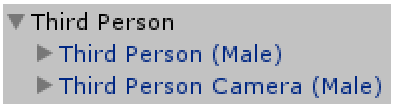
Third Person Character

Overview

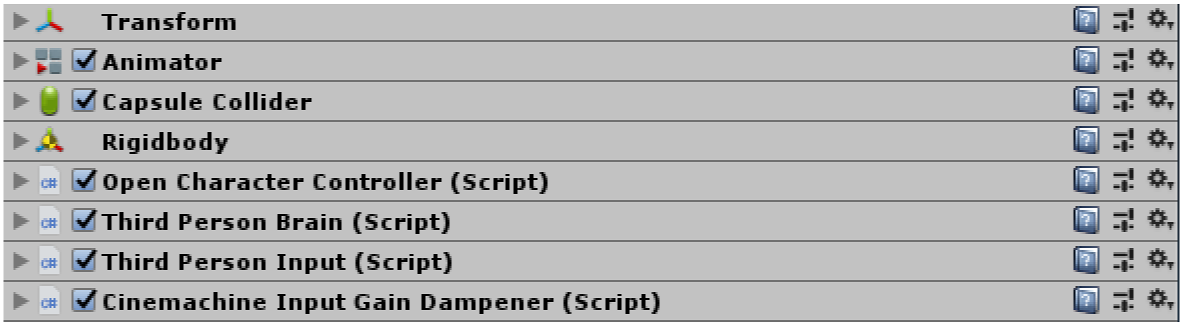
The Third Person character is by far the most complex system in the game, because of the advanced animation setup. It is similar to the First Person character in that it consists of three prefabs:



▼ Third Person
  ▶ Third Person (Male)
  ▶ Third Person Camera (Male)

1. **Third Person (Male/Female)** - contains the Character Brain and the humanoid avatar.
2. **Third Person Camera (Male/Female)** - management of the third person Cinemachine cameras.

Third Person prefab

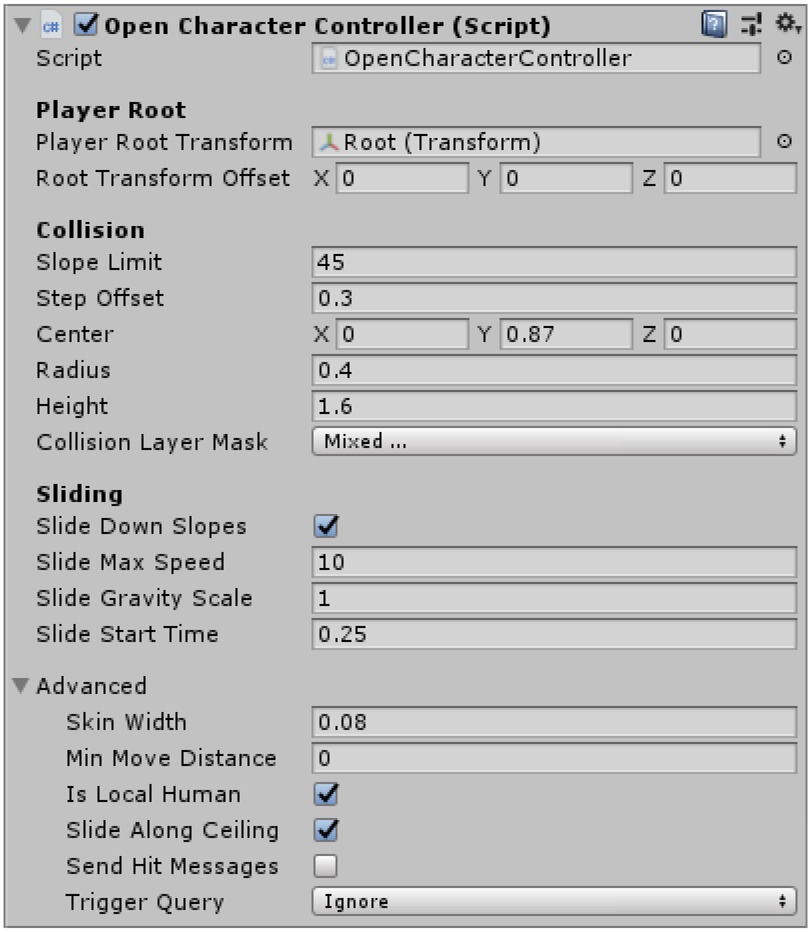


▶ Transform
▶ ☑ Animator
▶ ☑ Capsule Collider
▶ Rigidbody
▶ ☑ Open Character Controller (Script)
▶ ☑ Third Person Brain (Script)
▶ ☑ Third Person Input (Script)
▶ ☑ Cinemachine Input Gain Dampener (Script)

- **Animator** - the animator for handling the humanoid animations
- **Rigidbody, Capsule Collider and Open Character Controller** - handles the physical movement of the character. See the Open Character Controller documentation for more information
- **Third Person Brain** - the control component that ties input to animation to movement
- **Third Person Input** - the cross platform input system, see the documentation for more information.
- **Cinemachine Input Gain Dampener** - Cinemachine axis speed mode control.

> The Third Person Character derives its motion from the animations (Root Motion) allowing for minimal foot-sliding and more accurate movement.

Open Character Controller



☑ Open Character Controller (Script)

Script: OpenCharacterController

Player Root

Player Root Transform: Root (Transform)

Root Transform Offset: X 0 Y 0 Z 0

Collision

Slope Limit: 45

Step Offset: 0.3

Center: X 0 Y 0.87 Z 0

Radius: 0.4

Height: 1.6

Collision Layer Mask: Mixed ...

Sliding

Slide Down Slopes: ☑

Slide Max Speed: 10

Slide Gravity Scale: 1

Slide Start Time: 0.25

Advanced

Skin Width: 0.08

Min Move Distance: 0

Is Local Human: ☑

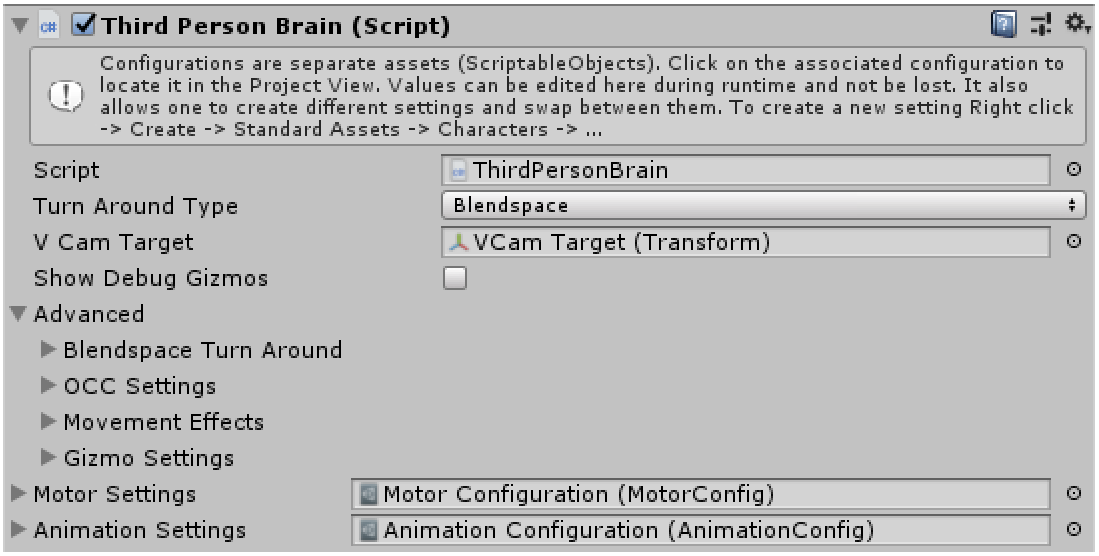
Slide Along Ceiling: ☑

Send Hit Messages: ☐

Trigger Query: Ignore

See the Open Character Controller documentation for more information.

Third Person Brain



☑ Third Person Brain (Script)

Configurations are separate assets (ScriptableObjects). Click on the associated configuration to locate it in the Project View. Values can be edited here during runtime and not be lost. It also allows one to create different settings and swap between them. To create a new setting Right click -> Create -> Standard Assets -> Characters -> ...

Script: ThirdPersonBrain
Turn Around Type: Blendspace
V Cam Target: VCam Target (Transform)
Show Debug Gizmos: □
Advanced
- Blendspace Turn Around
- OCC Settings
- Movement Effects
- Gizmo Settings

Motor Settings: Motor Configuration (MotorConfig)
Animation Settings: Animation Configuration (AnimationConfig)

Third Person Brain has the following concepts:

- **Third Person Motor** - handles the physical movement of the character
  - The Motor Settings ScriptableObject can be accessed from the ThirdPersonBrain
- It provides the **Animator** with conditioned parameters for animation
  - The **AnimationSettings** ScriptableObject can be accessed from the ThirdPersonBrain
- **Turnarounds**: how the character handles rapid turns (e.g. 90 degree or 180 degree turns)
  - **Turnaround Type**: turnarounds can either be responsive using the Blendspace or they can be realistic using Animation
  - **Turnaround Behaviour**: shows the configuration based on which Turnaround Type is selected
- **Third Person Movement Event Handler** - the mechanism of broadcasting footsteps and movement effects such as sounds and particles during jumping and landing. For more information on Movement Events see the documentation
- **Character Controller Adapter** - handles movement via the Open Character Controller, this is described in detail in the physic documentation.

