SurfaceType59	Water
SurfaceType60	Metal
SurfaceType61	Grass
SurfaceType62	Concrete

(Physical Surface Settings Inside Unreal Engine)

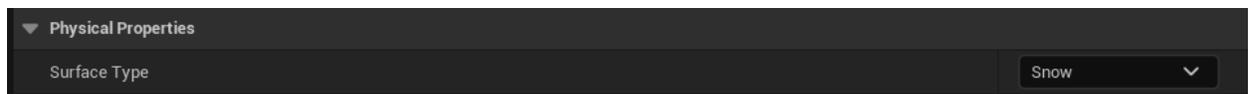
Creating and Linking Physical Material Assets

For each new physical surface you define, you need to create a corresponding **Physical Material Asset**:

1. **Create a Physical Material:**
 - In the Content Browser, right-click and select **Create -> Physics -> Physical Material**.
 - Name the asset to match the surface type (e.g., PM_Mud, PM_Gravel).
2. **Set the Surface Type:**
 - Open the physical material asset.
 - Under the **Physical Properties** section, set the **Surface Type** to your newly defined physical surface.
3. **Assign the Physical Material:**
 - Apply the physical material asset to the appropriate material assets or static meshes in your project.



(Included Physical Material Assets)



(Physical Properties Setting Inside Physical Material Asset)

By completing these steps, you ensure that the Footstep SFX System recognizes and utilizes the new physical surfaces, enabling specific audio effects based on the material properties of your game's environment.

3.2 USING AND CUSTOMIZING METASOUND TEMPLATES

Section Overview

This section explains how to utilize the included MetaSound templates to create and customize your own footstep sound systems. The templates provide powerful tools for building dynamic and flexible audio setups directly within the engine, offering extensive control over pitch, volume, and layering. They also have the included SoundClass and Attenuation assets already assigned.

Included MetaSound Systems and Templates

The Footstep SFX System includes two MetaSound systems and two MetaSound templates:

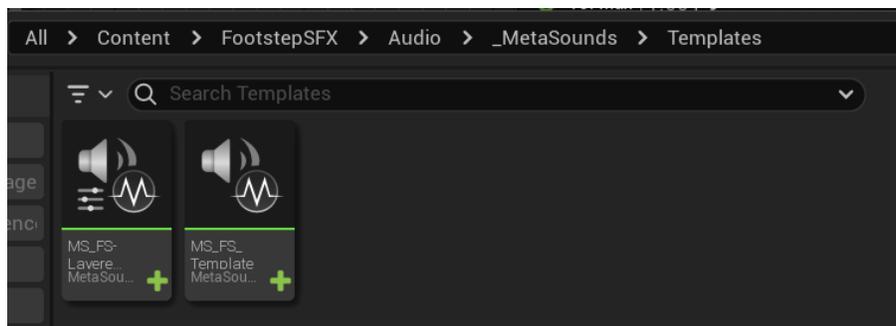
1. MetaSound Systems:

- **Random One-Shot Function:** Plays a random sound from an array of audio files.
- **Layered One-Shot Function:** Plays multiple sound layers simultaneously, with independent control over each layer.

Note: These systems are core functions used by other MetaSounds and should not be modified.

2. MetaSound Templates:

- **MS_FS_Template:** A template based on the Random One-Shot Function.
- **MS_FS_Layered_Template:** A template that combines up to four sound layers, each with independent controls.

