Cinemachine
Welcome to Cinemachine

Cinemachine is a suite of ‘smart’ procedural camera modules which allow you to define the shot and they'll dynamically follow your direction.

Set up shots which track and compose motion in realtime, like AI camera operators. The procedural nature makes them bug-resistant as they always work to make the shot based on your direction. That’s great for gameplay, but they’re also amazingly fast for cutscenes. Change an animation, a vehicle speed, ground terrain - whatever - and Cinemachine will dynamically make the shot. You can use really telephoto lenses and not have to update the cutscene if things change.

The Brain

The idea is to tell the Unity camera where to go and what to do in response to gameplay and scene evolution. To do this, the first step is to give the Unity Camera a CinemachineBrain behaviour. Think of that as the movie director, who chooses among the available camera shots for the one that tells the best story at any given moment.

The next step is to populate your scene with intelligent CinemachineVirtualCamera objects. These can be thought of as skilled cameramen who are given instructions about what game objects to target, and how to follow them. It is the brain's job to monitor these and choose the most appropriate one to track at any given frame.

Virtual Cameras

A virtual camera is not a camera. Instead, it can be thought of as a camera controller, not unlike a cameraman. It can drive the Unity Camera and control its position, orientation, lens settings, and PostProcessing effects. Each Virtual Camera owns its own Cinemachine Component Pipeline, through which you provide the instructions for dynamically tracking specific game objects.

A virtual camera is very lightweight, and does no rendering of its own. It merely tracks interesting GameObjects, and positions itself accordingly. A typical game can have dozens of virtual cameras,
each set up to follow a particular character or capture a particular event.

A Virtual Camera can be in any of three states:

- **Live**: The virtual camera is actively controlling the Unity Camera. The virtual camera is tracking its targets and being updated every frame.
- **Standby**: The virtual camera is tracking its targets and being updated every frame, but no Unity Camera is actively being controlled by it. This is the state of a virtual camera that is enabled in the scene but perhaps at a lower priority than the Live virtual camera.
- **Disabled**: The virtual camera is present but disabled in the scene. It is not actively tracking its targets and so consumes no processing power. However, the virtual camera can be made live from the Timeline.

The Unity Camera can be driven by any virtual camera in the scene. The game logic can choose the virtual camera to make live by manipulating the virtual cameras' *enabled* flags and their priorities, based on game logic.

In order to be driven by a virtual camera, the Unity Camera must have a `CinemachineBrain` behaviour, which will select the most eligible virtual camera based on its priority or on other criteria, and will manage blending.

### Blending

Camera Blending is a concept that brings the Virtual Cameras beyond the analogy of a cameraman. It is essentially an interpolation over time of one virtual camera position and state to another. If you think of virtual cameras as cameramen, then blending is a little like one cameraman smoothly passing the camera to another cameraman. You can specify the time over which to blend, as well as the blend curve shape. Note that a camera cut is just a zero-time blend.

The `CinemachineBrain` behaviour holds the instructions for blending. There you can specify the default blend (how virtual cameras transition from one being live to another), and also specific overrides to the default blend in the form of an instruction list: "when this vcam goes live, blend like this", or "when bending from this vcam to that one, do it like that", and so on.
Getting Started

The first thing to look at is the **CinemachineVirtualCamera** behaviour. That should lead you to the pipeline of **ICinemachineComponents** which serve as instructions for controlling the Position, Orientation, and Lens of the camera. Create some Virtual Cameras in your scene, set their targets, and play with their settings to see what they do. Also have a look at some of the example scenes included with the package.

The API generally reflects what you see in the inspector for the behaviours of the different sorts of Virtual Cameras and their components. Have a look at the corresponding classes, beginning with:

- **CinemachineVirtualCamera**
- **CinemachineBrain**
- **CinemachineClearShot**
- **CinemachineFreeLook**
- **CinemachineCollider**
- **CinemachineTransposer**
- **CinemachineComposer**

See Also

Other Resources

Version History

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Version History

The topics in this section describe the various changes made to the [TODO: Project Title] over the life of the project.

Version History

Select a version below to see a description of its changes.

- Version 2.0
- Version 2.1

See Also

Other Resources
Welcome to Cinemachine

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Version 2.0

Version 2.0 was released on July 23, 2017.

Changes in This Release

- This is a complete overhaul of Cinemachine 1.0, and contains many new features and implementations.

- Unfortunately it was not possible to maintain back-compatibility with Cinemachine 1, and so any Cinemachine content created with earlier versions will have to be re-created using this version.

- Please see the Cinemachine Forum for a complete list of changes in this release.

See Also

Other Resources
Version History

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Version 2.1

Version 2.1 was released on Nov 11, 2017.

Changes in This Release

- PostProcessing V2 is now supported.

- New Collider implementation. Curb feelers are gone, replaced by a clever camera-repositioning algorithm that will try to preserve camera height or distance from target (strategy is user-selectable).

- New CinemachineConfiner: confine a virtual camera to a simple bounding volume or PolygonCollider2D.

- New Framing Transposer. This is a special transposer that will respect composition and framing rules by moving the camera without rotating it. Takes only a Follow Target (no LookAt - this is important). Designed for Orthographic cameras, but will also work for Perspective cameras. If Follow target is a CinemachineTargetGroup, then will also provide Group Framing options.

- New CinemachinePOV Aim component. Camera aim is entirely controlled by user, using 2 input axes.

- New CinemachineMixingCamera. Drive a continuous blend of up to 8 virtual cameras from timeline or game logic. Create complex rigs with custom blends and expose them as ordinary vcams.

- New CinemachineBlendListCamera. A mini vcam sequencer that provides a quick and easy way to do AB camera moves on the fly.

- New CinemachineDollyCart behaviour, for moving anything along a path. No need to create dummy vcams just for that.

- New CinemachineSmoothPath component. You can use it instead of the old CinemachinePath. The difference is that the SmoothPath guarantees second-order continuity, which means
that there will never be any sudden camera rotation changes baked into the path tangents. Easier to use, too: no tangents to mess around with (they get automatically set for smoothness).

- Path now supports Distance Units in addition to Path units, making it easy to create steady motion.
- TrackedDolly: added ability to use Distance or Path units for path position.
- Transposer and TrackedDolly: added target angular damping on 3 axes.
- OrbitalTransposer and FreeLook: added angular damping and binding mode, same as Transposer.
- OrbitalTransopser and freelook: added checkbox to invert input axis.
- Transposer, OrbitalTransposer, and FreeLook: added new Simple Follow binding mode, which will follow the target using rotation as much as possible, changing position as little as possible. Like a lazy cameraman.
- Added IgnoreTimeScale option to Brain. Useful for snappy cameras even in slo-mo.
- Added Lookahead Time to composer. Composer will look at the point where it estimates the target will be at some time in the near future. This gives much more natural framing of a moving target.
- SmartUpdate is smarter: added support for Interpolation when target is animated by physics system.
- Added off-button for SaveDuringPlay.
- No SaveDuringPlay for vcam priority, LookAt and Follow targets, GroupTarget members.
- Added IsBlending API method to StateDrivenCamera and ClearShot.
- TargetGroup now has a user-selectable update method.
- TargetGroup now respects the weight when computing bounding box, so it's possible to gradually add or remove members by manipulating the weight.
- Clearshot: if randomize, then re-randomize whenever it becomes active.
- ClearShot: default blend is cut.
- ClearShot create menu: add a Collider by default.
- FollowZoom: min/max FOV defaults changed to 3/60.
- Composer damping range is now 0-20 instead of 0-100.
- Orbital and FreeLook: Heading Bias can now be animated on the timeline.
- Orbital and FreeLook: damping no longer interferes with camera response to user input. Axis movement bypasses all damping.
- TrackedDolly: added path position offset to Auto-Dolly. Stays on the path (unlike path offset, which is based on the path tangent and so can go off the path)
- Noise component inspector now has a dropdown for Profile presets, instead of directly allowing editing of the Profile asset.
- Added concept of Cinemachine Extension. Collider, confiner, PostProcessing, etc are now Extensions. They are available via a dropdown at the bottom of the inspection for virtual cameras. They will no longer appear in the standard Components menu.
- Time.timeScale = 0 is now supported. Pausing the game will also pause the virtual cameras.
- HardConstraint has been split into two settings: Do Nothing and Hard LookAt/Follow. Do Nothing will leave the camera's transform alone, ignoring any target that may have been set.
- CinemachineBrain no longer requires a Camera component. Can be used with any GameObject, making it possible to use Cinemachine to control the transforms of arbitrary objects.
- Improved logic for AxisState accel/decel. More realistic behaviour.
- SaveDuringPlay obsolete API fix for 2017.2.
- Fixed build errors when building for UWP.
- Clearshot and SDC: don’t reset state if deactivated.
- FreeLook destroy - no more orphan rigs.
- Fixed strange build error that only showed up in MonoDevelop.
- FreeLook was not respecting X-axis accel and decel. Heading speed had to be crazy high. Now same as Orbital (warning: may have to re-tune settings on existing FreeLooks).
- Recenter to target heading was not moving smoothly in some circumstances.
- Collider raycasts no longer hit triggers.
- Noise: handle variable deltaTime gracefully.
- State-Driven-Camera: don’t generate errors when animated target is inactive.
- Several jitter and judder issues resolved.

See Also

Other Resources
Version History

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# Namespaces

<table>
<thead>
<tr>
<th>Namespace</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cinemachine</td>
<td>This namespace holds the Core Cinemachine code.</td>
</tr>
<tr>
<td>Cinemachine.Utility</td>
<td>Cinemachine non-domain utilities and Unity extensions and helpers</td>
</tr>
</tbody>
</table>

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
# Cinemachine Namespace

This namespace holds the Core Cinemachine code.

## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![ ] CinemachineBasicMultiChannelPerlin</td>
<td>As a part of the Cinemachine Pipeline implementing the Noise stage, this component adds Perlin Noise to the Camera state, in the Correction of the CameraState. It is created by using a predefined noise profile asset. You can scale this in amplitude or in time, to produce a family of different noises using the same profile.</td>
</tr>
<tr>
<td>![ ] CinemachineBlend</td>
<td>Describes a blend between 2 Cinemachine Virtual Cameras, and holds the blend.</td>
</tr>
<tr>
<td>![ ] CinemachineBlendDefinitionPropertyAttribute</td>
<td>Property applied to CinemachineBlendDefinition. Used for custom drawing in the inspector.</td>
</tr>
<tr>
<td>![ ] CinemachineBlenderSettings</td>
<td>Asset that defines the rules for blending between Virtual Cameras.</td>
</tr>
</tbody>
</table>
CinemachineBlendListCamera

This is a virtual “manager” that manages a collection of Virtual Cameras. When a camera goes live, these child vcams are enabled, one after another, holding each camera for a designated time. The last camera is held indefinitely.

CinemachineBrain

CinemachineBrain is the link between the Unity Camera and the Cinemachine Virtual Cameras in the scene. It monitors the priority stack to choose the current Camera, and if necessary, if necessary... importantly, it monitors Camera state to the attached Unity Camera. CinemachineBrain is also the place where rules for blending between virtual cameras are defined. Camera blending is an interpolation over time of one virtual camera to another. If you think of virtual cameras as cameramen, then blending is a little like one cameraman smoothly passing the camera to another, time over time. That is to say, the blend is a zero-time blend.
CinemachineBrainBrainEvent  Event with a parameter

CinemachineBrainVcamEvent  Event with a ICinemachineCamera parameter

CinemachineClearShot  Cinemachines "manager camera" and manages Camera game. When Live, it check the children, the one with shot and may be a very powerful tool.

child cameras have CinemachineCollider extensions, scene for target obstructions, optimal target other items, assessment assessment to the ClearShot parent, who will then choose the best one. You can use this to set multi-camera coverage of a scene, and be assured that a clear shot of the always be available. You can also define custom blends between ClearShot children.

CinemachineCollider  An add-on module for Cinemachines
that post-processes position of the virtual camera. Based on the supplied settings, the Collider will attempt to preserve the Line of Sight of the virtual camera by moving objects that obstruct the view. Additionally, the Collider can be used to assess the shot quality and report this as a field in the camera State.

### CinemachineComponentBase
An abstract mutator acting on a Cinemachine Virtual Camera.

### CinemachineComposer
This is a Cinemachine Composer in the Aim section of the component pipeline. Its job is to aim the camera at the target object with offsets, damping, and composition rules. The composer does not change the camera's position; it pans and tilts the virtual camera, in order to get the desired framing. To move the camera, you have to use the camera's Body Section.

### CinemachineConfiner
An add-on module for Cinemachine Virtual Camera that post-processes position of the virtual camera. It will confine the camera position to the volume specified...
CinemachineCore

A singleton that manages complete lists of Cinemachine Brain, Cinemachine Virtual Cameras, and the priority queue services to keep track whether Cinemachine Virtual Cameras have each frame.

CinemachineDollyCart

This is a very simple behaviour that constrains its transform to a Cinemachine Path. It is used to animate any objects along a path or as a Follow target for Cinemachine Virtual Cameras.

CinemachineExtension

Base class for a Cinemachine Virtual Camera extension module. Hooks into the Cinemachine Pipeline.

CinemachineExternalCamera

This component exposes a non-cinemachine camera to the cinemachine system to participate in blends as a component alongside an existing Unity Camera component.

CinemachineFollowZoom

An add-on module for Cinemachine Virtual Camera that adjusts zoom to keep the target object at a constant size on the screen regardless of...
This is a Cinemachine Framing Transposer Component. It is part of the (Camera ... position.

It is a Cinemachine Component in the Body section of the component pipeline. Its job is to position the camera in a fixed screen-space relationship to the vcam’s Follow target object, with offsets and damping. The camera will be first moved along the camera’s Z axis until the desired distance from the vcam’s Follow target. The camera will then be moved along the camera’s XY plane until the Follow target is at the desired point on the camera’s screen. The FramingTransposer will only change the camera’s position in space. It will not otherwise aim the camera. This component will work with the vcam’s LookAt target being null. The Follow target will define what the camera is looking at. If this component is used with a Cinemachine Target Group, then additional controls will be available to frame the entire group. This component is designed for orthographic cameras and works equally well with perspective cameras used in 3D environments.
towards a 3rd person experience.
around its subject with three separate camera rings around it, each has its own composer, and lens settings. Depending on the camera's position along a spline connecting the three, these settings are interpolated to give the final camera position and state.

**CinemachineGroupComposer**

This is a Cinemachine Component in the Aim section of the component pipeline. Its job is to aim the camera at a target object, with configurable offsets, damping, and composition rules. In addition, if the target is a CinemachineTargetGroup, the behaviour will ensure that the entire group of targets is framed properly.

**CinemachineHardLockToTarget**

This is a Cinemachine Component in the Aim section of the component pipeline. Its job is to place the camera on the Follow Target.

**CinemachineHardLookAt**

This is a Cinemachine Component in the Aim section of the component pipeline. Its job is to aim the camera hard at the LookAt.
CinemachineMixingCamera

CinemachineMixingCamera is a "manager camera" that takes on the state of the weighted average of the states of its child virtual cameras. Rather than a dynamic array, we do it this way in order to support weight animation from the Timeline and animate arrays of slots.

CinemachineOrbitalTransposer

This is a Cinemachine Component in the Body section of the component pipeline. It positions the camera in a variable relationship to the target object, with offsets and damping. Typically used to implement a camera that can accept player input from an input device, like a joystick. The OrbitalTransposer introduces the concept of __Heading__, which is the direction the target is moving. OrbitalTransposer will attempt to position the camera in relationship to the heading, which is by default directly behind the target.
control the default relationship by adjusting the Heading Bias setting. If you attach an input controller to the Orbital Transposer, the player can adjust the camera position in relation to the target. This allows the player to any spot on the orbit around the target.

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachinePath</td>
<td>Defines a world-space path consisting of an array of waypoints, each with position, tangent, and roll settings. Bezier interpolation performed between waypoints, to get a smooth and continuous path.</td>
</tr>
<tr>
<td>CinemachinePathBase</td>
<td>Abstract base class for a world-space path, camera dolly track.</td>
</tr>
<tr>
<td>CinemachinePathBaseAppearance</td>
<td>This class holds the settings that control how the path is displayed in the editor scene view. The path is not visible in the game view.</td>
</tr>
<tr>
<td>CinemachinePipeline</td>
<td>Internal container for CinemachineComponentBase. Does nothing.</td>
</tr>
<tr>
<td>CinemachinePOV</td>
<td>This is a CinemachineComponent in the Aim section of the component pipeline. Its job is to aim the camera.</td>
</tr>
</tbody>
</table>
Camera in response to the user's mouse or joystick input. The composer does not change the camera's position, pan and tilt, this is, in order to get the desired framing. To move the camera, you have to use the camera's Body section.

**CinemachineSmoothPath**

Defines a world-space path consisting of an array of waypoints, each of which has position and roll settings. Interpolation between the waypoints results in smooth and continuous path. The path will pass through all waypoints, and unlike CinemachinePath, first order continuity is guaranteed.

**CinemachineStateDrivenCamera**

This is a virtual camera "manager" that manages a collection of Virtual Cameras. Virtual cameras are mapped to individual states in an animation state machine, allowing you to associate specific virtual cameras with specific animation states. If that state is active in the state machine, the associated virtual camera will activates. You can define custom blends and transitions between virtual cameras. In order to use this behaviour, you
animated target (i.e., animated with a state machine) to drive the behavior.

**CinemachineTargetGroup**

Defines a group of target objects, each with a weight. The weight is used when calculating the average position of the group. Higher-weighted members of the group will count more.

**CinemachineTrackedDolly**

A Cinemachine Body component that constrains camera motion. Cinemachine Tracked Dolly can move along the path of a Cinemachine Path. Behaviour can operate in two modes: manual and Auto-Dolly positioning. In Manual mode, the camera's position is specified by animating the Path Position field. In Auto-Dolly mode, the Path Position field is animated automatically every frame by finding the position on the path that's closest to the virtual camera's Follow target.

**CinemachineTransposer**

This is a Cinemachine Body section pipeline. Its job is to position the camera in a fixed relationship to the vcam's Follow target.
with offsets and damping. Transposer will only change the camera's position in space. It will not re-orient or otherwise aim the camera. To aim the camera, you need to instruct the Aim section of its pipeline.

CinemachineVirtualCamera

This behavior is intended to be attached to an empty Transform GameObject within the Unity scene. The Virtual Camera will animate its Transform according to the rules of its CinemachineComponent pipeline (Aim, Body, Noise).

When the virtual camera is not active, the Unity camera will assume the position and orientation of the virtual camera. However, the virtual camera is not a camera. Instead, it can be thought of as a camera controller or cameraman. It can drive the Unity Camera and control its position, orientation, lens settings, and PostProcessing effects. Each virtual camera owns its own CinemachineComponent pipeline, through which you provide instructions for tracking specific game objects. The virtual camera is very lightweight, as it does not render its own data. Instead, it simply tracks interesting GameObjects and positions itself accordingly.
A typical game of virtual cameras is to follow a particular character and capture a particular event. Virtual Camera can be in any of three states: virtual camera is actively controlling the Unity Camera. The virtual camera is tracking its targets and being updated every frame. **Standby**: The virtual camera is tracking its targets and being updated, but no Unity Camera is actively being controlled by it. This is the state of a virtual camera that is enabled in the scene but perhaps at a lower priority than the Live virtual camera. **Disabled**: The virtual camera is present but disabled in the scene. It is not actively tracking its targets and consumes no processing power. Camera can be driven by any virtual camera in the scene. The game logic can choose the virtual camera to make live by manipulating the virtual cameras' enabled flags and their priorities, based on game logic.

In order to be driven by a virtual camera, the Unity Camera must have a CinemachineBrain behaviour, which will select the most eligible virtual camera based on its priority.
CinemachineVirtualCameraBase

Base class for a Virtual Camera within the Unity scene. Intended to be attached to an empty Transform GameObject. Inherited classes can be standalone virtual cameras such as CinemachineVirtualCamera, or meta-cameras such as CinemachineClearShot or CinemachineFreeLook.

CinemachineVirtualCameraBase exposes a Priority property. When the behaviour is in the game, the Virtual Camera is automatically placed in a queue maintained by the static CinemachineCore singleton. The queue is sorted by priority. When a Unity camera is equipped with a CinemachineBrain behaviour, the brain will choose the camera at the head of the queue. If you have multiple Unity cameras with CinemachineBrain behaviours (for example, in a split-screen context), the brain can choose the camera at the head of the queue by setting the culling flags on the virtual cameras. The culling mask of the Camera will be used for the brain. There is nothing that prevents a virtual camera from controlling multiple Unity cameras.
DocumentationSortingAttribute

Attribute to control the automatic generation of documentation.

LensSettingsPropertyAttribute

Property applied to LensSettings.

drawing in the inspector.

NoiseSettings

This is an asset that defines a noise profile. The shape of the noise as a function of time can be arbitrarily complex by combining different base perlin noise frequencies at different amplitudes. Amplitudes should be chosen with care, to ensure an interesting noise quality that is not obviously repetitive.

Mathematically, any arbitrary periodic curve can be broken down into a fixed-amplitude sine-waves added together. Fourier decomposition is the basis of much signal processing. It doesn't really have much to do with this asset, but it's super interesting!

NoSaveDuringPlayAttribute

Suppresses play-mode-save for a field. Use if there are fields in the class that shouldn't be
Invoke play-mode-save attribute for class. This class’s fields will be scanned upon exiting play mode, and its property values will be applied to the scene object. This is a stopgap measure that will become obsolete once Unity implements play-mode-save in a more general way.

## Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AxisState</strong></td>
<td>Axis state for defining to react to player input. The settings here control the responsiveness of the axis to player input.</td>
</tr>
</tbody>
</table>
| **CameraState** | The output of the Cinemachine engine for a specific virtual camera. Information in this struct can be blended, and provides what is needed to calculate an appropriate camera position, orientation, and lens setting. Raw values are what the Cinemachine
behaviours generate. The correction channel holds perturbations to the raw values - e.g. noise or smoothing, or obstacle avoidance corrections. Coirrections are not considered when making time-based calculations such as damping. The final position and orientation is the combination of raw values and corrections.

<table>
<thead>
<tr>
<th>CameraStateCustomBlendable</th>
<th>Opaque struct representing extra blendable stuff and its weight. The base system ignores this data - it is intended for extension modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBlendDefinition</td>
<td>Definition of a Camera blend struct holds the information necessary to generate a suitable AnimationCurve for a Cinemachine Blend.</td>
</tr>
<tr>
<td>CinemachineBlenderSettingsCustomBlend</td>
<td>Container specifying how two specific Cinemachine Virtual Cameras blend together.</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CinemachineBlendListCameraInstruction</td>
<td>This represents a single entry in the instruction list of the BlendListCamera.</td>
</tr>
<tr>
<td>CinemachineFreeLookOrbit</td>
<td>Defines the height and radius of the Rig orbit.</td>
</tr>
<tr>
<td>CinemachineOrbitalTransposerHeading</td>
<td>How the &quot;forward&quot; direction is defined. Orbital offset is in relation to the forward direction.</td>
</tr>
<tr>
<td>CinemachineOrbitalTransposerRecentering</td>
<td>Controls how automatic orbit recentering occurs.</td>
</tr>
<tr>
<td>CinemachinePathWaypoint</td>
<td>A waypoint along the path.</td>
</tr>
<tr>
<td>CinemachineSmoothPathWaypoint</td>
<td>A waypoint along the path.</td>
</tr>
<tr>
<td>CinemachineStateDrivenCameraInstruction</td>
<td>This represents a single instruction to the StateDrivenCamera. It associates a state from the state machine with</td>
</tr>
</tbody>
</table>
Virtual Camera also holds activation tuning parameters.

<table>
<thead>
<tr>
<th>Interface</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineTargetGroupTarget</td>
<td>Holds the information that represents a member of the group.</td>
</tr>
<tr>
<td>CinemachineTrackedDollyAutoDolly</td>
<td>Controls how automatic dolling occurs.</td>
</tr>
<tr>
<td>LensSettings</td>
<td>Describes the field of view and clip plane for a camera. This generally mirrors the Unity Camera's lens settings, will be used to drive the Unity camera when the vcarm is active.</td>
</tr>
<tr>
<td>NoiseSettingsNoiseParams</td>
<td>Describes the behaviour for a channel of noise.</td>
</tr>
<tr>
<td>NoiseSettingsTransformNoiseParams</td>
<td>Contains the behaviour of noise settings for all 3 cardinal axes of the camera.</td>
</tr>
</tbody>
</table>
ICinemachineCamera  An abstract representation of a virtual camera which lives within the Unity scene

Delegates

<table>
<thead>
<tr>
<th>Delegate</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineCoreAxisInputDelegate</td>
</tr>
<tr>
<td>CinemachineFreeLookCreateRigDelegate</td>
</tr>
<tr>
<td>CinemachineFreeLookDestroyRigDelegate</td>
</tr>
<tr>
<td>CinemachineVirtualCameraCreatePipelineDelegate</td>
</tr>
<tr>
<td>CinemachineVirtualCameraDestroyPipelineDelegate</td>
</tr>
</tbody>
</table>
## Enumerations

<table>
<thead>
<tr>
<th>Enumeration</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBlendDefinitionStyle</td>
</tr>
<tr>
<td>CinemachineBrainUpdateMethod</td>
</tr>
<tr>
<td>CinemachineColliderResolutionStrategy</td>
</tr>
<tr>
<td>Class Name</td>
</tr>
<tr>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>CinemachineConfinerMode</td>
</tr>
<tr>
<td>CinemachineCoreStage</td>
</tr>
<tr>
<td>CinemachineCoreUpdateFilter</td>
</tr>
<tr>
<td>CinemachineDollyCartUpdateMethod</td>
</tr>
<tr>
<td>CinemachineFramingTransposerAdjustmentMode</td>
</tr>
<tr>
<td>CinemachineFramingTransposerFramingMode</td>
</tr>
</tbody>
</table>
CinemachineGroupComposerAdjustmentMode

CinemachineGroupComposerFramingMode

CinemachineOrbitalTransposerHeadingHeadingDefinition

CinemachinePathBasePositionUnits

CinemachineTargetGroupPositionMode

CinemachineTargetGroupRotationMode

CinemachineTargetGroupUpdateMethod
CinemachineTrackedDollyCameraUpMode

CinemachineTransposerBindingMode

DocumentationSortingAttributeLevel

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
AxisState Structure

Axis state for defining to react to player input. The settings here control the responsiveness of the axis to player input.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll)  
Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
[SerializableAttribute]  
[DocumentationSortingAttribute(6.4f, DocumentationSortingAttribute)]  
public struct AxisState
```

The **AxisState** type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AxisState</strong></td>
<td>Constructor with specific values</td>
</tr>
</tbody>
</table>

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SetThresholds</strong></td>
<td>Sets the constraints by which this axis will operate on</td>
</tr>
<tr>
<td><strong>Update</strong></td>
<td>Updates the state of this axis</td>
</tr>
</tbody>
</table>
based on the axis defined by AxisState.m_AxisName

<table>
<thead>
<tr>
<th>Fields</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AccelTime</td>
<td>The amount of time in seconds it takes to accelerate to MaxSpeed with the supplied Axis at its maximum value</td>
</tr>
<tr>
<td>m_DecelTime</td>
<td>The amount of time in seconds it takes to decelerate the axis to zero if the supplied axis is in a neutral position</td>
</tr>
<tr>
<td>m_InputAxisName</td>
<td>The name of this axis as specified in Unity Input manager. Setting to an empty string will disable the automatic updating of this axis</td>
</tr>
<tr>
<td>m_InputAxisValue</td>
<td>The value of the input axis. A value of 0 means no input. You can drive this directly from a custom input system, or you can set the Axis Name and have the value driven by the internal Input Manager</td>
</tr>
<tr>
<td>m_InvertAxis</td>
<td>If checked, then the raw value</td>
</tr>
</tbody>
</table>
of the input axis will be inverted before it is used.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_MaxSpeed</td>
<td>How fast the axis value can travel. Increasing this number makes the behaviour more responsive to joystick input</td>
</tr>
<tr>
<td>Value</td>
<td>The current value of the axis</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
AxisState Constructor

Constructor with specific values

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>`public AxisState(</td>
<td></td>
</tr>
<tr>
<td>float <code>maxSpeed</code>,</td>
<td></td>
</tr>
<tr>
<td>float <code>accelTime</code>,</td>
<td></td>
</tr>
<tr>
<td>float <code>decelTime</code>,</td>
<td></td>
</tr>
<tr>
<td>float <code>val</code>,</td>
<td></td>
</tr>
<tr>
<td>string <code>name</code>,</td>
<td></td>
</tr>
<tr>
<td>bool <code>invert</code></td>
<td></td>
</tr>
<tr>
<td>)</td>
<td></td>
</tr>
</tbody>
</table>

**Parameters**

**maxSpeed**

Type: **System.Single**

[Missing <param name="maxSpeed"/> documentation for  

**accelTime**

Type: **System.Single**

[Missing <param name="accelTime"/> documentation for  

**decelTime**

Type: **System.Single**

[Missing <param name="decelTime"/> documentation for  

**val**

[Missing <param name="val"/> documentation for  
Type: SystemSingle
[Missing <param name="val"/> documentation for

name
Type: SystemString
[Missing <param name="name"/> documentation for

invert
Type: SystemBoolean
[Missing <param name="invert"/> documentation for

See Also

Reference
AxisState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
AxisState Methods

The `AxisState` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SetThresholds</td>
<td>Sets the constraints by which this axis will operate on</td>
</tr>
<tr>
<td>Update</td>
<td>Updates the state of this axis based on the axis defined by <code>AxisState.m_AxisName</code></td>
</tr>
<tr>
<td>Validate</td>
<td>Call from OnValidate: Make sure the fields are sensible</td>
</tr>
</tbody>
</table>

## See Also

Reference
- `AxisState` Structure
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
AxisStateSetThresholds Method

Sets the constraints by which this axis will operate on

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public void SetThresholds(
    float minValue,
    float maxValue,
    bool wrapAround
)
```

Parameters

- `minValue`
  Type: SystemSingle
  The lowest value this axis can achieve

- `maxValue`
  Type: SystemSingle
  The highest value this axis can achieve

- `wrapAround`
  Type: SystemBoolean
  If true, values commanded greater than mMaxValue or less than mMinValue will wrap around. If false, the value will be clamped within the range.

See Also

Reference
AxisState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
**AxisStateUpdate Method**

Updates the state of this axis based on the axis defined by AxisState.m_AxisName

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```
public bool Update(
    float deltaTime
)
```

**JavaScript**

```
// Not supported
```

### Parameters

deltaTime  
Type: System.Single  
Delta time in seconds

### Return Value

Type: Boolean  
Returns true if this axis' input was non-zero this Update, false otherwise

### See Also

Reference  
AxisState Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
AxisStateValidate Method

Call from OnValidate: Make sure the fields are sensible

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public void Validate()
```

See Also

Reference
AxisState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
AxisState Fields

The AxisState type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AccelTime</td>
<td>The amount of time in seconds it takes to accelerate to MaxSpeed with the supplied Axis at its maximum value</td>
</tr>
<tr>
<td>m_DecelTime</td>
<td>The amount of time in seconds it takes to decelerate the axis to zero if the supplied axis is in a neutral position</td>
</tr>
<tr>
<td>m_InputAxisName</td>
<td>The name of this axis as specified in Unity Input manager. Setting to an empty string will disable the automatic updating of this axis</td>
</tr>
<tr>
<td>m_InputAxisValue</td>
<td>The value of the input axis. A value of 0 means no input. You can drive this directly from a custom input system, or you can set the Axis Name and have the value driven by the internal Input Manager</td>
</tr>
<tr>
<td>m_InvertAxis</td>
<td>If checked, then the raw value of the input axis will be</td>
</tr>
</tbody>
</table>
inverted before it is used.