Third Person Motor

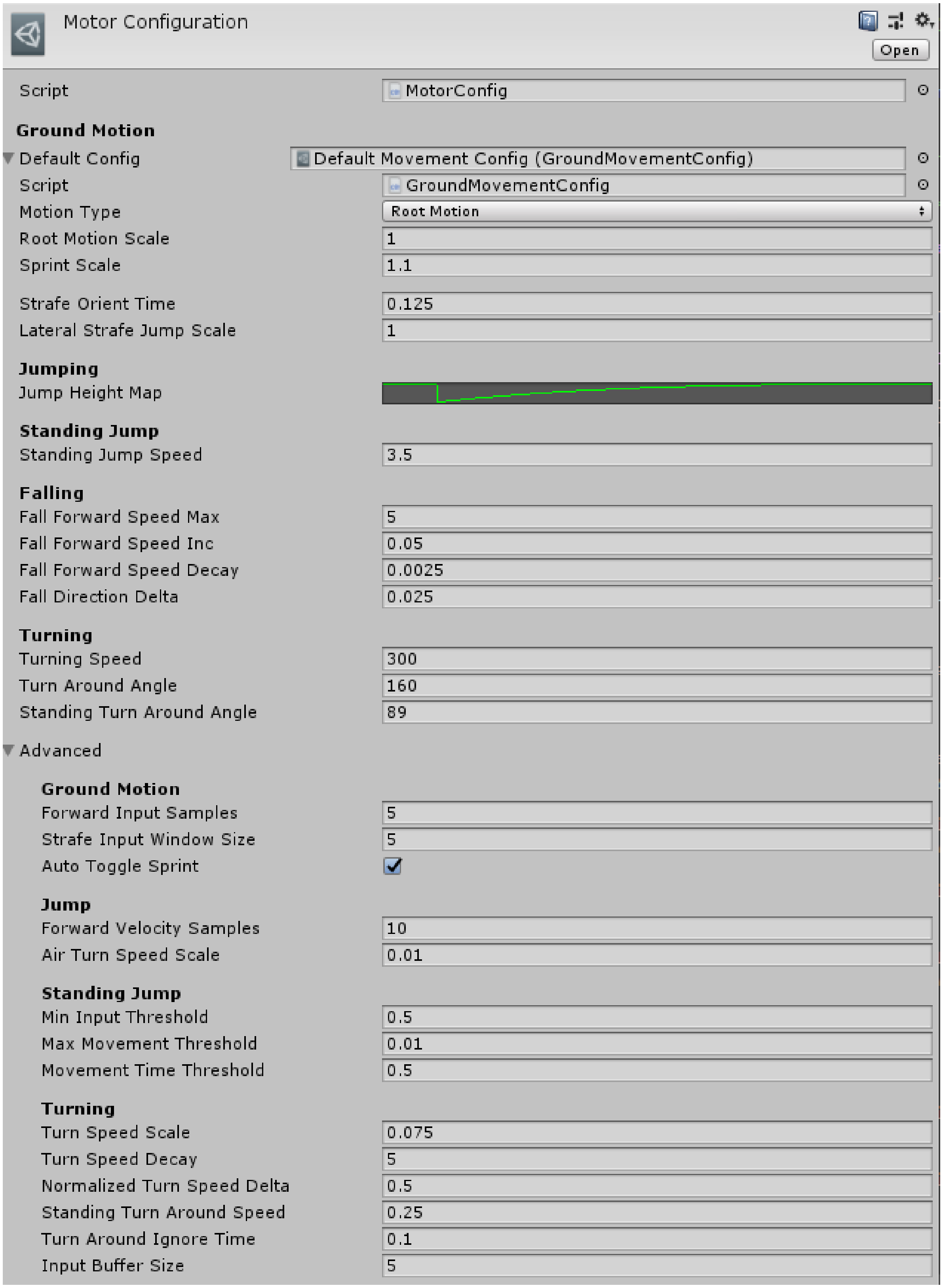
The Third Person Motor handles the character movement state:

- **Exploration Movement -** the forward of the character is not locked to the camera and the character turns to face the direction of the movement input relative to the camera. E.g. Breath Of The Wild
- **Strafe Movement -** the forward of the character is locked to the camera and the character will strafe when lateral movement input is applied. E.g. Gears Of War.

The Third Person Motor also handles the translation of the character in physical space. This is done during the OnAnimatorMove method as the character moves using Root Motion to ensure realistic movement and minimise foot-sliding. The Third Person Motor also handles movement in the aerial states (jumping and falling), which does not use the Root Motion of those animations, but rather derives the movement from the movement at the moment that the character becomes ungrounded. The Third Person Motor has a ScriptableObject for its configuration, this can be located on the ThirdPersonBrain.

> By using ScriptableObjects for configuration, there is a separate asset that can be edited at run-time, which will allow changes to persist. Another advantage is the ability to create different setups (e.g. realistic vs arcade movement) that can be swapped quickly and easily.

Motor Configuration



Script	MotorConfig
Ground Motion	
▼ Default Config	Default Movement Config (GroundMovementConfig)
Script	GroundMovementConfig
Motion Type	Root Motion
Root Motion Scale	1
Sprint Scale	1.1
Strafe Orient Time	0.125
Lateral Strafe Jump Scale	1
Jumping	
Jump Height Map	
Standing Jump	
Standing Jump Speed	3.5
Falling	
Fall Forward Speed Max	5
Fall Forward Speed Inc	0.05
Fall Forward Speed Decay	0.0025
Fall Direction Delta	0.025
Turning	
Turning Speed	300
Turn Around Angle	160
Standing Turn Around Angle	89
▼ Advanced	
Ground Motion	
Forward Input Samples	5
Strafe Input Window Size	5
Auto Toggle Sprint	☑
Jump	
Forward Velocity Samples	10
Air Turn Speed Scale	0.01
Standing Jump	
Min Input Threshold	0.5
Max Movement Threshold	0.01
Movement Time Threshold	0.5
Turning	
Turn Speed Scale	0.075
Turn Speed Decay	5
Normalized Turn Speed Delta	0.5
Standing Turn Around Speed	0.25
Turn Around Ignore Time	0.1
Input Buffer Size	5

The Motor Configuration has the following fields:

Ground Motion

- **Default Config** - ScriptableObject containing the config that will drive ground motion. This default can be overridden by setting be desired config on the corresponding Locomotion Animator State.
  - The default behaviour is set to **Root Motion**. This will use the animations' motion to move the character. The **Root Motion Scale** can be used to adjust the movement before applying to to the character. The **Sprint Scale** is the value used to modify normalized forward speed while sprinting to allow a sprint animation to play with in the locomotion blend tree.
  - The other option for movement is **Specified Speed**. This will define the character's **Max Speed** relative to input. **Max Speed Type** can be set to **Float** or **Curve**. The **Speed Delta** defines the speed at which the character's current movement speed is allowed to move towards the max speed. The **Sprint Scale** is the value used to modify the max speed while sprinting.
- **Strafe Orient Time -** time it takes for the character to turn and face the camera orientation when Strafe Mode has been entered
- **Lateral Strafe Jump Scale -** this is a scale applied to the speed of a lateral strafe jump.

Jumping

- **Jump Height Map** - how high the character jumps based on the normalized forward speed.

Standing Jump

- **Standing Jump Speed** - forward speed applied during standing jump.

Falling

- **Fall Forward Max Speed** - the maximum forward speed of the character, while falling
- **Fall Forward Speed Inc** - the rate at which falling forward speed can increase
- **Fall Forward Speed Decay**- the rate at which falling forward speed can decrease
- **Fall Direction Delta** - the speed at which fall direction can change.

Turning

- **Turning Speed** - the degrees per second that the character can turn
- **Turn Around Angle** - the minimum angle required to trigger a turnaround while the character is moving
- **Stationary Turn Around Angle** - the minimum angle required to trigger a turnaround while the character is stationary.

Advanced Settings

Adjusting these settings without having a full understanding of their function could yield unexpected results. Defaults are recommended.

Ground Motion

- **Sprint Speed Modifier** - Used to extend the locomotion blend tree during sprinting, setting the normalized speed to 1 + this value
- **Forward Input Samples** - the number of samples for averaging forward input. Increasing this results in a smoother movement at the sacrifice of responsiveness
- **Strafe Input Window Size** - the number of samples for averaging the strafe input. Increasing this results in a smoother movement at the sacrifice of responsiveness
- **Auto Toggle Sprint** - indicates whether or not sprint should auto disable when the left analog stick of a gamepad input device is released.

Jumping

- **Jump Velocity Samples** - the number of samples used to track the current forward velocity of the character, to be used as the forward velocity while jumping
- **Air Turn Speed Scale** - how much of the Turning Speed is applied during the turn.

Standing Jump

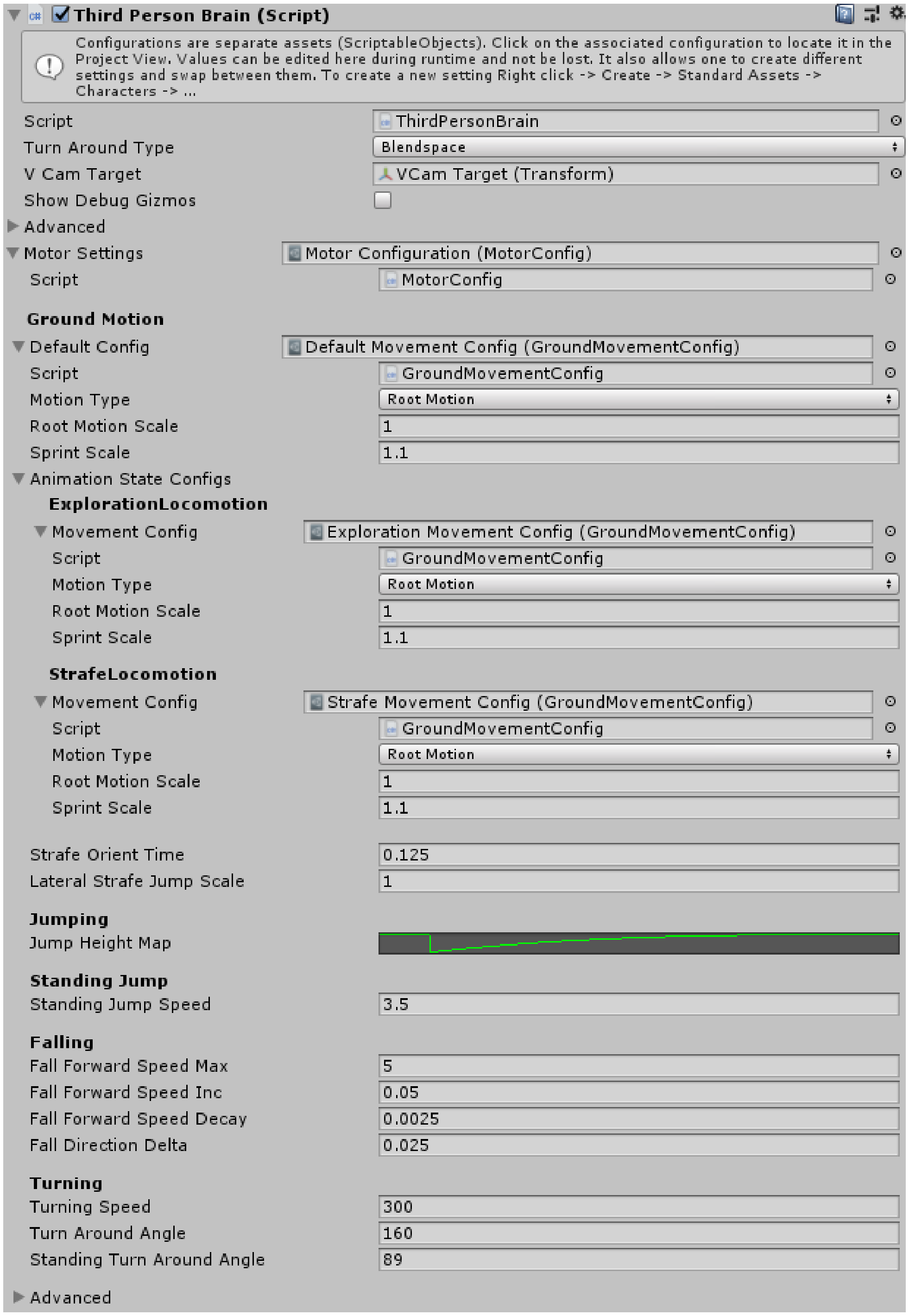
- **Min Input Threshold** - the minimum input allowed to trigger a standing forward jump
- **Max Movement Threshold** - the maximum character forward movement that can trigger a standing jump
- **Movement Time Threshold** - How long after a character starts moving that a standing jump can still be initiated.

