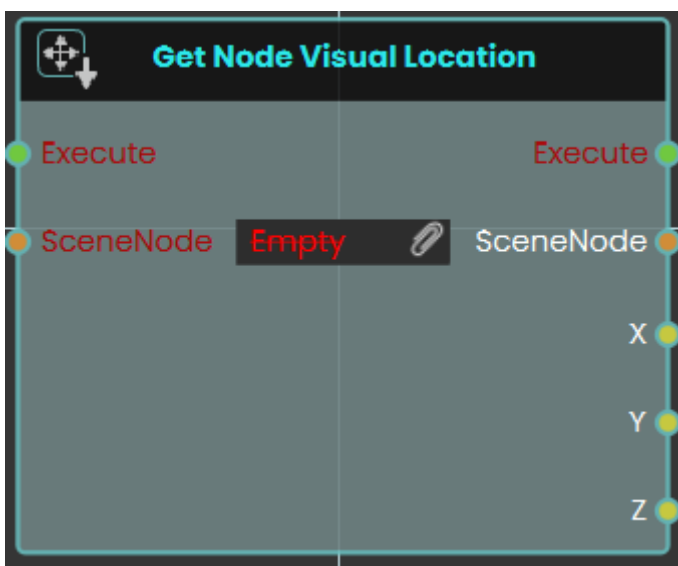


SceneNode \ Transform \ Location

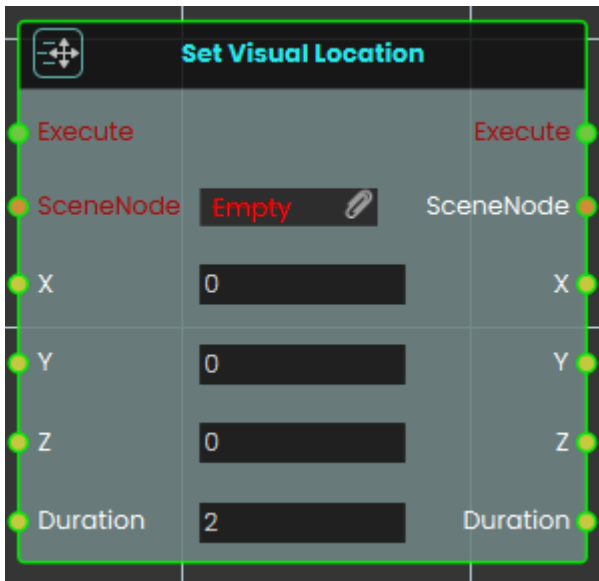
📍 Visual \ Get Node Visual Location



The **Get Node Visual Location** node enables the system to retrieve the exact X, Y, and Z coordinates of a SceneNode based on its absolute visual representation in the 3D scene.

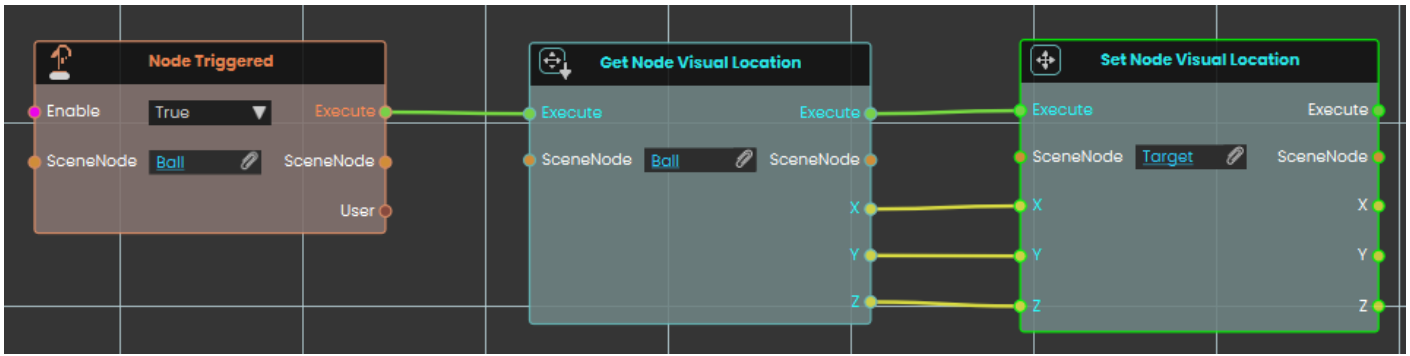
Unlike standard location nodes that calculate position relative to a parent object, this node fetches the true visual position of the object within the environment, completely unaffected by the scene's parent-child hierarchy system.

Visual \ Set Visual Location



The **Set Node Visual Location** node enables the system to instantly move a SceneNode to specific X, Y, and Z coordinates within the absolute visual space of the 3D scene. By directly altering where the object appears visually, this node completely bypasses any positional offsets or transformations that the object might normally inherit from its parent hierarchy.