<table>
<thead>
<tr>
<th>m_MaxSpeed</th>
<th>How fast the axis value can travel. Increasing this number makes the behaviour more responsive to joystick input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>The current value of the axis</td>
</tr>
</tbody>
</table>

See Also

Reference
AxisState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
AxisState\_AccelTime Field

The amount of time in seconds it takes to accelerate to MaxSpeed with the supplied Axis at its maximum value

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("The amount of time in seconds to accelerate to MaxSpeed with the supplied Axis at its maximum value")]
public float m\_AccelTime
``` |  |

### Field Value

**Type:** Single

### See Also

**Reference**
- AxisState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
AxisStatem_DecelTime Field

The amount of time in seconds it takes to decelerate the axis to zero if the supplied axis is in a neutral position.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("The amount of time in seconds
public float m_DecelTime

Field Value

Type: Single

### See Also

**Reference**

- AxisState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
AxisState\_InputAxisName

Field

The name of this axis as specified in Unity Input manager. Setting to an empty string will disable the automatic updating of this axis.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

C#  
```csharp
[FormerlySerializedAsAttribute("m_AxisName")]
[TooltipAttribute("The name of this axis as specified in Unity Input manager. Setting to an empty string will disable the automatic updating of this axis")]
public string m_InputAxisName
```

### Field Value

**Type:** String

## See Also

**Reference**
- AxisState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
AxisStatem_InputAxisValue Field

The value of the input axis. A value of 0 means no input. You can drive this directly from a custom input system, or you can set the Axis Name and have the value driven by the internal Input Manager.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[NoSaveDuringPlayAttribute]
[TooltipAttribute("The value of the input axis. You can drive this directly from a custom input system, or you can set the Axis Name and have the value driven by the internal Input Manager")]
public float m_InputAxisValue
```

**Field Value**

Type: **Single**

**See Also**

Reference
- AxisState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
**AxisStatem_InvertAxis Field**

If checked, then the raw value of the input axis will be inverted before it is used.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[NoSaveDuringPlayAttribute]
[TooltipAttribute("If checked, then the raw value of the input axis will be inverted before it is used")]
public bool m_InvertAxis
```

Field Value  
Type: **Boolean**

### See Also

**Reference**  
 AxisState Structure  
 Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
AxisStateMaxSpeed Field

How fast the axis value can travel. Increasing this number makes the behaviour more responsive to joystick input.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    | JavaScript
---    |----------
[TooltipAttribute("The maximum speed of this axis"]/n
public float m_MaxSpeed

Field Value
Type: Single

See Also

Reference
AxisState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
AxisStateValue Field

The current value of the axis

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[NoSaveDuringPlayAttribute]
[TooltipAttribute("The current value of the axis.
public float Value
```

Field Value  
Type: **Single**

### See Also

**Reference**  
AxisState Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraState Structure

The output of the Cinemachine engine for a specific virtual camera. The information in this struct can be blended, and provides what is needed to calculate an appropriate camera position, orientation, and lens setting. Raw values are what the Cinemachine behaviours generate. The correction channel holds perturbations to the raw values - e.g. noise or smoothing, or obstacle avoidance corrections. Corrections are not considered when making time-based calculations such as damping. The Final position and orientation is the combination of the raw values and their corrections.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

```csharp
public struct CameraState
```

The `CameraState` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CorrectedOrientation</td>
<td>Orientation with correction applied.</td>
</tr>
<tr>
<td>CorrectedPosition</td>
<td>Position with correction applied.</td>
</tr>
<tr>
<td>Default</td>
<td>State with default values</td>
</tr>
<tr>
<td><strong>FinalOrientation</strong></td>
<td>Orientation with correction and dutch applied. This is what the final camera gets.</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>FinalPosition</strong></td>
<td>Position with correction applied. This is what the final camera gets.</td>
</tr>
<tr>
<td><strong>HasLookAt</strong></td>
<td>Returns true if this state has a valid ReferenceLookAt value.</td>
</tr>
<tr>
<td><strong>Lens</strong></td>
<td>Camera Lens Settings.</td>
</tr>
<tr>
<td><strong>NumCustomBlendables</strong></td>
<td>The number of custom blendables that will be applied to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules</td>
</tr>
<tr>
<td><strong>OrientationCorrection</strong></td>
<td>Orientation correction. This will be added to the raw orientation. This value doesn't get fed back into the system when calculating the next frame. Can be noise, or smoothing, or both, or something else.</td>
</tr>
<tr>
<td><strong>PositionCorrection</strong></td>
<td>Position correction. This will be added to the raw position. This value doesn't get fed back into the system when</td>
</tr>
</tbody>
</table>
calculating the next frame. Can be noise, or smoothing, or both, or something else.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RawOrientation</strong></td>
<td>Raw (un-corrected) world space orientation of this camera</td>
</tr>
<tr>
<td><strong>RawPosition</strong></td>
<td>Raw (un-corrected) world space position of this camera</td>
</tr>
<tr>
<td><strong>ReferenceLookAt</strong></td>
<td>The world space focus point of the camera. What the camera wants to look at. There is a special constant define to represent &quot;nothing&quot;. Be careful to check for that (or check the HasLookAt property).</td>
</tr>
<tr>
<td><strong>ReferenceUp</strong></td>
<td>Which way is up. World space unit vector.</td>
</tr>
<tr>
<td><strong>ShotQuality</strong></td>
<td>Subjective estimation of how &quot;good&quot; the shot is. Larger values mean better quality. Default is 1.</td>
</tr>
</tbody>
</table>

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddCustomBlendable</td>
<td>Add a custom blendable</td>
</tr>
</tbody>
</table>
to the pot for eventual application to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules.

---

### GetCustomBlendable
Get a custom blendable that will be applied to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules.

### Lerp
Intelligently blend the contents of two states.

---

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![s]kNoPoint</td>
<td>This constant represents &quot;no point in space&quot; or &quot;no direction&quot;.</td>
</tr>
</tbody>
</table>

---

## See Also

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CameraState Properties

The CameraState type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CorrectedOrientation</td>
<td>Orientation with correction applied.</td>
</tr>
<tr>
<td>CorrectedPosition</td>
<td>Position with correction applied.</td>
</tr>
<tr>
<td>Default</td>
<td>State with default values</td>
</tr>
<tr>
<td>FinalOrientation</td>
<td>Orientation with correction and dutch applied. This is what the final camera gets.</td>
</tr>
<tr>
<td>FinalPosition</td>
<td>Position with correction applied. This is what the final camera gets.</td>
</tr>
<tr>
<td>HasLookAt</td>
<td>Returns true if this state has a valid ReferenceLookAt value.</td>
</tr>
<tr>
<td>Lens</td>
<td>Camera Lens Settings.</td>
</tr>
<tr>
<td>NumCustomBlendables</td>
<td>The number of custom blendables that will be applied to the camera. The base system manages but otherwise</td>
</tr>
</tbody>
</table>
ignores this data - it is intended for extension modules

<table>
<thead>
<tr>
<th>OrientationCorrection</th>
<th>Orientation correction. This will be added to the raw orientation. This value doesn't get fed back into the system when calculating the next frame. Can be noise, or smoothing, or both, or something else.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PositionCorrection</td>
<td>Position correction. This will be added to the raw position. This value doesn't get fed back into the system when calculating the next frame. Can be noise, or smoothing, or both, or something else.</td>
</tr>
<tr>
<td>RawOrientation</td>
<td>Raw (un-corrected) world space orientation of this camera</td>
</tr>
<tr>
<td>RawPosition</td>
<td>Raw (un-corrected) world space position of this camera</td>
</tr>
<tr>
<td>ReferenceLookAt</td>
<td>The world space focus point of the camera. What the camera wants to look at. There is a special constant define to represent &quot;nothing&quot;. Be careful to check for</td>
</tr>
</tbody>
</table>
that (or check the HasLookAt property).

<table>
<thead>
<tr>
<th>ReferenceUp</th>
<th>Which way is up. World space unit vector.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShotQuality</td>
<td>Subjective estimation of how &quot;good&quot; the shot is. Larger values mean better quality. Default is 1.</td>
</tr>
</tbody>
</table>

**See Also**

- Reference
- CameraState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraStateCorrectedOrientation Property

Orientation with correction applied.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public Quaternion CorrectedOrientation { get; }
```

**Property Value**  
Type: Quaternion

**See Also**

Reference  
CameraState Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraStateCorrectedPosition
Property

Position with correction applied.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
public Vector3 CorrectedPosition { get; }
```

Property Value
Type: Vector3

### See Also

Reference
CameraState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraStateDefault Property

State with default values

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public static CameraState Default { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value  
Type: CameraState

**See Also**

Reference  
CameraState Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraStateFinalOrientation Property

Orientation with correction and dutch applied. This is what the final camera gets.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public Quaternion FinalOrientation { get; }
```

Property Value
Type: `Quaternion`

See Also

Reference
CameraState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraStateFinalPosition Property

Position with correction applied. This is what the final camera gets.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public Vector3 FinalPosition { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Vector3

See Also

Reference
CameraState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraStateHasLookAt Property

Returns true if this state has a valid ReferenceLookAt value.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
public bool HasLookAt { get; }
``` | |

**Property Value**  
Type: Boolean

### See Also

- Reference
  - CameraState Structure
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraStateLens Property

Camera Lens Settings.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public LensSettings Lens { get; set; }
```

Property Value  
Type: **LensSettings**

### See Also

**Reference**  
CameraState Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
**CameraStateNumCustomBlendables**: Property

The number of custom blendables that will be applied to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
public int NumCustomBlendables { get; }
```

**Property Value**

Type: `Int32`

**See Also**

**Reference**

CameraState Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraStateOrientationCorrection Property

Orientation correction. This will be added to the raw orientation. This value doesn't get fed back into the system when calculating the next frame. Can be noise, or smoothing, or both, or something else.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```
public Quaternion OrientationCorrection { get; set; }
```

**JavaScript**

```
public Quaternion OrientationCorrection { get; set; }
```

### Property Value

**Type:** Quaternion

### See Also

**Reference**

- CameraState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraStatePositionCorrection Property

Position correction. This will be added to the raw position. This value doesn't get fed back into the system when calculating the next frame. Can be noise, or smoothing, or both, or something else.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public Vector3 PositionCorrection {
    get;
    set;
}
```

### Property Value

Type: `Vector3`

### See Also

**Reference**

- CameraState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraStateRawOrientation Property

Raw (un-corrected) world space orientation of this camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public Quaternion RawOrientation { get; set; }
```

**Property Value**  
Type: Quaternion

**See Also**

- Reference  
  CameraState Structure  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraStateRawPosition Property

Raw (un-corrected) world space position of this camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>public Vector3 RawPosition { get; set; }</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: *Vector3*

**See Also**

Reference

- CameraState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CameraStateReferenceLookAt

Property

The world space focus point of the camera. What the camera wants to look at. There is a special constant define to represent "nothing". Be careful to check for that (or check the HasLookAt property).

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▶ Syntax

```c#
public Vector3 ReferenceLookAt { get; set; }
```

Property Value

Type: **Vector3**

▶ See Also

Reference

CameraState Structure

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CameraStateReferenceUp

Property

Which way is up. World space unit vector.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public Vector3 ReferenceUp { get; set; }</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: `Vector3`

⚠️ See Also

Reference
- CameraState Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraStateShotQuality Property

Subjective estimation of how "good" the shot is. Larger values mean better quality. Default is 1.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#       JavaScript

```csharp
public float ShotQuality { get; set; }
```

Property Value
Type: Single

See Also

Reference
CameraState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraState Methods

The `CameraState` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddCustomBlendable</td>
<td>Add a custom blendable to the pot for eventual application to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules</td>
</tr>
<tr>
<td>GetCustomBlendable</td>
<td>Get a custom blendable that will be applied to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules</td>
</tr>
<tr>
<td>Lerp</td>
<td>Intelligently blend the contents of two states.</td>
</tr>
</tbody>
</table>

### See Also

- **Reference**
  - CameraState Structure
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraStateAddCustomBlendable Method

Add a custom blendable to the pot for eventual application to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

![Syntax](https://unity3d.com/legal/terms-of-service)

Syntax

```csharp
public void AddCustomBlendable(
    CameraStateCustomBlendable b
)
```

Parameters

*b*

Type: `CinemachineCameraStateCustomBlendable`  
The custom blendable to add. If b.m_Custom is the same as an already-added custom blendable, then they will be merged and the weights combined.

### See Also

Reference

- CameraState Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraState.GetCustomBlendable Method

Get a custom blendable that will be applied to the camera. The base system manages but otherwise ignores this data - it is intended for extension modules.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public CameraStateCustomBlendable GetCustomBlendable(int index)
```

### Parameters

- **index**  
  Type: System.Int32  
  Which one to get. Must be in range [0...NumCustomBlendables)

### Return Value

- **Type:** CameraStateCustomBlendable  
  The custom blendable at the specified index.

## See Also

- **Reference**  
  CameraState Structure  
  Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraStateLerp Method

Intelligently blend the contents of two states.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public static CameraState Lerp(
    CameraState stateA,
    CameraState stateB,
    float t
)
```

## Parameters

- **stateA**  
  Type: **CinemachineCameraState**  
  The first state, corresponding to t=0

- **stateB**  
  Type: **CinemachineCameraState**  
  The second state, corresponding to t=1

- **t**  
  Type: **SystemSingle**  
  How much to interpolate. Internally clamped to 0..1

## Return Value

Type: **CameraState**  
Linearly interpolated CameraState

## See Also
Reference

CameraState Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraState Fields

The `CameraState` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kNoPoint</td>
<td>This constant represents &quot;no point in space&quot; or &quot;no direction&quot;.</td>
</tr>
</tbody>
</table>

### See Also

Reference
- `CameraState Structure`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CameraStatekNoPoint Field

This constant represents "no point in space" or "no direction".

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public static Vector3 kNoPoint
```

### Field Value

Type: **Vector3**

### See Also

- Reference
  - CameraState Structure
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CameraStateCustomBlendable Structure

Opaque structure represent extra blendable stuff and its weight. The base system ignores this data - it is intended for extension modules

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public struct CustomBlendable
```

The `CameraStateCustomBlendable` type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="CameraStateCustomBlendable" /></td>
<td>Constructor with specific values</td>
</tr>
</tbody>
</table>

Top

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="m_Custom" /></td>
<td>The custom stuff that the extention module will consider</td>
</tr>
</tbody>
</table>
m_Weight  The weight of the custom stuff. Must be 0...1

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CameraStateCustomBlendable
Constructor

Constructor with specific values

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public CustomBlendable(</td>
<td></td>
</tr>
<tr>
<td>Object custom,</td>
<td></td>
</tr>
<tr>
<td>float weight</td>
<td></td>
</tr>
<tr>
<td>)</td>
<td></td>
</tr>
</tbody>
</table>

### Parameters

- **custom**
  - **Type:** Object
  - The custom stuff that the extension module will consider

- **weight**
  - **Type:** System.Single
  - The weight of the custom stuff. Must be 0...1

### See Also

- Reference  
  - CameraStateCustomBlendable Structure  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CustomBlendable Fields

The **CameraStateCustomBlendable** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Custom</td>
<td>The custom stuff that the extension module will consider</td>
</tr>
<tr>
<td>m_Weight</td>
<td>The weight of the custom stuff. Must be 0...1</td>
</tr>
</tbody>
</table>

### See Also

**Reference**

- CameraStateCustomBlendable Structure
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CameraStateCustomBlendablem_Custom

Field

The custom stuff that the extension module will consider

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public</code></td>
<td><code>Object</code></td>
</tr>
</tbody>
</table>

Field Value

Type: **Object**

⚠️ See Also

Reference

CameraStateCustomBlendable Structure  
Cinemachine Namespace

Visit the Cinematic Forum  
https://unity3d.com/legal/terms-of-service
CameraStateCustomBlendablem_Weight

The weight of the custom stuff. Must be 0...1

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public float m_Weight</code></td>
<td></td>
</tr>
</tbody>
</table>

### Field Value

Type: *Single*

### See Also

**Reference**
- CameraStateCustomBlendable Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin

Class

As a part of the Cinemachine Pipeline implementing the Noise stage, this component adds Perlin Noise to the Camera state, in the Correction channel of the CameraState. The noise is created by using a predefined noise profile asset. This defines the shape of the noise over time. You can scale this in amplitude or in time, to produce a large family of different noises using the same profile.

Inheritance Hierarchy

- System
- Object
- Component
- Behaviour
- MonoBehaviour
- Cinemachine
- CinemachineComponentBase
- CinemachineBasicMultiChannelPerlin

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#     JavaScript

```csharp
[DocumentationSortingAttribute(8f, DocumentationAttribute(""))]
[RequireComponent(typeof( CinemachinePipeline ))]
[SaveDuringPlayAttribute]
public class CinemachineBasicMultiChannelPerlin :
```

The CinemachineBasicMultiChannelPerlin type exposes the following members.
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if the component is valid, i.e. it has a noise definition and is enabled. (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Noise stage (Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

Top
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies noise to the Correction channel if the delta time is greater than 0. Other (Overrides CinemachineComponentBaseMutateSingle).</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position dragged from the user. The class implementation does nothing. (Inherited from CinemachineComponentBase)</td>
</tr>
</tbody>
</table>

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AmplitudeGain</td>
<td>Gain to apply to the amplitudes defined in the settings asset.</td>
</tr>
<tr>
<td>m_FrequencyGain</td>
<td>Scale factor to apply to the frequencies defined in the settings asset.</td>
</tr>
<tr>
<td>m_NoiseProfile</td>
<td>Serialized property for referencing a NoiseSettings asset</td>
</tr>
</tbody>
</table>

## See Also

Reference
Cinemachine Namespace
CinemachineNoiseSettings

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The `CinemachineBasicMultiChannelPerlin` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| **FollowTarget** | Returns the owner vcam's Follow target.  
(Inherited from `CinemachineComponentBase`.) |
| **IsValid**   | True if the component is valid, i.e. it has a noise definition and is enabled.  
(Overrides `CinemachineComponentBaseIsValid`.) |
| **LookAtTarget** | Returns the owner vcam's LookAt target.  
(Inherited from `CinemachineComponentBase`.) |
| **Stage**     | Get the Cinemachine Pipeline stage that this component implements.  
Always returns the Noise stage  
(Overrides `CinemachineComponentBaseStage`). |
| **VcamState** | Returns the owner vcam's CameraState.  
(Inherited from `CinemachineComponentBase`.) |
VirtualCamera

Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)

See Also

Reference
CinemachineBasicMultiChannelPerlin Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin Property

True if the component is valid, i.e. it has a noise definition and is enabled.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override bool IsValid { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**
Type: **Boolean**

### See Also

**Reference**
- CinemachineBasicMultiChannelPerlin Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin
Property

Get the Cinemachine Pipeline stage that this component implements. Always returns the Noise stage

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

**C#**

```csharp
public override CinemachineCoreStage Stage { get; }
```

**JavaScript**

### Property Value

Type: `CinemachineCoreStage`

### See Also

**Reference**

- CinemachineBasicMultiChannelPerlin Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin Methods

The `CinemachineBasicMultiChannelPerlin` type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies noise to the Correction channel if the delta time is greater than 0. Otherwise, does nothing. (Overrides <code>CinemachineComponentBase.MutateCameraState(CameraState Single)</code>.)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. The class implementation does nothing. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
</tbody>
</table>

**See Also**

Reference
- `CinemachineBasicMultiChannelPerlin Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin
Method

Applies noise to the Correction channel of the CameraState if the delta
time is greater than 0. Otherwise, does nothing.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
C#     JavaScript

public override void MutateCameraState(  
    ref CameraState curState,  
    float deltaTime  
)
```

### Parameters

- **curState**  
  - Type: CinemachineCameraState  
  - The current camera state

- **deltaTime**  
  - Type: System.Single  
  - How much to advance the perlin noise generator. Noise is only applied if this value is greater than or equal to 0

### See Also

- Reference  
  - CinemachineBasicMultiChannelPerlin Class  
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBasicMultiChannelPerlin Fields

The CinemachineBasicMultiChannelPerlin type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AmplitudeGain</td>
<td>Gain to apply to the amplitudes defined in the settings asset.</td>
</tr>
<tr>
<td>m_FrequencyGain</td>
<td>Scale factor to apply to the frequencies defined in the settings asset.</td>
</tr>
<tr>
<td>m_NoiseProfile</td>
<td>Serialized property for referencing a NoiseSettings asset</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineBasicMultiChannelPerlin Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin Field

Gain to apply to the amplitudes defined in the settings asset.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[TooltipAttribute("Gain to apply to the amplitudes defined in the NoiseSettings asset. 1 is normal. Setting this to 0 completely mutes the noise.")]
public float m_AmplitudeGain` |  |

### Field Value
Type: Single

### See Also

**Reference**  
CinemachineBasicMultiChannelPerlin Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBasicMultiChannelPerlin Field

Scale factor to apply to the frequencies defined in the settings asset.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public float m_FrequencyGain
```

Field Value  
Type: Single

### See Also

- Reference  
  CinemachineBasicMultiChannelPerlin Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBasicMultiChannelPerlin Field

Serialized property for referencing a NoiseSettings asset

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[HideInInspector]
[TooltipAttribute("The asset containing the Noise Profile")]
[FormerlySerializedAsAttribute("m_Definition")]
public NoiseSettings m_NoiseProfile
```

**JavaScript**

```javascript
//
```

### Field Value

Type: NoiseSettings

### See Also

**Reference**

- CinemachineBasicMultiChannelPerlin Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBlend Class

Describes a blend between 2 Cinemachine Virtual Cameras, and holds the current state of the blend.

▲ Inheritance Hierarchy

System\Object  Cinemachine\CinemachineBlend

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▲ Syntax

```
public class CinemachineBlend
```

The CinemachineBlend type exposes the following members.

▲ Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚡</td>
<td>CinemachineBlend</td>
</tr>
</tbody>
</table>

▲ Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🎥 BlendCurve</td>
<td>The curve that describes the way the blend transitions over time from the first camera to the second. X-</td>
</tr>
</tbody>
</table>
axis is time in seconds over which the blend takes place and Y axis is blend weight (0..1)

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BlendWeight</strong></td>
<td>The current weight of the blend. This is an evaluation of the BlendCurve at the current time relative to the start of the blend. 0 means camA, 1 means camB.</td>
</tr>
<tr>
<td><strong>CamA</strong></td>
<td>First camera in the blend</td>
</tr>
<tr>
<td><strong>CamB</strong></td>
<td>Second camera in the blend</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Text description of the blend, for debugging</td>
</tr>
<tr>
<td><strong>Duration</strong></td>
<td>Duration in seconds of the blend. This is given read from the BlendCurve.</td>
</tr>
<tr>
<td><strong>IsComplete</strong></td>
<td>True if the time relative to the start of the blend is greater than or equal to the blend duration</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>Validity test for the blend. True if both cameras are defined.</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>Compute the blended CameraState for the current time in the blend.</td>
</tr>
<tr>
<td><strong>TimeInBlend</strong></td>
<td>The current time relative to the start of the blend</td>
</tr>
</tbody>
</table>

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>UpdateCameraState</td>
<td>Make sure the source cameras get updated.</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>Uses</td>
<td>Does the blend use a specific Cinemachine Virtual Camera?</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlend Constructor

Construct a blend

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public CinemachineBlend(
    ICinemachineCamera a,
    ICinemachineCamera b,
    AnimationCurve curve,
    float duration,
    float t
)
```

## Parameters

- **a**  
  Type: CinemachineICinemachineCamera  
  First camera

- **b**  
  Type: CinemachineICinemachineCamera  
  Second camera

- **curve**  
  Type: AnimationCurve  
  Blend curve

- **duration**  
  Type: SystemSingle  
  [Missing <param name="duration"/> documentation for "M:Cinemachine.CinemachineBlend.#ctor(Cinemachine.ICinemachineCamera,Cin

- **t**
Type: SystemSingle
Current time in blend, relative to the start of the blend

See Also

Reference
CinemachineBlend Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlend Properties

The `CinemachineBlend` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BlendCurve</strong></td>
<td>The curve that describes the way the blend transitions over time from the first camera to the second. X-axis is time in seconds over which the blend takes place and Y axis is blend weight (0..1)</td>
</tr>
<tr>
<td><strong>BlendWeight</strong></td>
<td>The current weight of the blend. This is an evaluation of the BlendCurve at the current time relative to the start of the blend. 0 means camA, 1 means camB.</td>
</tr>
<tr>
<td><strong>CamA</strong></td>
<td>First camera in the blend</td>
</tr>
<tr>
<td><strong>CamB</strong></td>
<td>Second camera in the blend</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Text description of the blend, for debugging</td>
</tr>
<tr>
<td><strong>Duration</strong></td>
<td>Duration in seconds of the blend. This is given read from the BlendCurve.</td>
</tr>
<tr>
<td><strong>IsComplete</strong></td>
<td>True if the time relative to the start of the blend is greater than or equal to the blend duration</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>Validity test for the blend. True if both cameras are defined.</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>Compute the blended CameraState for the current time in the blend.</td>
</tr>
<tr>
<td><strong>TimeInBlend</strong></td>
<td>The current time relative to the start of the blend</td>
</tr>
</tbody>
</table>

**See Also**

Reference
- CinemachineBlend Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendBlendCurve

Property

The curve that describes the way the blend transitions over time from the first camera to the second. X-axis is time in seconds over which the blend takes place and Y axis is blend weight (0..1)

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public AnimationCurve BlendCurve { get; set; }
```

### Property Value

Type: *AnimationCurve*

### See Also

Reference

*CinemachineBlend Class*

*Cinemachine Namespace*

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendBlendWeight Property

The current weight of the blend. This is an evaluation of the BlendCurve at the current time relative to the start of the blend. 0 means camA, 1 means camB.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public float BlendWeight { get; }
```

Property Value
Type: **Single**

### See Also

**Reference**
- CinemachineBlend Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendCamA

Property

First camera in the blend

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public ICinemachineCamera CamA { get; set; }</code></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: `ICinemachineCamera`

## See Also

**Reference**

- CinemachineBlend Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendCamB

Property

Second camera in the blend

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public ICinemachineCamera CamB { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: ICinemachineCamera

**See Also**

Reference

- CinemachineBlend Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBlendDescription Property

Text description of the blend, for debugging

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```
C#  JavaScript
```  

```
public string Description { get; }
```  

**Property Value**

Type: **String**

**See Also**

Reference  
CinemachineBlend Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendDuration Property

Duration in seconds of the blend. This is given read from the BlendCurve.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public float Duration { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**  
Type: Single

### See Also

**Reference**  
CinemachineBlend Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendIsComplete Property

True if the time relative to the start of the blend is greater than or equal to the blend duration

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public bool IsComplete { get; }
```

Property Value
Type: Boolean

See Also

Reference
CinemachineBlend Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendIsValid
Property

Validity test for the blend. True if both cameras are defined.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public bool IsValid { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Boolean

See Also

Reference
CinemachineBlend Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendState Property

Compute the blended CameraState for the current time in the blend.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public CameraState State { get; }
```

Property Value  
Type: CameraState

**See Also**

- Reference
  - CinemachineBlend Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlend.TimeInBlend Property

The current time relative to the start of the blend

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public float TimeInBlend { get; set; }
```

**Property Value**  
Type: Single

**See Also**

Reference  
CinemachineBlend Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlend Methods

The CinemachineBlend type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UpdateCameraState</td>
<td>Make sure the source cameras get updated.</td>
</tr>
<tr>
<td>Uses</td>
<td>Does the blend use a specific Cinemachine Virtual Camera?</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineBlend Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendUpdateCameraState Method

Make sure the source cameras get updated.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

### C#  
```csharp
public void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

### JavaScript

See Also

**Reference**
- CinemachineBlend Class  
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendUses Method

Does the blend use a specific Cinemachine Virtual Camera?

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public bool Uses(
    ICinemachineCamera cam
)
```

### Parameters

**cam**
- Type: `CinemachineICinemachineCamera`
- The camera to test

### Return Value

Type: `Boolean`
True if the camera is involved in the blend

### See Also

Reference
- CinemachineBlend Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendDefinition Structure

Definition of a Camera blend. This struct holds the information necessary to generate a suitable AnimationCurve for a Cinemachine Blend.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(10.2f, DocumentationSortingAttribute
public struct CinemachineBlendDefinition
```

The CinemachineBlendDefinition type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBlendDefinition</td>
<td>Constructor</td>
</tr>
</tbody>
</table>

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlendCurve</td>
<td>An AnimationCurve specifying the</td>
</tr>
</tbody>
</table>
interpolation duration and value for this camera blend. The time of the last key frame is assumed to be the duration of the blend. Y-axis values must be in range [0,1] (internally clamped within Blender) and time must be in range of [0, +infinity)

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Style</td>
<td>The shape of the blend curve.</td>
</tr>
<tr>
<td>m_Time</td>
<td>The duration (in seconds) of the blend</td>
</tr>
</tbody>
</table>

See Also

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendDefinition

Constructor

Constructor

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
public CinemachineBlendDefinition(
    CinemachineBlendDefinitionStyle style,
    float time
)
``` | ```javascript
public CinemachineBlendDefinition(style: CinemachineBlendDefinitionStyle, time: float)
``` |

### Parameters

- **style**
  - Type: CinemachineBlendDefinitionStyle  
  - The shape of the blend curve.

- **time**
  - Type: System.Single  
  - The duration (in seconds) of the blend

### See Also

- Reference
  - CinemachineBlendDefinition Structure
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendDefinition
Properties

The CinemachineBlendDefinition type exposes the following members.

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlendCurve</td>
<td>An AnimationCurve specifying the interpolation duration and value for this camera blend. The time of the last key frame is assumed to be the duration of the blend. Y-axis values must be in range [0,1] (internally clamped within Blender) and time must be in range of [0, +infinity)</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineBlendDefinition Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendDefinition.BlendCurve Property

An AnimationCurve specifying the interpolation duration and value for this camera blend. The time of the last key frame is assumed to be the duration of the blend. Y-axis values must be in range [0,1] (internally clamped within Blender) and time must be in range of [0, +infinity)

**Namespace**: Cinemachine  
**Assembly**: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public AnimationCurve BlendCurve { get; }</td>
<td></td>
</tr>
</tbody>
</table>

### Property Value

**Type**: AnimationCurve

### See Also

**Reference**
- CinemachineBlendDefinition Structure
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendDefinition Fields

The `CinemachineBlendDefinition` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Style</td>
<td>The shape of the blend curve.</td>
</tr>
<tr>
<td>m_Time</td>
<td>The duration (in seconds) of the blend</td>
</tr>
</tbody>
</table>

### See Also

- **Reference**
  - `CinemachineBlendDefinition Structure`
  - `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendDefinitionm_Style Field

The shape of the blend curve.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

### C#

```csharp
[TooltipAttribute("Shape of the blend curve")]
public CinemachineBlendDefinitionStyle m_Style
```

Field Value

Type: CinemachineBlendDefinitionStyle

See Also

Reference

CinemachineBlendDefinition Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendDefinition.m_Time Field

The duration (in seconds) of the blend

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://via.placeholder.com/150" alt="Code Snippet" /></td>
<td><img src="https://via.placeholder.com/150" alt="Code Snippet" /></td>
</tr>
</tbody>
</table>

Field Value  
Type: Single

## See Also

Reference  
CinemachineBlendDefinition Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendDefinitionStyle Enumeration

Supported predefined shapes for the blend curve.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[DocumentationSortingAttribute(10.21f, DocumentationSortingAttribute)</code></td>
<td></td>
</tr>
<tr>
<td><code>public enum Style</code></td>
<td></td>
</tr>
</tbody>
</table>

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cut</td>
<td>0</td>
<td>Zero-length blend</td>
</tr>
<tr>
<td>EaseInOut</td>
<td>1</td>
<td>S-shaped curve, giving a gentle and smooth transition</td>
</tr>
<tr>
<td>EaseIn</td>
<td>2</td>
<td>Linear out of the outgoing shot, and easy into the incoming</td>
</tr>
<tr>
<td>EaseOut</td>
<td>3</td>
<td>Easy out of the outgoing shot, and linear into the incoming</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Easy out of the outgoing, and hard into the incoming</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>----------------------------------------------------</td>
</tr>
<tr>
<td><strong>HardIn</strong></td>
<td>4</td>
<td><strong>HardOut</strong> 5</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Hard out of the outgoing, and easy into the incoming</td>
</tr>
<tr>
<td><strong>Linear</strong></td>
<td>6</td>
<td>Linear blend. Mechanical-looking.</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

[Cinemachine Namespace](https://unity3d.com/legal/terms-of-service)

Visit the Cinemachine Forum

Cinemachine
CinemachineBlendDefinitionPropertyAttribute

Class

Property applied to CinemachineBlendDefinition. Used for custom drawing in the inspector.

Inheritance Hierarchy

System
  Object
  SystemAttribute
    PropertyAttribute
      Cinemachine
        CinemachineBlendDefinitionPropertyAttribute

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

Public sealed class CinemachineBlendDefinitionPropertyAttribute

The CinemachineBlendDefinitionPropertyAttribute type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBlendDefinitionPropertyAttribute</td>
<td></td>
</tr>
</tbody>
</table>

See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendDefinitionPropertyAttribute Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineBlendDefinitionPropertyAttribute.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

► Syntax

C#      JavaScript

```csharp
public CinemachineBlendDefinitionPropertyAttribute()
```

► See Also

Reference
CinemachineBlendDefinitionPropertyAttribute Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlenderSettings Class

Asset that defines the rules for blending between Virtual Cameras.

Inheritance Hierarchy

- **System**
  - **Object**
    - **ScriptableObject**
      - **Cinemachine**
        - **CinemachineBlenderSettings**

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll)  
**Version:** 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(10f, DocumentationSortingAttribute)]
public sealed class CinemachineBlenderSettings : 
```

The CinemachineBlenderSettings type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="CinemachineBlenderSettings" /></td>
<td>CinemachineBlenderSettings</td>
</tr>
</tbody>
</table>

Top
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetBlendCurveForVirtualCameras</td>
<td>Attempts to find a blend curve which matches the to and from cameras as specified. If no match is found, the function returns either the default blend for this Blender or NULL depending on the state returnDefaultOnNoMatch.</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kBlendFromAnyCameraLabel</td>
<td>Internal API for the inspector editopr: a label to represent any camera</td>
</tr>
<tr>
<td>m_CustomBlends</td>
<td>The array containing explicitly defined blends between two Virtual Cameras</td>
</tr>
</tbody>
</table>

### See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettings Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineBlenderSettings.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public CinemachineBlenderSettings()
```

See Also

Reference
CinemachineBlenderSettings Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettings

Methods

The CinemachineBlenderSettings type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetBlendCurveForVirtualCameras</td>
<td>Attempts to find a blend curve which matches the to and from cameras as specified. If no match is found, the function returns either the default blend curve for this Blender or NULL depending on the state returnDefaultOnNoMatch.</td>
</tr>
</tbody>
</table>

Top

See Also

Reference

CinemachineBlenderSettings Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettingsGetBlendCurveForVirtualCameras Method

Attempts to find a blend curve which matches the to and from cameras as specified. If no match is found, the function returns either the default blend for this Blender or NULL depending on the state of returnDefaultOnNoMatch.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public AnimationCurve GetBlendCurveForVirtualCameras(string fromCameraName, string toCameraName, AnimationCurve defaultCurve)
```

Parameters

fromCameraName
Type: System.String
The game object name of the from camera
toCameraName
Type: System.String
The game object name of the to camera
defaultCurve
Type: AnimationCurve
Curve to return if no curve found. Can be NULL.
Return Value
Type: AnimationCurve

See Also

Reference
CinemachineBlenderSettings Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The `CinemachineBlenderSettings` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>kBlendFromAnyCameraLabel</code></td>
<td>Internal API for the inspector editor: a label to represent any camera</td>
</tr>
<tr>
<td><code>m_CustomBlends</code></td>
<td>The array containing explicitly defined blends between two Virtual Cameras</td>
</tr>
</tbody>
</table>

### See Also

- Reference
  - `CinemachineBlenderSettings Class`
  - `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlenderSettings.kBlendFromAnyCamera Label

Internal API for the inspector editor: a label to represent any camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public const string kBlendFromAnyCameraLabel = "ANY CAMERA";
```

**Field Value**  
Type: String

**See Also**

- Reference  
  CinemachineBlenderSettings Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlenderSettingsm_CustomBlends Field

The array containing explicitly defined blends between two Virtual Cameras

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[TooltipAttribute("The array containing explicit")]
public CinemachineBlenderSettingsCustomBlend[] m_ |

Field Value  
Type: CinemachineBlenderSettingsCustomBlend

### See Also

Reference  
CinemachineBlenderSettings Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettingsCustom Structure

Container specifying how two specific Cinemachine Virtual Cameras blend together.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(10.1f, DocumentationSortingAttribute)
public struct CustomBlend
```

The `CinemachineBlenderSettingsCustomBlend` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Blend</td>
<td></td>
</tr>
<tr>
<td>m_From</td>
<td></td>
</tr>
<tr>
<td>m_To</td>
<td></td>
</tr>
</tbody>
</table>

See Also
Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CustomBlend Fields

The CinemachineBlenderSettingsCustomBlend type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Blend</td>
<td></td>
</tr>
<tr>
<td>m_From</td>
<td></td>
</tr>
<tr>
<td>m_To</td>
<td></td>
</tr>
</tbody>
</table>

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettingsCustomField

[Missing <summary> documentation for "F:Cinemachine.CinemachineBlenderSettings.CustomBlend.m_Blend"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;Blend curve definition&quot;)] public CinemachineBlendDefinition m_Blend</td>
<td></td>
</tr>
</tbody>
</table>

Field Value
Type: CinemachineBlendDefinition

See Also

Reference
CinemachineBlenderSettingsCustomBlend Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettingsCustomBlend Field

[Missing <summary> documentation for "F:Cinemachine.CinemachineBlenderSettings.CustomBlend.m_From"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
C#       JavaScript
[TooltipAttribute("When blending from this camera")
public string m_From
```

Field Value
Type: String

See Also

Reference
CinemachineBlenderSettingsCustomBlend Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlenderSettingsCustomBlend Field

[Missing <summary> documentation for "F:Cinemachine.CinemachineBlenderSettings.CustomBlend.m_To"]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("When blending to this camera")]
public string m_To |

**Field Value**  
Type: String

### See Also

**Reference**  
CinemachineBlenderSettingsCustomBlend Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera Class

This is a virtual camera "manager" that owns and manages a collection of child Virtual Cameras. When the camera goes live, these child vcams are enabled, one after another, holding each camera for a designated time. Blends between cameras are specified. The last camera is held indefinitely.