Turning

- **Turn Speed Scale** - used for affecting how much of the -1 to 1 range of normalizedTurningSpeed is used
- **Turn Speed Decay** - Rate at which normalized turn speed will return to zero when there is no turn input
- **Normalized Turn Speed Delta** - the speed at which the normalized turning speed can change
- **Standing Turnaround Speed** - A forward movement less than this would allow a standing turnaround
- **Turn Around Ignore Time** - time in seconds that input will be ignored after a rapid turn
- **Input Buffer Size** - the number of frames of input that will be used to determine if a turnaround was triggered.

Modifying the Motor Configuration

The Motor Configuration can be modified directly on the ScriptableObject asset or through the ThirdPersonBrain component. If the latter is your method of choice extra options are added to configure the Animation State Configs found in the Animator on the LocomotionAnimatorStates.



Third Person Brain (Script)

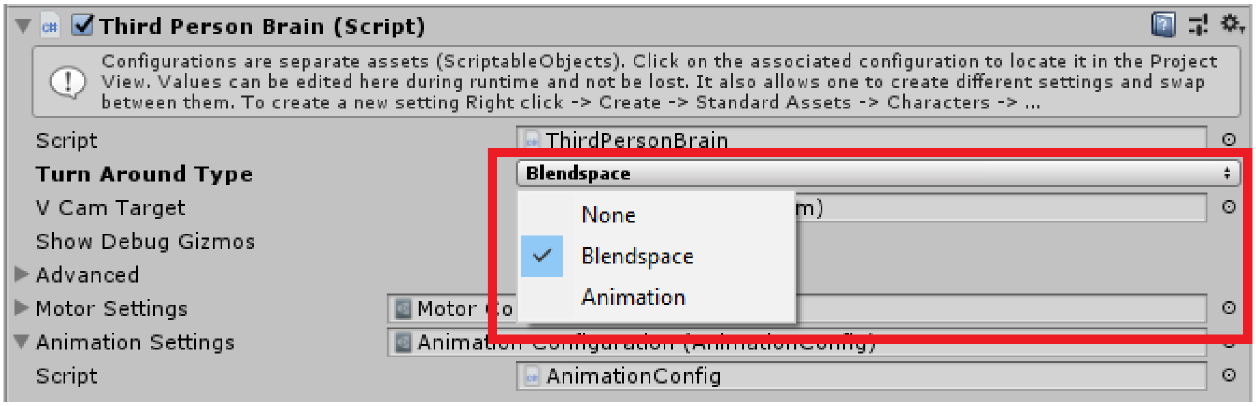
Configurations are separate assets (ScriptableObjects). Click on the associated configuration to locate it in the Project View. Values can be edited here during runtime and not be lost. It also allows one to create different settings and swap between them. To create a new setting Right click -> Create -> Standard Assets -> Characters -> ...

Script	ThirdPersonBrain
Turn Around Type	Blendspace
V Cam Target	VCam Target (Transform)
Show Debug Gizmos	□
Advanced	
Motor Settings	Motor Configuration (MotorConfig)
Script	MotorConfig
Ground Motion	
Default Config	Default Movement Config (GroundMovementConfig)
Script	GroundMovementConfig
Motion Type	Root Motion
Root Motion Scale	1
Sprint Scale	1.1
Animation State Configs	
ExplorationLocomotion	
Movement Config	Exploration Movement Config (GroundMovementConfig)
Script	GroundMovementConfig
Motion Type	Root Motion
Root Motion Scale	1
Sprint Scale	1.1
StrafeLocomotion	
Movement Config	Strafe Movement Config (GroundMovementConfig)
Script	GroundMovementConfig
Motion Type	Root Motion
Root Motion Scale	1
Sprint Scale	1.1
Strafe Orient Time	0.125
Lateral Strafe Jump Scale	1
Jumping	
Jump Height Map	
Standing Jump	
Standing Jump Speed	3.5
Falling	
Fall Forward Speed Max	5
Fall Forward Speed Inc	0.05
Fall Forward Speed Decay	0.0025
Fall Direction Delta	0.025
Turning	
Turning Speed	300
Turn Around Angle	160
Standing Turn Around Angle	89
Advanced	

Turnaround

Turnaround is how the character handles rapid (90/180 degree) turns. There are three modes of operation for turnarounds:

1. **None** - the character ignores rapid turns and slowly turns in an arch
2. **Blendspace** - the character uses the locomotion blend-tree (described in the animation setup), as well as turning the character. This is for a character where responsiveness is prioritised over visual fidelity
3. **Animation** - the character uses animations setup for turnarounds. This is for a character where visual fidelity is prioritised over responsiveness.



Third Person Brain (Script)

Configurations are separate assets (ScriptableObjects). Click on the associated configuration to locate it in the Project View. Values can be edited here during runtime and not be lost. It also allows one to create different settings and swap between them. To create a new setting Right click -> Create -> Standard Assets -> Characters -> ...

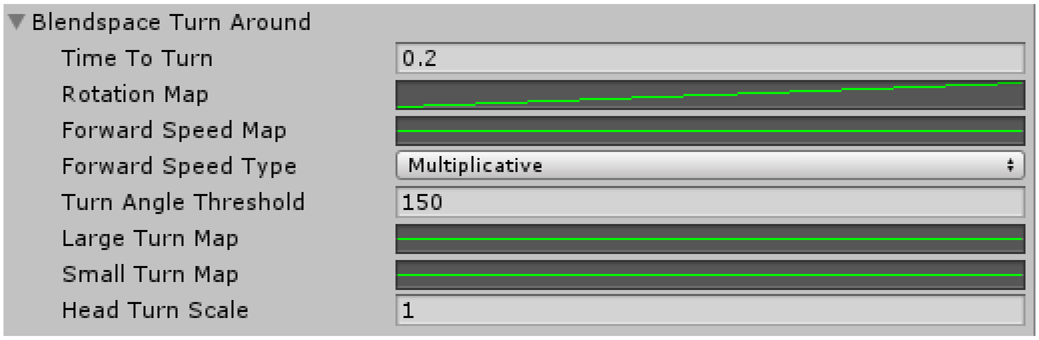
Script	ThirdPersonBrain
Turn Around Type	Blendspace
V Cam Target	m)
Show Debug Gizmos	
Advanced	
Motor Settings	Motor Co
Animation Settings	Animation Configuration (AnimationConfig)
Script	AnimationConfig

In the inspector only the configuration settings for the selected type are displayed.

Blendspace Turnaround Behaviour

The Blendspace Turnaround Behaviour manipulates two animation parameters that affect the locomotion blend-tree:

1. **TurningSpeed** - a fixed value, which influences how much lean the character will have for the duration of the turnaround
2. **ForwardSpeed** - this is manipulated via an animation curve and will influence what parts of the forward locomotion of the blend-tree will be played during the turnaround.



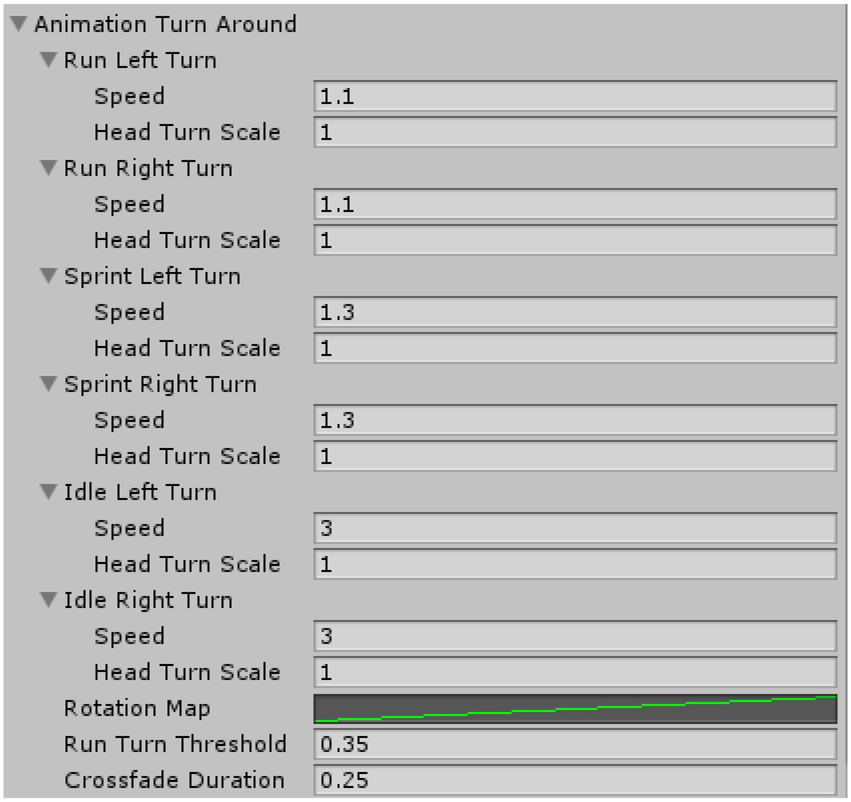
Blendspace Turn Around

Time To Turn	0.2
Rotation Map	
Forward Speed Map	
Forward Speed Type	Multiplicative
Turn Angle Threshold	150
Large Turn Map	
Small Turn Map	
Head Turn Scale	1

The configuration of the Blendspace Turnaround Behaviour has the following fields:

- **Time To Turn** - duration of the turnaround in seconds. Default value is 0.2
- **Rotation Map** - an animation curve that defines how the character rotates over the duration of the turnaround
- **Forward Speed Map -** the character's current normalized forward speed is cached when turnaround is started. This animation curve defines how the normalized forward speed is changed over the duration of the turnaround
- **Forward Speed Type -** an enum that defines how the Forward Speed is applied to the cached normalized forward speed to calculate the new normalized forward speed, i.e. is the result of the Forward Speed evaluation multiplied or added to the cached normalized forward speed. Default value is Multiplicative
- **Turn Angle Threshold -** an angle in degrees that is used to distinguish between a 90 degree turn and a 180 degree turn. Default value is 150
- **Large Turn Map -** the forward movement of the character is cached when the turnaround is started. The character is then moved at some portion of this speed in the direction of the turnaround destination angle (for the duration of the turnaround). This is the proportion of the movement that is applied during a 180 degree turn
- **Small Turn Map -** this is the proportion of movement (mentioned above) that is applied during a 90 degree turn
- **Head Turn Scale -** proportion of head turning that is done during turnaround.