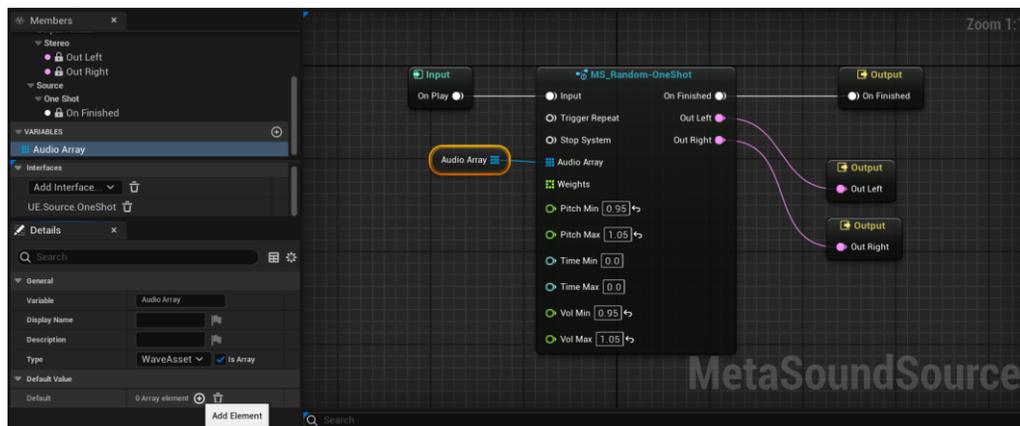


(Included MetaSound Templates)

MS_FS_Template

The **MS_FS_Template** is designed for straightforward sound setups:

- Contains a random one-shot function with an array of audio files.
- To use, simply drop your audio files into the array, and the system will randomly play them.
- Automatically applies small variations in pitch and volume for added realism.
- Exposed settings allow for easy adjustments to pitch, volume, and randomized delays, enabling quick customization for individual systems.



(MS_FS_Template)

MS_FS_Layered_Template

The **MS_FS_Layered_Template** is more advanced, allowing you to combine up to four audio layers:

- Each layer has its own audio array and controls for pitch, delay, and volume.
- Includes master volume controls for each layer, enabling precise sound mixing.

This template is ideal for creating complex sound systems. For example:

- Imagine a scenario with **Wet Concrete** or **Wet Grass** surfaces. You can mix concrete or grass sounds with light water splashes by layering the audio.
- Adjust the volumes to ensure the water sound is subtle, preserving the core surface audio while adding an immersive wet effect.



(MS_FS-Layered_Template)

Customization Options

The layered template is a **MetaSound Source Preset** that requires overriding default values to edit:

1. Select the input you want to modify (e.g., **FS_Layer** for audio arrays, **Master Vol**, **Pitch Min/Max**, **Time Min/Max**, **Vol Min/Max** for each layer).
2. In the details panel, enable **Override Inherited Default** for the selected input.
3. Make custom changes, such as adding audio files to the array or adjusting the default values.

You can also extend min/max ranges for values like the master volume, allowing greater flexibility (e.g., increasing a layer's master volume beyond 1.0).

Benefits of MetaSound Templates

- **Dynamic Layering:** Combine up to four sound layers for unique sound systems directly in the engine.
- **Independent Control:** Adjust pitch, delay, volume, and more for each layer, giving you full creative control.
- **Endless Possibilities:** Generate custom audio mixtures dynamically at runtime for immersive and varied soundscapes.

By leveraging these MetaSound templates, you can create robust and highly customizable footstep sound systems that adapt seamlessly to your game's environments and characters.

3.3 ASSIGNING ADDITIONAL SOUND SYSTEMS FOR CUSTOM SURFACES

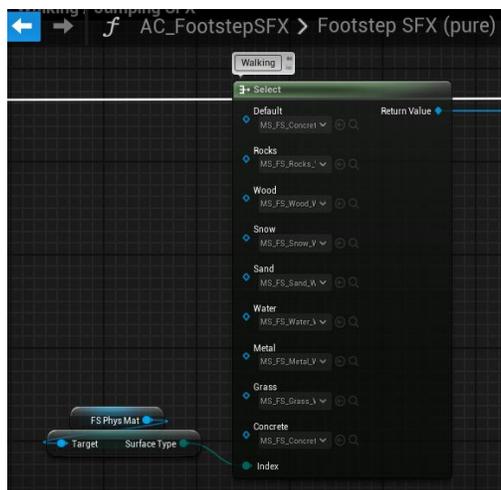
Section Overview

This section explains how to assign newly created sound systems for custom physical surfaces to the **FootstepSFX Component**. By completing this setup, the system will recognize and utilize your custom audio assets for specific surfaces, enhancing the depth and customization of your game's audio.