Inheritance Hierarchy

System
  Object
  Component
    Behaviour
      MonoBehaviour
        Cinemachine
          CinemachineVirtualCameraBase
            CinemachineCinemachineBlendListCamera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    JavaScript

```csharp
[DocumentationSortingAttribute(13f, Documentationator
[ExecuteInEditMode]
[DisallowMultipleComponent]
[AddComponentMenu("Cinemachine/CinemachineBlendLi
public class CinemachineBlendListCamera : Cinemac
```

The CinemachineBlendListCamera type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBlendListCamera</td>
<td></td>
</tr>
</tbody>
</table>

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>The list of child cameras. These are just the immediate children in the hierarchy.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug information. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the current Follow target. If parent is non-null and no specific Follow defined for this camera (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>LiveChild</td>
<td>Get the current &quot;best&quot; child virtual camera. Would be chosen if the State Driven Camera were active.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the current LookAt target. LookAt if parent is non-null and</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera.</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-camera situations.</td>
</tr>
<tr>
<td>PreviousStateIsInvalid</td>
<td>Set this to force the next update to ignore deltaTime and reset itself.</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera, which determines its placement.</td>
</tr>
<tr>
<td>State</td>
<td>The State of the current live child.</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed, for upgrading legacy.</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the Virtual Camera behaviour.</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is a live child of this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is a Live camera if it shares the highest priority in the queue with its peers, and the most recent one goes to the top of the priority subqueue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Makes sure the internal child cache is up to date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransformChildrenChanged</td>
<td>Makes sure the internal child cache is up to date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>(Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields, when changed in inspector. Base method is called, ValidatingStreamVersion will be valid.</td>
</tr>
</tbody>
</table>
PreUpdateChildCameras

This is called prior to the vcams are updating on FixedUpdate, LateUpdate instead. Base class implementation does nothing.

RemovePostPipelineStageHook

Remove a Pipeline stage hook.

ResolveFollow

Returns this vcam's Follow target, or if that is null, will return null.

ResolveLookAt

Returns this vcam's LookAt target, or if that is null, will return null.

Start

Base class implementation does nothing.

Update

Base class implementation makes sure the priority queue remains up-to-date.

UpdateCameraState

Called by CinemachineCore Core at designated update time updates all the children, chooses the best one, and implements any required blending.

ValidateInstructions

Internal API for the inspector editor.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It support the soon-to-be-removed Cinemachine Debugger in the Editor.</td>
</tr>
</tbody>
</table>
- **m_ChildCameras**: Internal API for the editor. Use this field

- **m_EnableAllChildCameras**: Force all child cameras to be enabled. This is useful for animating them in Timeline, but consumes extra resources.

- **m_ExcludedPropertiesInInspector**: Inspector control - Use for hiding sections of the Inspector UI.

- **m_Follow**: Default object for the children that want to move with the body target, if not specified in a child rig. May be empty.

- **m_Instructions**: The set of instructions associating virtual cameras with states. Set of instructions for enabling child cameras.

- **m_LockStageInInspector**: Inspector control - Use for enabling sections of the UI.

- **m_LookAt**: Default object for the children to look at (the target), if not specified in a child rig. May be empty.

- **m_Priority**: The priority will determine which camera becomes active based on...
the state of other cameras. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_ShowDebugText</td>
<td>When enabled, the current camera and blend will be indicated in the game debugging.</td>
</tr>
<tr>
<td>OnPostPipelineStage</td>
<td>A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCamera Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineBlendListCamera.#ctor"]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th></th>
<th>C#</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>public CinemachineBlendListCamera()</td>
</tr>
</tbody>
</table>

### See Also

- Reference
  - CinemachineBlendListCamera Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
## CinemachineBlendListCamera Properties

The `CinemachineBlendListCamera` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>The list of child cameras. These are just the immediate children in the hierarchy.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug information. (Overrides <code>CinemachineVirtualCameraBase.Description</code>)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the current Follow target. If no specific Follow is defined for this camera, get parent's Follow if parent is non-null. (Overrides <code>CinemachineVirtualCameraBase.Follow</code>)</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>LiveChild</td>
<td>Get the current &quot;best&quot; child virtual camera, that would be chosen if the State Driven Camera were active.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides <code>CinemachineVirtualCameraBase.LiveChildOrSelf</code>)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the current LookAt target.</td>
</tr>
</tbody>
</table>
LookAt if parent is non-null and LookAt defined for this camera
(Overrides CinemachineVirtualCameraBase)

Name
Get the name of the Virtual Camera. The implementation returns the owner GameObject's name.
(Inherited from CinemachineVirtualCameraBase)

ParentCamera
Support for meta-virtual-cameras. In a situation where a virtual camera is the public face of a private army of which it manages on its own. To find the VirtualCamera owner, if any, the parent vcam. The method gets the VirtualCamera owner, if any.
(Inherited from CinemachineVirtualCameraBase)

PreviousStateIsValid
Set this to force the next update to use deltaTime and reset itself
(Inherited from CinemachineVirtualCameraBase)

Priority
Get the Priority of the virtual camera. This determines its placement in the CinemachineCore's queue of eligible shots.
(Inherited from CinemachineVirtualCameraBase)

State
The State of the current live child camera.
(Overrides CinemachineVirtualCameraBase)

ValidatingStreamVersion
Version that was last streamed, for upgrading legacy
(Inherited from CinemachineVirtualCameraBase)

VirtualCameraGameObject
The GameObject owner of the Virtual Camera behaviour.
(Inherited from CinemachineVirtualCameraBase)
See Also

Reference
CinemachineBlendListCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCameraChildCameras

Property

The list of child cameras. These are just the immediate children in the hierarchy.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#   JavaScript

```csharp
public CinemachineVirtualCameraBase[] ChildCameras
```

Property Value

Type: CinemachineVirtualCameraBase

See Also

Reference

- CinemachineBlendListCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCameraDescription Property

Gets a brief debug description of this virtual camera, for use when displaying debug info

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public override string Description { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value  
Type: **String**  
Implements **ICinemachineCameraDescription**

### See Also

**Reference**  
CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCameraFollow

Property

Get the current Follow target. Returns parent's Follow if parent is non-null and no specific Follow defined for this camera

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override Transform Follow { get; set; }
```

**Property Value**

Type: **Transform**

Implements

**ICinemachineCameraFollow**

**See Also**

Reference

- CinemachineBlendListCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera.IsBlending Property

Is there a blend in progress?

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

- **Syntax**

  ```csharp
  public bool IsBlending { get; }
  ```

  ```javascript
  // Not available in JavaScript
  ```

- **Property Value**
  
  Type: **Boolean**

- **See Also**

  - Reference
    - CinemachineBlendListCamera Class
    - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraLiveChild Property

Get the current "best" child virtual camera, that would be chosen if the State Driven Camera were active.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public ICinemachineCamera LiveChild { get; set; }
```

Property Value  
Type: ICinemachineCamera

### See Also

Reference  
- CinemachineBlendListCamera Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraLiveC Property

Return the live child.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override ICinemachineCamera LiveChildOrSelf
```

**Property Value**

Type: ICinemachineCamera  
Implements ICinemachineCameraLiveChildOrSelf

**See Also**

Reference  
CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraLookAt

Property

Get the current LookAt target. Returns parent’s LookAt if parent is non-null and no specific LookAt defined for this camera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override Transform LookAt { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Transform
Implements ICinemachineCameraLookAt

See Also

Reference
- CinemachineBlendListCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraState Property

The State of the current live child

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```csharp
public override CameraState State { get; }
```

Property Value
Type: CameraState

Implements
ICinemachineCameraState

See Also

Reference
CinemachineBlendListCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The **CinemachineBlendListCamera** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all its peers. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is live. If the vcam is the most recent one of its peers, it will become live. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, this method moves the most recent one to the top of the priority subqueue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Makes sure the internal child cache is up-to-date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Function Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>OnTransformChildrenChanged</strong></td>
<td>Makes sure the internal child cache is up to date.</td>
</tr>
<tr>
<td><strong>OnTransformParentChanged</strong></td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnTransitionFromCamera</strong></td>
<td>(Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnValidate</strong></td>
<td>Enforce bounds for fields when changed in inspector.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>PreUpdateChildCameras</strong></td>
<td>This is called prior to the child cameras being updated on FixedUpdate. LateUpdate instead. Base class implementation does nothing.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>RemovePostPipelineStageHook</strong></td>
<td>Remove a Pipeline stage hook callback.</td>
</tr>
<tr>
<td></td>
<td>(Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ResolveFollow</strong></td>
<td>Returns this vcam's Follow target, or if that is null, will return null.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ResolveLookAt</strong></td>
<td>Returns this vcam's LookAt target, or if that is null, will return null.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Base class implementation does nothing.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Update</strong></td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>UpdateCameraState</strong></td>
<td>Called by CinemachineCore, updates all the children, and implements any required blending.</td>
</tr>
<tr>
<td></td>
<td>(Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ValidateInstructions</strong></td>
<td>Internal API for the inspector editor.</td>
</tr>
</tbody>
</table>

Top
See Also

Reference
- CinemachineBlendListCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera.IsLiveChild Method

Check whether the vcam a live child of this camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#

```csharp
public override bool IsLiveChild(
    ICinemachineCamera vcam
)
```

JavaScript

```
public override bool IsLiveChild(
    ICinemachineCamera vcam
)
```

Parameters

vcam
Type: CinemachineICinemachineCamera
The Virtual Camera to check

Return Value
Type: Boolean
True if the vcam is currently actively influencing the state of this vcam

Implements
ICinemachineCamera.IsLiveChild(ICinemachineCamera)

See Also

Reference
CinemachineBlendListCamera Class
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraOnEnable Method

Makes sure the internal child cache is up to date

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#       JavaScript

```csharp
protected override void OnEnable()
```

See Also

Reference
- CinemachineBlendListCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera OnTransformChildrenChanged Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

C#  
```csharp
public void OnTransformChildrenChanged()
```

JavaScript  
```
```

**See Also**

Reference  
CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera OnTransitionFromCamera Method

[Missing <summary> documentation for "M:Cinemachine.CinemachineBlendListCamera.OnTransitionFromCamera(Cinemachine.ICinemachineCamera)"

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override void OnTransitionFromCamera( ICinemachineCamera fromCam )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Parameters**

*fromCam*
Type: Cinemachine!ICinemachineCamera

[Missing <param name="fromCam"/> documentation for "M:Cinemachine.CinemachineBlendListCamera.OnTransitionFromCamera(Cinemachine.ICinemachineCamera)"

**Implements**
ICinemachineCameraOnTransitionFromCamera(I CIMachineCamera

**See Also**

Reference
CinemachineBlendListCamera Class
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCameraRemovePostPipelineStageHook Method

Remove a Pipeline stage hook callback. Make sure it is removed from all the children.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public override void RemovePostPipelineStageHook(CinemachineVirtualCameraBaseOnPostPipelineStageDelegate d)
```

**JavaScript**

```javascript
```

### Parameters

- **d**
  - Type: `CinemachineVirtualCameraBaseOnPostPipelineStageDelegate`
  - The delegate to remove.

### See Also

- **Reference**
  - CinemachineBlendListCamera Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCameraUpdateCameraState Method

Called by CinemachineCore at designated update time so the vcam can position itself and track its targets. This implementation updates all the children, chooses the best one, and implements any required blending.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public override void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

### Parameters

`worldUp`  
Type: `Vector3`  
Default world Up, set by the CinemachineBrain

`deltaTime`  
Type: `SystemSingle`  
Delta time for time-based effects (ignore if less than or equal to 0)

Implements  
ICinemachineCameraUpdateCameraState(Vector3, Single)

### See Also
Reference
CinemachineBlendListCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCameraValidateInstructions

Method

Internal API for the inspector editor.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

⚠️ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public void ValidateInstructions()</td>
<td></td>
</tr>
</tbody>
</table>

⚠️ See Also

Reference

CinemachineBlendListCamera Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera

Fields

The CinemachineBlendListCamera type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It support the soon-to-be-removed Cinemachine Debugger Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_ChildCameras</td>
<td>Internal API for the editor. Use this field</td>
</tr>
<tr>
<td>m_EnableAllChildCameras</td>
<td>Force all child cameras to be enabled. This is useful if animating them in Timeline, but consumes extra resources.</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use for hiding sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_Follow</td>
<td>Default object for the child rig (if children wants to move with a body target), if not specified, use the body target.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_Instructions</td>
<td>The set of instructions associating virtual cameras with specific states.</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Use for enabling sections of the UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_LookAt</td>
<td>Default object for the children to look at (the target), if not specified in the rig. May be empty.</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_ShowDebugText</td>
<td>When enabled, the current camera and blend will be indicated in the game window for debugging.</td>
</tr>
<tr>
<td>OnPostPipelineStage</td>
<td>A delegate to hook into the state calculation pipeline. Implementation must be designed to call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
See Also

Reference
  CinemachineBlendListCamera Class
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera

Field

Internal API for the editor. Do not use this field

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

**Syntax**

```c#
[SerializeField]
[HideInInspector]
[NoSaveDuringPlayAttribute]
public CinemachineVirtualCameraBase[] m_ChildCameras
```

Field Value

Type: CinemachineVirtualCameraBase

**See Also**

Reference

CinemachineBlendListCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera.m_EnableAllChildCameras Field

Force all child cameras to be enabled. This is useful if animating them in Timeline, but consumes extra resources.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| [TooltipAttribute("Force all child cameras to be enabled. This is useful if animating them in Timeline, but consumes extra resources")]

```csharp
public bool m_EnableAllChildCameras
```

**Field Value**  
Type: **Boolean**

### See Also

**Reference**  
CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBlendListCamera_Fo Field

Default object for the camera children wants to move with (the body target), if not specified in a child rig. May be empty

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[TooltipAttribute("Default object for the camera (the body target), if not specified in a child rig. May be empty.")]
public Transform m_Follow
```

**JavaScript**

```javascript
// No corresponding JavaScript attribute
```

### Field Value

Type: `Transform`

### See Also

Reference

CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera::m_Instructions

Field

The set of instructions associating virtual cameras with states. The set of instructions for enabling child cameras.

Namespace:  Cinemachine
Assembly:  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  

```csharp
[TooltipAttribute("The set of instructions for enabling child cameras.
public CinemachineBlendListCameraInstruction[] m_
``` 

JavaScript  

```javascript
```

Field Value

Type:  CinemachineBlendListCameraInstruction

See Also

Reference

CinemachineBlendListCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera::LookAt Field

Default object for the camera children to look at (the aim target), if not specified in a child rig. May be empty.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("Default object for the camera aim target, if not specified in a child rig. May be empty.")]
| public Transform m_LookAt |

**Field Value**  
**Type:** Transform

### See Also

Reference  
CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCamera_ShField

When enabled, the current camera and blend will be indicated in the game window, for debugging

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[TooltipAttribute("When enabled, the current child camera and blend will be indicated in the game window, for debugging")]
public bool m_ShowDebugText
```

**JavaScript**

```javascript
public bool m_ShowDebugText
```

### Field Value

Type: **Boolean**

### See Also

**Reference**  
CinemachineBlendListCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraInstruction Structure

This represents a single entry in the instruction list of the BlendListCamera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```c#
[SerializableAttribute]
public struct Instruction
```

The `CinemachineBlendListCameraInstruction` type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Blend</td>
<td>How to blend to the next virtual camera in the list (if any)</td>
</tr>
<tr>
<td>m_Hold</td>
<td>How long to wait (in seconds) before activating the next virtual camera in the list (if any)</td>
</tr>
<tr>
<td>m_VirtualCamera</td>
<td>The virtual camera to activate when this instruction becomes</td>
</tr>
</tbody>
</table>
Instruction Fields

The `CinemachineBlendListCameraInstruction` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Blend</td>
<td>How to blend to the next virtual camera in the list (if any)</td>
</tr>
<tr>
<td>m_Hold</td>
<td>How long to wait (in seconds) before activating the next virtual camera in the list (if any)</td>
</tr>
<tr>
<td>m_VirtualCamera</td>
<td>The virtual camera to activate when this instruction becomes active</td>
</tr>
</tbody>
</table>

See Also

Reference

- `CinemachineBlendListCameraInstruction Structure`
- `Cinemachine Namespace`

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraInstruction Field

How to blend to the next virtual camera in the list (if any)

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[CinemachineBlendDefinitionPropertyAttribute]
[TooltipAttribute("How to blend to the next virtual camera in the list (if any)")]
public CinemachineBlendDefinition m_Blend
```

**Field Value**  
Type: CinemachineBlendDefinition

**See Also**

Reference  
CinemachineBlendListCameraInstruction Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraInstruction

Field

How long to wait (in seconds) before activating the next virtual camera in the list (if any)

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("How long to wait (in seconds)")
public float m_Hold
|

Field Value

Type: Single

See Also

Reference

CinemachineBlendListCameraInstruction Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBlendListCameraInstruction Field

The virtual camera to activate when this instruction becomes active

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>Language</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>C#</td>
<td><code>public CinemachineVirtualCameraBase m_VirtualCamera</code></td>
</tr>
<tr>
<td>JavaScript</td>
<td><code>m_VirtualCamera</code></td>
</tr>
</tbody>
</table>

**Field Value**  
Type: CinemachineVirtualCameraBase

### See Also

**Reference**  
CinemachineBlendListCameraInstruction Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrain Class

CinemachineBrain is the link between the Unity Camera and the Cinemachine Virtual Cameras in the scene. It monitors the priority stack to choose the current Virtual Camera, and blend with another if necessary. Finally and most importantly, it applies the Virtual Camera state to the attached Unity Camera. The CinemachineBrain is also the place where rules for blending between virtual cameras are defined. Camera blending is an interpolation over time of one virtual camera position and state to another. If you think of virtual cameras as cameramen, then blending is a little like one cameraman smoothly passing the camera to another cameraman. You can specify the time over which to blend, as well as the blend curve shape. Note that a camera cut is just a zero-time blend.

Inheritance Hierarchy

- System
  - SystemObject
    - Object
      - Component
        - Behaviour
          - MonoBehaviour
            - CinemachineBrain

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(0f, DocumentationSortingAttribute.Direction.Descending)]
[DisallowMultipleComponent]
[AddComponentMenu("Cinemachine/CinemachineBrain")]
[SaveDuringPlayAttribute]
public class CinemachineBrain : MonoBehaviour
```
The **CinemachineBrain** type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBrain</td>
<td>Top</td>
</tr>
</tbody>
</table>

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveBlend</td>
<td>Get the current blend in progress. Returns null if none.</td>
</tr>
<tr>
<td>ActiveVirtualCamera</td>
<td>Get the current active virtual camera.</td>
</tr>
<tr>
<td>CurrentCameraState</td>
<td>The current state applied to the unity camera (may be the result of a blend)</td>
</tr>
<tr>
<td>DefaultWorldUp</td>
<td>Get the default world up for the virtual cameras.</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>OutputCamera</td>
<td>Get the Unity Camera that is attached to this GameObject. This is the camera that will be controlled by the brain.</td>
</tr>
</tbody>
</table>
**SoloCamera**
API for the Unity Editor. Show this camera no matter what. This is static, and so affects all Cinemachine brains.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetSoloGUIColor</td>
<td>API for the Unity Editor.</td>
</tr>
<tr>
<td>IsLive</td>
<td>True if the ICinemachineCamera the current active camera, or part of a current blend, either directly or indirectly because its parents are live.</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_CameraActivatedEvent</td>
<td>This event will fire whenever a virtual camera goes live. If a blend is involved, then the event will fire on the first frame of the blend</td>
</tr>
<tr>
<td>m_CameraCutEvent</td>
<td>This event will fire whenever a virtual camera goes live and there is no blend</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_CustomBlends</td>
<td>This is the asset which contains custom settings for specific blends.</td>
</tr>
<tr>
<td>m_DefaultBlend</td>
<td>The blend which is used if you don't explicitly define a blend between two Virtual Cameras.</td>
</tr>
<tr>
<td>m_IgnoreTimeScale</td>
<td>When enabled, the cameras will always respond in real-time to user input and damping, even if the game is running in slow motion.</td>
</tr>
<tr>
<td>m_ShowCameraFrustum</td>
<td>When enabled, shows the camera's frustum in the scene view.</td>
</tr>
<tr>
<td>m_ShowDebugText</td>
<td>When enabled, the current camera and blend will be indicated in the game window, for debugging.</td>
</tr>
<tr>
<td>m_UpdateMethod</td>
<td>Depending on how the target objects are animated, adjust the update method to minimize the potential jitter. Use FixedUpdate if all your targets are animated with for</td>
</tr>
</tbody>
</table>
RigidBody animation. SmartUpdate will choose the best method for each virtual camera, depending on how the target is animated.

| **m_WorldUpOverride** | If set, this object's Y axis will define the worldspace Up vector for all the virtual cameras. This is useful in top-down game environments. If not set, Up is worldspace Y. |

**Top**

**See Also**

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrain Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineBrain.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public CinemachineBrain()</td>
<td></td>
</tr>
</tbody>
</table>

See Also

Reference

CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrain Properties

The CinemachineBrain type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveBlend</td>
<td>Get the current blend in progress. Returns null if none.</td>
</tr>
<tr>
<td>ActiveVirtualCamera</td>
<td>Get the current active virtual camera.</td>
</tr>
<tr>
<td>CurrentCameraState</td>
<td>The current state applied to the unity camera (may be the result of a blend)</td>
</tr>
<tr>
<td>DefaultWorldUp</td>
<td>Get the default world up for the virtual cameras.</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>OutputCamera</td>
<td>Get the Unity Camera that is attached to this GameObject. This is the camera that will be controlled by the brain.</td>
</tr>
<tr>
<td>SoloCamera</td>
<td>API for the Unity Editor. Show this camera no matter what. This is static, and so affects all Cinemachine brains.</td>
</tr>
</tbody>
</table>
See Also

Reference
CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBrainActiveBlend Property

Get the current blend in progress. Returns null if none.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```  
``` | ```  
``` |

```csharp
public CinemachineBlend ActiveBlend { get; }
```

Property Value  
Type: CinemachineBlend

### See Also

Reference  
CinemachineBrain Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainActiveVirtualCamera Property

Get the current active virtual camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```
public ICinemachineCamera ActiveVirtualCamera {
```

**JavaScript**

```
function ActiveVirtualCamera() {
```

Property Value

Type: ICinemachineCamera

### See Also

**Reference**

CinemachineBrain Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBrainCurrentCameraState

Property

The current state applied to the unity camera (may be the result of a blend)

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

```csharp
public CameraState CurrentCameraState { get; }
```

Property Value

Type: CameraState

⚠️ See Also

Reference

CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainDefaultWorldUp

Property

Get the default world up for the virtual cameras.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
public Vector3 DefaultWorldUp { get; }
```

Property Value  
Type: **Vector3**

See Also

Reference  
CinemachineBrain Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrain.IsBlending Property

Is there a blend in progress?

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public bool IsBlending { get; }
```

Property Value
Type: Boolean

See Also

Reference
CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrain.OutputCamera Property

Get the Unity Camera that is attached to this GameObject. This is the camera that will be controlled by the brain.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```
public Camera OutputCamera { get; }
```

**See Also**

- **Reference**  
- CinemachineBrain Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainSoloCamera Property

API for the Unity Editor. Show this camera no matter what. This is static, and so affects all Cinemachine brains.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#

```csharp
public static ICinemachineCamera SoloCamera { get }
```

JavaScript

```javascript
//
```

Property Value
Type: ICinemachineCamera

See Also

Reference
CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBrain Methods

The CinemachineBrain type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetSoloGUIColor</td>
<td>API for the Unity Editor.</td>
</tr>
<tr>
<td>IsLive</td>
<td>True if the ICinemachineCamera the current active camera, or part of a current blend, either directly or indirectly because its parents are live.</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBrainGetSoloGUIColor Method

API for the Unity Editor.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public static Color GetSoloGUIColor()</code></td>
<td></td>
</tr>
</tbody>
</table>

#### Return Value

**Type:** Color  
Color used to indicate that a camera is in Solo mode.

### See Also

**Reference**  
CinemachineBrain Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBrainIsLive Method

True if the ICinemachineCamera the current active camera, or part of a current blend, either directly or indirectly because its parents are live.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public bool IsLive(
    ICinemachineCamera vcam
)
```

### Parameters

**vcam**

Type: CinemachineICinemachineCamera
The camera to test whether it is live

### Return Value

Type: Boolean
True if the camera is live (directly or indirectly) or part of a blend in progress.

## See Also

**Reference**

CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrain Fields

The **CinemachineBrain** type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_CameraActivatedEvent</td>
<td>This event will fire whenever a virtual camera goes live. If a blend is involved, then the event will fire on the first frame of the blend.</td>
</tr>
<tr>
<td>m_CameraCutEvent</td>
<td>This event will fire whenever a virtual camera goes live and there is no blend.</td>
</tr>
<tr>
<td>m_CustomBlends</td>
<td>This is the asset which contains custom settings for specific blends.</td>
</tr>
<tr>
<td>m_DefaultBlend</td>
<td>The blend which is used if you don't explicitly define a blend between two Virtual Cameras.</td>
</tr>
<tr>
<td>m_IgnoreTimeScale</td>
<td>When enabled, the cameras will always respond in real-time to user input and</td>
</tr>
</tbody>
</table>
damping, even if the game is running in slow motion

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![m_ShowCameraFrustum]</td>
<td>When enabled, shows the camera's frustum in the scene view.</td>
</tr>
<tr>
<td>![m_ShowDebugText]</td>
<td>When enabled, the current camera and blend will be indicated in the game window, for debugging.</td>
</tr>
<tr>
<td>![m_UpdateMethod]</td>
<td>Depending on how the target objects are animated, adjust the update method to minimize the potential jitter. Use FixedUpdate if all your targets are animated with for RigidBody animation. SmartUpdate will choose the best method for each virtual camera, depending on how the target is animated.</td>
</tr>
<tr>
<td>![m_WorldUpOverride]</td>
<td>If set, this object's Y axis will define the worldspace Up vector for all the virtual cameras. This is useful in top-down game environments. If not set, Up is...</td>
</tr>
</tbody>
</table>
See Also

Reference

CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_CameraActivatedEvent Field

This event will fire whenever a virtual camera goes live. If a blend is involved, then the event will fire on the first frame of the blend.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
[TooltipAttribute("This event will fire whenever a virtual camera goes live. If a blend is involved, then the event will fire on the first frame of the blend.")]
public CinemachineBrainVcamEvent m_CameraActivatedEvent;
```

Field Value  
Type: CinemachineBrainVcamEvent

**See Also**

Reference  
CinemachineBrain Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_CameraCutEvent Field

This event will fire whenever a virtual camera goes live and there is no blend

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

| C#            | JavaScript
|---------------|-------------|
| ```
[TooltipAttribute("This event will fire whenever a virtual camera goes live and there is no blend")]
public CinemachineBrainBrainEvent m_CameraCutEvent
``` |

### Field Value

**Type:** CinemachineBrainBrainEvent

### See Also

- Reference
  - CinemachineBrain Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_CustomBlends Field

This is the asset which contains custom settings for specific blends.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public CinemachineBlenderSettings m_CustomBlends
```

Field Value
Type: CinemachineBlenderSettings

See Also

Reference
CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_DefaultBlend Field

The blend which is used if you don't explicitly define a blend between two Virtual Cameras.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[CinemachineBlendDefinitionPropertyAttribute]
[TooltipAttribute("The blend that is used in case")]
public CinemachineBlendDefinition m_DefaultBlend
```

### Field Value

**Type:** CinemachineBlendDefinition

### See Also

- **Reference**  
  CinemachineBrain Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_IgnoreTimeScale Field

When enabled, the cameras will always respond in real-time to user input and damping, even if the game is running in slow motion

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

[TooltipAttribute("When enabled, the cameras will always respond in real-time to user input and damping, even if the game is running in slow motion")]

**Syntax**

```csharp
public bool m_IgnoreTimeScale
```

**Field Value**  
**Type:** Boolean

**See Also**

**Reference**  
- CinemachineBrain Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_ShowCameraFrustum Field

When enabled, shows the camera's frustum in the scene view.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
[TooltipAttribute("When enabled, the camera's frustum will be shown at all times in the scene view")]
public bool m_ShowCameraFrustum
``` |

Field Value  
Type: **Boolean**

### See Also

- **Reference**  
  - CinemachineBrain Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_ShowDebugText Field

When enabled, the current camera and blend will be indicated in the game window, for debugging.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public bool m_ShowDebugText | [TooltipAttribute("When enabled, the current camera and blend will be indicated in the game window, for debugging")]

Field Value

Type: **Boolean**

### See Also

**Reference**

- CinemachineBrain Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineBrainm_UpdateMethod Field

Depending on how the target objects are animated, adjust the update method to minimize the potential jitter. Use FixedUpdate if all your targets are animated with RigidBody animation. SmartUpdate will choose the best method for each virtual camera, depending on how the target is animated.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="C# code" /></td>
<td><img src="image" alt="JavaScript code" /></td>
</tr>
</tbody>
</table>

Field Value

Type: CinemachineBrainUpdateMethod

### See Also

**Reference**

CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainm_WorldUpOverride Field

If set, this object's Y axis will define the worldspace Up vector for all the virtual cameras. This is useful in top-down game environments. If not set, Up is worldspace Y.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("If set, this object's Y axis v public Transform m_WorldUpOverride

Field Value
Type: Transform

See Also

Reference
CinemachineBrain Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainBrainEvent Class

Event with a CinemachineBrain parameter

Inheritance Hierarchy

```
System \< Object \< UnityEventBase \< UnityEvent \< CinemachineBrain \< CinemachineBrainBrainEvent
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[SerializableAttribute]
public class BrainEvent : UnityEvent<CinemachineBrainBrainEvent
```

The CinemachineBrainBrainEvent type exposes the following members.

Constructors

```
CinemachineBrainBrainEvent
```

Methods
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddListener(UnityAction\text{UTP})</td>
<td>(Inherited from \text{UnityEventCinemach}\text{l})</td>
</tr>
<tr>
<td>AddListener(Object, MethodInfo)</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>FindMethod_Impl</td>
<td>(Inherited from \text{UnityEventCinemach}\text{l})</td>
</tr>
<tr>
<td>GetPersistentEventCount</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>GetPersistentMethodName</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>GetPersistentTarget</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>Invoke(Object)</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>Invoke(UTP)</td>
<td>(Inherited from \text{UnityEventCinemach}\text{l})</td>
</tr>
<tr>
<td>RegisterPersistentListener</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>RemoveAllListeners</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>RemoveListener(UnityAction\text{UTP})</td>
<td>(Inherited from \text{UnityEventCinemach}\text{l})</td>
</tr>
<tr>
<td>RemoveListener(Object, MethodInfo)</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>SetPersistentListenerState</td>
<td>(Inherited from \text{UnityEventBase.})</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>ToString</code></td>
<td>(Inherited from <code>UnityEventBase</code>.)</td>
</tr>
<tr>
<td><code>ValidateRegistration(MethodInfo, Object, PersistentListenerMode)</code></td>
<td>(Inherited from <code>UnityEventBase</code>.)</td>
</tr>
<tr>
<td><code>ValidateRegistration(MethodInfo, Object, PersistentListenerMode, Type)</code></td>
<td>(Inherited from <code>UnityEventBase</code>.)</td>
</tr>
</tbody>
</table>

**Explicit Interface Implementations**

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>UnityEngine#ISerializationCallbackReceiver#OnAfterDeserialize</code></td>
</tr>
<tr>
<td><code>UnityEngine#ISerializationCallbackReceiver#OnBeforeSerialize</code></td>
</tr>
</tbody>
</table>

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineBrainBrainEvent Constructor


**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public BrainEvent()</code></td>
<td></td>
</tr>
</tbody>
</table>

### See Also

Reference  
CinemachineBrainBrainEvent Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
BrainEvent Methods

The CinemachineBrainBrainEvent type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="AddListener(UnityActionUTP)" /></td>
<td>(Inherited from UnityEventCinemac1</td>
</tr>
<tr>
<td><img src="image" alt="AddListener(Object, MethodInfo)" /></td>
<td>(Inherited from UnityEventBase.)</td>
</tr>
<tr>
<td><img src="image" alt="FindMethod_Impl" /></td>
<td>(Inherited from UnityEventCinemac1</td>
</tr>
<tr>
<td><img src="image" alt="GetPersistentEventCount" /></td>
<td>(Inherited from UnityEventBase.)</td>
</tr>
<tr>
<td><img src="image" alt="GetPersistentMethodName" /></td>
<td>(Inherited from UnityEventBase.)</td>
</tr>
<tr>
<td><img src="image" alt="GetPersistentTarget" /></td>
<td>(Inherited from UnityEventBase.)</td>
</tr>
<tr>
<td><img src="image" alt="Invoke(Object)" /></td>
<td>(Inherited from UnityEventBase.)</td>
</tr>
<tr>
<td><img src="image" alt="Invoke(UTP)" /></td>
<td>(Inherited from UnityEventCinemac1</td>
</tr>
<tr>
<td><img src="image" alt="RegisterPersistentListener" /></td>
<td>(Inherited from UnityEventBase.)</td>
</tr>
<tr>
<td><img src="image" alt="RemoveAllListeners" /></td>
<td>(Inherited from</td>
</tr>
</tbody>
</table>
UnityEventBase.