Animation Turnaround Behaviour



▼ Animation Turn Around
- ▼ Run Left Turn
  - Speed: 1.1
  - Head Turn Scale: 1
- ▼ Run Right Turn
  - Speed: 1.1
  - Head Turn Scale: 1
- ▼ Sprint Left Turn
  - Speed: 1.3
  - Head Turn Scale: 1
- ▼ Sprint Right Turn
  - Speed: 1.3
  - Head Turn Scale: 1
- ▼ Idle Left Turn
  - Speed: 3
  - Head Turn Scale: 1
- ▼ Idle Right Turn
  - Speed: 3
  - Head Turn Scale: 1
- Rotation Map
- Run Turn Threshold: 0.35
- Crossfade Duration: 0.25

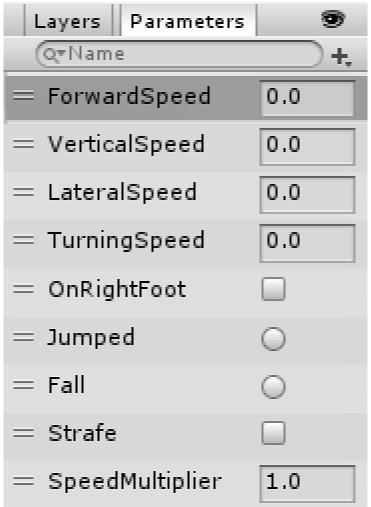
The Animation Turnaround Behaviour has the following fields that adjust its behaviour:

- There are settings for each animation turnaround:
  - Run Left Turn, Run Right Turn, Sprint Left Turn, Sprint Right Turn, Idle Left Turn and Idle Right Turn
  - The behaviour will determine which animation turn around to play based on the state of the character
  - The **Speed** field is the speed the animator will be set to during the animation
  - The **Head Turn Scale** field is a modifier applied to the head turning during that turnaround
- **Rotation Map -** The rotation during the turnaround can be modified by the curve.

  Ensure that the curve x axis starts at 0.0 and ends at 1.0 for a full smooth rotation
- **Run Turn Threshold -** A normalized forward speed between this value and 1 will trigger the correct run turnaround around animation. A value lower will trigger an idle turn around. A value greater than 1 triggers the sprint turn around.
- **Crossfade Duration -** This is the normalized duration of the crossfade into the selected turnaround animation.

Animator

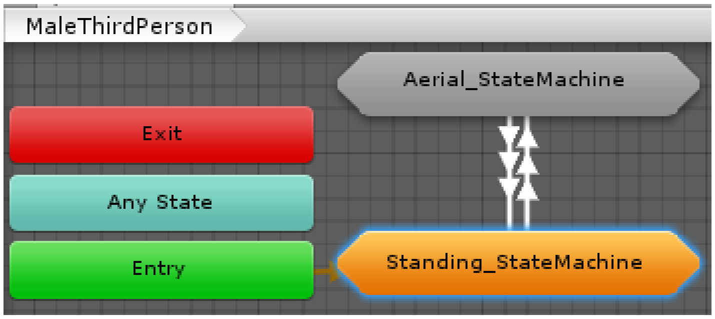
The ThirdPersonBrain handles the character animator. It listens for various events from the character physics and motor to change animator parameters and in some cases crossfades.



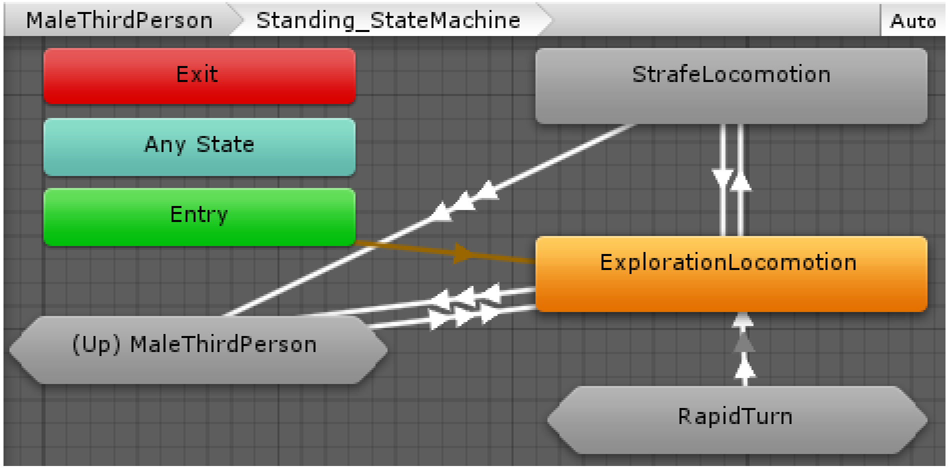
ForwardSpeed	0.0
VerticalSpeed	0.0
LateralSpeed	0.0
TurningSpeed	0.0
OnRightFoot	□
Jumped	○
Fall	○
Strafe	□
SpeedMultiplier	1.0

There are the following animator parameters:

- **ForwardSpeed** - Used by the locomotion blend trees
- **VerticalSpeed** - Used by the physics jump blend trees
- **LateralSpeed** - Used by the locomotion blend trees
- **TurningSpeed** - Used by the locomotion blend trees
- **OnRightFoot** - Used to determine which jump state to enter
- **Jumped** - Used to enter the jump states
- **Fall** - Used to enter the fall states
- **Strafe** - Used to enter/exit the strafe state
- **Speed Multiplier** - Used to scale animator speed.



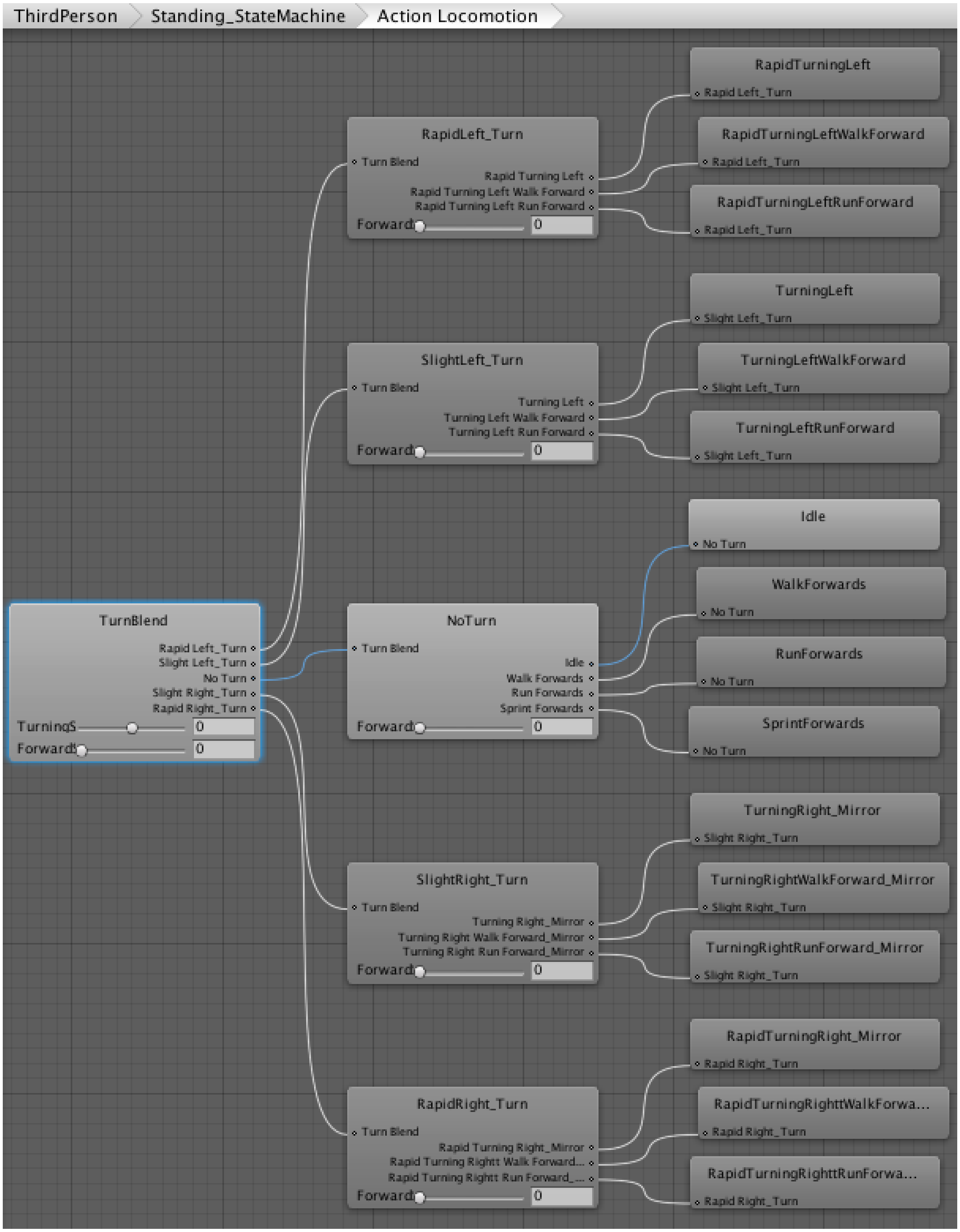
This is the very top layer of the Animator. It contains two state machines representing the possible types of movement.