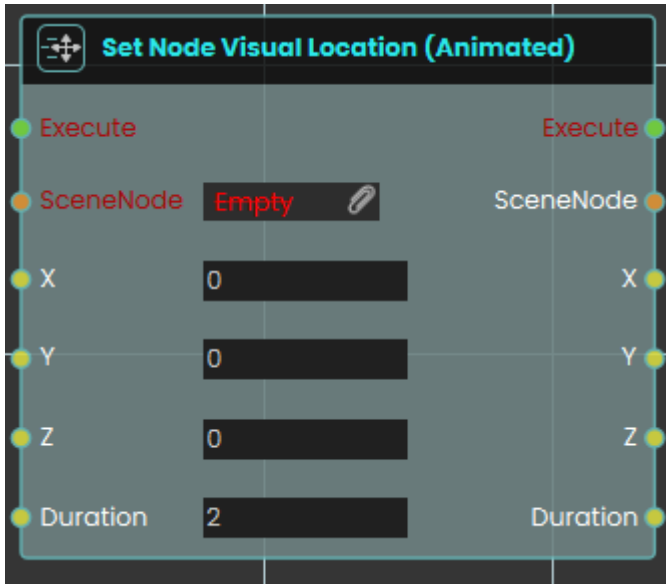
Example



In this example, an interaction is set up to match the positions of two independent objects.

1. Upon interacting with the "Ball" object, the **Node Triggered** event initiates the execution flow.
2. The **Get Node Visual Location** node extracts the true visual X, Y, and Z coordinates of the "Ball" in the scene.
3. These coordinate values are then fed directly into the X, Y, and Z inputs of a **Set Node Visual Location** node assigned to a "Target" object. This instantly moves the Target to the exact visual location of the Ball, regardless of how deeply nested either object is within the scene's hierarchy.

Visual \ Set Node Visual Location (Animated)



The **Set Node Visual Location (Animated)** node enables the system to smoothly move a SceneNode to specific X, Y, and Z coordinates within the absolute visual space of the 3D scene over a set period.

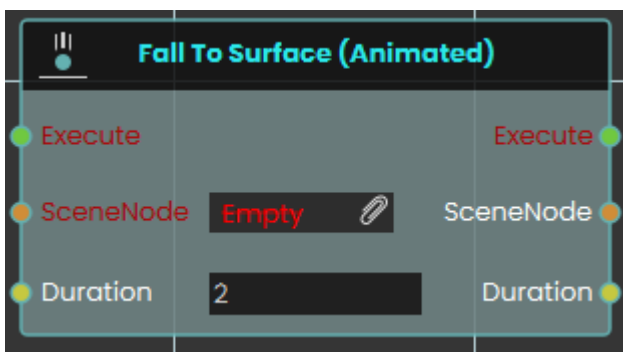
Similar to the standard **Set Node Visual Location** node, this movement completely bypasses any positional offsets or transformations inherited from the object's parent hierarchy. However, this animated variant includes an additional **Duration** parameter. Instead of instantly snapping the object to the new coordinates, this node creates a smooth transition (tweening) from its current visual location to the target location over the specified number of seconds.

Fall To Surface



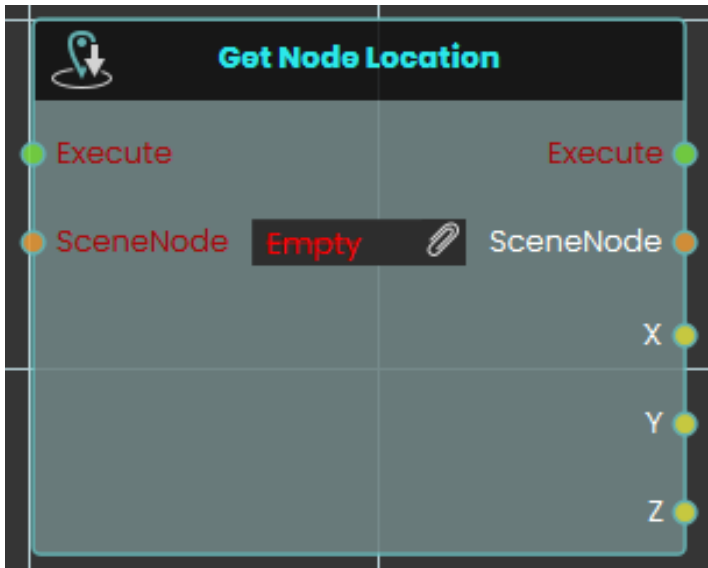
The **Fall To Surface node** instantly repositions a specific 3D object directly onto the nearest 3D surface located immediately below it in the scene. When activated, the node takes the targeted SceneNode input and immediately updates its vertical placement to simulate a sudden, instantaneous drop, resting the object perfectly against the underlying geometry.