Assigning Sound Systems to Physical Surfaces

To link your custom sound systems with their corresponding physical surfaces:

1. **Open the AC_FootstepSFX Blueprint:**
 - Navigate to the **AC_FootstepSFX Component** in your project.
2. **Locate the Footstep SFX Function:**
 - Under the **Functions** window, expand the **Data** category.
 - Find the function labeled **Footstep SFX** and open it.
3. **Configure the Select Nodes:**
 - Inside the function, you will see four **Select Nodes** corresponding to different actions: **Walking**, **Jumping**, **Running**, and **Landing**.
 - Each node allows you to assign a sound system to a specific physical surface.
4. **Add Your Custom Physical Surface:**
 - If your custom physical surface is not listed, ensure it has been added to the engine settings as described in [Section 3.1](#).
 - Right-click on the **Select Node** and choose **Refresh Nodes**.
 - Your newly added physical surface should now appear in the dropdown menu.
5. **Assign the Custom Sound System:**
 - For each action (walking, running, jumping, landing), select the appropriate physical surface from the dropdown.
 - Assign your custom sound system (e.g., sound cues or MetaSound assets) to each surface.



(Select Nodes Used For Assigning Sound Systems To Physical Surfaces)

Best Practices

- **Test Your Assignments:** After assigning the sound systems, test them in your game to ensure the correct sounds are playing for each physical surface.
- **Use Consistent Naming:** Maintain clear and descriptive names for your custom physical surfaces and sound systems to make configuration easier.
- **Leverage MetaSound Templates:** If your custom surface involves complex audio requirements, consider using the **MetaSound templates** described in [Section 3.2](#).

Example Use Case

Imagine you've added a custom surface called **Wet Grass** and created a unique sound system for it. By following the steps above:

- You can ensure that whenever the player walks, runs, jumps, or lands on **Wet Grass**, the custom sound system will be triggered.
- The result is a seamless and immersive audio experience that dynamically responds to the environment.

By assigning your custom sound systems to the **FootstepSFX Component**, you unlock the full potential of your game's audio design, enabling tailored soundscapes for any physical surface in your project.

4.0 HUD CONTROLS

Section Overview

This section explains how to use the controls displayed on the player HUD. These controls are provided as part of the demo to help showcase the capabilities of the Footstep SFX System. While not integral to the core functionality of the system, they are useful for testing and demonstrating specific features during development.

Controls Overview

- **To Run:** Hold the **Shift** key to make the player character run. This allows you to test the transition between walking and running sound effects.
- **To Solo Footstep Audio:** Press the **"1"** key to isolate footstep sounds, muting all other audio. This is useful for focusing on the footstep SFX without distractions from background sounds.
- **To Visualize Audio:** Press the **"2"** key to enable or disable audio visualization. This feature shows real-time visual feedback for audio playback, helping you analyze the timing and volume of footstep sounds.
- **To Toggle "Spawn Attached":** Press the **"3"** key to toggle the **Spawn Attached** setting on or off. This lets you test how footstep audio behaves when attached to the player versus when it's spawned at the line trace location.
- **To Spawn/Despawn AI Enemy:** Press the **"4"** key to spawn or despawn an AI enemy. This allows you to test how the Footstep SFX System works with AI-controlled characters.



(HUD Controls)

Purpose and Use

These controls are designed to make it easier to test and demonstrate the system’s functionality in various scenarios:

- **Real-Time Adjustments:** Quickly switch between different settings to observe how they affect footstep audio.
- **Debugging and Testing:** Identify potential issues with sound playback or settings.
- **Showcase Features:** Demonstrate the system’s flexibility and adaptability during presentations or testing sessions.

Best Practices

- Use the **Solo Footstep Audio** and **Visualize Audio** features to fine-tune the timing and volume of your footstep sounds.
- Experiment with the **Spawn Attached** toggle to decide which mode works best for your project.
- Utilize the **Spawn/Despawn AI Enemy** option to ensure the system functions properly for AI characters in addition to player characters.

By using these HUD controls, you can efficiently test and refine the Footstep SFX System, ensuring it performs optimally in your project.

5.0 FAQ – COMMON ISSUES AND EASY FIXES

Section Overview

This section addresses frequently asked questions, common issues, and their solutions. It will be expanded as the system evolves and we gather feedback from users.

Q: I just migrated this system to my existing project, and I'm getting compilation errors in the AC_FootstepSFX Blueprint Component. Why is this happening?