The standing state machine handles locomotion. There is a blend tree of Exploration Locomotion for forward unlocked movement and another for Strafe Locomotion or forward locked movement. There is also a state machine for rapid turns. This rapid turn state machine is used when the ThirdPersonBrain’s turnaround type is set to animation.

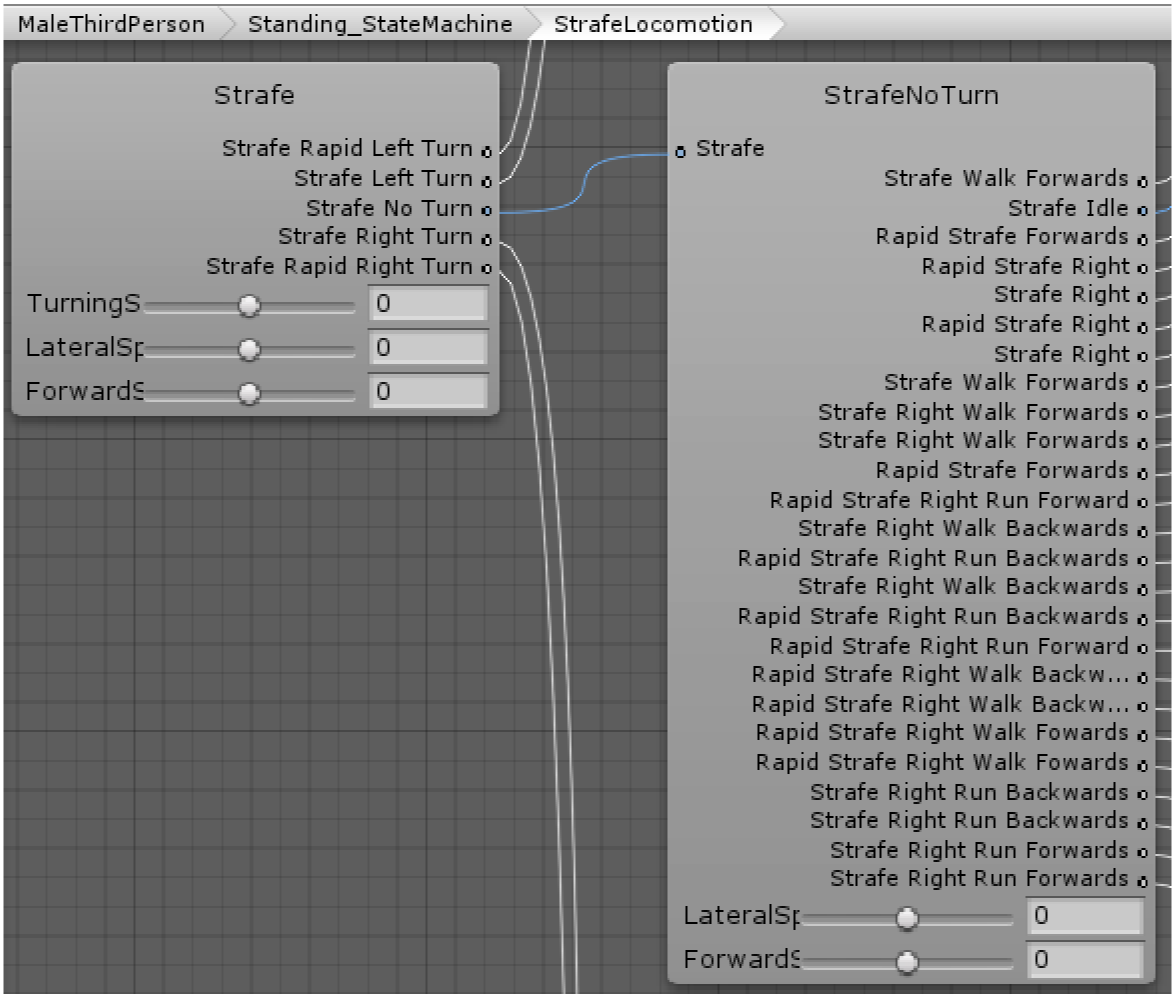
For both locomotion and strafe there is a 1D blend tree based on turning speed that is used to determine which blend tree to enter: *NoTurn, SlightLeft_Turn, RapidLeft_Turn, SlightRight_Turn, RapidRight_Turn*.

Exploration Locomotion



The blend trees use ForwardSpeed.

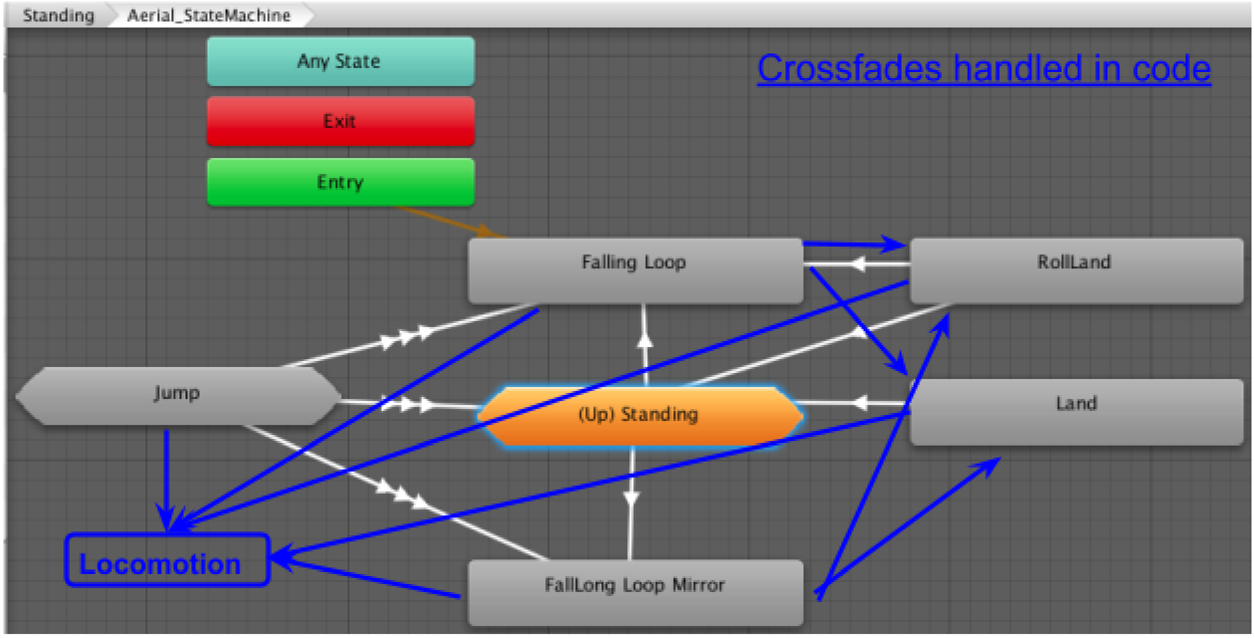
Strafe Locomotion


MaleThirdPerson
Standing_StateMachine
StrafeLocomotion
Strafe
Strafe Rapid Left Turn
Strafe Left Turn
Strafe No Turn
Strafe Right Turn
Strafe Rapid Right Turn
StrafeNoTurn
Strafe
Strafe Walk Forwards
Strafe Idle
Rapid Strafe Forwards
Rapid Strafe Right
Strafe Right
Rapid Strafe Right
Strafe Right
Strafe Walk Forwards
Strafe Right Walk Forwards
Strafe Right Walk Forwards
Rapid Strafe Forwards
Rapid Strafe Right Run Forward
Strafe Right Walk Backwards
Rapid Strafe Right Run Backwards
Strafe Right Walk Backwards
Rapid Strafe Right Run Backwards
Rapid Strafe Right Run Forward
Rapid Strafe Right Walk Backw...
Rapid Strafe Right Walk Backw...
Rapid Strafe Right Walk Fowards
Rapid Strafe Right Walk Fowards
Strafe Right Run Backwards
Strafe Right Run Backwards
Strafe Right Run Forwards
Strafe Right Run Forwards

TurningS 0
LateralSp 0
ForwardS 0

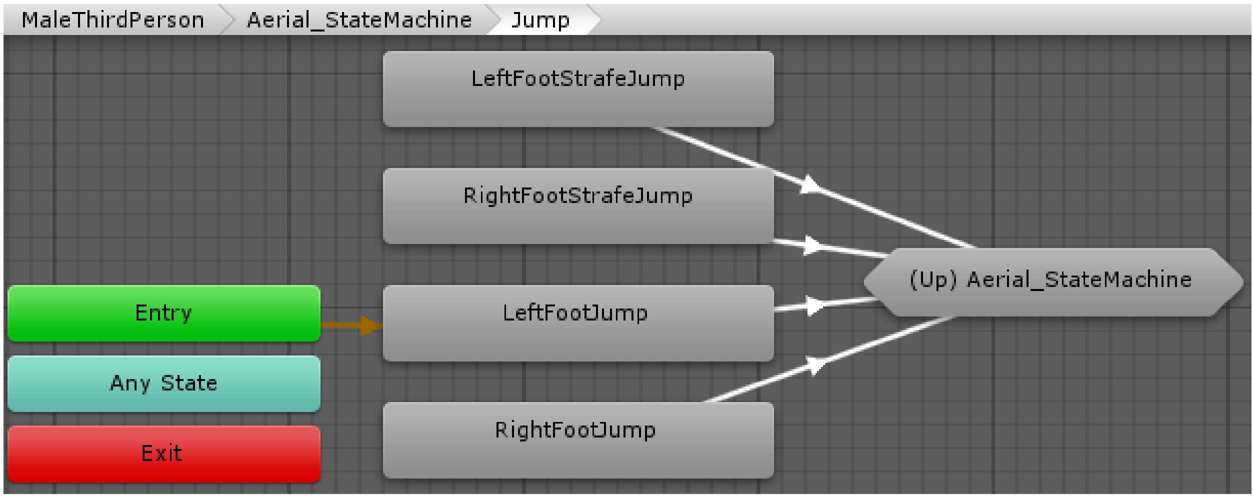
LateralSp 0
ForwardS 0

The blend trees use ForwardSpeed and LateralSpeed as parameters.