- **RemoveListener(UnityAction)**
  - (Inherited from UnityEventCinematic)

- **RemoveListener(Object, MethodInfo)**
  - (Inherited from UnityEventBase.)

- **SetPersistentListenerState**
  - (Inherited from UnityEventBase.)

- **ToString**
  - (Inherited from UnityEventBase.)

- **ValidateRegistration(MethodInfo, Object, PersistentListenerMode)**
  - (Inherited from UnityEventBase.)

- **ValidateRegistration(MethodInfo, Object, PersistentListenerMode, Type)**
  - (Inherited from UnityEventBase.)

### Explicit Interface Implementations

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnityEngine#ISerializationCallbackReceiver#OnAfterDeserialize</td>
</tr>
<tr>
<td>UnityEngine#ISerializationCallbackReceiver#OnBeforeSerialize</td>
</tr>
</tbody>
</table>

### See Also

**Reference**

CinemachineBrainBrainEvent Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainUpdateMethod Enumeration

This enum defines the options available for the update method.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[DocumentationSortingAttribute(0.1f, DocumentationSortingAttribute)
public enum UpdateMethod

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FixedUpdate</td>
<td>0</td>
<td>Virtual cameras are updated in sync with the Physics module, in FixedUpdate</td>
</tr>
<tr>
<td>LateUpdate</td>
<td>1</td>
<td>Virtual cameras are updated in MonoBehaviour LateUpdate.</td>
</tr>
<tr>
<td>SmartUpdate</td>
<td>2</td>
<td>Virtual cameras are updated according to how the target is</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainVcamEvent Class

Event with a ICinemachineCamera parameter

Inheritance Hierarchy

```
System
  Object
  UnityEventBase
    UnityEvent
      ICinemachineCamera
      Cinemachine
        CinemachineBrainVcamEvent
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[SerializableAttribute]
public class VcamEvent : UnityEvent<ICinemachineCamera>
```

The CinemachineBrainVcamEvent type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineBrainVcamEvent</td>
<td></td>
</tr>
</tbody>
</table>

Methods
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddListener(UnityActionUTP)</td>
<td>(Inherited from UnityEventICinemac)</td>
</tr>
<tr>
<td>AddListener(Object, MethodInfo)</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>FindMethod_Impl</td>
<td>(Inherited from UnityEventICinemac)</td>
</tr>
<tr>
<td>GetPersistentEventCount</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>GetPersistentMethodName</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>GetPersistentTarget</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>Invoke(Object)</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>Invoke(UTP)</td>
<td>(Inherited from UnityEventICinemac)</td>
</tr>
<tr>
<td>RegisterPersistentListener</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>RemoveAllListeners</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>RemoveListener(UnityActionUTP)</td>
<td>(Inherited from UnityEventICinemac)</td>
</tr>
<tr>
<td>RemoveListener(Object, MethodInfo)</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>SetPersistentListenerState</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>ToString</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>ValidateRegistration(MethodInfo, Object, PersistentListenerMode)</td>
<td>(Inherited from Unity)</td>
</tr>
<tr>
<td>ValidateRegistration(MethodInfo, Object, PersistentListenerMode,</td>
<td>(Inherited from Unity)</td>
</tr>
</tbody>
</table>
Explicit Interface Implementations

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnityEngine#ISerializationCallbackReceiver#OnAfterDeserialize</td>
</tr>
<tr>
<td>UnityEngine#ISerializationCallbackReceiver#OnBeforeSerialize</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineBrainVcamEvent Constructor


Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#           JavaScript
public VcamEvent()                       Copy

See Also

Reference
CinemachineBrainVcamEvent Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
The `CinemachineBrain.VcamEvent` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddListener(UnityAction&lt;UTP&gt;)</td>
<td>(Inherited from <code>UnityEvent.ICinemachineCamera</code>)</td>
</tr>
<tr>
<td>AddListener(Object, MethodInfo)</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>FindMethod_Impl</td>
<td>(Inherited from <code>UnityEvent.ICinemachineCamera</code>)</td>
</tr>
<tr>
<td>GetPersistentEventCount</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>GetPersistentMethodName</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>GetPersistentTarget</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>Invoke(Object)</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>Invoke(UTP)</td>
<td>(Inherited from <code>UnityEventICinemac</code>)</td>
</tr>
<tr>
<td>RegisterPersistentListener</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>RemoveAllListeners</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
<tr>
<td>RemoveListener(UnityAction&lt;UTP&gt;)</td>
<td>(Inherited from <code>UnityEventICinemac</code>)</td>
</tr>
<tr>
<td>RemoveListener(Object, MethodInfo)</td>
<td>(Inherited from <code>UnityEventBase</code>)</td>
</tr>
</tbody>
</table>
SetPersistentListenerState (Inherited from Unity)

ToString (Inherited from Unity)

ValidateRegistration(MethodInfo, Object, PersistentListenerMode) (Inherited from Unity)

ValidateRegistration(MethodInfo, Object, PersistentListenerMode, Type) (Inherited from Unity)

Top

明显的接口实现

<table>
<thead>
<tr>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnityEngine#ISerializationCallbackReceiver#OnAfterDeserialize</td>
</tr>
<tr>
<td>UnityEngine#ISerializationCallbackReceiver#OnBeforeSerialize</td>
</tr>
</tbody>
</table>

See Also

Reference

CinemachineBrainVcamEvent Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShot Class

Cinemachine ClearShot is a "manager camera" that owns and manages a set of Virtual Camera gameObject children. When Live, the ClearShot will check the children, and choose the one with the best quality shot and make it Live. This can be a very powerful tool. If the child cameras have CinemachineCollider extensions, they will analyze the scene for target obstructions, optimal target distance, and other items, and report their assessment of shot quality back to the ClearShot parent, who will then choose the best one. You can use this to set up complex multi-camera coverage of a scene, and be assured that a clear shot of the target will always be available. If multiple child cameras have the same shot quality, the one with the highest priority will be chosen. You can also define custom blends between the ClearShot children.

Inheritance Hierarchy

- SystemObject
- Object
- Component
- Behaviour
- MonoBehaviour
  - CinemachineCinemachineVirtualCameraBase
  - CinemachineCinemachineClearShot

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(12f, DocumentationAttribute) ]
[ExecuteInEditMode]
[DisallowMultipleComponent]
[AddComponentMenu("Cinemachine/CinemachineClearShot")]`
public class CinemachineClearShot : CinemachineVirtualCameraBase

The CinemachineClearShot type exposes the following members.

## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineClearShot</td>
<td>Top</td>
</tr>
</tbody>
</table>

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>The list of child cameras. These are immediate children in the hierarchy.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this camera, for use when displaying debug info. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the current Follow target. If Follow is active, the parent's Follow is also active if parent is non-null and has no specific Follow defined for this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>LiveChild</td>
<td>Get the current &quot;best&quot; child virtual camera. This would be chosen if the ClearShot camera were active.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>LookAt</strong></td>
<td>Get the current LookAt target. LookAt if parent is non-null and no specific LookAt defined for this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Get the name of the Virtual Camera. The implementation returns the owner GameObject's name. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ParentCamera</strong></td>
<td>Support for meta-virtual-camera situation where a virtual camera is the public face of a private army of which it manages on its own. The VirtualCamera owner, if any, are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>PreviousStateIsValid</strong></td>
<td>Set this to force the next update deltaTime and reset itself. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>Get the Priority of the virtual camera. This determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>The CameraState of the currently live child. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ValidatingStreamVersion</strong></td>
<td>Version that was last streamed. Legacy. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
VirtualCameraGameObject  The GameObject owner of the behaviour. (Inherited from CinemachineVi

Top

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVi</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy. (Inherited from CinemachineVi</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcamera is the most recent Live camera, if it shares the highest priority with its peers. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, the most recent one goes to the top of the priority subqueue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Makes sure the internal child cache is up-to-date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransformChildrenChanged</td>
<td>Makes sure the internal child cache is up-to-date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Method Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields when changed in inspector. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the child cameras being updated on FixedUpdate. If you are updating on LateUpdate instead, Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ResetRandomization</td>
<td>If RandomizeChoice is enabled, call this to re-randomize the children next frame.</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcam's Follow target, or if that is null, will return</td>
</tr>
<tr>
<td>ResolveLookAt</td>
<td>Returns this vcam's LookAt target, or if that is null, will return</td>
</tr>
<tr>
<td>Start</td>
<td>Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Update</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>UpdateCameraState</td>
<td>Called by CinemachineCore at designated update time. Updates all the children, chooses the best one, and implements any required blending. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

**Top**

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is there to support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_ActivateAfter</td>
<td>Wait this many seconds before activating a new child camera.</td>
</tr>
<tr>
<td>m_ChildCameras</td>
<td>Internal API for the editor to use. Use this filed.</td>
</tr>
<tr>
<td>m_CustomBlends</td>
<td>This is the asset which contains custom settings for specific blends.</td>
</tr>
<tr>
<td>m_DefaultBlend</td>
<td>The blend which is used if you don't explicitly define a blend between two Virtual Cameras.</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use for hiding sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_Follow</td>
<td>Default object for the children to move with (the body target), if not specified in a child camera. May be empty.</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Use for enabling sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_LookAt</td>
<td>Default object for the children to look at (the aim point).</td>
</tr>
<tr>
<td></td>
<td>m_MinDuration</td>
</tr>
<tr>
<td>---</td>
<td>---------------</td>
</tr>
<tr>
<td></td>
<td><strong>m_MinDuration</strong></td>
</tr>
<tr>
<td></td>
<td>An active camera must be active for at least this many seconds, unless a higher-priority camera wants to activate.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>m_Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>m_Priority</strong></td>
</tr>
<tr>
<td></td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>m_RandomizeChoice</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>m_RandomizeChoice</strong></td>
</tr>
<tr>
<td></td>
<td>If checked, camera choice will be randomized if multiple cameras are equally desirable. Child list order will be used.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>m_ShowDebugText</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>m_ShowDebugText</strong></td>
</tr>
<tr>
<td></td>
<td>When enabled, the current camera and blend will be indicated in the game window for debugging.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>OnPostPipelineStage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>OnPostPipelineStage</strong></td>
</tr>
<tr>
<td></td>
<td>A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

Top
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShot Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineClearShot.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp javascript
public CinemachineClearShot()
```

See Also

Reference
- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The **CinemachineClearShot** type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>The list of child cameras. These immediate children in the hierarchy.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description camera, for use when displaying (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the current Follow target. If parent is non-null and no specific defined for this camera (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>LiveChild</td>
<td>Get the current &quot;best&quot; child virtual camera, that would be chosen if the ClearShot camera were active.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the current LookAt target. LookAt if parent is non-null and</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>LookAt defined for this camera (Overrides CinemachineVirtualCameraBase)</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. The implementation returns the owner GameObject's name. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-camera situations where a virtual camera is the public face of a private army of which it manages on its own. This method gets the VirtualCamera owner, if any. Are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to deltaTime and reset itself (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera, which determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>State</td>
<td>The CameraState of the currently live child (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed, for upgrading legacy (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
See Also

Reference

CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotChildCamera Property

The list of child cameras. These are just the immediate children in the hierarchy.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public CinemachineVirtualCameraBase[] ChildCameras</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: CinemachineVirtualCameraBase

**See Also**

Reference
- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotDescription Property

Gets a brief debug description of this virtual camera, for use when displaying debug info

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public override string Description { get; }
```

**JavaScript**

See Also

- Reference
  - CinemachineClearShot Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotFollow Property

Get the current Follow target. Returns parent's Follow if parent is non-null and no specific Follow defined for this camera

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>public override Transform Follow { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

#### Property Value

Type: **Transform**

Implements

ICinemachineCameraFollow

### See Also

**Reference**

CinemachineClearShot Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineClearShot.IsBlending Property

Is there a blend in progress?

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public bool IsBlending { get; }
```

Property Value

Type: Boolean

**See Also**

Reference
- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotLiveChild

Property

Get the current "best" child virtual camera, that would be chosen if the ClearShot camera were active.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
public ICinemachineCamera LiveChild { get; set; }
```

Property Value

Type: ICinemachineCamera

**See Also**

Reference

CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotLiveChildOrSelf

Property

Return the live child.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override ICinemachineCamera LiveChildOrSelf</td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: ICinemachineCamera

Implements

ICinemachineCameraLiveChildOrSelf

### See Also

Reference

CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotLookAt Property

Get the current LookAt target. Returns parent's LookAt if parent is non-null and no specific LookAt defined for this camera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public override Transform LookAt { get; set; }
```

Property Value
Type: Transform
Implements ICinemachineCameraLookAt

See Also

Reference
- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotState Property

The CameraState of the currently live child

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>c# public override CameraState State { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: CameraState  
Implements ICinemachineCameraState

**See Also**

Reference  
CinemachineClearShot Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
## CinemachineClearShot Methods

The **CinemachineClearShot** type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddPostPipelineStageHook</strong></td>
<td>A delegate to hook into pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>InvokePostPipelineStageCallback</strong></td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>IsLiveChild</strong></td>
<td>Check whether the vcam is a live child of this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>MoveToTopOfPrioritySubqueue</strong></td>
<td>When multiple virtual cameras have the highest priority, there is a Live camera if it shares the highest priority in the queue with its peers and the most recent one goes to the top of the priority subqueue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnDestroy</strong></td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnDisable</strong></td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnEnable</strong></td>
<td>Makes sure the internal child cache is up-to-date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnTransformChildrenChanged</strong></td>
<td>Makes sure the internal child cache is up-to-date.</td>
</tr>
</tbody>
</table>
| **OnTransformParentChanged**                             | Base class implementation makes sure the priority queue remains up-to-date.
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live.</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields when changed in inspector.</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of child cameras.</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook callback.</td>
</tr>
<tr>
<td>ResetRandomization</td>
<td>If RandomizeChoice is enabled, call this to re-randomize the children.</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcam's Follow target, or if null, will return.</td>
</tr>
<tr>
<td>ResolveLookAt</td>
<td>Returns this vcam's LookAt target, or if null, will return.</td>
</tr>
<tr>
<td>Start</td>
<td>Base class implementation does nothing.</td>
</tr>
<tr>
<td>Update</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>UpdateCameraState</td>
<td>Called by CinemachineCore, updates all the children, and selects the best.</td>
</tr>
</tbody>
</table>

See Also
Reference

CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShot.IsLiveChild Method

Check whether the vcam a live child of this camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override bool IsLiveChild(I CIMachineCamera vcam)
```

### Parameters

**vcam**  
Type: Cinemachine.I CIMachineCamera  
The Virtual Camera to check

### Return Value

Type: Boolean  
True if the vcam is currently actively influencing the state of this vcam

Implementes

ICinemachineCamera.IsLiveChild(I CIMachineCamera)

### See Also

Reference  
CinemachineClearShot Class  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineClearShotOnEnable Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**
```
protected override void OnEnable()
```

**JavaScript**

### See Also

Reference  
CinemachineClearShot Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotOnTransformChildrenChanged Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public void OnTransformChildrenChanged()
```

**JavaScript**

### See Also

**Reference**

- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineClearShotOnTransition Method

Notification that this virtual camera is going live. This implementation resets the child randomization.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public override void OnTransitionFromCamera(I CinemachineCamera fromCam)</code></td>
<td></td>
</tr>
</tbody>
</table>

### Parameters

**fromCam**  
Type: `CinemachineICinemachineCamera`  
The camera being deactivated. May be null.

### Implements

`ICinemachineCameraOnTransitionFromCamera(I CinemachineCamera)`

### See Also

**Reference**  
CinemachineClearShot Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotRemovePostPipelineStageHook Method

Remove a Pipeline stage hook callback. Make sure it is removed from all the children.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override void RemovePostPipelineStageHook(CinemachineVirtualCameraBaseOnPostPipelineStageDelegate d)</td>
<td></td>
</tr>
</tbody>
</table>

#### Parameters

- **d**
  - Type: CinemachineVirtualCameraBaseOnPostPipelineStageDelegate  
  - The delegate to remove.

### See Also

- Reference
  - CinemachineClearShot Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotResetRandomization Method

If RandomizeChoice is enabled, call this to re-randomize the children next frame. This is useful if you want to freshen up the shot.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

```csharp
public void ResetRandomization()
```

**See Also**

- Reference
  - CinemachineClearShot Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotUpdateCameraState Method

Called by CinemachineCore at designated update time so the vcam can position itself and track its targets. This implementation updates all the children, chooses the best one, and implements any required blending.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public override void UpdateCameraState(  
| Vector3 worldUp,  
| float deltaTime  
| ) |

Parameters

worldUp
Type: Vector3
Default world Up, set by the CinemachineBrain
deltaTime
Type: System.Single
Delta time for time-based effects (ignore if less than 0)

Implements
ICinemachineCameraUpdateCameraState(Vector3, Single)

See Also
Reference
CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineClearShot Fields

The **CinemachineClearShot** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It support the soon-to-be removed Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_ActivateAfter</td>
<td>Wait this many seconds before activating a new child camera</td>
</tr>
<tr>
<td>m_ChildCameras</td>
<td>Internal API for the editor. Use this filed.</td>
</tr>
<tr>
<td>m_CustomBlends</td>
<td>This is the asset which contains custom settings for specific blends</td>
</tr>
<tr>
<td>m_DefaultBlend</td>
<td>The blend which is used if you don't explicitly define a blend between two Virtual Cameras</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use sections of the Inspector. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_Follow</td>
<td>Default object for the children wants to move</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Use for enabling sections of the UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_LookAt</td>
<td>Default object for the children to look at (the target), if not specified in a child camera. May be empty.</td>
</tr>
<tr>
<td>m_MinDuration</td>
<td>An active camera must be active for at least this many seconds, unless a higher-priority camera wants to activate.</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_RandomizeChoice</td>
<td>If checked, camera choice will be randomized if multiple cameras are equally desirable. Child list order will be used.</td>
</tr>
<tr>
<td>m_ShowDebugText</td>
<td>When enabled, the current camera and blend will be indicated in the game window for debugging.</td>
</tr>
<tr>
<td>OnPostPipelineStage</td>
<td>A delegate to hook into the state calculation pipeline.</td>
</tr>
</tbody>
</table>
Implementation must be sure to call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage (Inherited from CinemachineVirtualCameraBase).
CinemachineClearShot.m_ActivateAfter Field

Wait this many seconds before activating a new child camera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#

```csharp
[TooltipAttribute("Wait this many seconds before activating a new child camera")]
public float m_ActivateAfter
```

JavaScript

```
```

Field Value
Type: Single

See Also

Reference
CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_ChildCam

Field

Internal API for the editor. Do not use this field.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[SerializeField]
[HideInInspector]
[NoSaveDuringPlayAttribute]
public CinemachineVirtualCameraBase[] m_ChildCameras
``` |

**Field Value**  
Type: *CinemachineVirtualCameraBase*

### See Also

**Reference**  
- CinemachineClearShot Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_CustomBlends Field

This is the asset which contains custom settings for specific blends

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
</table>

```csharp
[HideInInspector]
public CinemachineBlenderSettings m_CustomBlends
```

**Field Value**
Type: CinemachineBlenderSettings

### See Also

**Reference**

CinemachineClearShot Class  
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_DefaultBlendField

The blend which is used if you don't explicitly define a blend between two Virtual Cameras

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
[CinemachineBlendDefinitionPropertyAttribute]
[TooltipAttribute("The blend which is used if you
don't explicitly define a blend between
two Virtual Cameras")]
public CinemachineBlendDefinition m_DefaultBlend;
```

**Field Value**

Type: CinemachineBlendDefinition

**See Also**

<table>
<thead>
<tr>
<th>Reference</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineClearShot Class</td>
<td></td>
</tr>
<tr>
<td>Cinemachine Namespace</td>
<td></td>
</tr>
</tbody>
</table>

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_Follow Field

Default object for the camera children wants to move with (the body target), if not specified in a child camera. May be empty.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
[TooltipAttribute("Default object for the camera
[NoSaveDuringPlayAttribute]
public Transform m_Follow
```

Field Value  
Type: **Transform**

## See Also

**Reference**  
CinemachineClearShot Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineClearShotm_LookAt Field

Default object for the camera children to look at (the aim target), if not specified in a child camera. May be empty.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Default object for the camera
[NoSaveDuringPlayAttribute]
public Transform m_LookAt
``` |            |

Field Value

Type: Transform

See Also

Reference
CinemachineClearShot Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_MinDuration Field

An active camera must be active for at least this many seconds, unless a higher-priority camera wants to activate

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[TooltipAttribute("An active camera must be active for at least this many seconds, unless a higher-priority camera wants to activate")]
public float m_MinDuration
```

**Field Value**  
**Type:** Single

**See Also**

**Reference**  
- CinemachineClearShot Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_RandomizeChoice Field

If checked, camera choice will be randomized if multiple cameras are equally desirable. Otherwise, child list order will be used.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[TooltipAttribute("If checked, camera choice will be randomized if multiple cameras are equally desirable. Otherwise, child list order will be used.")]
public bool m_RandomizeChoice
```

Field Value

Type: Boolean

**See Also**

Reference

- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineClearShotm_ShowsDebug

Field

When enabled, the current camera and blend will be indicated in the game window, for debugging

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;When enabled, the current child camera and blend will be indicated in the game window, for debugging&quot;)][1][NoSaveDuringPlayAttribute]</td>
<td>public bool m_ShowDebugText</td>
</tr>
</tbody>
</table>

### Field Value

Type: **Boolean**

### See Also

**Reference**
- CinemachineClearShot Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCollider Class

An add-on module for Cinemachine Virtual Camera that post-processes the final position of the virtual camera. Based on the supplied settings, the Collider will attempt to preserve the line of sight with the LookAt target of the virtual camera by moving away from objects that will obstruct the view. Additionally, the Collider can be used to assess the shot quality and report this as a field in the camera State.

Inheritance Hierarchy

System
  Object
  Component
    Behaviour
      MonoBehaviour
        CinemachineCinemachineExtension
        CinemachineCinemachineCollider

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
[DocumentationSortingAttribute(15f, DocumentationAttribute)]
[ExecuteInEditMode]
[AddComponentMenu("")]
[SaveDuringPlayAttribute]
public class CinemachineCollider : CinemachineExtension

The CinemachineCollider type exposes the following members.

Properties
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DebugPaths</td>
<td>Inspector API for debugging collision resolution path</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineExtension.)</td>
</tr>
</tbody>
</table>

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline implementation (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>CameraWasDisplaced</td>
<td>See whether the virtual camera has been moved by the collider</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extra state information only, not runtime! (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed for all the vcams, define a class to hold state information (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>IsTargetObscured</td>
<td>See whether an object is blocking the camera</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Cleanup (Overrides CinemachineExtension)</td>
</tr>
<tr>
<td>PostPipelineStageCallback</td>
<td>Callback to the collision resolution shot evaluation (Overrides CinemachineExtensionPostPipelineStageCallback)</td>
</tr>
</tbody>
</table>
### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AvoidObstacles</td>
<td>When enabled, will attempt to resolve situations where the line of sight to the target is blocked by an obstacle</td>
</tr>
<tr>
<td>m_CameraRadius</td>
<td>Camera will try to maintain this distance from any obstacle. Increase this value if you are seeing inside obstacles due to a large FOV on the camera.</td>
</tr>
<tr>
<td>m_CollideAgainst</td>
<td>The Unity layer mask against which the collider will raycast.</td>
</tr>
<tr>
<td>m_Damping</td>
<td>The gradualness of collision resolution.</td>
</tr>
</tbody>
</table>
Higher numbers will move the camera more gradually away from obstructions.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_DistanceLimit</strong></td>
<td>The raycast distance to test for when checking if the line of sight to this camera's target is clear.</td>
</tr>
<tr>
<td><strong>m_MaximumEffort</strong></td>
<td>Upper limit on how many obstacle hits to process. Higher numbers may impact performance. In most environments, 4 is enough.</td>
</tr>
<tr>
<td><strong>m_MinimumDistanceFromTarget</strong></td>
<td>Obstacles closer to the target than this will not be seen</td>
</tr>
<tr>
<td><strong>m_OptimalTargetDistance</strong></td>
<td>If greater than zero, a higher score will be given to shots when the target is closer to this distance. Set</td>
</tr>
</tbody>
</table>
### m_Strategy

This is the strategy used to determine how the Collider will attempt to preserve sight of the target. You can set this to zero to disable this feature.

<table>
<thead>
<tr>
<th>m_Strategy</th>
<th>The way in which the Collider will attempt to preserve sight of the target.</th>
</tr>
</thead>
</table>

### See Also

**Reference**

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineCollider Properties

The CinemachineCollider type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DebugPaths</td>
<td>Inspector API for debugging collision resolution path</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineExtension.)</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineCollider Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineColliderDebugPaths Property

Inspector API for debugging collision resolution path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public List&lt;List&lt;Vector3&gt;&gt; DebugPaths { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**  
Type: `List<List<Vector3>>`

### See Also

Reference  
- CinemachineCollider Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
The **CinemachineCollider** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Awake</strong></td>
<td>Connect to virtual camera pipeline implementation (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>CameraWasDisplaced</td>
<td>See whether the virtual camera has been moved by the collider</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extra states information for all vcams only, not runtime!       (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed on manager cams and will apply to all the vcams' children, define a class to hold your state information (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>IsTargetObscured</td>
<td>See whether an object is blocking the camera's view of the target</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Cleanup (Overrides CinemachineExtension)</td>
</tr>
<tr>
<td>PostPipelineStageCallback</td>
<td>Callback to the collision resolution (Overrides CinemachineExtensionPostPipelineCallback)</td>
</tr>
</tbody>
</table>

Top
See Also

Reference

CinemachineCollider Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineColliderCameraWasDisplaced Method

See whether the virtual camera has been moved by the collider

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public bool CameraWasDisplaced(
    CinemachineVirtualCameraBase vcam
)
```

Parameters

`vcam`

Type: CinemachineVirtualCameraBase
The virtual camera in question. This might be different from the virtual camera that owns the collider, in the event that the camera has children

Return Value

Type: Boolean
True if the virtual camera has been displaced due to collision or target obstruction

See Also

Reference
CinemachineCollider Class
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineColliderIsTargetObscured Method

See whether an object is blocking the camera's view of the target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public bool IsTargetObscured(
    ICinemachineCamera vcam
)
```

### Parameters

**vcam**  
Type: `CinemachineICinemachineCamera`  
The virtual camera in question. This might be different from the virtual camera that owns the collider, in the event that the camera has children.

### Return Value

Type: `Boolean`  
True if something is blocking the view.

### See Also

Reference  
CinemachineCollider Class  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineCollider OnDestroy Method

Cleanup

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected override void OnDestroy()
```

### See Also

**Reference**
- CinemachineCollider Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineColliderPostPipelineStageCallback Method

Callback to the collision resolution and shot evaluation

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
protected override void PostPipelineStageCallback(
    CinemachineVirtualCameraBase vcam,
    CinemachineCore.Stage stage,
    ref CameraState state,
    float deltaTime)
```

**Parameters**

- **vcam**  
  Type: Cinemachine.CinemachineVirtualCameraBase  
  [Missing <param name="vcam"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C

- **stage**  
  Type: Cinemachine.CinemachineCore.Stage  
  [Missing <param name="stage"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C

- **state**  
  Type: Cinemachine.CameraState  
  [Missing <param name="state"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C

- **deltaTime**  
  Type: System.Single  
  [Missing <param name="deltaTime"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C
```

**C#**

```csharp
protected override void PostPipelineStageCallback(
    CinemachineVirtualCameraBase vcam,
    CinemachineCore.Stage stage,
    ref CameraState state,
    float deltaTime)
```

**JavaScript**

```javascript
protected override function PostPipelineStageCallback(
    vcam: CinemachineVirtualCameraBase,
    stage: CinemachineCore.Stage,
    state: CameraState,
    deltaTime: float)
```

**Parameters**

- **vcam**  
  Type: Cinemachine.CinemachineVirtualCameraBase  
  [Missing <param name="vcam"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C

- **stage**  
  Type: Cinemachine.CinemachineCore.Stage  
  [Missing <param name="stage"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C

- **state**  
  Type: Cinemachine.CameraState  
  [Missing <param name="state"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C

- **deltaTime**  
  Type: System.Single  
  [Missing <param name="deltaTime"/> documentation for  
  "M:Cinemachine.CinemachineCollider.PostPipelineStageCallback(Cinemachine.C
```
**deltaTime**

Type: **System.Single**


See Also

Reference

- **CinemachineCollider Class**
- **Cinemachine Namespace**

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
## CinemachineCollider Fields

The `CinemachineCollider` type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>m_AvoidObstacles</code></td>
<td>When enabled, will attempt to resolve situations where the line of sight to the target is blocked by an obstacle.</td>
</tr>
<tr>
<td><code>m_CameraRadius</code></td>
<td>Camera will try to maintain this distance from any obstacle. Increase this value if you are seeing inside obstacles due to a large FOV on the camera.</td>
</tr>
<tr>
<td><code>m_CollideAgainst</code></td>
<td>The Unity layer mask against which the collider will raycast.</td>
</tr>
<tr>
<td><code>m_Damping</code></td>
<td>The</td>
</tr>
</tbody>
</table>
Gradualness of collision resolution. Higher numbers will move the camera more gradually away from obstructions.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_DistanceLimit</td>
<td>The raycast distance to test for when checking if the line of sight to this camera's target is clear.</td>
</tr>
<tr>
<td>m_MaximumEffort</td>
<td>Upper limit on how many obstacle hits to process. Higher numbers may impact performance. In most environments, 4 is enough.</td>
</tr>
<tr>
<td>m_MinimumDistanceFromTarget</td>
<td>Obstacles closer to the target than this will not be seen</td>
</tr>
<tr>
<td>m_OptimalTargetDistance</td>
<td>If greater than zero, a higher score will be given to shots</td>
</tr>
</tbody>
</table>
when the target is closer to this distance. Set this to zero to disable this feature.

- **m_Strategy**
  The way in which the Collider will attempt to preserve sight of the target.

---

**See Also**

Reference
- CinemachineCollider Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineColliderm_AvoidObstacles Field

When enabled, will attempt to resolve situations where the line of sight to the target is blocked by an obstacle

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**  
```csharp
[SpaceAttribute]
[TooltipAttribute("When enabled, will attempt to resolve situations where the line of sight to the target is blocked by an obstacle")]
[FormerlySerializedAsAttribute("m_PreserveLineOfSight")]
public bool m_AvoidObstacles
```

**JavaScript**

### Field Value

Type: **Boolean**

### See Also

**Reference**  
- CinemachineCollider Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineColliderm_CameraRad Field

Camera will try to maintain this distance from any obstacle. Increase this value if you are seeing inside obstacles due to a large FOV on the camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[TooltipAttribute("Camera will try to maintain this distance from any obstacle. Try to keep this value small. Increase it if you are seeing inside obstacles due to a large FOV on the camera.")]
public float m_CameraRadius
```

Field Value
Type: Single

See Also

Reference
CinemachineCollider Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCollider.m_CollideAgainst Field

The Unity layer mask against which the collider will raycast.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[fieldAttribute("Obstacle Detection")]
[fieldAttribute("The Unity layer mask against which the collider will raycast")]
public LayerMask m_CollideAgainst
```

**Field Value**  
**Type:** LayerMask

**See Also**

**Reference**  
CinemachineCollider Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineColliderm_Damping

Field

The gradualness of collision resolution. Higher numbers will move the camera more gradually away from obstructions.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public float m_Damping</td>
<td></td>
</tr>
</tbody>
</table>

Field Value

Type: Single

See Also

Reference
CinemachineCollider Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineColliderm_DistanceLimit Field

The raycast distance to test for when checking if the line of sight to this camera's target is clear.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("The maximum raycast distance when checking if the line of sight to this camera's target is clear.
If the setting is 0 or less, the current actual distance to target will be used.")]
[FormerlySerializedAsAttribute("m_LineOfSightFeelerDistance")]
public float m_DistanceLimit |

### Field Value

**Type:** Single

### See Also

**Reference**

- CinemachineCollider Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Field

Upper limit on how many obstacle hits to process. Higher numbers may impact performance. In most environments, 4 is enough.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

**C#**

```csharp
[RangeAttribute(1f, 10f)]
[TooltipAttribute("Upper limit on how many obstacle hits to process. Higher numbers may impact performance. In most environments, 4 is enough.")]
public int m_MaximumEffort
```

**JavaScript**

```javascript
// Field Value
Field Value
Type: Int32

### See Also

**Reference**

CinemachineCollider Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCollider MinimumDistanceFromTarget Field

Obstacles closer to the target than this will not be seen

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
    [TooltipAttribute("Obstacles closer to the target
    public float m_MinimumDistanceFromTarget
``` |

**Field Value**

Type: Single

### See Also

**Reference**

CinemachineCollider Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCollider_OptimalTargetField

If greater than zero, a higher score will be given to shots when the target is closer to this distance. Set this to zero to disable this feature.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
|                         | [HeaderAttribute("Shot Evaluation")]
|                         | [TooltipAttribute("If greater than zero, a higher |
| public float m_OptimalTargetDistance | score will be given to shots when the target is |
|                         | closer to this distance.
|                         | Set this to zero to disable this feature.

Field Value

Type: Single

### See Also

**Reference**

- CinemachineCollider Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineColliderm_Strategy Field

The way in which the Collider will attempt to preserve sight of the target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("The way in which the Collider
public CinemachineColliderResolutionStrategy m_Strat
```|  |

Field Value  
Type: CinemachineColliderResolutionStrategy

### See Also

**Reference**  
CinemachineCollider Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineColliderResolutionStrategy Enumeration

The way in which the Collider will attempt to preserve sight of the target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public enum ResolutionStrategy
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PullCameraForward</td>
<td>0</td>
<td>Camera will be pulled forward along its Z axis until it is in front of the nearest obstacle</td>
</tr>
<tr>
<td>PreserveCameraHeight</td>
<td>1</td>
<td>In addition to pulling the camera forward, an effort will be made to</td>
</tr>
</tbody>
</table>
return the camera to its original height

PreserveCameraDistance  2

In addition to pulling the camera forward, an effort will be made to return the camera to its original distance from the target

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBase Class

An abstract representation of a mutator acting on a Cinemachine Virtual Camera

Inheritance Hierarchy

```
SystemObject     Object
    Component
        Behaviour
            MonoBehaviour
                CinemachineCinemachineComponentBase
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[DocumentationSortingAttribute(24f, DocumentationSortingAttribute)
public abstract class CinemachineComponentBase :
```

The CinemachineComponentBase type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineComponentBase</td>
<td></td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target.</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>Returns true if this object is enabled and set up to produce results.</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target.</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>What part of the pipeline this fits into</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam's CameraState.</td>
</tr>
<tr>
<td><strong>VirtualCamera</strong></td>
<td>Get the associated CinemachineVirtualCameraBase</td>
</tr>
</tbody>
</table>

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MutateCameraState</strong></td>
<td>Mutates the camera state. This state will later be applied to the camera.</td>
</tr>
<tr>
<td><strong>OnPositionDragged</strong></td>
<td>API for the editor, to process a position drag from the user. Base class implementation does</td>
</tr>
</tbody>
</table>
nothing.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epsilon</td>
<td>Useful constant for very small floats</td>
</tr>
</tbody>
</table>

## See Also

Reference

Cinemachine Namespace

## Inheritance Hierarchy

SystemObject

Object

Component

Behaviour

MonoBehaviour

Cinemachine

CinemachineComponentBase

CinemachineBasicMultiChannelPerlin

CinemachineComposer

CinemachineFramingTransposer

CinemachineHardLockToTarget

CinemachineHardLookAt

CinemachinePOV

CinemachineTrackedDolly

CinemachineTransposer

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineComponentBase Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineComponentBase.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

➤ Syntax

```
protected CinemachineComponentBase()
```

➤ See Also

Reference
CinemachineComponentBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineComponentBase Properties

The CinemachineComponentBase type exposes the following members.