A: This issue is likely due to missing physical surface settings in your engine. Refer to [Section 1.1](#) for instructions on setting up physical surface settings. After completing the setup, refresh the nodes with errors, and the problem should be resolved.

Q: I'm using Automatic Mode, but the timing of the footsteps is off. How do I fix this?

A: Ensure you're using the **Walk/Run Rate** variables and the **Max Running Speed** setting as outlined in [Section 2.2](#). These settings are crucial for achieving proper footstep timing in Automatic Mode.

That said, Automatic Mode is designed primarily for prototyping or characters where animation timing is less critical, such as first-person characters. For third-person or AI characters, where precise animation timing is essential, we recommend switching to **Standard Mode** and setting up Animation Notifies as described in [Section 2.3](#).

Q: Why aren't my footstep sounds playing at all?

A: Ensure the Footstep SFX Component is properly attached to your character. If using Standard Mode, confirm that Animation Notifies are correctly placed and aligned with your character's footstep animations, as described in [Section 2.3](#). If using Automatic Mode, check the Walk/Run Rates and Max Running Speed settings as outlined in [Section 2.2](#).

Q: My custom physical surface isn't triggering the correct sounds. What did I miss?

A: Verify that your custom physical surface is properly defined and added to the **DefaultEngine.ini** file or through the engine's Physics settings, as explained in [Section 3.1](#). Also, ensure you've created a corresponding Physical Material Asset and assigned it to the material or static mesh in your project.

Q: Why is my custom sound system not playing for my custom surface?

A: Make sure your custom sound system is assigned to the correct physical surface in the **Footstep SFX** function inside the **AC_FootstepSFX Blueprint**, as detailed in [Section 3.3](#). If the surface doesn't appear in the Select Nodes, refresh them by right-clicking the node and choosing **Refresh Nodes**.

Q: How can I mix physical surface sounds with custom character sounds?

A: Enable **Dual Functionality** in the Footstep SFX Component, as explained in [Section 2.4](#). This allows you to combine physical surface sounds (e.g., snow, concrete) with character-specific sounds for a more immersive experience.

Q: Why are my footsteps cutting off when transitioning between audio volumes?

A: If you're using multiple audio volumes in your map, consider enabling the **Spawn Attached** option, as described in [Section 2.5](#). This ensures footstep sounds follow the character, avoiding abrupt audio culling caused by leaving the audio volume boundary.

Q: Can I use this system for non-human characters like animals or robots?

A: Absolutely! The Footstep SFX System is highly flexible and can be customized for any character type. You can:

- **Use the Default Physical Surface Sounds for general environmental effects.**
- **Assign unique sounds specific to the character's nature (e.g., robotic clangs, animal paws) using the Custom SFX settings ([Section 2.7](#)).**
- **Combine both using Dual Functionality Mode, allowing custom character sounds to play alongside physical surface sounds for a more dynamic and immersive audio experience.**

Q: Why are my footstep sounds too quiet or too loud?

A: Adjust the **Default Volume Multiplier** or **Custom Volume Multiplier** in the Footstep SFX Component, as outlined in [Section 2.6](#). These settings allow you to fine-tune the volume of physical surface and custom sounds independently.

Have additional questions?

If you encounter issues or have questions not listed here, feel free to join our public Discord community. We're happy to help and may even include your questions in future updates to this documentation.

6.0 PATCH NOTES & UPDATES

Section Overview

This section provides a record of patch notes and updates for the Footstep SFX System. Here, you'll find details about system changes, new features, and compatibility improvements to help you stay up to date with the latest developments.

December 21, 2024

- **Version:** 1.0
- **Description:** Initial release of the Footstep SFX System and documentation.
 - Compatible with Unreal Engine **5.4** and **5.5**.
 - Includes core functionality, such as Automatic and Standard Modes, Custom SFX settings, and support for physical surfaces.
 - Features two MetaSound templates (**MS_FS_Template** and **MS_FS_Layered_Template**) for creating dynamic and customizable sound systems.
 - Comprehensive documentation covering setup, customization, and troubleshooting.

Future Updates

As the system evolves, this section will be updated with patch notes for new features, enhancements, and bug fixes. If you have suggestions or encounter issues, please join our public Discord to provide feedback—we welcome your input to improve the system.