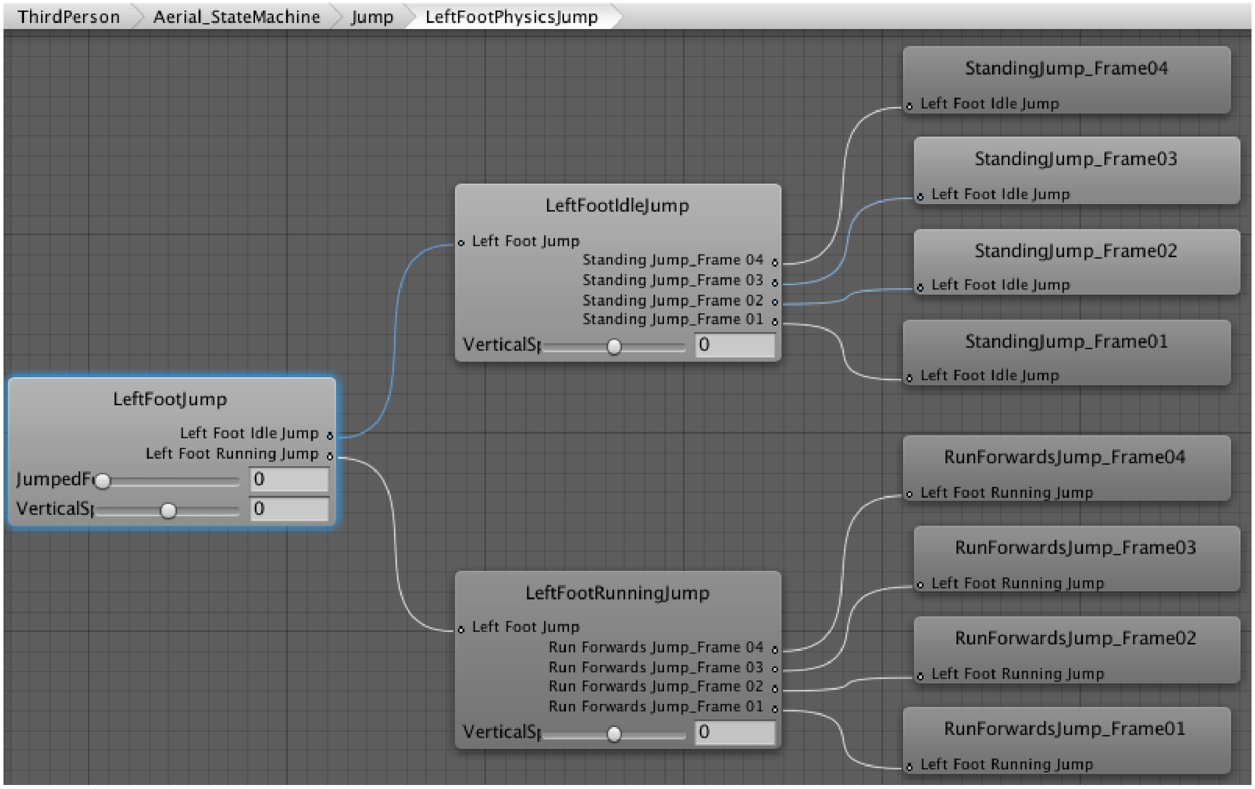


The aerial state machine is entered when the character becomes ungrounded through jumping or falling. Some transitions are handled in code by the ThirdPersonAnimationController. These can been seen in the image above. The configurations for these crossfade transitions can be changed on the Animation Configuration, detailed below.

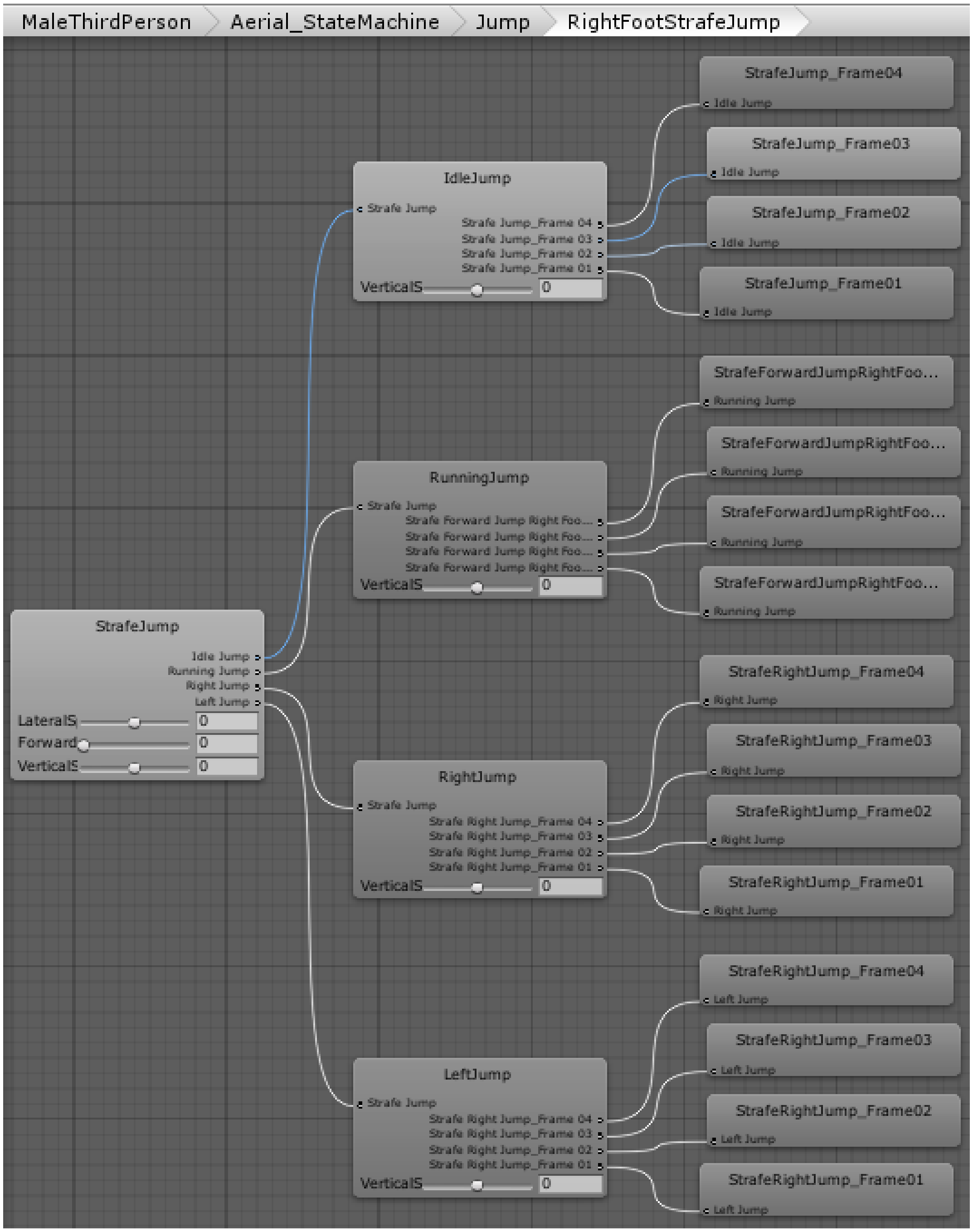
The jump state machine has four jump blend trees. One set is for root motion jumps and the other for physics based jumps. The transition out of a jump state is handled by a crossfade transition triggered by a land or a fall.



Exploration Jump

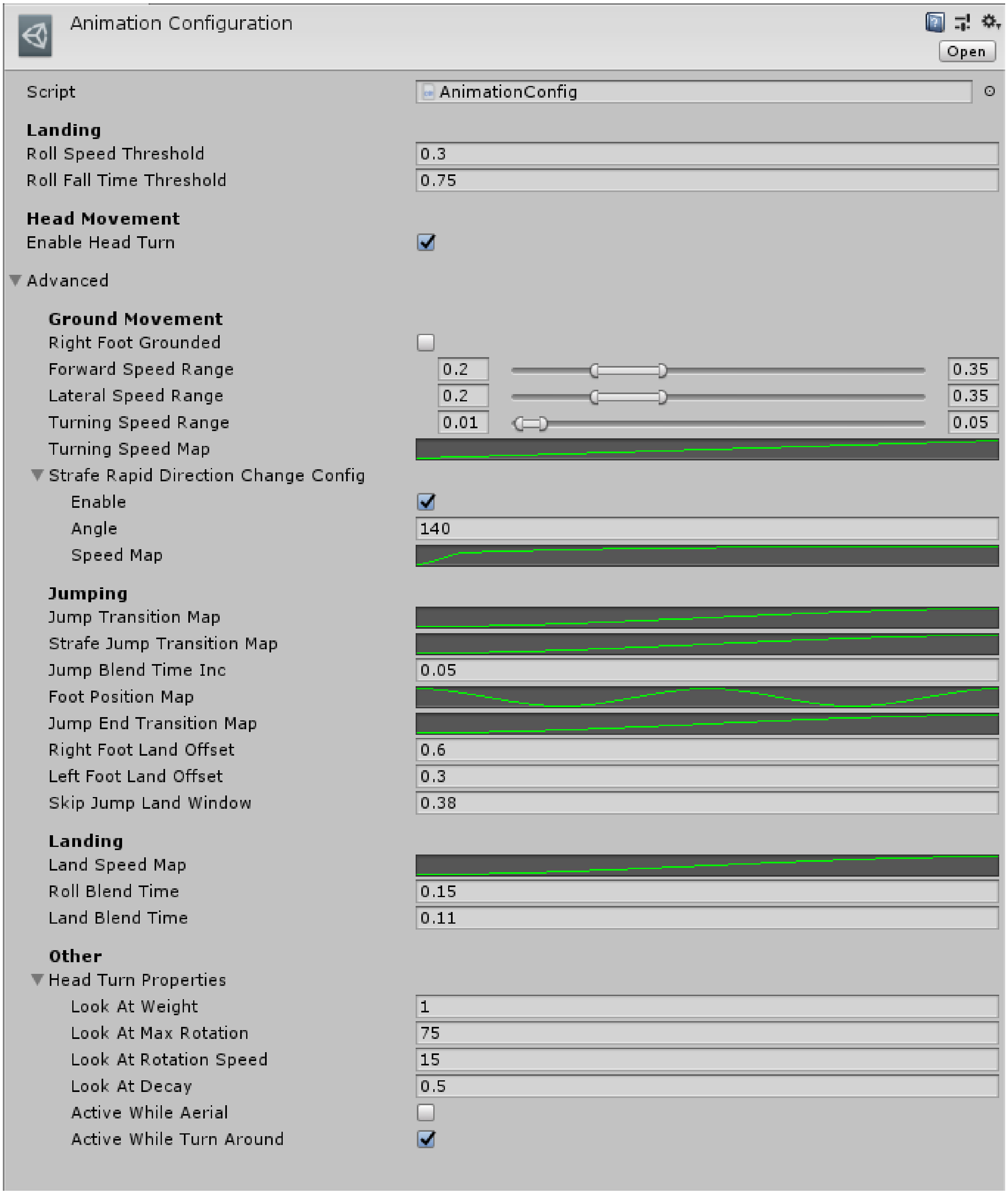


Strafe Jump



These blend trees are made up of frames of different jumps. As normalized vertical speed goes from 1 to -1 the frames blend to the appropriate poses. This allows a jump to be any length/duration with a smooth animation.