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target.</td>
</tr>
<tr>
<td>IsValid</td>
<td>Returns true if this object is enabled and set up to produce results.</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target.</td>
</tr>
<tr>
<td>Stage</td>
<td>What part of the pipeline this fits into</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState.</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineComponentBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBaseFollowTarget Property

Returns the owner vcam's Follow target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public Transform FollowTarget { get; }
```

**Property Value**  
Type: Transform

**See Also**

Reference  
CinemachineComponentBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineComponentBaseIsValid Property

Returns true if this object is enabled and set up to produce results.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public abstract bool IsValid { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Boolean

See Also

Reference
CinemachineComponentBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBase.LookAtTarget Property

Returns the owner vcam's LookAt target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public Transform LookAtTarget { get; }
```

### Property Value

**Type:** Transform

### See Also

Reference  
CinemachineComponentBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGetComponentBaseStage Property

What part of the pipeline this fits into

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public abstract CinemachineCoreStage Stage { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: `CinemachineCoreStage`

## See Also

**Reference**

- CinemachineGetComponentBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBase.VcamState Property

Returns the owner vcam's CameraState.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**(2.0.0.0)**

**Syntax**

```csharp
public CameraState VcamState { get; }
```

**Property Value**  
Type: CameraState

**See Also**

**Reference**  
CinemachineComponentBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComponentBaseVirtualCamera Property

Get the associated CinemachineVirtualCameraBase

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public CinemachineVirtualCameraBase VirtualCamera</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**  
Type:  CinemachineVirtualCameraBase

### See Also

**Reference**  
CinemachineComponentBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComponentBase

Methods

The `CinemachineComponentBase` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Mutates the camera state. This state will later be applied to the camera.</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. Base class implementation does nothing.</td>
</tr>
</tbody>
</table>

See Also

Reference
- `CinemachineComponentBase Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBase MutateCameraState Method

Mutates the camera state. This state will later be applied to the camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

### C#

```csharp
public abstract void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)
```

### JavaScript

```javascript
//
```

## Parameters

- **curState**
  - Type: CinemachineCameraState
  - Input state that must be mutated

- **deltaTime**
  - Type: System.Single
  - Delta time for time-based effects (ignore if less than 0)

## See Also

- Reference  
  CinemachineComponentBase Class  
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBaseOnPositionDragged Method

API for the editor, to process a position drag from the user. Base class implementation does nothing.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public virtual void OnPositionDragged(
    Vector3 delta
)
```

**JavaScript**

```javascript
// No equivalent in JavaScript
```

### Parameters

- **delta**
  - Type: `Vector3`
  - The amount dragged this frame

### See Also

**Reference**

- CinemachineComponentBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComponentBase

Fields

The CinemachineComponentBase type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epsilon</td>
<td>Useful constant for very small floats</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineComponentBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComponentBaseEpsilon Field

Useful constant for very small floats

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected const float Epsilon = 0.0001f</code></td>
<td></td>
</tr>
</tbody>
</table>

### Field Value

Type: Single

### See Also

Reference
- [CinemachineComponentBase Class](#)
- [Cinemachine Namespace](#)

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposer Class

This is a CinemachineComponent in the Aim section of the component pipeline. Its job is to aim the camera at the vcam's LookAt target object, with configurable offsets, damping, and composition rules. The composer does not change the camera's position. It will only pan and tilt the camera where it is, in order to get the desired framing. To move the camera, you have to use the virtual camera's Body section.

Inheritance Hierarchy

- **SystemObject**
  - **Object**
    - **Component**
      - **Behaviour**
        - **MonoBehaviour**
          - **CinemachineCinemachineComponentBase**
            - **CinemachineCinemachineComposer**
              - **CinemachineCinemachineGroupComposer**

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[DocumentationSortingAttribute(3f, Documentation$]
[ExecuteInEditMode]
[AddComponentMenu("")]
[RequireComponent(typeof(CinemachinePipeline))]
[SaveDuringPlayAttribute]
public class CinemachineComposer : CinemachineCon
```

The **CinemachineComposer** type exposes the following members.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>HardGuideRect</td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a LookAt defined (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>SoftGuideRect</td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage (Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTrackedPoint</td>
<td>Apply the target offsets to the target location.</td>
</tr>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly. (Overrides CinemachineComponentBase::MutateCameraState(CameraState Single)).</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. (Inherited from CinemachineComponentBase::OnPositionDragged).</td>
</tr>
</tbody>
</table>

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BiasX</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_BiasY</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_DeadZoneHeight</td>
<td>Camera will not rotate vertically if the target is within this range of the position</td>
</tr>
<tr>
<td>m_DeadZoneWidth</td>
<td>Camera will not rotate</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_HorizontalDamping</td>
<td>How aggressively the camera tries to follow the target in the screen-horizontal direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.</td>
</tr>
<tr>
<td>m_LookaheadSmoothing</td>
<td>Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag.</td>
</tr>
<tr>
<td>m_LookaheadTime</td>
<td>This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the</td>
</tr>
</tbody>
</table>
future. Note that this setting is sensitive to noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_ScreenX</td>
<td>Horizontal screen position for target. The camera will rotate to the position the tracked object here</td>
</tr>
<tr>
<td>m_ScreenY</td>
<td>Vertical screen position for target. The camera will rotate to position the tracked object here</td>
</tr>
<tr>
<td>m_SoftZoneHeight</td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed</td>
</tr>
<tr>
<td>m_SoftZoneWidth</td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed</td>
</tr>
<tr>
<td><strong>m_TrackedObjectOffset</strong></td>
<td>Target offset from the object's center in LOCAL space which the Composer tracks. Use this to fine-tune the tracking target position when the desired area is not in the tracked object's center</td>
</tr>
<tr>
<td><strong>m_VerticalDamping</strong></td>
<td>How aggressively the camera tries to follow the target in the screen-vertical direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.</td>
</tr>
<tr>
<td><strong>OnGUICallback</strong></td>
<td>Used by the Inspector Editor to display on-screen guides.</td>
</tr>
</tbody>
</table>

**See Also**

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComposer Properties

The **CinemachineComposer** type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemamachineComponentBase.)</td>
</tr>
<tr>
<td><strong>HardGuideRect</strong></td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a LookAt defined (Overrides CinemaemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemaemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>SoftGuideRect</strong></td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage (Overrides CinemaemachineComponentBaseStage.)</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam's</td>
</tr>
</tbody>
</table>
CameraState.  
(Inherited from CinemachineComponentBase.)

| VirtualCamera | Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.) |

See Also

Reference
- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComposerHardGuideRect

Property

Internal API for the inspector editor

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public Rect HardGuideRect { get; set; }</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: **Rect**

### See Also

Reference

CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposerIsValid Property

True if component is enabled and has a LookAt defined

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

- **C#**
  ```csharp
  public override bool IsValid { get; }
  ```

- **JavaScript**

**Property Value**  
Type: Boolean

### See Also

**Reference**  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposerSoftGuideRect

Property

Internal API for the inspector editor

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▶ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public Rect SoftGuideRect { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: Rect

▶ See Also

Reference

CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposerStage Property

Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
public override CinemachineCoreStage Stage {
    get;
}
```

**Property Value**

Type: `CinemachineCoreStage`

**See Also**

Reference

- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
# CinemachineComposer Methods

The *CinemachineComposer* type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTrackedPoint</td>
<td>Apply the target offsets to the target location.</td>
</tr>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly.</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user.</td>
</tr>
</tbody>
</table>

## See Also

Reference
- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineComposerGetTrackedPoint Method

Apply the target offsets to the target location.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected virtual Vector3 GetTrackedPoint(
    Vector3 lookAt
)
```

### Parameters

**lookAt**  
Type: `Vector3`  
The unoffset LookAt point

### Return Value

Type: `Vector3`  
The LookAt point with the offset applied

### See Also

Reference  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposer MutateCameraState

Method

Applies the composer rules and orients the camera accordingly

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public override void MutateCameraState(  
  ref CinemachineCameraState curState,  
  float deltaTime  
) | |

### Parameters

- **curState**
  - Type: CinemachineCameraState  
  - The current camera state

- **deltaTime**
  - Type: System.Single  
  - Used for calculating damping. If less than zero, then target will snap to the center of the dead zone.

### See Also

- Reference: CinemachineComposer Class, Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineComposer Fields

The CinemachineComposer type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BiasX</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_BiasY</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_DeadZoneHeight</td>
<td>Camera will not rotate vertically if the target is within this range of the position</td>
</tr>
<tr>
<td>m_DeadZoneWidth</td>
<td>Camera will not rotate horizontally if the target is within this range of the position</td>
</tr>
<tr>
<td>m_HorizontalDamping</td>
<td>How aggressively the camera tries to follow the target in the screen-horizontal direction. Small numbers are more responsive, rapidly orienting the camera to</td>
</tr>
</tbody>
</table>
keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.

<table>
<thead>
<tr>
<th>m_LookaheadSmoothing</th>
<th>Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_LookaheadTime</td>
<td>This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly.</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_ScreenX</td>
<td>Horizontal screen position for target. The camera will rotate to the position the tracked object here</td>
</tr>
<tr>
<td>m_ScreenY</td>
<td>Vertical screen position for target, The camera will rotate to position the tracked object here</td>
</tr>
<tr>
<td>m_SoftZoneHeight</td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed</td>
</tr>
<tr>
<td>m_SoftZoneWidth</td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed</td>
</tr>
<tr>
<td>m_TrackedObjectOffset</td>
<td>Target offset from the object's center in LOCAL space which the Composer tracks. Use this to fine-tune the tracking target position when the desired area is not in the tracked object's center</td>
</tr>
</tbody>
</table>
m_VerticalDamping

How aggressively the camera tries to follow the target in the screen-vertical direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.

OnGUICallback

Used by the Inspector Editor to display on-screen guides.

See Also

Reference
- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_BiasX Field

A non-zero bias will move the target position away from the center of the soft zone

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public float m_BiasX
```

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_BiasY Field

A non-zero bias will move the target position away from the center of the soft zone.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public float m_BiasY
```

### Field Value

Type: **Single**

## See Also

**Reference**  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_DeadZoneHeight Field

Camera will not rotate vertically if the target is within this range of the position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
[RangeAttribute(0f, 1f)]
[TooltipAttribute("Camera will not rotate vertically if the target is within this range of the position.")]
public float m_DeadZoneHeight
```

Field Value  
Type: Single

### See Also

- Reference  
  - CinemachineComposer Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposer

Field

Camera will not rotate horizontally if the target is within this range of the position.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```
[RangeAttribute(0f, 1f)]
[TooltipAttribute("Camera will not rotate horizontally if the target is within this range of the position.")]
public float m_DeadZoneWidth
```

### Field Value

**Type:** Single

### See Also

**Reference**

- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineComposer.

Field

How aggressively the camera tries to follow the target in the screen-horizontal direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.

_namespace:_ Cinemachine

_Assembly:_ Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[SpaceAttribute] [RangeAttribute(0f, 20f)] [TooltipAttribute(&quot;How aggressively the camera tries to follow the target in the screen-horizontal direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.&quot;)]</td>
<td>public float m_HorizontalDamping</td>
</tr>
</tbody>
</table>

Field Value

_Type:_ Single

See Also

Reference

CinemachineComposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineComposer m_LookaheadSmoothing Field

Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("Controls the smoothness of the
lookahead algorithm.
Larger values smooth out jittery predictions and also increase prediction lag")]
[RangeAttribute(3f, 30f)]
public float m_LookaheadSmoothing
```

Field Value  
Type: **Single**

### See Also

**Reference**  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposerField

This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ **Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public float m_LookaheadTime</code></td>
<td></td>
</tr>
</tbody>
</table>

Field Value

Type: **Single**

⚠️ **See Also**

**Reference**

- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineComposer m_ScreenX Field

Horizontal screen position for target. The camera will rotate to the position the tracked object here.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
```csharp
[SpaceAttribute]
[RangeAttribute(0f, 1f)]
[TooltipAttribute("Horizontal screen position for target. The camera will rotate to position the tracked object here.")]
public float m_ScreenX
```

JavaScript

Field Value

Type: Single

See Also

Reference
- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_ScreenY Field

Vertical screen position for target, The camera will rotate to to position the tracked object here

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[RangeAttribute(0f, 1f)]
[TooltipAttribute("Vertical screen position for target, The camera will rotate to position the tracked object here.")]
public float m_ScreenY
```

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineComposer.m_SoftZoneHeight

Field

When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[RangeAttribute(0f, 2f)]
[TooltipAttribute("When target is within this region, camera will gradually rotate vertically to re-align towards the desired position, depending on the damping speed.")]
public float m_SoftZoneHeight
```

Field Value
Type: Single

See Also

Reference
CinemachineComposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_SoftZoneWidth Field

When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
|[RangeAttribute(0f, 2f)]  
[TooltipAttribute("When target is within this region, camera will gradually rotate horizontally to re-align towards the desired position, depending on the damping speed.")]|

```csharp
public float m_SoftZoneWidth
```

### Field Value

**Type:** Single

### See Also

**Reference**
- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_TrackedObjectOffset Field

Target offset from the object's center in LOCAL space which the Composer tracks. Use this to fine-tune the tracking target position when the desired area is not in the tracked object's center.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

⚠️ Syntax

```csharp
[TooltipAttribute("Target offset from the target
public Vector3 m_TrackedObjectOffset
```

Field Value
Type: **Vector3**

⚠️ See Also

Reference
CinemachineComposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineComposer.m_VerticalDamping Field

How aggressively the camera tries to follow the target in the screen-vertical direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to follow the target in the screen-vertical direction.
Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.")]
public float m_VerticalDamping
```

**JavaScript**

```javascript
/*
  How aggressively the camera tries to follow the target in the screen-vertical direction.
  Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.
*/

public float m_VerticalDamping
```

### Field Value

**Type:** Single

### See Also

**Reference**

- CinemachineComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum

CinemachineComposerOnGUICallback Field

Used by the Inspector Editor to display on-screen guides.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[NoSaveDuringPlayAttribute]</code></td>
<td><code>[HideInInspector]</code></td>
</tr>
<tr>
<td><strong>public</strong> Action OnGUICallback</td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**  
**Type:** Action

**See Also**

**Reference**  
CinemachineComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineConfiner Class

An add-on module for Cinemachine Virtual Camera that post-processes the final position of the virtual camera. It will confine the virtual camera's position to the volume specified in the Bounding Volume field.

Inheritance Hierarchy

```
SystemObject  Object
  Component
   Behaviour
      MonoBehaviour
        CinemachineCinemachineExtension
          CinemachineCinemachineConfiner
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(22f, DocumentationSortingAttribute)
[ExecuteInEditMode]
[AddComponentMenu("" )]
[SaveDuringPlayAttribute]
public class CinemachineConfiner : CinemachineExtension
```

The CinemachineConfiner type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>

Copy

Copy

Copy

Copy
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsValid</td>
<td>Check if the bounding volume is defined</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineExtension.)</td>
</tr>
</tbody>
</table>

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline implementation (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>CameraWasDisplaced</td>
<td>See whether the virtual camera has been moved.</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extra states only, not runtime! (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed for all the vcams children, vcams can define a class to hold your states (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>InvalidatePathCache</td>
<td>Call this if the bounding shape has changed at runtime.</td>
</tr>
</tbody>
</table>
OnDestroy

Disconnect from virtual camera base implementation
(Inherited from CinemachineExtension)

PostPipelineStageCallback

Callback to the camera confining
(Overrides CinemachineExtensionPostPipelineStageCallback(CinemachineVirtualCameraBase, CinemachineCoreStage, CameraState)

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BoundingShape2D</td>
<td>The 2D shape within which the camera is to be contained.</td>
</tr>
<tr>
<td>m_BoundingVolume</td>
<td>The volume within which the camera is to be contained.</td>
</tr>
<tr>
<td>m_ConfineMode</td>
<td>The confiner can operate using a 2D bounding shape or a 3D bounding volume</td>
</tr>
<tr>
<td>m_ConfineScreenEdges</td>
<td>If camera is orthographic, screen edges will be confined to the volume.</td>
</tr>
<tr>
<td>m_Damping</td>
<td>How gradually to return the camera to the bounding volume if it goes beyond the borders</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineConfiner Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineConfiner.#ctor"]

Namespace: Cinemachine  
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

```csharp
public CinemachineConfiner()
```

⚠️ See Also

Reference  
CinemachineConfiner Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
The CinemachineConfiner type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsValid</td>
<td>Check if the bounding volume is defined</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineExtension.)</td>
</tr>
</tbody>
</table>

### See Also

Reference

- CinemachineConfiner Class
- Cinemachine Namespace
CinemachineConfinerIsValid Property

Check if the bounding volume is defined

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public bool IsValid { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**
Type: Boolean

### See Also

Reference  
CinemachineConfiner Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
The *CinemachineConfiner* type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline implementation (Inherited from <em>CinemachineExtension</em>)</td>
</tr>
<tr>
<td>CameraWasDisplaced</td>
<td>See whether the virtual camera has been moved by the confiner</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extra state information for all vcams only, not runtime! (Inherited from <em>CinemachineExtension</em>)</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed on manager cams, define a class to hold your state information for all the vcam children, vcam-specific state information (Inherited from <em>CinemachineExtension</em>)</td>
</tr>
<tr>
<td>InvalidatePathCache</td>
<td>Call this if the bounding shape's points change at runtime</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Disconnect from virtual camera pipeline (Inherited from <em>CinemachineExtension</em>)</td>
</tr>
<tr>
<td>PostPipelineStageCallback</td>
<td>Callback to the camera confining pipeline (Overrides <em>CinemachineExtensionPostPipelineStageCallback(CinemachineVirtualCameraBase, CinemachineCore.Stage, CameraState)</em>, <em>CinemachineExtensionPostPipelineStageCallback(CinemachineCamera, CinemachineCore.Stage, CameraState)</em>)</td>
</tr>
</tbody>
</table>
See Also

Reference
CinemachineConfiner Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineConfinerCameraWasDisplaced Method

See whether the virtual camera has been moved by the confiner

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public bool CameraWasDisplaced(
    CinemachineVirtualCameraBase vcam
)
```

### Parameters

**vcam**  
Type: `CinemachineVirtualCameraBase`  
The virtual camera in question. This might be different from the virtual camera that owns the confiner, in the event that the camera has children

### Return Value

Type: `Boolean`  
True if the virtual camera has been repositioned

### See Also

Reference  
CinemachineConfiner Class  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineConfinerInvalidatePathCache Method

Call this if the bounding shape's points change at runtime

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public void InvalidatePathCache()</code></td>
<td></td>
</tr>
</tbody>
</table>

### See Also

**Reference**  
CinemachineConfiner Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineConfinerPostPipelineStageCallback Method

Callback to the camera confining

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
  protected override void PostPipelineStageCallback(
    CinemachineVirtualCameraBase vcam,
    CinemachineCoreStage stage,
    ref CameraState state,
    float deltaTime
  )
``` | [Copy] |

### Parameters

**vcam**
- **Type:** CinemachineVirtualCameraBase  
  [Missing <param name="vcam"/>] documentation for  

**stage**
- **Type:** CinemachineCoreStage  
  [Missing <param name="stage"/>] documentation for  

**state**
- **Type:** CinemachineCameraState  
  [Missing <param name="state"/>] documentation for  
  "M:Cinemachine.CinemachineConfiner.PostPipelineStageCallback(Cinemachine."
**deltaTime**

Type: System.Single


---

**See Also**

Reference

CinemachineConfiner Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineConfiner Fields

The CinemachineConfiner type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BoundingShape2D</td>
<td>The 2D shape within which the camera is to be contained.</td>
</tr>
<tr>
<td>m_BoundingVolume</td>
<td>The volume within which the camera is to be contained.</td>
</tr>
<tr>
<td>m_ConfineMode</td>
<td>The confiner can operate using a 2D bounding shape or a 3D bounding volume</td>
</tr>
<tr>
<td>m_ConfineScreenEdges</td>
<td>If camera is orthographic, screen edges will be confined to the volume.</td>
</tr>
<tr>
<td>m_Damping</td>
<td>How gradually to return the camera to the bounding volume if it goes beyond the borders</td>
</tr>
</tbody>
</table>

See Also
Reference
CinemachineConfine Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineConfiner m_BoundingShape2D Field

The 2D shape within which the camera is to be contained.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;The 2D shape within which the camera is to be contained&quot;)] public Collider2D m_BoundingShape2D</td>
<td></td>
</tr>
</tbody>
</table>

Field Value

Type: **Collider2D**

**See Also**

Reference

CinemachineConfiner Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineConfiner
Field

The volume within which the camera is to be contained.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public Collider m_BoundingVolume
```

Field Value
Type: Collider

See Also

Reference
CinemachineConfiner Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineConfiner

Field

The confiner can operate using a 2D bounding shape or a 3D bounding volume

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td></td>
</tr>
<tr>
<td>CinemachineConfinerMode m_ConfineMode</td>
<td></td>
</tr>
</tbody>
</table>

Field Value

Type: CinemachineConfinerMode

### See Also

**Reference**

CinemachineConfiner Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineConfiner.m_ConfineScreenEdges

Field

If camera is orthographic, screen edges will be confined to the volume.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("If camera is orthographic, scr
public bool m_ConfineScreenEdges
```

### Field Value

Type: Boolean

### See Also

**Reference**  
CinemachineConfiner Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineConfiner Field

How gradually to return the camera to the bounding volume if it goes beyond the borders

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[TooltipAttribute("How gradually to return the camera to the bounding volume if it goes beyond the borders.
Higher numbers are more gradual.")]
[RangeAttribute(0f, 10f)]
public float m_Damping
```

Field Value
Type: Single

See Also

Reference
CinemachineConfiner Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineConfinerMode Enumeration

The confiner can operate using a 2D bounding shape or a 3D bounding volume

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public enum Mode
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confine2D</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Confine3D</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

### See Also

Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCore Class

A singleton that manages complete lists of CinemachineBrain and, Cinemachine Virtual Cameras, and the priority queue. Provides services to keeping track of whether Cinemachine Virtual Cameras have been updated each frame.

Inheritance Hierarchy

- System
  - Object
  - Cinemachine
    - CinemachineCore

Namespace:  Cinemachine
Assembly:  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public sealed class CinemachineCore
```

The CinemachineCore type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineCore</td>
<td></td>
</tr>
</tbody>
</table>

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BrainCount</td>
<td>Access the array of active</td>
</tr>
</tbody>
</table>
CinemachineBrains in the scene

- **Instance**
  Get the singleton instance

- **VirtualCameraCount**
  List of all active Cinemachine Virtual Cameras for all brains. This list is kept sorted by priority.

---

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FindPotentialTargetBrain</td>
<td>Try to find a CinemachineBrain to associate with a Cinemachine Virtual Camera. The first CinemachineBrain in which this Cinemachine Virtual Camera is live will be used. If none, then the first active CinemachineBrain will be used. Brains with OutputCamera == null will not be returned. Final result may be null.</td>
</tr>
<tr>
<td>GenerateCameraActivationEvent</td>
<td>Signal that the virtual has been activated. If the camera is live, then all CinemachineBrains</td>
</tr>
</tbody>
</table>
that are showing it will send an activation event.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GenerateCameraCutEvent</td>
<td>Signal that the virtual camera's content is discontinuous WRT the previous frame. If the camera is live, then all CinemachineBrains that are showing it will send a cut event.</td>
</tr>
<tr>
<td>GetActiveBrain</td>
<td>Access the array of active CinemachineBrains in the scene without generating garbage</td>
</tr>
<tr>
<td>GetVcamUpdateStatus</td>
<td>Internal use only</td>
</tr>
<tr>
<td>GetVirtualCamera</td>
<td>Access the array of active ICinemachineCamera in the scene without generating garbage</td>
</tr>
<tr>
<td>IsLive</td>
<td>Is this virtual camera currently actively controlling any Camera?</td>
</tr>
</tbody>
</table>

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>

**Top**
GetInputAxis

Delegate for overriding Unity's default input system. If you set this, then your delegate will be called instead of System.Input.GetAxis(axisName) whenever in-game user input is needed.

kStreamingVersion

Data version string. Used to upgrade from legacy projects.

kVersionString

Human-readable Cinemachine Version.

sShowHiddenObjects

If true, show hidden Cinemachine objects, to make manual script mapping possible.

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCore Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineCore.#ctor"]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public CinemachineCore()
```

### See Also

**Reference**
CinemachineCore Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCore Properties

The CinemachineCore type exposes the following members.

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BrainCount</td>
<td>Access the array of active CinemachineBrains in the scene</td>
</tr>
<tr>
<td>Instance</td>
<td>Get the singleton instance</td>
</tr>
<tr>
<td>VirtualCameraCount</td>
<td>List of all active Cinemachine Virtual Cameras for all brains. This list is kept sorted by priority.</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineCore Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreBrainCount Property

Access the array of active CinemachineBrains in the scene

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
<tbody>
<tr>
<td>public int BrainCount { get; }</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: Int32

### See Also

- Reference
  - CinemachineCore Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCoreInstance Property

Get the singleton instance

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public static CinemachineCore Instance { get; }
```

#### Property Value

Type: CinemachineCore

### See Also

**Reference**

- CinemachineCore Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCoreVirtualCameraCount Property

List of all active Cinemachine Virtual Cameras for all brains. This list is kept sorted by priority.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public int VirtualCameraCount { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: **Int32**

### See Also

**Reference**

- CinemachineCore Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
## CinemachineCore Methods

The **CinemachineCore** type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="FindPotentialTargetBrain" /></td>
<td>Try to find a CinemachineBrain to associate with a Cinemachine Virtual Camera. The first CinemachineBrain in which this Cinemachine Virtual Camera is live will be used. If none, then the first active CinemachineBrain will be used. Brains with OutputCamera == null will not be returned. Final result may be null.</td>
</tr>
<tr>
<td><img src="image" alt="GenerateCameraActivationEvent" /></td>
<td>Signal that the virtual has been activated. If the camera is live, then all CinemachineBrains that are showing it will send an activation event.</td>
</tr>
<tr>
<td><img src="image" alt="GenerateCameraCutEvent" /></td>
<td>Signal that the virtual</td>
</tr>
</tbody>
</table>
camera's content is discontinuous WRT the previous frame. If the camera is live, then all CinemachineBrains that are showing it will send a cut event.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetActiveBrain</td>
<td>Access the array of active CinemachineBrains in the scene without generating garbage</td>
</tr>
<tr>
<td>GetVcamUpdateStatus</td>
<td>Internal use only</td>
</tr>
<tr>
<td>GetVirtualCamera</td>
<td>Access the array of active ICinemachineCamera in the scene without generating garbage</td>
</tr>
<tr>
<td>IsLive</td>
<td>Is this virtual camera currently actively controlling any Camera?</td>
</tr>
</tbody>
</table>

See Also

Reference

CinemachineCore Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineCoreFindPotentialTargetBrain Method

Try to find a CinemachineBrain to associate with a Cinemachine Virtual Camera. The first CinemachineBrain in which this Cinemachine Virtual Camera is live will be used. If none, then the first active CinemachineBrain will be used. Brains with OutputCamera == null will not be returned. Final result may be null.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public CinemachineBrain FindPotentialTargetBrain(
    ICinemachineCamera vcam
)
```

**JavaScript**

```javascript
public CinemachineBrain FindPotentialTargetBrain(
    ICinemachineCamera vcam
)
```

### Parameters

**vcam**  
Type: `CinemachinelICinemachineCamera`  
Virtual camera whose potential brain we need.

### Return Value

Type: `CinemachineBrain`  
First CinemachineBrain found that might be appropriate for this vcam, or null

### See Also
Reference

CinemachineCore Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreGenerateCameraActivationEvent Method

Signal that the virtual has been activated. If the camera is live, then all CinemachineBrains that are showing it will send an activation event.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public void GenerateCameraActivationEvent(
    ICinemachineCamera vcam
)
```

Parameters

- `vcam` Type: CinemachineICinemachineCamera

See Also

Reference
CinemachineCore Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCore.GenerateCameraCutEvent Method

Signal that the virtual camera's content is discontinuous WRT the previous frame. If the camera is live, then all CinemachineBrains that are showing it will send a cut event.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public void GenerateCameraCutEvent(I CinemachineICamera vcam)</code></td>
<td></td>
</tr>
</tbody>
</table>

### Parameters

**vcam**

Type: `CinemachineICinemachineCamera`  
[Missing <param name="vcam"/> documentation for "M:Cinemachine.CinemachineCore.GenerateCameraCutEvent(Cinemachine.ICinemachineCamera)"

### See Also

**Reference**

CinemachineCore Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCoreGetActiveBrain Method

Access the array of active CinemachineBrains in the scene without generating garbage

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public CinemachineBrain GetActiveBrain(
    int index
)
```

**JavaScript**

```javascript
public CinemachineBrain GetActiveBrain(
    int index
)
```

### Parameters

**index**

- **Type:** System.Int32  
  Index of the brain to access, range 0-BrainCount

### Return Value

- **Type:** CinemachineBrain  
  The brain at the specified index

### See Also

**Reference**

- CinemachineCore Class  
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreGetVcamUpdateStatus Method

Internal use only

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public CinemachineCoreUpdateFilter GetVcamUpdateStatus(I CIMachineCamera vcam)
```

Parameters

`vcam`
Type: [CinemachineICinemachineCamera](/doc/Cinemachine/ICinemachineCamera)

[Missing <param name="vcam"/> documentation for "M:Cinemachine.CinemachineCore.GetVcamUpdateStatus(Cinemachine.ICinematiceCamera)"

Return Value

Type: [CinemachineCoreUpdateFilter](/doc/Cinemachine/CinemachineCoreUpdateFilter)

[Missing <returns> documentation for "M:Cinemachine.CinemachineCore.GetVcamUpdateStatus(Cinemachine.ICinematiceCamera)"

See Also

Reference
[CinemachineCore Class](/doc/Cinemachine/CinemachineCore)
[Cinemachine Namespace](/doc/Cinemachine/Namespace)
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreGetVirtualCamera Method

Access the array of active ICinemachineCamera in the scene without generating garbage

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public ICinemachineCamera GetVirtualCamera(
    int index
)
```

### Parameters

- **index**
  - Type: **SystemInt32**  
    - Index of the camera to access, range 0-VirtualCameraCount

### Return Value

- Type: **ICinemachineCamera**  
  - The virtual camera at the specified index

### See Also

- **Reference**
  - CinemachineCore Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineCoreIsLive Method

Is this virtual camera currently actively controlling any Camera?

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public bool IsLive(
    ICinemachineCamera vcam
)
```

**JavaScript**

### Parameters

**vcam**

Type: CinemachineICinemachineCamera

[Missing <param name="vcam"/> documentation for "M:Cinemachine.CinemachineCore.IsLive(Cinemachine.ICinemachineCamera)""]

### Return Value

Type: Boolean

[Missing <returns> documentation for "M:Cinemachine.CinemachineCore.IsLive(Cinemachine.ICinemachineCamera)""]

### See Also

Reference

- CinemachineCore Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineCore Fields

The *CinemachineCore* type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GetInputAxis</code></td>
<td>Delegate for overriding Unity's default input system. If you set this, then your delegate will be called instead of <code>System.Input.GetAxis(axisName)</code> whenever in-game user input is needed.</td>
</tr>
<tr>
<td><code>kStreamingVersion</code></td>
<td>Data version string. Used to upgrade from legacy projects</td>
</tr>
<tr>
<td><code>kVersionString</code></td>
<td>Human-readable Cinemachine Version</td>
</tr>
<tr>
<td><code>sShowHiddenObjects</code></td>
<td>If true, show hidden Cinemachine objects, to make manual script mapping possible.</td>
</tr>
</tbody>
</table>

---

Top

See Also

Reference

- CinemachineCore Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreGetInputAxis Field

Delegate for overriding Unity's default input system. If you set this, then your delegate will be called instead of System.Input.GetAxis(axisName) whenever in-game user input is needed.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public static CinemachineCoreAxisInputDelegate GetInputAxis;
```

**JavaScript**

