

# Camera Animation

**Helper function prefix:** CameraAnimation\_, CameraAnimationNode\_

## General

The camera animation system is used to move the player camera along a specific path while still allowing some degree of interaction.

## Instructions

To use the system, place CameraAnimation nodes (Area → CameraAnimation) in the scene and give each of them the name of the animation followed by an underscore and their placement within the animation (such as WakeUpAnimation\_1, WakeUpAnimation\_2 etc). Note that if you duplicate (Ctrl-D) a node, the duplicate will automatically get the next available number as its suffix.

Rotate the nodes so that their Z-axes point in the direction you want the camera to look at that part of the animation.

In script, call the helper function CameraAnimation\_Begin() to start the animation.

## Node properties

### MaxYaw/MaxPitch

Dictates how far the player is allowed to rotate the camera away from the target forward rotation. This value is interpolated between nodes.

### LookSpeedMultiplier

How fast the player can rotate the camera.

### ForwardTime

The time it takes to move from this node to the next at full speed forward.

### BackwardTime

The time it takes to move from the next node to this one at full reverse speed.

### AutoMovement

Dictates how the animation behaves when no movement input is given, or when movement input is disabled. 0 stops movement, 1 is as if given full forward speed (animation will finish after [ForwardTime] seconds), -1 is as if given full reverse speed.

### LookMoveDistance

The length between this node and the next the player can move just by looking toward the next node, given as a value between 0 and 1.

### LookMoveMaxAngle

The maximum angle away from the next node the player can look and still move forward. Only valid if LookMoveDistance > 0.

## **InteractiveMovement**

If checked, the player can control the movement between this node and the next. If unchecked, the AutoMovement property has complete control over the movement and should not be set to 0.

## **CrouchOnExit**

If checked, the player will automatically crouch if the animation is ended at this node.

## **InitToCamera**

If checked, the node is given the orientation of the camera when the animation starts. This is very useful when you want the animation to have a smooth beginning, or you want to make sure the player is returned to where they were when the end node is reached.

## **UsePlayerEyeline**

If ticked, instead of using the Y-value of the node itself, the camera system uses the Y-value that the player's eyeline should be at that point (i.e. from the ground/floor upwards).

## **CallbackFunc**

Function in the level script that should be called when reaching this node. The syntax is bool FuncName(const tString &in asEntity, int alDirection), where asEntity is the name of the node and alDirection is 1 if moving forward and -1 if moving backward.

## **Important Functions**

### **CameraAnimation\_Begin**

Start a camera animation.

### **CameraAnimation\_End**

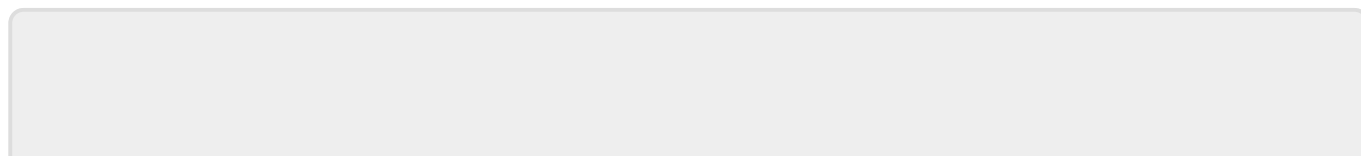
Stop a camera animation.

### **CameraAnimationNode\_SetAutoMove**

Allows you to pause or continue animation at a particular node.

### **CameraAnimationNode\_SetInteractiveMovement**

Allows you to enable the player to progress within a particular node.



From:  
<https://oldwiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link:  
[https://oldwiki.frictionalgames.com/hpl3/game/scripting/user\\_modules/camera\\_animation?rev=1442406200](https://oldwiki.frictionalgames.com/hpl3/game/scripting/user_modules/camera_animation?rev=1442406200)

Last update: **2015/09/16 13:23**