Animation Configuration



Script	AnimationConfig	
Landing		
Roll Speed Threshold	0.3	
Roll Fall Time Threshold	0.75	
Head Movement		
Enable Head Turn	☑	
Advanced		
Ground Movement		
Right Foot Grounded	□	
Forward Speed Range	0.2	0.35
Lateral Speed Range	0.2	0.35
Turning Speed Range	0.01	0.05
Turning Speed Map		
Strafe Rapid Direction Change Config		
Enable	☑	
Angle	140	
Speed Map		
Jumping		
Jump Transition Map		
Strafe Jump Transition Map		
Jump Blend Time Inc	0.05	
Foot Position Map		
Jump End Transition Map		
Right Foot Land Offset	0.6	
Left Foot Land Offset	0.3	
Skip Jump Land Window	0.38	
Landing		
Land Speed Map		
Roll Blend Time	0.15	
Land Blend Time	0.11	
Other		
Head Turn Properties		
Look At Weight	1	
Look At Max Rotation	75	
Look At Rotation Speed	15	
Look At Decay	0.5	
Active While Aerial	□	
Active While Turn Around	☑	

Landing

- **Roll Speed Threshold** - a forward speed higher than this will trigger a roll on land
- **Roll Fall Time Threshold** - a fall time greater than this will trigger a roll. Less than this will transition to locomotion.

Head Movement

- **Enabled Head Turn** - indicates whether the dynamic head turn should be enabled.

Advanced Settings

Ground Movement

- **Right Foot Grounded** - a toggle for whether the right foot start as grounded. Default is left foot
- **Forward Speed Range** - references the parameter name and a min and max interpolation time to use when adjusting the parameter value
- **Lateral Speed Range** - references the parameter name and a min and max interpolation time to use when adjusting the parameter value
- **Turning Speed Range** - references the parameter name and a min and max interpolation time to use when adjusting the parameter value
- **Turning Speed Map** - curve used to remap raw normalized turning speed.

Strafe Rapid Direction Change Config

- **Enable** - indicates whether a strafe rapid direction change should be detected and smoothed. This should only be enabled if opposing strafe animations are reverses of each other. eg walk backwards is walk forward played at a -1 speed
- **Angle** - input change angle threshold used to trigger a strafe rapid direction change
- **Speed Curve** - curve used to change animator movement speeds during a strafe rapid direction change.

Jumping

- **Jump Transition Map** - curve used to determine the cross fade duration of the transition into the exploration jump animation state
- **Strafe Jump Transition Map** - curve used to determine the cross fade duration of the transition into the strafe jump animation state
- **Jump Blend Time Inc** - time to add to the jump blend duration based on current grounded foot's position
- **Foot Position Map** - curve used to evaluate the current foot’s position in order to add Jump Blend Time Inc
- **Jump End Transition Map** - curve used to determine the cross fade duration of the transition from the jump animation state into the locomotion animation state
- **Right Foot Land Offset & Left Foot Land Offset** - cross fade cycle offset for transition into locomotion state after a jump

- **Skip Jump Land Window** - time in seconds allowed between jumps to create a skip effect.

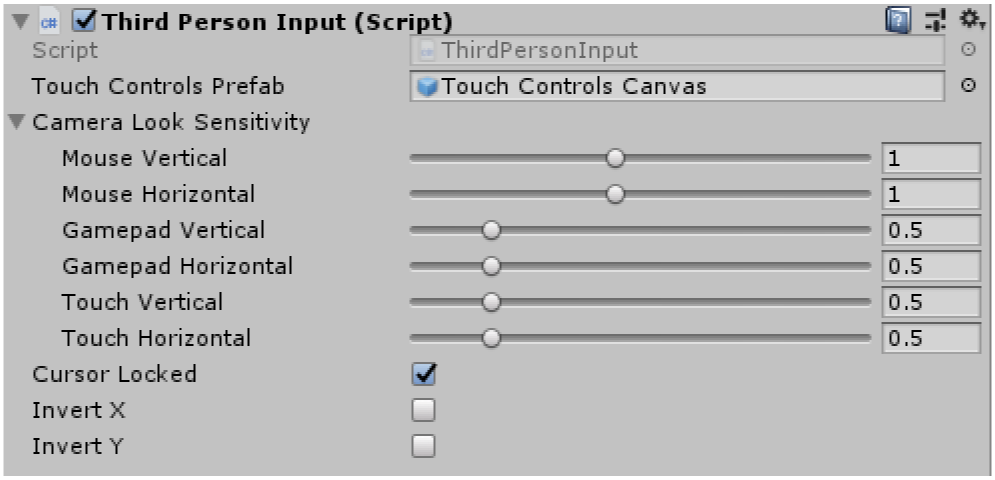
Landing

- **Land Speed Map** - curve used to determine the land animation speed
- **Roll Blend Time -** normalized time used for the cross fade into the roll animation state
- **Land Bend Time** - normalized time used for the cross fade into the land animation state.

Other

- Head Turn Properties
    - **Look At Weight** - the global weight of Animator.LookAt. See: https://docs.unity3d.com/ScriptReference/Animator.SetLookAtWeight.html
    - **Look At Max Rotation -** the maximum angle the head can turn.
    - **Look At Rotation Speed -** angles per second.
    - **Active While Aerial** - whether the dynamic head turn should be enabled during aerial states
    - **Active Turn Around -** whether the dynamic head turn should be enabled during a turnaround.

Third Person Input



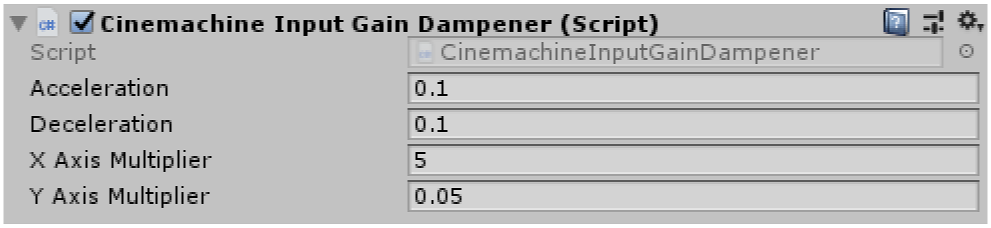
☑ **Third Person Input (Script)**

Script	ThirdPersonInput
Touch Controls Prefab	Touch Controls Canvas
Camera Look Sensitivity	
Mouse Vertical	1
Mouse Horizontal	1
Gamepad Vertical	0.5
Gamepad Horizontal	0.5
Touch Vertical	0.5
Touch Horizontal	0.5
Cursor Locked	☑
Invert X	□
Invert Y	□

The Third Person Input component has the following fields:

1. **Touch Controls Prefab** - prefab that will be instantiated into the scene for mobile touch controls UI.
2. **Camera Look Sensitivity** - Adjust the camera vertical and horizontal sensitivity for mouse, gamepad and touch look input.
3. **Cursor Locked** - if checked then the mouse cursor is locked to the center of the screen, which is useful for first person shooters.
4. **Invert X** - inverts the horizontal look input.
5. **Invert Y** - inverts the vertical look input

Cinemachine Input Gain Dampener



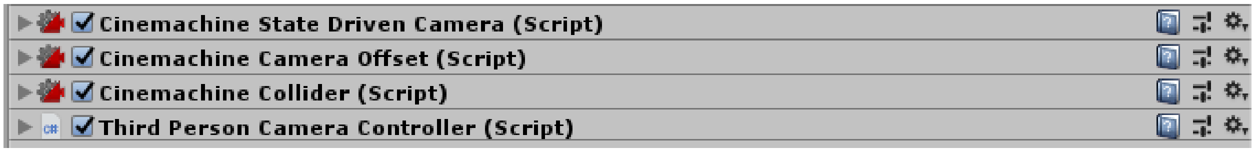
☑ **Cinemachine Input Gain Dampener (Script)**

Script	CinemachineInputGainDampener
Acceleration	0.1
Deceleration	0.1
X Axis Multiplier	5
Y Axis Multiplier	0.05

The Cinemachine Input Gain Dampener component has the following fields:

1) **Acceleration -** The gamepad look input acceleration when using Cinemachine Input Gain speed mode.
2) **Deceleration -** The gamepad look input deceleration value when using Cinemachine Input Gain speed mode.
3) **X Axis Multiplier -** Multiplies the X axis look value to compensate for different axis movement ranges.
4) **Y Axis Multiplier -** Multiplies the X axis look value to compensate for different axis movement ranges.