```
CinemachineCore.AxisInputDelegate GetInputAxis;
```

Field Value  
Type: CinemachineCoreAxisInputDelegate

### See Also

Reference  
CinemachineCore Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCorekStreamingVersion Field

Data version string. Used to upgrade from legacy projects

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public static readonly int kStreamingVersion</td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**

Type: Int32

### See Also

**Reference**  
CinemachineCore Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCorekVersionString Field

Human-readable Cinemachine Version

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▶ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public static readonly string kVersionString</code></td>
<td></td>
</tr>
</tbody>
</table>

Field Value  
Type:  **String**

▶ See Also

Reference  
**CinemachineCore Class**  
**Cinemachine Namespace**

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineCoresShowHiddenObjects Field

If true, show hidden Cinemachine objects, to make manual script mapping possible.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public static bool sShowHiddenObjects</td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**

**Type:** Boolean

### See Also

**Reference**
- CinemachineCore Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineCoreAxisInputDelegate Delegate

Delegate for overriding Unity's default input system. Returns the value of the named axis.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public delegate float AxisInputDelegate(</td>
<td></td>
</tr>
<tr>
<td>string axisName</td>
<td></td>
</tr>
<tr>
<td>)</td>
<td></td>
</tr>
</tbody>
</table>

Parameters

axisName
  Type: System.String

Return Value
  Type: Single

See Also

Reference
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreStage Enumeration

Stages in the Cinemachine Component pipeline, used for UI organization. This enum defines the pipeline order.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public enum Stage
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body</td>
<td>0</td>
<td>Second stage: position the camera in space</td>
</tr>
<tr>
<td>Aim</td>
<td>1</td>
<td>Third stage: orient the camera to point at the target</td>
</tr>
<tr>
<td>Noise</td>
<td>2</td>
<td>Final stage: apply noise (this is done separately, in the Correction channel of the CameraState)</td>
</tr>
</tbody>
</table>

### See Also
Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineCoreUpdateFilter
Enumeration

Internal use only

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#       JavaScript

```csharp
public enum UpdateFilter
```

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Late</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Any</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineDollyCart Class

This is a very simple behaviour that constrains its transform to a CinemachinePath. It can be used to animate any objects along a path, or as a Follow target for Cinemachine Virtual Cameras.

Inheritance Hierarchy

- SystemObject
  - Object
    - Component
      - Behaviour
        - MonoBehaviour
          - CinemachineCinemachineDollyCart

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#

```csharp
[DocumentationSortingAttribute(21f, DocumentationAttribute]
[ExecuteInEditMode]
public class CinemachineDollyCart : MonoBehaviour
```

JavaScript

The CinemachineDollyCart type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineDollyCart</td>
<td></td>
</tr>
</tbody>
</table>
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Path</td>
<td>The path to follow</td>
</tr>
<tr>
<td>m_Position</td>
<td>The cart's current position on the path, in distance units</td>
</tr>
<tr>
<td>m_PositionUnits</td>
<td>How to interpret the Path Position</td>
</tr>
<tr>
<td>m_Speed</td>
<td>Move the cart with this speed</td>
</tr>
<tr>
<td>m_UpdateMethod</td>
<td>When to move the cart, if Velocity is non-zero</td>
</tr>
</tbody>
</table>

### See Also

**Reference**

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineDollyCart Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineDollyCart.#ctor"]

Namespace: Cinemachine  
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#  JavaScript  Copy

```csharp
public CinemachineDollyCart()
```

See Also

Reference  
CinemachineDollyCart Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineDollyCart Fields

The CinemachineDollyCart type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Path</td>
<td>The path to follow</td>
</tr>
<tr>
<td>m_Position</td>
<td>The cart's current position on the path, in distance units</td>
</tr>
<tr>
<td>m_PositionUnits</td>
<td>How to interpret the Path Position</td>
</tr>
<tr>
<td>m_Speed</td>
<td>Move the cart with this speed</td>
</tr>
<tr>
<td>m_UpdateMethod</td>
<td>When to move the cart, if Velocity is non-zero</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineDollyCart Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineDollyCartm_Path Field

The path to follow

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ![TooltipAttribute("The path to follow")]]
| public CinemachinePathBase m_Path |

### Field Value

**Type:** CinemachinePathBase

## See Also

- **Reference**
  - CinemachineDollyCart Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
**CinemachineDollyCartm_Position Field**

The cart's current position on the path, in distance units

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[TooltipAttribute("The position along the path at which the cart will be placed. This can be animated directly or, if the value is set to zero, will be updated automatically. The value is interpreted according to the Position Units setting.")]
[FormerlySerializedAsAttribute("m_CurrentDistance")]

public float m_Position
```

**Field Value**  
Type: Single

### See Also

- Reference  
  CinemachineDollyCart Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineDollyCartm_PositionUnits Field

How to interpret the Path Position

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[TooltipAttribute("How to interpret the Path Position Units.
If set to Path Units, values are as follows:
0 represents the beginning of the path.
1 represents a configurable separation between the waypoints.
If set to Distance, then Path Position represents distance along the path.")]
public CinemachinePathBasePositionUnits m_PositionUnits
```

Field Value
Type: CinemachinePathBasePositionUnits

See Also

Reference
CinemachineDollyCart Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineDollyCartm_Speed Field

Move the cart with this speed

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript

```csharp
[TooltipAttribute("Move the cart with this speed
[FormerlySerializedAsAttribute("m_Velocity")]
public float m_Speed
```

Field Value
Type: Single

See Also

Reference
CinemachineDollyCart Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineDollyCartm_UpdateMethod Field

When to move the cart, if Velocity is non-zero

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;When to move the cart, if Velocity is non-zero&quot;)]&lt;br&gt;public CinemachineDollyCartUpdateMethod m_UpdateMethod</td>
<td></td>
</tr>
</tbody>
</table>

Field Value  
Type: CinemachineDollyCartUpdateMethod

### See Also

Reference  
CinemachineDollyCart Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineDollyCartUpdateMethod Enumeration

This enum defines the options available for the update method.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

### Syntax

*C#*  
```csharp
public enum UpdateMethod
```

### Members

<table>
<thead>
<tr>
<th>Member Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update</td>
<td>0</td>
<td>Updated in normal MonoBehaviour Update.</td>
</tr>
<tr>
<td>FixedUpdate</td>
<td>1</td>
<td>Updated in sync with the Physics module, in FixedUpdate</td>
</tr>
</tbody>
</table>

### See Also

- **Reference**  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineExtension Class

Base class for a Cinemachine Virtual Camera extension module. Hooks into the Cinemachine Pipeline.

Inheritance Hierarchy

```
SystemObject  Object
  Component
    Behaviour
      MonoBehaviour
        CinemachineExtension
        CinemachineCollider
        CinemachineConfiner
        CinemachineFollowZoom
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```
[DocumentationSortingAttribute(23f, DocumentationSortingAttribute)
public abstract class CinemachineExtension : MonoBehaviour]
```

The CinemachineExtension type exposes the following members.

Constructors

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineExtension</td>
<td></td>
</tr>
</tbody>
</table>
```
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase</td>
</tr>
</tbody>
</table>

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline. Override implementations must call this base implementation</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Ineffecient method to get all extra state infor for all vcams. Intended for Editor use only, not runtime!</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed on manager cams and will in that case be called for all the vcam children, vcam-specific state information should be stored here. Just define a class to hold your state info and</td>
</tr>
</tbody>
</table>
use it exclusively when calling this.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnDestroy</td>
<td>Disconnect from virtual camera pipeline. Override implementations must call this base implementation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Callback</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PostPipelineStageCallback</td>
<td>This callback will be called after the virtual camera has implemented each stage in the pipeline. This method may modify the referenced state. If deltaTime less than 0, reset all state info and perform no damping.</td>
</tr>
</tbody>
</table>

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epsilon</td>
<td>Useful constant for very small floats</td>
</tr>
</tbody>
</table>

**See Also**

Reference
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtension
Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineExtension.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript Copy

```csharp
protected CinemachineExtension()
```

See Also

Reference

CinemachineExtension Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtension Properties

The `CinemachineExtension` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase</td>
</tr>
</tbody>
</table>

### See Also

Reference
- `CinemachineExtension` Class
- `Cinemachine` Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtensionVirtualCamera

Property

Get the associated CinemachineVirtualCameraBase

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td>CinemachineVirtualCameraBase VirtualCamera:</td>
</tr>
</tbody>
</table>

**Property Value**

Type: CinemachineVirtualCameraBase

### See Also

**Reference**

- CinemachineExtension Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
## CinemachineExtension Methods

The **CinemachineExtension** type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline. Override implementations must call this base implementation</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extra state info for all vcams. Intended for Editor use only, not runtime!</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed on manager cams and will in that case be called for all the vcam children, vcam-specific state information should be stored here. Just define a class to hold your state info and use it exclusively when calling this.</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Disconnect from</td>
</tr>
</tbody>
</table>
virtual camera pipeline. Override implementations must call this base implementation.

PostPipelineStageCallback

This callback will be called after the virtual camera has implemented each stage in the pipeline. This method may modify the referenced state. If deltaTime less than 0, reset all state info and perform no damping.

See Also

Reference

CinemachineExtension Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtensionAwake Method

Connect to virtual camera pipeline. Override implementations must call this base implementation

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected virtual void Awake()
```

### See Also

**Reference**  
CinemachineExtension Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineExtension.GetAllExtraStates Method

Inefficient method to get all extra state info for all vcams. Intended for Editor use only, not runtime!

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
protected List<T> GetAllExtraStates<T>()
where T : class, new()
``` | [Copy](#) |

**Type Parameters**

T

**Return Value**

Type: List<T>

[Missing <returns> documentation for "M:Cinemachine.CinemachineExtension.GetAllExtraStates`1"]

### See Also

**Reference**

CinemachineExtension Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineExtension.GetExtraState Method

Because extensions can be placed on manager cams and will in that case be called for all the vcam children, vcam-specific state information should be stored here. Just define a class to hold your state info and use it exclusively when calling this.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
protected T GetExtraState<T>(
    ICinemachineCamera vcam
)
where T : class, new()
```

Parameters

`vcam`
Type: CinemachineICinemachineCamera

Type Parameters

`T`

Return Value
Type: `T`

See Also

Reference

CinemachineExtension Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtensionOnDestroy Method

Disconnect from virtual camera pipeline. Override implementations must call this base implementation.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
protected virtual void OnDestroy()
```

**JavaScript**


### See Also

Reference

- CinemachineExtension Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineExtensionPostPipelineStageCallback Method

This callback will be called after the virtual camera has implemented each stage in the pipeline. This method may modify the referenced state. If deltaTime less than 0, reset all state info and perform no damping.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
C# | JavaScript
---|---
``` 

```csharp
protected abstract void PostPipelineStageCallback(
    CinemachineVirtualCameraBase vcam,
    CinemachineCore.Stage stage,
    ref CameraState state,
    float deltaTime
)
``` 

Parameters

vcam
Type: Cinemachine.CinemachineVirtualCameraBase

stage
Type: Cinemachine.CinemachineCore.Stage

state
```
Type: CinemachineCameraState

[Missing param name="state"/> documentation for "M:Cinemachine.CinemachineExtension.PostPipelineStageCallback(Cinemachine

deltaTime
Type: System.Single

[Missing param name="deltaTime"/> documentation for "M:Cinemachine.CinemachineExtension.PostPipelineStageCallback(Cinemachine

See Also

Reference
CinemachineExtension Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtension Fields

The `CinemachineExtension` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌐ε</td>
<td><strong>Epsilon</strong> – Useful constant for very small floats</td>
</tr>
</tbody>
</table>

### See Also

**Reference**
- CinemachineExtension Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExtensionEpsilon Field

Useful constant for very small floats

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
C#  JavaScript

protected const float Epsilon = 0.0001f
```

Field Value
Type: **Single**

### See Also

- **Reference**
- CinemachineExtension Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineExternalCamera Class

This component will expose a non-cinemachine camera to the cinemachine system, allowing it to participate in blends. Just add it as a component alongside an existing Unity Camera component.

Inheritance Hierarchy

- System
- Object
- Component
- Behaviour
- MonoBehaviour
  - Cinemachine
  - CinemachineVirtualCameraBase
  - CinemachineCinemachineExternalCamera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(14f, DocumentationSortingAttribute)
[RequireComponent(typeof(Camera))]
[DisallowMultipleComponent]
[ExecuteInEditMode]
[AddComponentMenu("Cinemachine/CinemachineExternalCamera")]
public class CinemachineExternalCamera : Cinemachine
```

The CinemachineExternalCamera type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineExternalCamera</td>
<td></td>
</tr>
</tbody>
</table>

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info.</td>
</tr>
<tr>
<td>Follow</td>
<td>This vcam defines no targets.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Just returns self.</td>
</tr>
<tr>
<td>LookAt</td>
<td>The object that the camera is looking at.</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Base implementation returns the owner GameObject's name.</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-camera in the situation where a virtual camera is used.</td>
</tr>
</tbody>
</table>
fact the public face of a private virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to ignore deltaTime and reset itself.</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. This determines its placement in CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>State</td>
<td>Get the CameraState, as we are about to construct one from the Unity Camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed, for upgrading legacy. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the Camera behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage.</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageCallback for this camera, and up the hierarchy for all parent cameras (if any).</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is a live child of this camera.</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is sometimes the need to push one to the top, making it the current Live camera if it shares the highest priority with its peers. Use this method to push a vcam to the top of its priority peers. If it already is the current Live camera, then this vcam will become the new Live camera.</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation ensures the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Base class implementation adds the virtual camera to the priority queue.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>OnTransformParentChanged</code></td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>OnTransitionFromCamera</code></td>
<td>Notification that this virtual camera is going live. Implementation must be called by any overridden method. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>OnValidate</code></td>
<td>Enforce bounds for fields when changed in inspector. Base class implementation at the beginning of overridden method. After base method is called, ValidatingStreamVersion will be valid. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>PreUpdateChildCameras</code></td>
<td>This is called prior to the cameras, in order to allow the children to update. It does not necessarily be called prior to every FixedUpdate, but it might be called on LateUpdate. Does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>RemovePostPipelineStageHook</code></td>
<td>Remove a Pipeline stage hook. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>ResolveFollow</code></td>
<td>Returns this vcam's Follow target, or if that is null, will return the parent vcam's Follow target. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>ResolveLookAt</code></td>
<td>Returns this vcam's LookAt target, or if that is null, will return the parent vcam's LookAt target. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>Start</code></td>
<td>Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>Update</code></td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
Construct a CameraState (Overrides CinemachineVirtualCamera Single).

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It support the soon-to-be-removed Cinemachine Debugger Editor. (Inherited from CinemachineVirtualCamera)</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Us sections of the Inspector UI. (Inherited from CinemachineVirtualCamera)</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Us enabling sections of the UI. (Inherited from CinemachineVirtualCamera)</td>
</tr>
<tr>
<td>m_LookAt</td>
<td>The object that the camera is looking at.</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCamera)</td>
</tr>
</tbody>
</table>
OnPostPipelineStage

A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage.

(Inherited from CinemachineVirtualCameraBase)

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExternalCamera Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineExternalCamera.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    JavaScript

```csharp
public CinemachineExternalCamera()
```

See Also

Reference
CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExternalCamera Properties

The CinemachineExternalCamera type exposes the following members.

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description virtual camera, for use when debug info (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>This vcam defines no targets (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Just returns self. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LookAt</td>
<td>The object that the camera is looking at (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Base implementation returns the GameObject's name. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-cameras.</td>
</tr>
</tbody>
</table>
the situation where a virtual camera acts as the public face of a private army of virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to ignore deltaTime and reset itself. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. This determines its placement in CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>State</td>
<td>Get the CameraState, as we are about to construct one from the Unity Camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed for upgrading legacy. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the Camera behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
See Also

Reference
CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExternalCameraFollow

Property

This vcam defines no targets

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override Transform Follow { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

- **Type:** Transform
- Implements ICinemachineCameraFollow

**See Also**

- Reference
  - CinemachineExternalCamera Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExternalCameraLookAt

Property

The object that the camera is looking at

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override Transform LookAt { get; set; }
```

**Property Value**

Type: Transform

Implements ICinemachineCameraLookAt

**See Also**

Reference

CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExternalCameraState Property

Get the CameraState, as we are able to construct one from the Unity Camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
public override CameraState State { get; }
``` |

**Property Value**
- **Type:** CameraState
- **Implements:** ICinemachineCameraState

### See Also

**Reference**
- CinemachineExternalCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineExternalCamera

Methods

The **CinemachineExternalCamera** type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Add Post Pipeline Stage Hook" /></td>
<td>A delegate to hook into the pipeline. See CinemachineStage.</td>
</tr>
<tr>
<td><img src="image" alt="Invoke Post Pipeline Stage Callback" /></td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all parent cameras (if any).</td>
</tr>
<tr>
<td><img src="image" alt="Is Live Child" /></td>
<td>Check whether the vcam is a live child of this camera. Class implementation always returns false.</td>
</tr>
<tr>
<td><img src="image" alt="Move To Top Of Priority Subqueue" /></td>
<td>When multiple virtual cameras have the highest priority, there is sometimes the need to move one to the top of the priority queue with its peers. This happens automatically when a vcam is enabled: the most recent one goes to the top of its priority subqueue. Use this method to push a vcam to the top of its priority peers. If it and its peers share the highest priority, then this vcam will become Live.</td>
</tr>
<tr>
<td><img src="image" alt="On Destroy" /></td>
<td>Base class implementation removes the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Base class implementation adds the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live.</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields when changed in inspector.</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of the child cameras, in order to allow the children to update prior to this method being called.</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook.</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcam's Follow target, or if that is null, will return the parent vcam's Follow target.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ResolveLookAt</td>
<td>Returns this vcam's LookAt target, or the parent vcam's LookAt target.</td>
</tr>
<tr>
<td>Start</td>
<td>Base class implementation does nothing.</td>
</tr>
<tr>
<td>Update</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>UpdateCameraState</td>
<td>Construct a CameraState (Overrides CinemachineVirtualCameraBase.Single).</td>
</tr>
</tbody>
</table>

See Also

Reference

CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineExternalCameraUpdateCameraState Method

Construct a CameraState object from the Unity Camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

**C#**

```csharp
public override void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

**JavaScript**

```javascript
//
```

### Parameters

**worldUp**  
Type: Vector3

**deltaTime**  
Type: System.Single

Implements  
ICinemachineCameraUpdateCameraState(Vector3, Single)

### See Also

Reference
CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineExternalCamera

Fields

The `CinemachineExternalCamera` type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CinemachineGUIDebuggerCallback</code></td>
<td>This is deprecated. It support the soon-to-b Cinemachine Debugg Editor. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><code>m_ExcludedPropertiesInInspector</code></td>
<td>Inspector control - Us sections of the Inspector (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><code>m_LockStageInInspector</code></td>
<td>Inspector control - Us enabling sections of the Inspector (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><code>m_LookAt</code></td>
<td>The object that the camera looking at.</td>
</tr>
<tr>
<td><code>m_Priority</code></td>
<td>The priority will determine which camera becomes active based on the state of other cameras. Higher numbers indicate higher priority. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
</tbody>
</table>
OnPostPipelineStage

A delegate to hook into the calculation pipeline. Implementation must be called after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage.

(Inherited from CinemachineVirtualCameraBase)

See Also

Reference
CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineExternalCamera

Field

The object that the camera is looking at.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("The object that the camera is looking at.	Setting this will improve the quality to and from this camera")]
public Transform m_LookAt |

Field Value

Type: Transform

See Also

Reference

CinemachineExternalCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoom Class

An add-on module for Cinemachine Virtual Camera that adjusts the FOV of the lens to keep the target object at a constant size on the screen, regardless of camera and target position.

Inheritance Hierarchy

System
  Object
  Component
    Behaviour
      MonoBehaviour
        Cinemachine
          CinemachineExtension
            CinemachineFollowZoom

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

The CinemachineFollowZoom type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CinemachineFollowZoom

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineExtension.)</td>
</tr>
</tbody>
</table>

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline. Implementation (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extra states only, not runtime! (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed on manager cameras, define a class to hold your state information (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Disconnect from virtual camera pipeline. Base implementation (Inherited from CinemachineExtension.)</td>
</tr>
<tr>
<td>PostPipelineStageCallback</td>
<td>Callback to perform the zoom adjustment (Overrides CinemachineExtensionPostPipelineStageCallback.)</td>
</tr>
</tbody>
</table>
## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Damping</td>
<td>Increase this value to soften the aggressiveness of the follow-zoom. Small numbers are more responsive, larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_MaxFOV</td>
<td>Will not generate an FOV target than this.</td>
</tr>
<tr>
<td>m_MinFOV</td>
<td>Will not generate an FOV smaller than this.</td>
</tr>
<tr>
<td>m_Width</td>
<td>The shot width to maintain, in world units, at target distance. FOV will be adjusted as far as possible to maintain this width at the target distance from the camera.</td>
</tr>
</tbody>
</table>

## See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoom Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineFollowZoom.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```C#
public CinemachineFollowZoom()
```

See Also

Reference
CinemachineFollowZoom Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoom Properties

The CinemachineFollowZoom type exposes the following members.

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineExtension.)</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineFollowZoom Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineFollowZoom

Methods

The `CinemachineFollowZoom` type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Awake</td>
<td>Connect to virtual camera pipeline implementation (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>GetAllExtraStatesT</td>
<td>Inefficient method to get all extensions, not runtime! (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>GetExtraStateT</td>
<td>Because extensions can be placed on all the vcams, define a class to hold your state (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Disconnect from virtual camera pipeline base implementation (Inherited from CinemachineExtension)</td>
</tr>
<tr>
<td>PostPipelineStageCallback</td>
<td>Callback to perform the zoom adjustment (Overrides CinemachineExtensionPostPipelineStageCallback)</td>
</tr>
</tbody>
</table>

See Also
Reference
CinemachineFollowZoom Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoomPostPipelineStageCallback Method

Callback to preform the zoom adjustment

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
protected override void PostPipelineStageCallback(
    CinemachineVirtualCameraBase vcam,
    CinemachineCore.Stage stage,
    ref CameraState state,
    float deltaTime
)
```

### Parameters

- **vcam**  
  Type: Cinemachine.CinemachineVirtualCameraBase  

- **stage**  
  Type: Cinemachine.CinemachineCore.Stage  

- **state**  
  Type: Cinemachine.CameraState  

- **deltaTime**  
  Type: System.Single  

---

*Note: The documentation snippet contains placeholder text.*
**deltaTime**

Type: System.Single


## See Also

**Reference**

- CinemachineFollowZoom Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineFollowZoom Fields

The `CinemachineFollowZoom` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Damping</td>
<td>Increase this value to soften the aggressiveness of the follow-zoom. Small numbers are more responsive, larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_MaxFOV</td>
<td>Will not generate an FOV target than this.</td>
</tr>
<tr>
<td>m_MinFOV</td>
<td>Will not generate an FOV smaller than this.</td>
</tr>
<tr>
<td>m_Width</td>
<td>The shot width to maintain, in world units, at target distance. FOV will be adjusted as far as possible to maintain this width at the target distance from the camera.</td>
</tr>
</tbody>
</table>

### See Also

**Reference**
- [CinemachineFollowZoom Class](#)
- [Cinemachine Namespace](#)
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoomm_Damping Field

Increase this value to soften the aggressiveness of the follow-zoom. Small numbers are more responsive, larger numbers give a more heavy slowly responding camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("Increase this value to soften aggression of the follow-zoom.
Small numbers are more responsive, larger numbers give a more heavy slowly responding camera.")]
public float m_Damping
```

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineFollowZoom Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoomm_MaxFOV

Field

Will not generate an FOV larger than this.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[RangeAttribute(1f, 179f)] [TooltipAttribute(&quot;Upper limit for the FOV that this behaviour will generate.&quot;)] public float m_MaxFOV</td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**

Type: **Single**

### See Also

- Reference
  - CinemachineFollowZoom Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoom\_MinFOV Field

Will not generate an FOV smaller than this.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[RangeAttribute(1f, 179f)]
[TooltipAttribute("Lower limit for the FOV that this behaviour will generate.")]
public float m_MinFOV
```

**JavaScript**

```javascript

```

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineFollowZoom Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFollowZoomm_Width Field

The shot width to maintain, in world units, at target distance. FOV will be adjusted as far as possible to maintain this width at the target distance from the camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

```csharp
public float m_Width
```

Field Value  
Type: Single

⚠️ See Also

Reference  
CinemachineFollowZoom Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer Class

This is a Cinemachine Component in the Body section of the component pipeline. Its job is to position the camera in a fixed screen-space relationship to the vcam's Follow target object, with offsets and damping. The camera will be first moved along the camera Z axis until the Follow target is at the desired distance from the camera's X-Y plane. The camera will then be moved in its XY plane until the Follow target is at the desired point on the camera's screen. The FramingTransposer will only change the camera's position in space. It will not re-orient or otherwise aim the camera. For this component to work properly, the vcam's LookAt target must be null. The Follow target will define what the camera is looking at. If the Follow target is a CinemachineTargetGroup, then additional controls will be available to dynamically adjust the camera's view in order to frame the entire group. Although this component was designed for orthographic cameras, it works equally well with perspective cameras and can be used in 3D environments.

Inheritance Hierarchy

```
SystemObject   Object
   Component
      Behaviour
          MonoBehaviour
              CinemachineCinemachineComponentBase
                  CinemachineCinemachineFramingTransposer
```

Namespace:  Cinemachine
Assembly:  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)
The CinemachineFramingTransposer type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineFramingTransposer</td>
<td></td>
</tr>
</tbody>
</table>

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>HardGuideRect</td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a valid Follow target. (Overrrides CinemachineComponentBaseIsValid)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_LastBounds</td>
<td>For editor visualization of the calculated bounding box of the group.</td>
</tr>
<tr>
<td>m_lastBoundsMatrix</td>
<td>For editor visualization of the calculated bounding box of the group.</td>
</tr>
<tr>
<td>SoftGuideRect</td>
<td>Internal API for the inspector editor.</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage. (Overrides CinemachineComponentBase.Stage.)</td>
</tr>
<tr>
<td>TargetGroup</td>
<td>Get Follow target as CinemachineTargetGroup, or null if target is not a group</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

**Methods**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Positions the virtual camera according to the transposer rules. (Overrides CinemachineComponentBaseMutateCameralStage.)</td>
</tr>
</tbody>
</table>
OnPositionDragged API for the editor, to process a position drag from the user. (Inherited from CinemachineComponentBase)

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AdjustmentMode</td>
<td>How to adjust the camera to get the desired framing</td>
</tr>
<tr>
<td>m_BiasX</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_BiasY</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_CameraDistance</td>
<td>The distance along the camera axis that will be maintained from the Follow target</td>
</tr>
<tr>
<td>m_DeadZoneDepth</td>
<td>The camera will not move along its z-axis if the Follow target is within this distance of the specified camera distance</td>
</tr>
<tr>
<td>m_DeadZoneHeight</td>
<td>Camera will not move</td>
</tr>
<tr>
<td><strong>m_DeadZoneWidth</strong></td>
<td>Camera will not move horizontally if the target is within this range of the position</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_GroupFramingMode</strong></td>
<td>What screen dimensions to consider when framing</td>
</tr>
<tr>
<td><strong>m_GroupFramingSize</strong></td>
<td>How much of the screen to fill with the bounding box of the targets.</td>
</tr>
<tr>
<td><strong>m_LookaheadSmoothing</strong></td>
<td>Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag</td>
</tr>
<tr>
<td><strong>m_LookaheadTime</strong></td>
<td>This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to noisy animation, and can amplify the noise,</td>
</tr>
</tbody>
</table>
resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_MaxDollyIn</td>
<td>How much closer to the target can the camera go?</td>
</tr>
<tr>
<td>m_MaxDollyOut</td>
<td>How much farther from the target can the camera go?</td>
</tr>
<tr>
<td>m_MaximumDistance</td>
<td>Set this to limit how far from the target the camera can get</td>
</tr>
<tr>
<td>m_MaximumFOV</td>
<td>If adjusting FOV, will not set the FOV higher than this</td>
</tr>
<tr>
<td>m_MaximumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it higher than this</td>
</tr>
<tr>
<td>m_MinimumDistance</td>
<td>Set this to limit how close to the target the camera can get</td>
</tr>
<tr>
<td>m_MinimumFOV</td>
<td>If adjusting FOV, will not set the FOV lower than this</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_MinimumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it lower than this.</td>
</tr>
<tr>
<td>m_ScreenX</td>
<td>Horizontal screen position for target. The camera will move to position the tracked object here.</td>
</tr>
<tr>
<td>m_ScreenY</td>
<td>Vertical screen position for target. The camera will move to position the tracked object here.</td>
</tr>
<tr>
<td>m_SoftZoneHeight</td>
<td>When target is within this region, camera will gradually move to realign towards the desired position, depending on the damping speed.</td>
</tr>
<tr>
<td>m_SoftZoneWidth</td>
<td>When target is within this region, camera will gradually move to realign towards the desired position, depending on the damping speed.</td>
</tr>
<tr>
<td>m_UnlimitedSoftZone</td>
<td></td>
</tr>
<tr>
<td>m_XDamping</td>
<td>How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more.</td>
</tr>
</tbody>
</table>
responsive, rapidly translating the camera to keep the target’s x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

| **m_YDamping** | How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors |
| **m_ZDamping** | How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more |
heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

<table>
<thead>
<tr>
<th></th>
<th>OnGUICallback</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Used by the Inspector Editor to display on-screen guides.</td>
</tr>
</tbody>
</table>

Open the Cinemachine Inspector

*See Also*

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineFramingTransposer.#ctor"]

Namespace:  Cinemachine
Assembly:  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public CinemachineFramingTransposer()
```

See Also

Reference
- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Properties

The **CinemachineFramingTransposer** type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td>HardGuideRect</td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a valid Follow target (Overrides <code>CinemachineComponentBaseIsValid</code>)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td>m_LastBounds</td>
<td>For editor visualization of the calculated bounding box of the group</td>
</tr>
<tr>
<td>m_lastBoundsMatrix</td>
<td>For editor visualization of the calculated bounding box of the group</td>
</tr>
<tr>
<td>SoftGuideRect</td>
<td>Internal API for the inspector editor</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage (Overrides CinemachineComponentBaseStage).</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>TargetGroup</strong></td>
<td>Get Follow target as CinemachineTargetGroup, or null if target is not a group</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>VirtualCamera</strong></td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

**See Also**

Reference

CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerHar
Property

Internal API for the inspector editor

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public Rect HardGuideRect { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value

Type: Rect

**See Also**

Reference

CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerIsValid Property

True if component is enabled and has a valid Follow target

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
public override bool IsValid {
    get;
}
``` |

**Property Value**

Type: Boolean

### See Also

**Reference**

CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer.m_LastBounds Property

For editor visualization of the calculated bounding box of the group

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
public Bounds m_LastBounds { get; }
``` |

**Property Value**

Type: **Bounds**

### See Also

**Reference**

CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Property

For editor visualization of the calculated bounding box of the group

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public Matrix4x4 m_lastBoundsMatrix { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: **Matrix4x4**

**See Also**

Reference

CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerSoftGuideRect

Internal API for the inspector editor

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public Rect SoftGuideRect { get; set; }</code></td>
<td></td>
</tr>
</tbody>
</table>

### Property Value

Type: *Rect*

### See Also

- **Reference**
  - CinemachineFramingTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public override CinemachineCoreStage Stage { get; }
```

Property Value
Type: CinemachineCoreStage

See Also

Reference
CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerTargetGroup Property

Get Follow target as CinemachineTargetGroup, or null if target is not a group

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

#### C#

```csharp
public CinemachineTargetGroup TargetGroup {
    get;
}
```

#### JavaScript

### See Also

**Reference**

CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Methods

The CinemachineFramingTransposer type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Positions the virtual camera according to the transposer rules. (Overrides CinemachineComponentBaseMutateCameraState(CameraState Single).)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. The class implementation does nothing. (Inherited from CinemachineComponentBaseOnPositionDragged (UnityEvent).)</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerMutateCameraState Method

Positions the virtual camera according to the transposer rules.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public override void MutateCameraState(</code></td>
<td><code>ref CameraState curState,</code></td>
</tr>
<tr>
<td></td>
<td><code>float deltaTime</code>)</td>
</tr>
</tbody>
</table>

### Parameters

- **curState**
  - Type: CinemachineCameraState  
  - The current camera state

- **deltaTime**
  - Type: System.Single  
  - Used for damping. If less than 0, no damping is done.

### See Also

- Reference  
  - CinemachineFramingTransposer Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
# CinemachineFramingTransposer Fields

The CinemachineFramingTransposer type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AdjustmentMode</td>
<td>How to adjust the camera to get the desired framing</td>
</tr>
<tr>
<td>m_BiasX</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_BiasY</td>
<td>A non-zero bias will move the target position away from the center of the soft zone</td>
</tr>
<tr>
<td>m_CameraDistance</td>
<td>The distance along the camera axis that will be maintained from the Follow target</td>
</tr>
<tr>
<td>m_DeadZoneDepth</td>
<td>The camera will not move along its z-axis if the Follow target is within this distance of the specified camera distance</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_DeadZoneHeight</td>
<td>Camera will not move vertically if the target is within this range of the position</td>
</tr>
<tr>
<td>m_DeadZoneWidth</td>
<td>Camera will not move horizontally if the target is within this range of the position</td>
</tr>
<tr>
<td>m_GroupFramingMode</td>
<td>What screen dimensions to consider when framing</td>
</tr>
<tr>
<td>m_GroupFramingSize</td>
<td>How much of the screen to fill with the bounding box of the targets.</td>
</tr>
<tr>
<td>m_LookaheadSmoothing</td>
<td>Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag</td>
</tr>
<tr>
<td>m_LookaheadTime</td>
<td>This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to</td>
</tr>
</tbody>
</table>
noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_MaxDollyIn</td>
<td>How much closer to the target can the camera go?</td>
</tr>
<tr>
<td>m_MaxDollyOut</td>
<td>How much farther from the target can the camera go?</td>
</tr>
<tr>
<td>m_MaximumDistance</td>
<td>Set this to limit how far from the target the camera can get</td>
</tr>
<tr>
<td>m_MaximumFOV</td>
<td>If adjusting FOV, will not set the FOV higher than this</td>
</tr>
<tr>
<td>m_MaximumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it higher than this</td>
</tr>
<tr>
<td>m_MinimumDistance</td>
<td>Set this to limit how close to the target the camera can get</td>
</tr>
<tr>
<td>m_MinimumFOV</td>
<td>If adjusting FOV, will not set the FOV lower than this</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>m_MinimumOrthoSize</code></td>
<td>If adjusting Orthographic Size, will not set it lower than this</td>
</tr>
<tr>
<td><code>m_ScreenX</code></td>
<td>Horizontal screen position for target. The camera will move to position the tracked object here</td>
</tr>
<tr>
<td><code>m_ScreenY</code></td>
<td>Vertical screen position for target. The camera will move to position the tracked object here</td>
</tr>
<tr>
<td><code>m_SoftZoneHeight</code></td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed</td>
</tr>
<tr>
<td><code>m_SoftZoneWidth</code></td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed</td>
</tr>
<tr>
<td><code>m_UnlimitedSoftZone</code></td>
<td></td>
</tr>
<tr>
<td><code>m_XDamping</code></td>
<td>How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more</td>
</tr>
</tbody>
</table>
responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

| m_YDamping | How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors |
| m_ZDamping | How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more |
heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

| 🎥 OnGUICallback | Used by the Inspector Editor to display on-screen guides. |

See Also

Reference
- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerField

How to adjust the camera to get the desired framing

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("How to adjust the camera to get the desired framing.
You can zoom, dolly in/out, or do both.")]
public CinemachineFramingTransposerAdjustmentMode m_AdjustmentMode
```