Fall To Surface (Animated)



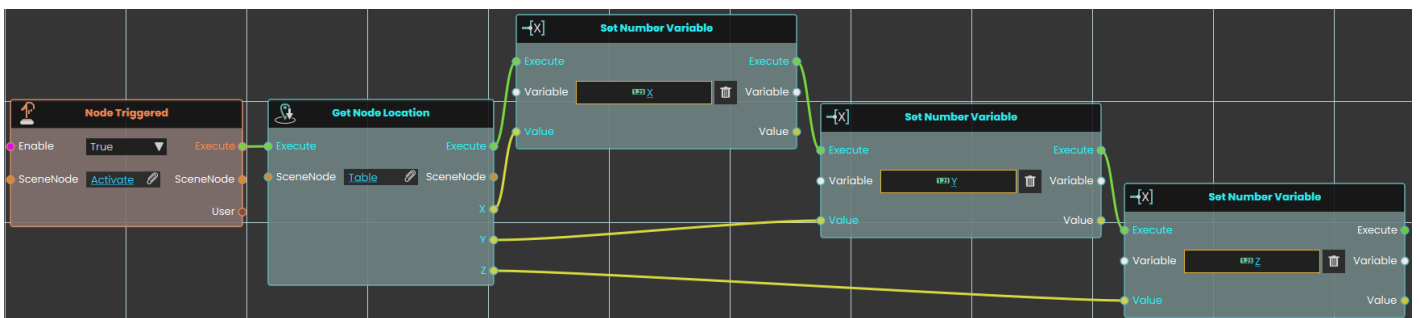
The **Fall To Surface (Animated) node** functions similarly to the standard fall node, but instead of an instant drop, it smoothly transitions the 3D object down to the nearest underlying surface over a defined period. Once activated, the node takes the targeted SceneNode and seamlessly tweens its vertical descent until it rests on the geometry below, completing the fall animation over the exact amount of time specified by the numerical Duration input (measured in seconds).

Get Node Location



The **Get Node Location** response obtains the X, Y, and Z coordinates of an object's location each time the event connected to it is triggered. Once the response is executed, the coordinates can be acquired through the **X**, **Y**, and **Z** ports.

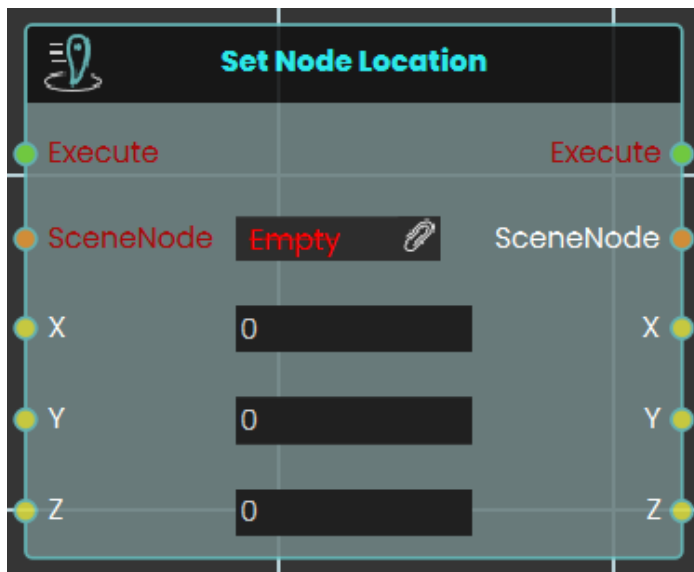
Example



In this example, a **Get Node Location** response is used to obtain the X, Y, and Z coordinates of an object named Table once the user triggers the object named Activate. The coordinates are then stored in variables named X, Y, and Z, which can be connected

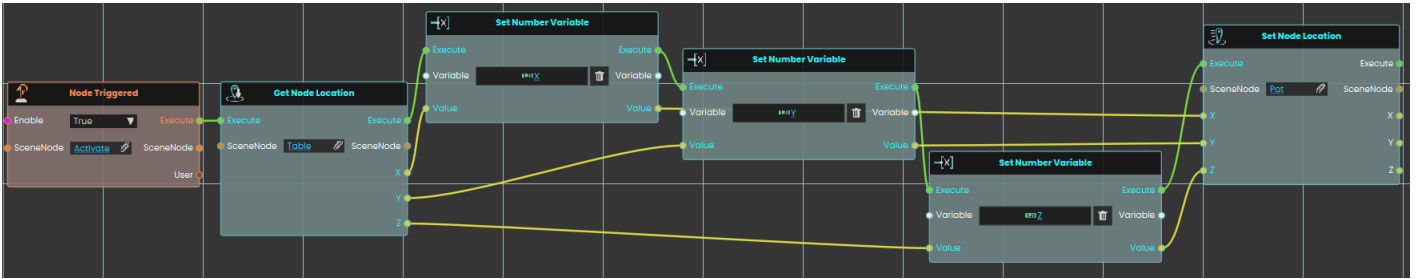
to a variable writer to be displayed during the VR Experience.