Third Person Cameras prefab



☑ Cinemachine State Driven Camera (Script)

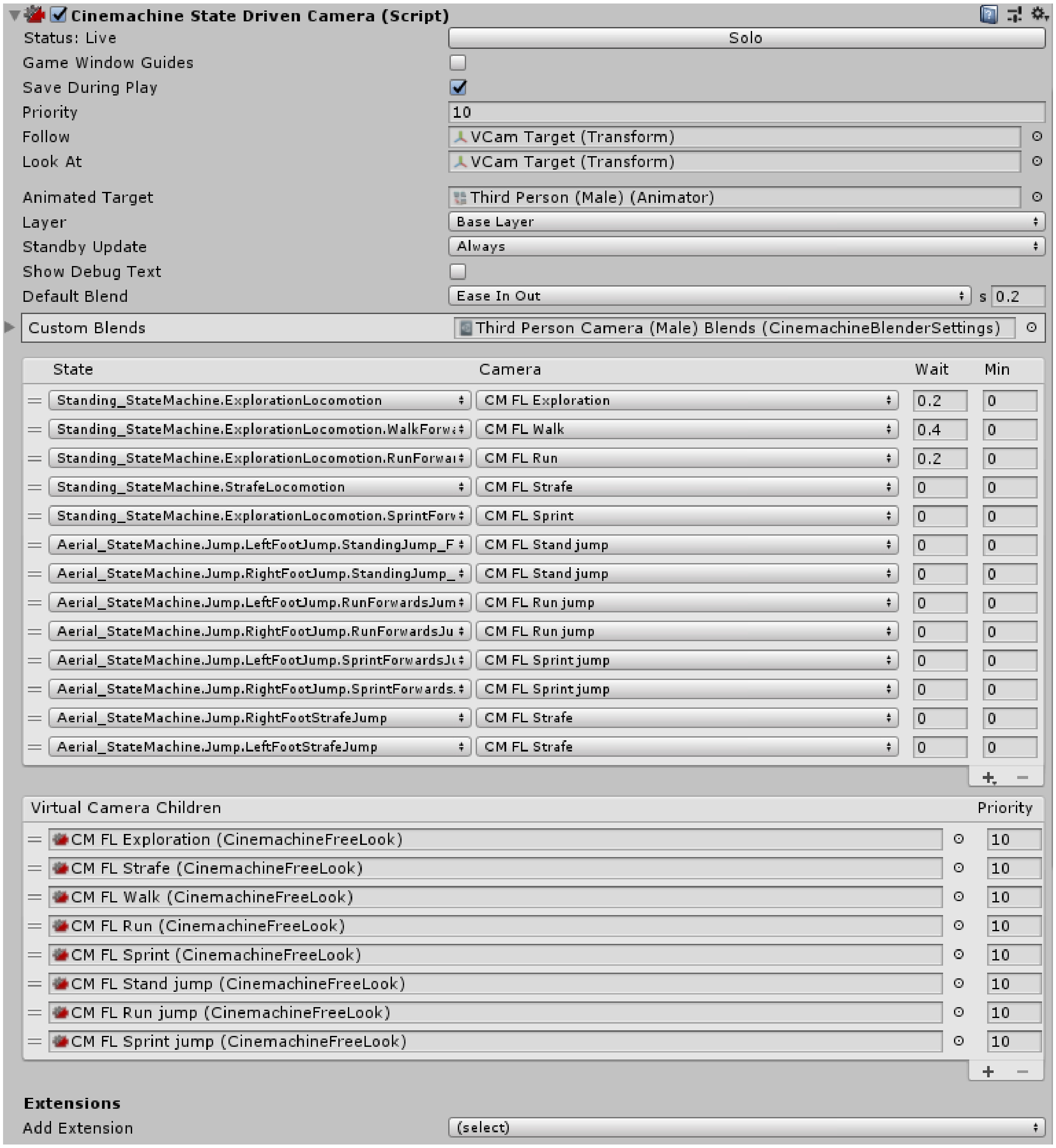
☑ Cinemachine Camera Offset (Script)

☑ Cinemachine Collider (Script)

☑ Third Person Camera Controller (Script)

- **Cinemachine State Driven Camera -** the parent State Driven Camera that contains the two primary states. These are the Exploration State and the Strafe State
- **Cinemachine Camera Offset** - adds an offset to the cameras
- **Cinemachine Collider** - adds a camera collider to all the Cinemachine FreeLook camera children of the State Driven Camera
- **Third Person Camera Controller -** displays a crosshair, handles recentering and does some CInemachine initialization.

Cinemachine State Driven Camera



Cinemachine State Driven Camera (Script)

Status: Live	Solo
Game Window Guides	□
Save During Play	☑
Priority	10
Follow	VCam Target (Transform)
Look At	VCam Target (Transform)
Animated Target	Third Person (Male) (Animator)
Layer	Base Layer
Standby Update	Always
Show Debug Text	□
Default Blend	Ease In Out s 0.2
Custom Blends	Third Person Camera (Male) Blends (CinemachineBlenderSettings)

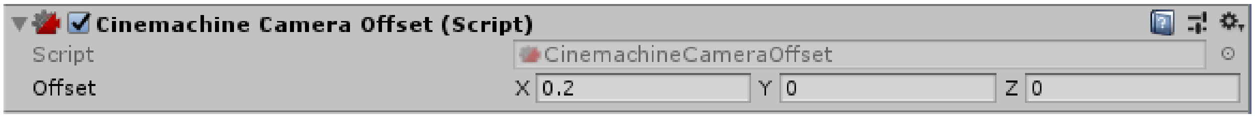
State	Camera	Wait	Min
Standing_StateMachine.ExplorationLocomotion	CM FL Exploration	0.2	0
Standing_StateMachine.ExplorationLocomotion.WalkForw	CM FL Walk	0.4	0
Standing_StateMachine.ExplorationLocomotion.RunForwa	CM FL Run	0.2	0
Standing_StateMachine.StrafeLocomotion	CM FL Strafe	0	0
Standing_StateMachine.ExplorationLocomotion.SprintForv	CM FL Sprint	0	0
Aerial_StateMachine.Jump.LeftFootJump.StandingJump_F	CM FL Stand jump	0	0
Aerial_StateMachine.Jump.RightFootJump.StandingJump_	CM FL Stand jump	0	0
Aerial_StateMachine.Jump.LeftFootJump.RunForwardsJum	CM FL Run jump	0	0
Aerial_StateMachine.Jump.RightFootJump.RunForwardsJu	CM FL Run jump	0	0
Aerial_StateMachine.Jump.LeftFootJump.SprintForwardsJu	CM FL Sprint jump	0	0
Aerial_StateMachine.Jump.RightFootJump.SprintForwards.	CM FL Sprint jump	0	0
Aerial_StateMachine.Jump.RightFootStrafeJump	CM FL Strafe	0	0
Aerial_StateMachine.Jump.LeftFootStrafeJump	CM FL Strafe	0	0

Virtual Camera Children	Priority
CM FL Exploration (CinemachineFreeLook)	10
CM FL Strafe (CinemachineFreeLook)	10
CM FL Walk (CinemachineFreeLook)	10
CM FL Run (CinemachineFreeLook)	10
CM FL Sprint (CinemachineFreeLook)	10
CM FL Stand jump (CinemachineFreeLook)	10
CM FL Run jump (CinemachineFreeLook)	10
CM FL Sprint jump (CinemachineFreeLook)	10

Extensions

Add Extension: (select)

Cinemachine Camera Offset

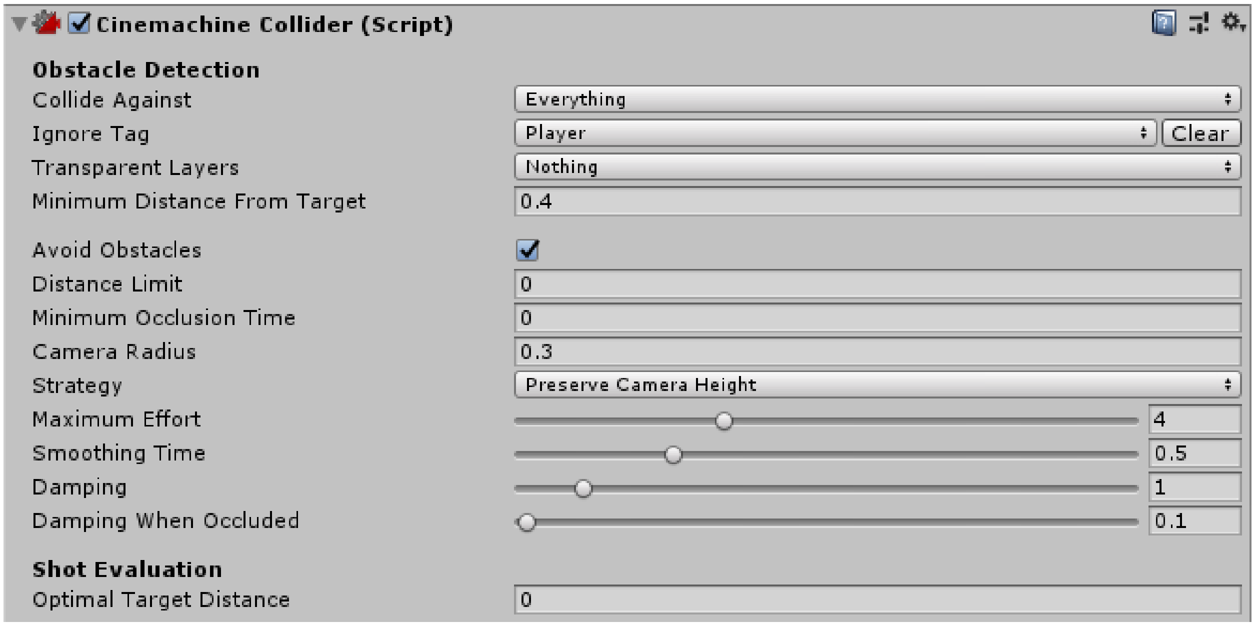


☑ **Cinemachine Camera Offset (Script)**

Script	CinemachineCameraOffset
Offset	X 0.2 Y 0 Z 0

This allows the parent Cinemachine State Driven camera to have an offset.

Cinemachine Collider



☑ **Cinemachine Collider (Script)**

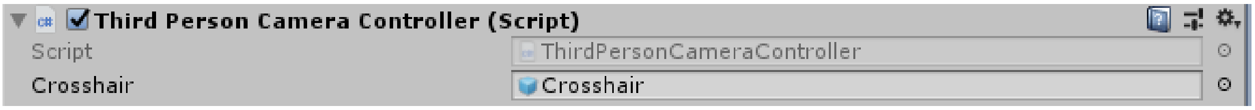
Obstacle Detection

Collide Against	Everything
Ignore Tag	Player Clear
Transparent Layers	Nothing
Minimum Distance From Target	0.4
Avoid Obstacles	☑
Distance Limit	0
Minimum Occlusion Time	0
Camera Radius	0.3
Strategy	Preserve Camera Height
Maximum Effort	4
Smoothing Time	0.5
Damping	1
Damping When Occluded	0.1

Shot Evaluation

Optimal Target Distance	0

Third Person Camera Controller



☑ Third Person Camera Controller (Script)

Script: ThirdPersonCameraController

Crosshair: Crosshair

The Third Person Camera Controller displays the crosshair when strafing and does camera recentering in exploration mode.

Third Person Camera Controller also finds the ThirdPersonBrain in the scene and automatically sets up the required fields for the Cinemachine Cameras.

- **Look At Target** - set using the VCam Target GameObject under the root of the third person character
- **Follow Target** - set using the VCam Target GameObject under the root of the third person character
- **Animated Target** - Set using the Animator component of the ThirdPersonBrain.