### See Also

- Reference
  - CinemachineFramingTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Field

A non-zero bias will move the target position away from the center of the soft zone

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[RangeAttribute(-0.5f, 0.5f)]
[TooltipAttribute("A non-zero bias will move the target position horizontally away from the center of the soft zone.")]
public float m_BiasX
``` |

Field Value
Type: Single

See Also

Reference
CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer

Field

A non-zero bias will move the target position away from the center of the soft zone

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

```csharp
[RangeAttribute(-0.5f, 0.5f)]
[TooltipAttribute("A non-zero bias will move the target position vertically away from the center of the soft zone.")]
public float m_BiasY
```

### Field Value

Type: **Single**

### See Also

**Reference**
CinemachineFramingTransposer Class
Cinemachine Namespace
CinemachineFramingTransposer Class Field

The distance along the camera axis that will be maintained from the Follow target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
[TooltipAttribute("The distance along the camera axis that will be maintained from the Follow target")]
public float m_CameraDistance
``` | ```javascript
Copy
```  

**Field Value**

Type: Single

### See Also

- Reference
  - CinemachineFramingTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Field

The camera will not move along its z-axis if the Follow target is within this distance of the specified camera distance

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[TooltipAttribute("The camera will not move along its z-axis if the Follow target is within this distance of the specified camera distance")] [FormerlySerializedAsAttribute("m_DistanceDeadZoneSize")]
public float m_DeadZoneDepth
```

**Field Value**

- **Type:** Single

**See Also**

- **Reference**
  - CinemachineFramingTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer Field

Camera will not move vertically if the target is within this range of the position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[RangeAttribute(0f, 1f)]`  
 `[TooltipAttribute("Camera will not move vertically if the target is within this range of the position.")]`  
 `public float m_DeadZoneHeight` |

**Field Value**  
Type: Single

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer FIELD

Camera will not move horizontally if the target is within this range of the position

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```
[SpaceAttribute]
[RangeAttribute(0f, 1f)]
[TooltipAttribute("Camera will not move horizontally if the target is within this range of the position.")]

public float m_DeadZoneWidth
```

### Field Value

Type: **Single**

### See Also

**Reference**
- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerField

What screen dimensions to consider when framing

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```
[SpaceAttribute]
[TooltipAttribute("What screen dimensions to consider when framing. Can be Horizontal, Vertical, or both")]
[FormerlySerializedAsAttribute("m_FramingMode")]
public CinemachineFramingTransposerFramingMode m_GroupFramingMode
```

**Field Value**

Type: CinemachineFramingTransposerFramingMode

**See Also**

Reference  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Field

How much of the screen to fill with the bounding box of the targets.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;The bounding box of the target public float m_GroupFramingSize)]</td>
<td></td>
</tr>
</tbody>
</table>

#### Field Value

Type: **Single**

### See Also

**Reference**

- CinemachineFramingTransposer Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer.M_LookaheadSmoothing

Field

Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
public float m_LookaheadSmoothing
```

Field Value

Type: **Single**

**See Also**

**Reference**
- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer m_LookaheadTime

This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public float m_LookaheadTime</code></td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**  
**Type:** Single

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer.m_MaxDollyIn Field

How much closer to the target can the camera go?

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

```csharp
[TooltipAttribute("The maximum distance toward t")]
public float m_MaxDollyIn
```

### Field Value

- **Type:** Single

### See Also

- **Reference**
  - CinemachineFramingTransposer Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer Field

How much farther from the target can the camera go?

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The maximum distance away the target that this behaviour is allowed to move the camera.")] public float m_MaxDollyOut
```

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Field

Set this to limit how far from the target the camera can get

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

**C#**

```csharp
[TooltipAttribute("Set this to limit how far from
public float m_MaximumDistance
```

**JavaScript**

```javascript
```

### Field Value

Type: Single

### See Also

**Reference**

CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Field

If adjusting FOV, will not set the FOV higher than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▶ Syntax

```csharp
[RangeAttribute(1f, 179f)]
[TooltipAttribute("If adjusting FOV, will not set FOV higher than this.")]
public float m_MaximumFOV
```

Field Value  
Type: Single

▶ See Also

Reference  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer

Field

If adjusting Orthographic Size, will not set it higher than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▶ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
</table>
| [TooltipAttribute("If adjusting Orthographic Size, will not set it higher than this.")]
public float m_MaximumOrthoSize |

Field Value  
Type: Single

▶ See Also

Reference  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer._Field

Set this to limit how close to the target the camera can get

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public float m_MinimumDistance
```

**Field Value**  
Type: Single

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer m_MinimumFOV Field

If adjusting FOV, will not set the FOV lower than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[RangeAttribute(1f, 179f)]
[TooltipAttribute("If adjusting FOV, will not set")]
public float m_MinimumFOV
``` | Copy |

### Field Value

Type: Single

### See Also

**Reference**
- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer

Field

If adjusting Orthographic Size, will not set it lower than this

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

C#  
public float m_MinimumOrthoSize

JavaScript

[TooltipAttribute("If adjusting Orthographic Size, will not set it lower than this.")]

Field Value
Type: Single

See Also

Reference
CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerField

Horizontal screen position for target. The camera will move to position the tracked object here

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[SpaceAttribute]
[RangeAttribute(0f, 1f)]
[TooltipAttribute("Horizontal screen position for target. The camera will move to position the tracked object here.")]
public float m_ScreenX
```

**JavaScript**

This field is not directly accessible in JavaScript.

### Field Value

Type: **Single**

### See Also

**Reference**

- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer.m_ScreenY Field

Vertical screen position for target, The camera will move to position the tracked object here

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[RangeAttribute(0f, 1f)]</code></td>
<td><code>[TooltipAttribute('&quot;Vertical screen position for target, The camera will move to position the tracked object here.&quot;')]</code></td>
</tr>
<tr>
<td>public float m_ScreenY</td>
<td>public float m_ScreenY</td>
</tr>
</tbody>
</table>

Field Value  
Type: **Single**

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposerField

When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>`[RangeAttribute(0f, 2f)]</td>
<td>`[RangeAttribute(0f, 2f)]</td>
</tr>
<tr>
<td>`[TooltipAttribute(&quot;When target is within this region, camera will gradually move vertically to re-align towards the desired position, depending on the damping speed.&quot;)]]</td>
<td>`[TooltipAttribute(&quot;When target is within this region, camera will gradually move vertically to re-align towards the desired position, depending on the damping speed.&quot;)]]</td>
</tr>
<tr>
<td>public float m_SoftZoneHeight</td>
<td>public float m_SoftZoneHeight</td>
</tr>
</tbody>
</table>

**Field Value**  
Type: **Single**

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer.m_SoftZoneWidth

When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public float m_SoftZoneWidth</code></td>
<td></td>
</tr>
</tbody>
</table>

Field Value

Type: *Single*

### See Also

**Reference**

CinemachineFramingTransposer Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer.m_UnlimitedSoftZone

[Missing <summary> documentation for "F:Cinemachine.CinemachineFramingTransposer.m_UnlimitedSoftZone"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#       JavaScript

```csharp
[SpaceAttribute]
[TooltipAttribute("If checked, then then soft zor
public bool m_UnlimitedSoftZone
```

Field Value
Type: Boolean

See Also

Reference
CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer.m_XDamping Field

How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

Namespace: Cinemachine
Assembly: Cinemachine.dll Version: 2.0.0.0

Syntax

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.")]
public float m_XDamping
```

Field Value
Type: Single

See Also

Reference
CinemachineFramingTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposer Field

How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain the offset in the Y-axis.
Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.")]
public float m_YDamping
```

Field Value

**Type:** Single

**See Also**

**Reference**

- CinemachineFramingTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFramingTransposer Field

How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

**C#**

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

```

```javascript
public float m_ZDamping
```

### See Also

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerOnGUIField

Used by the Inspector Editor to display on-screen guides.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[NoSaveDuringPlayAttribute]</td>
<td></td>
</tr>
<tr>
<td>[HideInInspector]</td>
<td></td>
</tr>
<tr>
<td><strong>public</strong> Action OnGUICallback</td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**  
**Type:** Action

**See Also**

**Reference**  
CinemachineFramingTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerAdj
Enumeration

How to adjust the camera to get the desired framing

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
public enum AdjustmentMode
```

**Members**

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ZoomOnly</strong></td>
<td>0</td>
<td>Do not move the camera, only adjust the FOV.</td>
</tr>
<tr>
<td><strong>DollyOnly</strong></td>
<td>1</td>
<td>Just move the camera, don't change the FOV.</td>
</tr>
<tr>
<td><strong>DollyThenZoom</strong></td>
<td>2</td>
<td>Move the camera as much as permitted by the ranges, then adjust the FOV if necessary to make the shot.</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFramingTransposerFramingMode Enumeration

What screen dimensions to consider when framing

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[DocumentationSortingAttribute(4.01f, DocumentationSortingAttribute)]
public enum FramingMode
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal</td>
<td>0</td>
<td>Consider only the horizontal dimension. Vertical framing is ignored.</td>
</tr>
<tr>
<td>Vertical</td>
<td>1</td>
<td>Consider only the vertical dimension. Horizontal framing is ignored.</td>
</tr>
<tr>
<td>HorizontalAndVertical</td>
<td>2</td>
<td>The larger of the</td>
</tr>
</tbody>
</table>
horizontal and vertical dimensions will dominate, to get the best fit.

| None | 3 | Don't do any framing adjustment |

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLook Class

A Cinemachine Camera geared towards a 3rd person camera experience. The camera orbits around its subject with three separate camera rigs defining rings around the target. Each rig has its own radius, height offset, composer, and lens settings. Depending on the camera's position along the spline connecting these three rigs, these settings are interpolated to give the final camera position and state.

Inheritance Hierarchy

```
SystemObject  Object
  Component
    Behaviour
      MonoBehaviour
        Cinemachine
            CinemachineVirtualCameraBase
                Cinemachine
                    CinemachineFreeLook
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(11f, Documentationator
[ExecuteInEditMode]
[DisallowMultipleComponent]
[AddComponentMenu("Cinemachine/CinemachineFreeLoc
public class CinemachineFreeLook : CinemachineVir
```

The CinemachineFreeLook type exposes the following members.

Constructors
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the current Follow target. If the Follow is non-null and a specific Follow defined for this camera (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Returns the rig with the greatest weight. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the current LookAt target. If the parent is non-null and a specific LookAt defined for this camera (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. The implementation returns the owner GameObject's name. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-camera situation where a virtual camera is the public face of a private army of cameras which it manages on its own.</td>
</tr>
</tbody>
</table>
the VirtualCamera owner, if any. Are implemented as Transform parent vcam. (Inherited from CinemachineVi

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update with deltaTime and reset itself (Inherited from CinemachineVi</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera, which determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from CinemachineVi</td>
</tr>
<tr>
<td>RigNames</td>
<td>Names of the 3 child rigs</td>
</tr>
<tr>
<td>State</td>
<td>The camera state, which will be a blend of the 3 child rig states (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed for upgrading legacy (Inherited from CinemachineVi</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the behaviour. (Inherited from CinemachineVi</td>
</tr>
</tbody>
</table>

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVi</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>GetLocalPositionForCameraFromInput</td>
<td>Returns the local position of the camera along the spline used to connect the current heading of the camera.</td>
</tr>
<tr>
<td>GetRig</td>
<td>Get a child rig</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all child camera rigs. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the camera is a live child of this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is a Live camera if it shares the highest priority with its peers and its peers share the highest priority. If the most recent one becomes live, it goes to the top of the priority subqueue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Makes sure that the child rigs get destroyed in an undo-friendly manner. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Updates the child rig cache. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>If we are transitioning from another FreeLook, grab the axis values from it. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields, when changed in inspector. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of the camera's child cameras. If we are updating on FixedUpdate or LateUpdate, this is not necessarily called. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
RemovePostPipelineStageHook

Removes a Pipeline PostStageHook.

(Overrides CinemachineVirtualCameraBase)

ResolveFollow

Returns this vcam's Follow target, or if that is null, will return null.

(Inherited from CinemachineVirtualCameraBase)

ResolveLookAt

Returns this vcam's LookAt target, or if that is null, will return null.

(Inherited from CinemachineVirtualCameraBase)

Start

Base class implementation does nothing.

(Inherited from CinemachineVirtualCameraBase)

Update

Base class implementation makes sure the priority queue remains up-to-date.

(Inherited from CinemachineVirtualCameraBase)

UpdateCameraState

Called by CinemachineCore at designated update time with updated, and a blend calculated, depending on the value of the Y axis.

(Overrides CinemachineVirtualCameraBase)

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is there to support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>CreateRigOverride</td>
<td>Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do the same thing as the CreatePipeline method in this class.</td>
</tr>
<tr>
<td><strong>s</strong></td>
<td>DestroyRigOverride</td>
</tr>
<tr>
<td>-------</td>
<td>---------------------</td>
</tr>
<tr>
<td></td>
<td>m_BindingMode</td>
</tr>
<tr>
<td></td>
<td>m_CommonLens</td>
</tr>
<tr>
<td></td>
<td>m_ExcludedPropertiesInInspector</td>
</tr>
<tr>
<td></td>
<td>m_Follow</td>
</tr>
<tr>
<td></td>
<td>m_Heading</td>
</tr>
<tr>
<td></td>
<td>m_Lens</td>
</tr>
<tr>
<td></td>
<td>m_LockStageInInspector</td>
</tr>
<tr>
<td><strong>m_LookAt</strong></td>
<td>Object for the camera look at (the aim target)</td>
</tr>
<tr>
<td><strong>m_Orbits</strong></td>
<td>The radius and height of the three orbiting rigs</td>
</tr>
<tr>
<td><strong>m_Priority</strong></td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>m_RecenterToTargetHeading</strong></td>
<td>Controls how automatic recentering of the X axis is accomplished</td>
</tr>
<tr>
<td><strong>m_SplineCurvature</strong></td>
<td></td>
</tr>
<tr>
<td><strong>m_XAxis</strong></td>
<td>The Horizontal axis. Value is 0..359. This is passed to the rigs' OrbitalTransposer component</td>
</tr>
<tr>
<td><strong>m_YAxis</strong></td>
<td>The Vertical axis. Value is 0..1. Chooses how to blend the child rigs</td>
</tr>
<tr>
<td><strong>OnPostPipelineStage</strong></td>
<td>A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLook Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineFreeLook.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td></td>
</tr>
<tr>
<td>CinemachineFreeLook()</td>
<td></td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The `CinemachineFreeLook` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
</tbody>
</table>
| Follow           | Get the current Follow target. If Follow is not defined, or the parent is null, return the parent’s Follow. (Overrides `CinemachineVirtualCameraBase`)
| LiveChildOrSelf  | Returns the rig with the greatest weight. (Overrides `CinemachineVirtualCameraBase`)                                                         |
| LookAt           | Get the current LookAt target. If LookAt is not defined, or the parent is null, return the parent’s LookAt. (Overrides `CinemachineVirtualCameraBase`) |
| Name             | Get the name of the Virtual Camera. In the base implementation returns the owner GameObject's name. (Inherited from `CinemachineVirtualCameraBase`) |
| ParentCamera     | Support for meta-virtual-camera.                                                                                                          |
situation where a virtual camera is the public face of a private army of cameras which it manages on its own. The VirtualCamera owner, if any, are implemented as Transform children of the parent vcam.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to ignore deltaTime and reset itself (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. This determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>RigNames</td>
<td>Names of the 3 child rigs</td>
</tr>
<tr>
<td>State</td>
<td>The camera state, which will be a blend of the child rig states (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed, for upgrading legacy (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the VirtualCamera behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

---

**See Also**

Reference

- CinemachineFreeLook Class
- Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookFollow
Property

Get the current Follow target. Returns parent's Follow if parent is non-null and no specific Follow defined for this camera.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override Transform Follow { get; set; }
```

<table>
<thead>
<tr>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Transform</code></td>
</tr>
</tbody>
</table>

Implements

ICinemachineCameraFollow

### See Also

Reference

CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookLiveChildOrSelf

Property

Returns the rig with the greatest weight

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript

```csharp
public override ICinemachineCamera LiveChildOrSelf
```

Property Value

Type: ICinemachineCamera
Implements ICinemachineCameraLiveChildOrSelf

See Also

Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookLookAt
Property

Get the current LookAt target. Returns parent's LookAt if parent is non-null and no specific LookAt defined for this camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  

public override Transform LookAt { get; set; }  

JavaScript

Property Value

Type: Transform
Implements
ICinemachineCameraLookAt

See Also

Reference

CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookRigNames

Property

Names of the 3 child rigs

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public static string[] RigNames { get; }
```

Property Value

Type: **String**

**See Also**

Reference

CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookState
Property

The camera state, which will be a blend of the child rig states

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override CameraState State { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: CameraState
Implements
ICinemachineCameraState

See Also

Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLook Methods

The **CinemachineFreeLook** type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![humans] AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] GetLocalPositionForCameraFromInput</td>
<td>Returns the local position of the camera along the spline used to connect the current heading of the Live camera if it shares the highest priority in the queue with its peers, then this vcam will become Live. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] GetRig</td>
<td>Get a child rig</td>
</tr>
<tr>
<td>![humans] InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all Live cameras. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] IsLiveChild</td>
<td>Check whether the Live child is a live child of this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is the most recent Live camera that shares the highest priority in the queue with its peers. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] OnDestroy</td>
<td>Makes sure that the child trigs get destroyed in an undo-friendly manner. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>![humans] OnEnable</td>
<td>Updates the child trig cache</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>If we are transitioning from another FreeLook, grab the axis values from it. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields, when changed in inspector. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of the vcum's child cameras, and a blend calculated, depending on the value of the Y axis. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook callback. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcum's Follow target, or if that is null, will return null. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ResolveLookAt</td>
<td>Returns this vcum's LookAt target, or if that is null, will return null. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Start</td>
<td>Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Update</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>UpdateCameraState</td>
<td>Called by CinemachineCore at designated update time.</td>
</tr>
</tbody>
</table>

See Also
Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookGetLocalPositionForCameraFromInput Method

Returns the local position of the camera along the spline used to connect the three camera rigs. Does not take into account the current heading of the camera (or its target)

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public Vector3 GetLocalPositionForCameraFromInput(float t)
```

### Parameters

- **t**  
  Type: `System.Single`  
  The t-value for the camera on its spline. Internally clamped to the value [0,1]

### Return Value

Type: `Vector3`  
The local offset (back + up) of the camera WRT its target based on the supplied t-value

### See Also

Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookGetRig Method

Get a child rig

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Parameters

- **i**  
  Type: `SystemInt32`  
  Rig index. Can be 0, 1, or 2

### Return Value

Type: `CinemachineVirtualCamera`  
The rig, or null if index is bad.

### See Also

**Reference**  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLook.IsLiveChild Method

Check whether the vcam a live child of this camera. Returns true if the child is currently contributing actively to the camera state.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public override bool IsLiveChild(I CIMachineCamera vcam)
```

**JavaScript**

```javascript
// Not provided
```

### Parameters

**vcam**

Type: CIMachineCameraICIMachineCamera  
The Virtual Camera to check

### Return Value

Type: Boolean  
True if the vcam is currently actively influencing the state of this vcam

**Implements**

ICIMachineCameraICIMachineCamera.IsLiveChild(ICIMachineCamera)

### See Also

Reference  
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLook OnDestroy Method

Makes sure that the child rigs get destroyed in an undo-friendly manner. Invalidates the rig cache.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected override void OnDestroy()</code></td>
<td></td>
</tr>
</tbody>
</table>

**See Also**

Reference  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookOnEnable Method

Updates the child rig cache

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected override void OnEnable()</code></td>
<td></td>
</tr>
</tbody>
</table>

### See Also

**Reference**
- CinemachineFreeLook Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookOnTransitionFromCamera Method

If we are transitioning from another FreeLook, grab the axis values from it.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override void OnTransitionFromCamera(
    ICinemachineCamera fromCam
)
```

### Parameters

`fromCam`  
Type: `CinemachineICinemachineCamera`  
The camera being deactivated. May be null.

### See Also

Reference  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookOnValidate Method

Enforce bounds for fields, when changed in inspector.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected override void OnValidate()</code></td>
<td></td>
</tr>
</tbody>
</table>

### See Also

**Reference**  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookPreUpdateChildCameras Method

This is called prior to the updating of the vcam's child cameras, in order to allow the parent to prepare its children. If the children are updating on FixedUpdate, then this will not necessarily be called prior to every FixedUpdate, but might be called on LateUpdate. This implementation pushes the axis values to the rigs.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public override void PreUpdateChildCameras(
    Vector3 worldUp,
    float deltaTime
)
```

Parameters

- **worldUp**
  Type: Vector3
  Default world Up, set by the CinemachineBrain
- **deltaTime**
  Type: System.Single
  Delta time for time-based effects (ignore if less than 0)

Implements
ICinemachineCameraPreUpdateChildCameras(Vector3, Single)

See Also
Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookRemovePostPipelineStageHook Method

Remove a Pipeline stage hook callback. Make sure it is removed from all the children.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public override void RemovePostPipelineStageHook(</code></td>
<td></td>
</tr>
<tr>
<td><code>CinemachineVirtualCameraBaseOnPostPipelineStageDelegate </code></td>
<td></td>
</tr>
<tr>
<td><code>)</code></td>
<td></td>
</tr>
</tbody>
</table>

**Parameters**

\(d\)

- **Type:** `CinemachineVirtualCameraBaseOnPostPipelineStageDelegate`  
The delegate to remove.

### See Also

**Reference**  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookUpdateCameraMethod

Called by CinemachineCore at designated update time so the vcam can position itself and track its targets. All 3 child rigs are updated, and a blend calculated, depending on the value of the Y axis.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

#### Syntax

```csharp
public override void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

**Parameters**

- **worldUp**
  - Type: `Vector3`
  - Default world Up, set by the CinemachineBrain

- **deltaTime**
  - Type: `System.Single`
  - Delta time for time-based effects (ignore if less than 0)

**Implements**

`ICinemachineCameraUpdateCameraState(Vector3, Single)`

#### See Also

Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLook Fields

The CinemachineFreeLook type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>CreateRigOverride</td>
<td>Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do the same thing as the CreatePipeline method in this class.</td>
</tr>
<tr>
<td>DestroyRigOverride</td>
<td>Override component pipeline destruction. This needs to be done by the editor to support Undo.</td>
</tr>
<tr>
<td>m_BindingMode</td>
<td>The coordinate space when interpreting the offset from the target.</td>
</tr>
<tr>
<td>m_CommonLens</td>
<td>If enabled, this lens setting will apply to all three child rigs; otherwise the child rig settings will be used.</td>
</tr>
<tr>
<td><strong>m_ExcludedPropertiesInInspector</strong></td>
<td>Inspector control - Use for hiding sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_Follow</strong></td>
<td>Object for the camera children wants to move with (the body target)</td>
</tr>
<tr>
<td><strong>m_Heading</strong></td>
<td>The definition of Forward. Camera will follow behind the body target</td>
</tr>
<tr>
<td><strong>m_Lens</strong></td>
<td>Specifies the lens properties of this Virtual Camera. This generally mirrors the Unity Camera's lens settings, and will be used to drive the Unity camera when the vcam is active</td>
</tr>
<tr>
<td><strong>m_LockStageInInspector</strong></td>
<td>Inspector control - Use for enabling sections of the UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>m_LookAt</strong></td>
<td>Object for the camera children to look at (the aim target)</td>
</tr>
<tr>
<td><strong>m_Orbits</strong></td>
<td>The radius and height of the three orbiting rigs</td>
</tr>
<tr>
<td><strong>m_Priority</strong></td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
### m_RecenterToTargetHeading
Controls how automatic recentering of the X axis is accomplished.

### m_SplineCurvature

### m_XAxis
The Horizontal axis. Value is 0..359. This is passed to the rigs’ OrbitalTransposer component.

### m_YAxis
The Vertical axis. Value is 0..1. Chooses how to blend the child rigs.

### OnPostPipelineStage
A delegate to hook into the calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage (Inherited from CinemachineVirtualCameraBase).

---

**Top**

### See Also

**Reference**
- CinemachineFreeLook Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookCreateRigOverride Field

Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do exactly the same thing as the CreatePipeline method in this class.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public static CinemachineFreeLookCreateRigDelegate
```

**JavaScript**

Field Value  
Type: CinemachineFreeLookCreateRigDelegate

### See Also

**Reference**  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookDestroyRigOverride Field

Override component pipeline destruction. This needs to be done by the editor to support Undo.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public static</code></td>
<td>CinemachineFreeLookDestroyRigDelegate</td>
</tr>
</tbody>
</table>

Field Value  
Type: `CinemachineFreeLookDestroyRigDelegate`

### See Also

- Reference  
  - CinemachineFreeLook Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookm_BindingMc Field

The coordinate space to use when interpreting the offset from the target

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[HeaderAttribute("Orbits")]
[TooltipAttribute("The coordinate space to use when interpreting the offset from the target. This is also used to set the camera's Up vector, which will be maintained when aiming the camera.")]

public CinemachineTransposerBindingMode m_BindingMc
``` |

### Field Value
Type: **CinemachineTransposerBindingMode**

### See Also

**Reference**  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookm_CommonL

Field

If enabled, this lens setting will apply to all three child rigs, otherwise the child rig lens settings will be used

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

```csharp
[TooltipAttribute("If enabled, this lens setting will apply to all three child rigs, otherwise the child rig lens settings will be used")]
[FormerlySerializedAsAttribute("m_UseCommonLensSetting")]
public bool m_CommonLens
```

Field Value

Type: **Boolean**

**See Also**

- Reference  
  - CinemachineFreeLook Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookm_Follow

Field

Object for the camera children wants to move with (the body target)

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  

```csharp
[TooltipAttribute("Object for the camera children wants to move with (the body target).")
[NoSaveDuringPlayAttribute]
public Transform m_Follow
```

Field Value

Type: Transform

See Also

Reference

- CinemachineFreeLook Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookm_Heading Field

The definition of Forward. Camera will follow behind

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[TooltipAttribute("The definition of Forward. Camera will follow behind.")]
public CinemachineOrbitalTransposerHeading m_Heading
```

Field Value

Type:  CinemachineOrbitalTransposerHeading

### See Also

**Reference**

- CinemachineFreeLook Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookm_Lens Field

Specifies the lens properties of this Virtual Camera. This generally mirrors the Unity Camera's lens settings, and will be used to drive the Unity camera when the vcam is active

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

🎉 **Syntax**

```csharp
[FormerlySerializedAsAttribute("m_LensAttributes")]
[TooltipAttribute("Specifies the lens properties of this Virtual Camera. This generally mirrors the Unity Camera's lens settings, and will be used to drive the Unity camera when the vcam is active")]
[LensSettingsPropertyAttribute]
public LensSettings m_Lens
```

Field Value
Type: **LensSettings**

🎉 **See Also**

Reference
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookm_LookAt Field

Object for the camera children to look at (the aim target)

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[TooltipAttribute("Object for the camera children
to look at (the aim target).")]
[NoSaveDuringPlayAttribute]
public Transform m_LookAt
```

Field Value
Type: Transform

See Also

Reference
CinemachineFreeLook Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookm_Orbits Field

The radius and height of the three orbiting rigs

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td>CinemachineFreeLook Orbit[] m_Orbits</td>
</tr>
</tbody>
</table>

Field Value  
Type: CinemachineFreeLook Orbit

**See Also**

Reference  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookm_RecenterT Field

Controls how automatic recentering of the X axis is accomplished

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[TooltipAttribute("Controls how automatic recentering of the X axis is accomplished")]

```csharp
public CinemachineOrbitalTransposerRecentering m_RecenterToTargetHeading
```

Field Value

Type:  CinemachineOrbitalTransposerRecentering

## See Also

**Reference**
- CinemachineFreeLook Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLook\textunderscore\_m\_SplineCurvature Field

[Missing \texttt{<summary>} documentation for \texttt{"F:Cinemachine.CinemachineFreeLook.m\_SplineCurvature"}]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
[TooltipAttribute("Controls how taut is the line
[RangeAttribute(0f, 1f)]
[FormerlySerializedAsAttribute("m\_SplineTension")]
public float m\_SplineCurvature
``` |

**Field Value**  
Type: Single

### See Also

**Reference**  
- CinemachineFreeLook Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookm_XAxis Field

The Horizontal axis. Value is 0..359. This is passed on to the rigs' OrbitalTransposer component

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="C# code" /></td>
<td><img src="image" alt="JavaScript code" /></td>
</tr>
</tbody>
</table>

**Field Value**  
**Type:** AxisState

### See Also

**Reference**
- CinemachineFreeLook Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookm_YAxis Field

The Vertical axis. Value is 0..1. Chooses how to blend the child rigs

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[HeaderAttribute("Axis Control")]
[TooltipAttribute("The Vertical axis. Value is 0..1. Chooses how to blend the child rigs")]
public AxisState m_YAxis
```

Field Value  
Type: AxisState

**See Also**

Reference  
CinemachineFreeLook Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookCreateRigDelegate

Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do exactly the same thing as the CreatePipeline method in this class.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public delegate CinemachineVirtualCamera CreateRigDelegate(
    CinemachineFreeLook vcam,
    string name,
    CinemachineVirtualCamera copyFrom
)
```

### Parameters

- **vcam**  
  Type: CinemachineFreeLook

- **name**  
  Type: System.String

- **copyFrom**  
  Type: CinemachineVirtualCamera

### Return Value

Type: CinemachineVirtualCamera

### See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookDestroyRigDelegate

Override component pipeline destruction. This needs to be done by the editor to support Undo.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public delegate void DestroyRigDelegate(
    GameObject rig
)
```

**JavaScript**

```
public delegate void DestroyRigDelegate(
    GameObject rig
)
```

### Parameters

- **rig**  
  Type: `GameObject`

### See Also

**Reference**  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookOrbit Structure

Defines the height and radius of the Rig orbit

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**  
```csharp
[SerializableAttribute]
public struct Orbit
```

**JavaScript**  

The CinemachineFreeLookOrbit type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Constructor" /> CinemachineFreeLookOrbit</td>
<td>Constructor with specific values</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Field" /> m_Height</td>
<td>Height relative to target</td>
</tr>
<tr>
<td><img src="image" alt="Field" /> m_Radius</td>
<td>Radius of orbit</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookOrbit Constructor

Constructor with specific values

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public Orbit(
    float h,
    float r
)
```

### Parameters

- **h**
  - Type: `System.Single`  

- **r**
  - Type: `System.Single`  

## See Also

- Reference:  
  - CinemachineFreeLookOrbit Structure  
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Orbit Fields

The CinemachineFreeLookOrbit type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Height</td>
<td>Height relative to target</td>
</tr>
<tr>
<td>m_Radius</td>
<td>Radius of orbit</td>
</tr>
</tbody>
</table>

See Also

Reference

CinemachineFreeLookOrbit Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineFreeLookOrbitm_Height Field

Height relative to target

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public float m_Height
```

Field Value  
Type:  Single

### See Also

Reference  
CinemachineFreeLookOrbit Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineFreeLookOrbitm_Radius Field

Radius of orbit

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public float m_Radius</code></td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**
Type: **Single**

### See Also

**Reference**
- CinemachineFreeLookOrbit Structure
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGroupComposer Class

This is a CinemachineComponent in the Aim section of the component pipeline. Its job is to aim the camera at a target object, with configurable offsets, damping, and composition rules. In addition, if the target is a CinemachineTargetGroup, the behaviour will adjust the FOV and the camera distance to ensure that the entire group of targets is framed properly.