Set Node Location



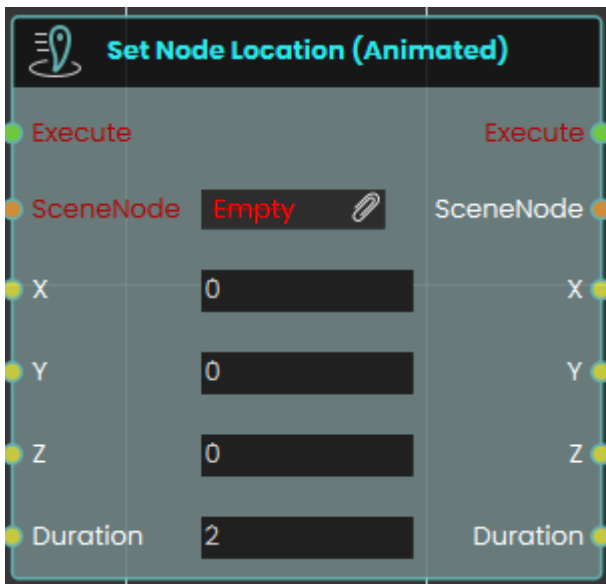
The **Set Node Location** response enables the user to set the location of an object to specific X, Y, and Z coordinates by either typing in the numerical values or by attaching a variable node to those ports. Once the response is executed, the object's new location is applied using the provided coordinates.

Example



In this example, a **Set Node Location** response is used to assign the X, Y, and Z coordinates stored in variables named X, Y, and Z to an object named Pot once the user triggers the object named Activate. The object's new location is set based on these coordinates, allowing the updated position to be reflected during the VR Experience.

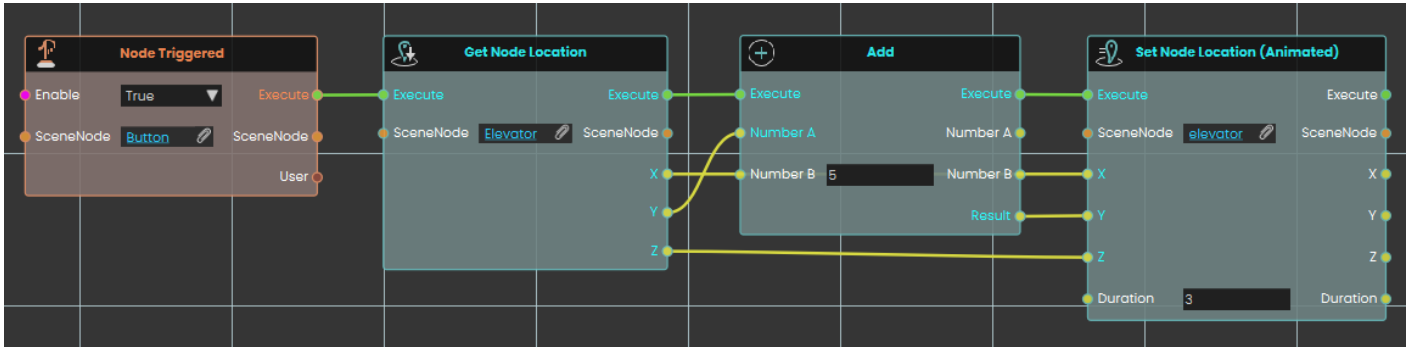
Set Node Location (Animated)



The **Set Node Location (Animated)** node enables the system to smoothly move a SceneNode to a specific global coordinate in the 3D space over a set period. this node creates a seamless tweening effect from the object's current position to its new target destination

This differs significantly from the standard **Set Node Location** node, which instantly snaps or teleports the object to the new coordinates without any transition. The animated variant is ideal for creating moving platforms, sliding doors, or moving vehicles.

Example



In this example, an interactive elevator mechanism is created.

1. Upon interacting with the button object, the **Node Triggered** event initiates the execution flow.
2. The **Get Node Location** node is used to retrieve the current X, Y, and Z global coordinates of the "Elevator" object.
3. To move the elevator up, the Y-axis coordinate is passed into an **Add** node, where 5 units are added to its current value. The X and Z coordinates remain unchanged.
4. Finally, the **Set Node Location (Animated)** node applies these new coordinates to the elevator object. The **Duration** parameter is set to 3, meaning it will take exactly 3 seconds for the elevator to smoothly transition (tween) to its new position 5 units higher.