Inheritance Hierarchy

- System
  - SystemObject
  - Object
  - Component
  - Behaviour
    - MonoBehaviour
      - CinemachineCinemachineComponentBase
      - CinemachineCinemachineComposer
      - CinemachineCinemachineGroupComposer

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public class CinemachineGroupComposer : Cinemachine
```
The **CinemachineGroupComposer** type exposes the following members.

## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGroupComposer</td>
<td></td>
</tr>
</tbody>
</table>

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>HardGuideRect</td>
<td>Internal API for the inspector editor (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a LookAt defined (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>m_LastBounds</td>
<td>For editor visualization of the calculated bounding box of the group</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTrackedPoint</td>
<td>Apply the target offsets to the target location. (Inherited from CinemachineComposer)</td>
</tr>
</tbody>
</table>
MutateCameraState

Applies the composer rules and orients accordingly
(Overrides CinemachineComposer::MutateCameraSingle.)

OnPositionDragged

API for the editor, to process a position drag from the user.
Base class implementation does nothing.
(Inherited from CinemachineComponent.

---

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AdjustmentMode</td>
<td>How to adjust the camera to get the desired framing</td>
</tr>
<tr>
<td>m_BiasX</td>
<td>A non-zero bias will move the target position away from the center of the soft zone (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_BiasY</td>
<td>A non-zero bias will move the target position away from the center of the soft zone (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_DeadZoneHeight</td>
<td>Camera will not rotate vertically if the target is within this range of the position (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_DeadZoneWidth</td>
<td>Camera will not rotate horizontally if the target is within this range of the position. Inherited from CinemachineComposer.</td>
</tr>
<tr>
<td>m_FrameDamping</td>
<td>How aggressively the camera tries to frame the group. Small numbers are more responsive.</td>
</tr>
<tr>
<td>m_FramingMode</td>
<td>What screen dimensions to consider when framing.</td>
</tr>
<tr>
<td>m_GroupFramingSize</td>
<td>How much of the screen to fill with the bounding box of the targets.</td>
</tr>
<tr>
<td>m_HorizontalDamping</td>
<td>How aggressively the camera tries to follow the target in the screen-horizontal direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors.</td>
</tr>
<tr>
<td><strong>m_LookaheadSmoothing</strong></td>
<td>Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_LookaheadTime</strong></td>
<td>This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td><strong>m_MaxDollyIn</strong></td>
<td>How much closer to the target can the camera go?</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_MaxDollyOut</td>
<td>How much farther from the target can the camera go?</td>
</tr>
<tr>
<td>m_MaximumDistance</td>
<td>Set this to limit how far from the target the camera can get</td>
</tr>
<tr>
<td>m_MaximumFOV</td>
<td>If adjusting FOV, will not set the FOV higher than this</td>
</tr>
<tr>
<td>m_MaximumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it higher than this</td>
</tr>
<tr>
<td>m_MinimumDistance</td>
<td>Set this to limit how close to the target the camera can get</td>
</tr>
<tr>
<td>m_MinimumFOV</td>
<td>If adjusting FOV, will not set the FOV lower than this</td>
</tr>
<tr>
<td>m_MinimumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it lower than this</td>
</tr>
<tr>
<td>m_ScreenX</td>
<td>Horizontal screen position for target. The camera will rotate to the position the tracked object here</td>
</tr>
<tr>
<td></td>
<td>(Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_ScreenY</td>
<td>Vertical screen position for target, The camera will rotate to to position</td>
</tr>
<tr>
<td><strong>m_SoftZoneHeight</strong></td>
<td>When target is within this region, camera will gradually move to realign towards the desired position, depending on the damping speed. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_SoftZoneWidth</strong></td>
<td>When target is within this region, camera will gradually move to realign towards the desired position, depending on the damping speed. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td><strong>m_TrackedObjectOffset</strong></td>
<td>Target offset from the object's center in LOCAL space which the Composer tracks. Use this to fine-tune the tracking target position when the desired area is not in the tracked object's center. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td><strong>m_VerticalDamping</strong></td>
<td>How aggressively the camera tries to follow the target in the screen-</td>
</tr>
</tbody>
</table>
vertical direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors. (Inherited from CinemachineComposer.)

<table>
<thead>
<tr>
<th>OnGUICallback</th>
</tr>
</thead>
<tbody>
<tr>
<td>Used by the Inspector Editor to display on-screen guides. (Inherited from CinemachineComposer.)</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer Constructor

Namespace: Cinemachine

Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

C#

```csharp
public CinemachineGroupComposer()
```

JavaScript


See Also

Reference
CinemachineGroupComposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# CinemachineGroupComposer Properties

The `CinemachineGroupComposer` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>HardGuideRect</strong></td>
<td>Internal API for the inspector editor (Inherited from <code>CinemachineComposer</code>.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a LookAt defined (Inherited from <code>CinemachineComposer</code>.)</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>m_LastBounds</strong></td>
<td>For editor visualization of the calculated bounding box of the group</td>
</tr>
<tr>
<td><strong>m_lastBoundsMatrix</strong></td>
<td>For editor visualization of the</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SoftGuideRect</td>
<td>Internal API for the inspector editor (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>TargetGroup</td>
<td>Get LookAt target as CinemachineTargetGroup, or null if target is not a group</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

**See Also**

Reference
- CinemachineGroupComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_LastBounds Property

For editor visualization of the calculated bounding box of the group

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public Bounds m_LastBounds { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**  
**Type:** Bounds

### See Also

**Reference**  
CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGroupComposer.m_lastBoundsMatrix

Property

For editor visualization of the calculated bounding box of the group

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

![Syntax](Copy)

### C#  
```
public Matrix4x4 m_lastBoundsMatrix { get; }
```

### JavaScript

#### Property Value

**Type:** `Matrix4x4`

### See Also

- Reference
- CinemachineGroupComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposerTargetGroup Property

Get LookAt target as CinemachineTargetGroup, or null if target is not a group

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▲ Syntax

C#  
```csharp
public CinemachineTargetGroup TargetGroup { get; }
```

JavaScript  
```javascript
// Not applicable
```

Property Value  
Type: CinemachineTargetGroup

▲ See Also

Reference  
CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGroupComposer

Methods

The CinemachineGroupComposer type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTrackedPoint</td>
<td>Apply the target offsets to the target location. (Inherited from CinemachineComposer)</td>
</tr>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly. (Overrides CinemachineComposerMutateCameraSingle.)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position dragged. Base class implementation does nothing. (Inherited from CinemachineComposer)</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineGroupComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGroupComposerMutateMethod

Applies the composer rules and orients the camera accordingly

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▲ Syntax

```csharp
public override void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)
```

**Parameters**

*curState*
- Type: CinemachineCameraState

*deltaTime*
- Type: System.Single
  - Used for calculating damping. If less than zero, then target will snap to the center of the dead zone.

▲ See Also

**Reference**
- CinemachineGroupComposer Class
- Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
The **CinemachineGroupComposer** type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AdjustmentMode</td>
<td>How to adjust the camera to get the desired framing</td>
</tr>
<tr>
<td>m_BiasX</td>
<td>A non-zero bias will move the target position away from the center of the soft zone (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_BiasY</td>
<td>A non-zero bias will move the target position away from the center of the soft zone (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_DeadZoneHeight</td>
<td>Camera will not rotate vertically if the target is within this range of the position (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_DeadZoneWidth</td>
<td>Camera will not rotate horizontally if the target is within this range of the position. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_FrameDamping</td>
<td>How aggressively the camera tries to frame the group. Small numbers are more responsive.</td>
</tr>
<tr>
<td>m_FramingMode</td>
<td>What screen dimensions to consider when framing.</td>
</tr>
<tr>
<td>m_GroupFramingSize</td>
<td>How much of the screen to fill with the bounding box of the targets.</td>
</tr>
<tr>
<td>m_HorizontalDamping</td>
<td>How aggressively the camera tries to follow the target in the screen-horizontal direction. Small numbers are more responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors. (Inherited from CinemachineComposer.)</td>
</tr>
</tbody>
</table>
m_LookaheadSmoothing Controls the smoothness of the lookahead algorithm. Larger values smooth out jittery predictions and also increase prediction lag (Inherited from CinemachineComposer.)

m_LookaheadTime This setting will instruct the composer to adjust its target offset based on the motion of the target. The composer will look at a point where it estimates the target will be this many seconds into the future. Note that this setting is sensitive to noisy animation, and can amplify the noise, resulting in undesirable camera jitter. If the camera jitters unacceptably when the target is in motion, turn down this setting, or animate the target more smoothly. (Inherited from CinemachineComposer.)

m_MaxDollyIn How much closer to the target can the camera go?

m_MaxDollyOut How much farther from the target can the
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_MaximumDistance</td>
<td>Set this to limit how far from the target the camera can get</td>
</tr>
<tr>
<td>m_MaximumFOV</td>
<td>If adjusting FOV, will not set the FOV higher than this</td>
</tr>
<tr>
<td>m_MaximumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it higher than this</td>
</tr>
<tr>
<td>m_MinimumDistance</td>
<td>Set this to limit how close to the target the camera can get</td>
</tr>
<tr>
<td>m_MinimumFOV</td>
<td>If adjusting FOV, will not set the FOV lower than this</td>
</tr>
<tr>
<td>m_MinimumOrthoSize</td>
<td>If adjusting Orthographic Size, will not set it lower than this</td>
</tr>
<tr>
<td>m_ScreenX</td>
<td>Horizontal screen position for target. The camera will rotate to the position the tracked object here (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_ScreenY</td>
<td>Vertical screen position for target. The camera will rotate to to position the tracked object here (Inherited from</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_SoftZoneHeight</td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_SoftZoneWidth</td>
<td>When target is within this region, camera will gradually move to re-align towards the desired position, depending on the damping speed. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_TrackedObjectOffset</td>
<td>Target offset from the object's center in LOCAL space which the Composer tracks. Use this to fine-tune the tracking target position when the desired area is not in the tracked object's center. (Inherited from CinemachineComposer.)</td>
</tr>
<tr>
<td>m_VerticalDamping</td>
<td>How aggressively the camera tries to follow the target in the screen-vertical direction. Small numbers are more</td>
</tr>
</tbody>
</table>
responsive, rapidly orienting the camera to keep the target in the dead zone. Larger numbers give a more heavy slowly responding camera. Using different vertical and horizontal settings can yield a wide range of camera behaviors. (Inherited from CinemachineComposer.)

| OnGUICallback | Used by the Inspector Editor to display on-screen guides. (Inherited from CinemachineComposer.) |

---

**See Also**

**Reference**
- CinemachineGroupComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_AdjustmentMode

How to adjust the camera to get the desired framing

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

#### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
    [TooltipAttribute("How to adjust the camera to get the desired framing.
    You can zoom, dolly in/out, or both.")]
    public CinemachineGroupComposerAdjustmentMode m_AdjustmentMode
``` |

Field Value
Type: `CinemachineGroupComposerAdjustmentMode`

#### See Also

**Reference**
- `CinemachineGroupComposer Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer m_FrameDamping Field

How aggressively the camera tries to frame the group. Small numbers are more responsive

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

![Syntax](#)

Field Value

Type: **Single**

**See Also**

**Reference**

CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer m_FramingMode

Field

What screen dimensions to consider when framing

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("What screen dimensions to consider when framing. Can be Horizontal, Vertical, or both")]
public CinemachineGroupComposerFramingMode m_FramingMode |

Field Value

Type: CinemachineGroupComposerFramingMode

See Also

Reference

CinemachineGroupComposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_GroupFramingSize

Field

How much of the screen to fill with the bounding box of the targets.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll)  
Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```
[SpatialAttribute]
[TooltipAttribute("The bounding box of the target
targets should occupy this amount of the	space.	1 means fill the whole	screen.	0.5 means fill	half the	screen, etc.")] public float m_GroupFramingSize
```

Field Value

Type: **Single**

**See Also**

- Reference  
  CinemachineGroupComposer Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_MaxDollyIn Field

How much closer to the target can the camera go?

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```
[TooltipAttribute("The maximum distance toward t")]
public float m_MaxDollyIn
```

**Field Value**  
Type: **Single**

**See Also**

**Reference**  
CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGroupComposer.m_MaxDollyOut Field

How much farther from the target can the camera go?

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The maximum distance away the target the behaviour is allowed to move the camera.")]
public float m_MaxDollyOut
```

**Field Value**  
**Type:** Single

### See Also

**Reference**  
- CinemachineGroupComposer Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_MaximumDistance Field

Set this to limit how far from the target the camera can get.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Set this to limit how far from
target the camera can get.")]
public float m_MaximumDistance
``` |

Field Value

Type: Single

### See Also

**Reference**

- CinemachineGroupComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer m_MaximumFOV Field

If adjusting FOV, will not set the FOV higher than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

![CinemachineGroupComposer m_MaximumFOV Field](image)

**Syntax**

C#  

```csharp
public float m_MaximumFOV
```

JavaScript

```javascript
[RangeAttribute(1f, 179f)]
[TooltipAttribute("If adjusting FOV, will not set FOV higher than this.")]
```

**Field Value**  
Type: **Single**

**See Also**

- Reference  
  - CinemachineGroupComposer Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_MaximumOrthoSize Field

If adjusting Orthographic Size, will not set it higher than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
[TooltipAttribute("If adjusting Orthographic Size,
will not set it higher than this.")]
public float m_MaximumOrthoSize
```

**Field Value**  
**Type:** Single

### See Also

**Reference**  
CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGroupComposer.m_MinimumDistance Field

Set this to limit how close to the target the camera can get

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Set this to limit how close to
public float m_MinimumDistance
``` |

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_MinimumFOV Field

If adjusting FOV, will not set the FOV lower than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
[RangeAttribute(1f, 179f)]
[TooltipAttribute("If adjusting FOV, will not set FOV lower than this.")]
public float m_MinimumFOV
``` | ```javascript
// No equivalent
``` |

**Field Value**

Type: **Single**

### See Also

**Reference**
- CinemachineGroupComposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposer.m_MinimumOrthoSize Field

If adjusting Orthographic Size, will not set it lower than this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
[TooltipAttribute("If adjusting Orthographic Size, will not set it lower than this.")]
public float m_MinimumOrthoSize
```

**Field Value**  
**Type:** Single

### See Also

**Reference**  
CinemachineGroupComposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposerAdjustmentMode Enumeration

How to adjust the camera to get the desired framing

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```csharp
public enum AdjustmentMode
```

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZoomOnly</td>
<td>0</td>
<td>Do not move the camera, only adjust the FOV.</td>
</tr>
<tr>
<td>DollyOnly</td>
<td>1</td>
<td>Just move the camera, don't change the FOV.</td>
</tr>
<tr>
<td>DollyThenZoom</td>
<td>2</td>
<td>Move the camera as much as permitted by the ranges, then adjust the FOV if necessary to make the shot.</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGroupComposerFramingMode Enumeration

What screen dimensions to consider when framing

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

C#  
```csharp
[DocumentationSortingAttribute(4.01f, DocumentationSortingAttribute)]
public enum FramingMode
```

JavaScript

## Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horizontal</td>
<td>0</td>
<td>Consider only the horizontal dimension. Vertical framing is ignored.</td>
</tr>
<tr>
<td>Vertical</td>
<td>1</td>
<td>Consider only the vertical dimension. Horizontal framing is ignored.</td>
</tr>
<tr>
<td>HorizontalAndVertical</td>
<td>2</td>
<td>The larger of the</td>
</tr>
</tbody>
</table>
horizontal and vertical dimensions will dominate, to get the best fit.

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLockToTarget Class

This is a CinemachineComponent in the Aim section of the component pipeline. Its job is to place the camera on the Follow Target.

Inheritance Hierarchy

- SystemObject
  - Object
    - Component
      - Behaviour
        - MonoBehaviour
          - Cinemachine
            - CinemachineComponentBase
              - CinemachineCinemachineHardLockToTarget

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(23f, Documentation)]
[AddComponentMenu("")]
[RequireComponent(typeof(CinemachinePipeline))]
[SaveDuringPlayAttribute]
public class CinemachineHardLockToTarget : Cinema
```

The CinemachineHardLockToTarget type exposes the following members.

Constructors
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineHardLockToTarget</td>
<td></td>
</tr>
<tr>
<td><strong>Top</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Properties</strong></td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="FollowTarget" /></td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><img src="image" alt="IsValid" /></td>
<td>True if component is enabled and has a LookAt defined (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td><img src="image" alt="LookAtTarget" /></td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><img src="image" alt="Stage" /></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage ( Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td><img src="image" alt="VcamState" /></td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><img src="image" alt="VirtualCamera" /></td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly (Overrides CinemachineComponentBase.MutateCameraState).</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. (Inherited from CinemachineComponentBase).</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLockToTarget Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineHardLockToTarget.#ctor"]

Namespace: Cinemachine  
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public CinemachineHardLockToTarget()</code></td>
<td></td>
</tr>
</tbody>
</table>

## See Also

- Reference  
  - CinemachineHardLockToTarget Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
The `CinemachineHardLockToTarget` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a LookAt defined. (Overrides <code>CinemachineComponentBaseIsValid</code>.)</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam’s LookAt target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage. (Overrides <code>CinemachineComponentBaseStage</code>.)</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam’s CameraState. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
</tbody>
</table>
VirtualCamera  Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)

See Also

Reference
CinemachineHardLockToTarget Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLockToTargetIsValid Property

True if component is enabled and has a LookAt defined

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override bool IsValid { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**
Type: Boolean

### See Also

**Reference**
CinemachineHardLockToTarget Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLockToTargetStage Property

Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

C#  
```csharp
public override CinemachineCoreStage Stage { get; }
```

### Property Value

Type: CinemachineCoreStage

### See Also

Reference  
- CinemachineHardLockToTarget Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
The `CinemachineHardLockToTarget` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly (Overrides <code>CinemachineComponentBase.MutateCameraState(CameraState Single)</code>).</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. The class implementation does nothing. (Inherited from <code>CinemachineComponent</code>).</td>
</tr>
</tbody>
</table>

## See Also

### Reference
- `CinemachineHardLockToTarget Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineHardLockToTargetMutateCameraState Method

Applies the composer rules and orients the camera accordingly

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public override void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)
```

Parameters

- `curState` : Type: CinemachineCameraState
  The current camera state
- `deltaTime` : Type: System.Single
  Used for calculating damping. If less than zero, then target will snap to the center of the dead zone.

See Also

Reference
- CinemachineHardLockToTarget Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineHardLookAt Class

This is a CinemachineComponent in the Aim section of the component pipeline. Its job is to aim the camera hard at the LookAt target.

Inheritance Hierarchy

```
System
    Object
    Component
    Behaviour
        MonoBehaviour
            Cinemachine
                CinemachineComponentBase
                    Cinemachine
                        CinemachineHardLookAt
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[DocumentationSortingAttribute(23f, Documentatio
[AddComponentMenu("")]
[RequireComponent(typeof(CinemachinePipeline))]
[SaveDuringPlayAttribute]
public class CinemachineHardLookAt : Cinemachine
```

The CinemachineHardLookAt type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineHardLookAt</td>
<td></td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a LookAt defined. (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage (Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>
Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly. (Overrides CinemachineComponentBaseMutateSingle.)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position dragged from the user. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLookAt Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineHardLookAt.#ctor"]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td>CinemachineHardLookAt()</td>
</tr>
</tbody>
</table>

### See Also

- Reference  
  CinemachineHardLookAt Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineHardLookAt Properties

The CinemachineHardLookAt type exposes the following members.

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a LookAt defined (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage (Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>
VirtualCamera

Get the associated
CinemachineVirtualCameraBase
(Inherited from
CinemachineComponentBase.)

See Also

Reference
CinemachineHardLookAt Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLookAtIsValid Property

True if component is enabled and has a LookAt defined

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public override bool IsValid { get; }
```

Property Value
Type: Boolean

See Also

Reference
CinemachineHardLookAt Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineHardLookAtStage Property

Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

C#  JavaScript

```csharp
public override CinemachineCoreStage Stage { get; }
```

Property Value
Type: CinemachineCoreStage

See Also

Reference
  CinemachineHardLookAt Class
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineHardLookAt Methods

The CinemachineHardLookAt type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies the composer rules and orients the camera accordingly (Overrrides CinemachineComponentBaseMutateSingle().)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

Top

## See Also

Reference

CinemachineHardLookAt Class

Cinemachine Namespace
Cinemachine
CinemachineHardLookAtMutateCamera Method

Applies the composer rules and orients the camera accordingly

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override void MutateCameraState(  
    ref CameraState curState,  
    float deltaTime
)
```

### Parameters

- **curState**  
  Type: CinemachineCameraState  
  The current camera state  

- **deltaTime**  
  Type: System.Single  
  Used for calculating damping. If less than zero, then target will snap to the center of the dead zone.

### See Also

- Reference  
  CinemachineHardLookAt Class  
  Cinemachine Namespace

Visit the Cinemachine Forum
CinemachineMixingCamera Class

CinemachineMixingCamera is a "manager camera" that takes on the state of the weighted average of the states of its child virtual cameras. A fixed number of slots are made available for cameras, rather than a dynamic array. We do it this way in order to support weight animation from the Timeline. Timeline cannot animate array elements.

Inheritance Hierarchy

```
System
  Object
  Component
    Behaviour
      MonoBehaviour
        Cinemachine
          CinemachineVirtualCameraBase
            CinemachineMixingCamera
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[DocumentationSortingAttribute(20f, DocumentationAttribute]
[ExecuteInEditMode]
[DisallowMultipleComponent]
[AddComponentMenu("Cinemachine/CinemachineMixingCamera")]
public class CinemachineMixingCamera : Cinemachine
```

The CinemachineMixingCamera type exposes the following members.
### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineMixingCamera</td>
<td></td>
</tr>
</tbody>
</table>

### Top

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>Get the cached list of child camera, just the immediate children in the hierarchy. Only the first entries of this list participate in the final blend, up to MaxCameras.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description of the virtual camera, for use when displaying debug information. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>Not used (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Not used (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Implementation returns the owner GameObject's name. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
**Support for meta-virtual-cameras.**

In a situation where a virtual camera is the public face of a private army of virtual cameras, which it manages on its own, the VirtualCamera owner, if any, are implemented as Transform children of the parent vcam. (Inherited from `CinemachineVirtualCameraBase`)

**PreviousStateIsValid**

Set this to force the next update of deltaTime and reset itself. (Inherited from `CinemachineVirtualCameraBase`)

**Priority**

Get the Priority of the virtual camera. It determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from `CinemachineVirtualCameraBase`)

**State**

The blended CameraState. (Overrides `CinemachineVirtualCameraBase`)

**ValidatingStreamVersion**

Version that was last streamed, for upgrading legacy. (Inherited from `CinemachineVirtualCameraBase`)

**VirtualCameraGameObject**

The GameObject owner of the behaviour. (Inherited from `CinemachineVirtualCameraBase`)

---

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AddPostPipelineStageHook</strong></td>
<td>A delegate to hook into the state calculation pipeline. See <code>CinemachineCore.Stage</code>. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
</tbody>
</table>
are updating LateUpdate instead.

RemovePostPipelineStageHook

Remove a Pipeline stage hook callback.

(Inherited from CinemachineRemovePostPipelineStageHook)

ResolveFollow

Returns this vcam’s Follow target, or if that is null, will return

(Inherited from CinemachineResolveFollow)

ResolveLookAt

Returns this vcam’s LookAt target, or if that is null, will return

(Inherited from CinemachineResolveLookAt)

SetWeight(Int32, Single)

Set the weight of the child at an index.

SetWeight(CinemachineVirtualCameraBase, Single)

Set the weight of the child CinemachineVirtualCameraBase.

Start

Base class implementation does nothing.

(Inherited from CinemachineStart)

Update

Base class implementation makes sure the priority queue remains up-to-date.

(Inherited from CinemachineUpdate)

UpdateCameraState

Called by CinemachineCore at designated update time computes and caches the weighted blend of the tracked cameras.

(Overridden from CinemachineUpdateCameraState)

ValidateListOfChildren

Rebuild the cached list of child cameras.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use for hiding sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Use for enabling sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_Weight0</td>
<td>Weight of the first tracked camera</td>
</tr>
<tr>
<td>m_Weight1</td>
<td>Weight of the second tracked camera</td>
</tr>
<tr>
<td>m_Weight2</td>
<td>Weight of the third tracked camera</td>
</tr>
<tr>
<td>m_Weight3</td>
<td>Weight of the fourth tracked camera</td>
</tr>
<tr>
<td>m_Weight4</td>
<td>Weight of the fifth tracked camera</td>
</tr>
<tr>
<td>m_Weight5</td>
<td>Weight of the sixth tracked camera</td>
</tr>
</tbody>
</table>

Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)
<table>
<thead>
<tr>
<th>m_Weight6</th>
<th>Weight of the seventh tracked camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Weight7</td>
<td>Weight of the eighth tracked camera</td>
</tr>
<tr>
<td>MaxCameras</td>
<td>The maximum number of cameras. If you want to add cameras, do it here in code, and be sure to add extra member variables and make the appropriate changes in <code>GetWeight()</code> and <code>SetWeight()</code>. The inspector will figure it out based on this value.</td>
</tr>
</tbody>
</table>

| OnPostPipelineStage | A delegate to hook into the calculation pipeline. Implementation must be called after each pipeline stage to allow other services to hook into the pipeline. See `CinemachineCore.Stage`. (Inherited from `CinemachineVirtualCameraBase`). |

### Top

#### See Also

**Reference**

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineMixingCamera.#ctor"]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

#### Syntax

**C#**

```csharp
public CinemachineMixingCamera()
```

**JavaScript**

#### See Also

- Reference  
  CinemachineMixingCamera Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
# CinemachineMixingCamera Properties

The `CinemachineMixingCamera` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>Get the cached list of child cameras. Only the first entries of this list participate in the final blend, up to MaxCameras.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug information. (Inherited from <code>CinemachineVirtualCameraBase</code>).</td>
</tr>
<tr>
<td>Follow</td>
<td>Not used (Overrides <code>CinemachineVirtualCameraBase</code>).</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides <code>CinemachineVirtualCameraBase</code>).</td>
</tr>
<tr>
<td>LookAt</td>
<td>Not used (Overrides <code>CinemachineVirtualCameraBase</code>).</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Implementation returns the owner GameObject's name. (Inherited from <code>CinemachineVirtualCameraBase</code>).</td>
</tr>
<tr>
<td><strong>ParentCamera</strong></td>
<td>Support for meta-virtual-camera situation where a virtual camera is the public face of a private army of virtual cameras, of which it manages on its own. The VirtualCamera owner, if any, are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>PreviousStateIsValid</strong></td>
<td>Set this to force the next update to ignore deltaTime and reset itself. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>Get the Priority of the virtual camera. It determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>The blended CameraState. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ValidatingStreamVersion</strong></td>
<td>Version that was last streamed. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>VirtualCameraGameObject</strong></td>
<td>The GameObject owner of the behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

## See Also

Reference
CinemachineMixingCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
CinemachineMixingCameraChildCameras Property

Get the cached list of child cameras. These are just the immediate children in the hierarchy. Note: only the first entries of this list participate in the final blend, up to MaxCameras.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public CinemachineVirtualCameraBase[] ChildCameras
```

Property Value

Type: CinemachineVirtualCameraBase

### See Also

**Reference**
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCameraFollow

Property

Not used

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public override Transform Follow { get; set; }
```

Property Value  
Type: **Transform**  
Implements  
**ICinemachineCameraFollow**

### See Also

Reference  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraLiveChildOrSelf

Property

Return the live child.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript

```csharp
public override ICinemachineCamera LiveChildOrSelf
```

Property Value

Type: ICinemachineCamera

Implements

ICinemachineCameraLiveChildOrSelf

See Also

Reference

CinemachineMixingCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera.LookAt

Property

Not used

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

🔹 Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>public override Transform LookAt { get; set; }</strong></td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: **Transform**
Implements
ICinemachineCameraLookAt

🔹 See Also

**Reference**
CinemachineMixingCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraState

Property

The blended CameraState

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

C#  

```csharp
public override CameraState State { get; }
```

### Property Value

Type: `CameraState`

**Implements**

`ICinemachineCameraState`

### See Also

**Reference**

- **CinemachineMixingCamera Class**
- **Cinemachine Namespace**

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera Methods

The CinemachineMixingCamera type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage. (Inherited from CinemachineBase)</td>
</tr>
<tr>
<td>GetWeight(Int32)</td>
<td>Get the weight of the child at an index.</td>
</tr>
<tr>
<td>GetWeight(CinemachineVirtualCameraBase)</td>
<td>Get the weight of the child CinemachineVirtualCameraBase.</td>
</tr>
<tr>
<td>InvalidateListOfChildren</td>
<td>Invalidate the cached list of child cameras.</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all. (Inherited from CinemachineBase)</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is a live child of this camera.</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is a live camera if the most recent one goes to the top of the priority subqueue, and its peers share the highest priority, then this vcam will become live. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from CinemachineBase)</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.  (Inherited from CinemachineBase)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Makes sure the internal child cache is up to date.</td>
</tr>
<tr>
<td>OnTransformChildrenChanged</td>
<td>Makes sure the internal child cache is up to date.</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live.</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Makes sure the weights are non-negative.</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of the vcam's child cameras.</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>LateUpdate instead, if the child cameras are updating on FixedUpdate, then this will not necessarily be called.</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook callback.</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcam's Follow target, or if that is null, will return.</td>
</tr>
<tr>
<td>ResolveLookAt</td>
<td>Returns this vcam's LookAt target, or if that is null, will return.</td>
</tr>
<tr>
<td>SetWeight(Int32, Single)</td>
<td>Set the weight of the child at an index.</td>
</tr>
<tr>
<td>SetWeight(CinemachineVirtualCameraBase, Singe)</td>
<td>Set the weight of the child CinemachineVirtualCameraBase.</td>
</tr>
<tr>
<td>Start</td>
<td>Base class implementation does nothing.</td>
</tr>
<tr>
<td>Update</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
</tbody>
</table>
### See Also

**Reference**
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraGetWeight Method

### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetWeight(Int32)</td>
<td>Get the weight of the child at an index.</td>
</tr>
<tr>
<td>GetWeight(CinemachineVirtualCameraBase)</td>
<td>Get the weight of the child CinemachineVirtualCameraBase.</td>
</tr>
</tbody>
</table>

### See Also

**Reference**
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraGetWeight Method (Int32)

Get the weight of the child at an index.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

## Syntax

### C#

```csharp
public float GetWeight(
    int index
)
```

### JavaScript

```javascript
// Not available in this context
```

## Parameters

**index**

Type: `System.Int32`
The child index. Only immediate CinemachineVirtualCameraBase children are counted.

## Return Value

Type: `Single`
The weight of the camera. Valid only if camera is active and enabled.

## See Also

- Reference
  - CinemachineMixingCamera Class
  - GetWeight Overload
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraGetWeight Method (CinemachineVirtualCameraBase)

Get the weight of the child CinemachineVirtualCameraBase.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public float GetWeight(
    CinemachineVirtualCameraBase vcam
)
```

**JavaScript**

```javascript
// Not applicable
```

### Parameters

**vcam**

Type: CinemachineVirtualCameraBase

The child camera.

### Return Value

Type: Single

The weight of the camera. Valid only if camera is active and enabled.

### See Also

**Reference**

CinemachineMixingCamera Class  
GetWeight Overload  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraInvalidate Method

Invalidate the cached list of child cameras.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected void InvalidateListOfChildren()</code></td>
<td></td>
</tr>
</tbody>
</table>

**See Also**

Reference  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCamera.IsLiveChild Method

Check whether the vcam a live child of this camera.

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override bool IsLiveChild(ICinemachineCamera vcam)
```

**Parameters**

*vcam*
Type: **CinemachineICinemachineCamera**
The Virtual Camera to check

**Return Value**
Type: **Boolean**
True if the vcam is currently actively influencing the state of this vcam

**Implements**
*CinemachineICinemachineCamera.IsLiveChild(IICinemachineCamera)*

**See Also**

Reference
*CinemachineMixingCamera Class*  
*Cinemachine Namespace*
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraOnEnable Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
protected override void OnEnable()
```

**See Also**

Reference
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraOnTransformChildrenChanged Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

C#  
```csharp
public void OnTransformChildrenChanged()
```

**See Also**

Reference  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraOnValidate Method

Makes sure the weights are non-negative

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected override void OnValidate()</code></td>
<td></td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCameraRemove Method

Remove a Pipeline stage hook callback. Make sure it is removed from all the children.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public override void RemovePostPipelineStageHook(
    CinemachineVirtualCameraBaseOnPostPipelineDelegate d
)
```

Parameters

\(d\)

Type: CinemachineVirtualCameraBaseOnPostPipelineDelegate

The delegate to remove.

See Also

Reference
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCameraSetWeight Method

### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌲 SetWeight(Int32, Single)</td>
<td>Set the weight of the child at an index.</td>
</tr>
<tr>
<td>🌲 SetWeight(CinemachineVirtualCameraBase, Single)</td>
<td>Set the weight of the child CinemachineVirtualCameraBase.</td>
</tr>
</tbody>
</table>

**Top**

### See Also

**Reference**
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCameraSetWeight Method (Int32, Single)

Set the weight of the child at an index.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public void SetWeight(
    int index,
    float w
)
```

### Parameters

- **index**
  - **Type:** SystemInt32  
  - The child index. Only immediate CinemachineVirtualCameraBase children are counted.

- **w**
  - **Type:** SystemSingle  
  - The weight to set. Can be any non-negative number.

## See Also

- **Reference**  
  - CinemachineMixingCamera Class  
  - SetWeight Overload  
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraSetWeight Method
(CinemachineVirtualCameraBase, Single)

Set the weight of the child CinemachineVirtualCameraBase.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public void SetWeight(
    CinemachineVirtualCameraBase vcam,
    float w
)
```

### Parameters

- **vcam**  
  Type: CinemachineVirtualCameraBase  
  The child camera.

- **w**  
  Type: System.Single  
  The weight to set. Can be any non-negative number.

### See Also

Reference  
CinemachineMixingCamera Class
SetWeight Overload
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraUpdateCameraMethod

 Called by CinemachineCore at designated update time so the vcam can position itself and track its targets. This implementation computes and caches the weighted blend of the tracked cameras.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

### Parameters

- **worldUp**
  - Type: `Vector3`  
  - Default world Up, set by the CinemachineBrain

- **deltaTime**
  - Type: `System.Single`  
  - Delta time for time-based effects (ignore if less than 0)

### Implements

ICinemachineCameraUpdateCameraState(Vector3, Single)

### See Also

Reference
CinemachineMixingCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameraValidateListOfChildren Method

Rebuild the cached list of child cameras.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
protected void ValidateListOfChildren()
```

**See Also**

Reference  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
## CinemachineMixingCamera Fields

The CinemachineMixingCamera type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="CinemachineGUIDebuggerCallback" /></td>
<td><strong>This is deprecated. It support the soon-to-be-removed Cinemachine Debugger in the Editor.</strong> (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><img src="image" alt="m_ExcludedPropertiesInInspector" /></td>
<td>Inspector control - Use sections of the Inspector. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><img src="image" alt="m_LockStageInInspector" /></td>
<td>Inspector control - Use enabling sections of the UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><img src="image" alt="m_Priority" /></td>
<td>The priority will determine which camera becomes active based on the state of other cameras. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>m_Weight0</strong></td>
<td>Weight of the first tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight1</strong></td>
<td>Weight of the second tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight2</strong></td>
<td>Weight of the third tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight3</strong></td>
<td>Weight of the fourth tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight4</strong></td>
<td>Weight of the fifth tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight5</strong></td>
<td>Weight of the sixth tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight6</strong></td>
<td>Weight of the seventh tracked camera</td>
</tr>
<tr>
<td><strong>m_Weight7</strong></td>
<td>Weight of the eighth tracked camera</td>
</tr>
<tr>
<td><strong>MaxCameras</strong></td>
<td>The maximum number of cameras. If you want to add cameras, do it here in the code, and be sure to add extra member variables and make the appropriate changes in GetWeight() and SetWeight(). The inspector will figure it out based on this value.</td>
</tr>
<tr>
<td><strong>OnPostPipelineStage</strong></td>
<td>A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage.</td>
</tr>
</tbody>
</table>
See Also

Reference
CinemachineMixingCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCamera

Field

Weight of the first tracked camera

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[TooltipAttribute("The weight of the first tracked camera")]
public float m_Weight0` | |

Field Value

Type: **Single**

### See Also

**Reference**
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCameram_Weight Field

Weight of the second tracked camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[TooltipAttribute("The weight of the second track")
public float m_Weight1
```

### Field Value

Type: **Single**

### See Also

- **Reference**  
  - CinemachineMixingCamera Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera Field

Weight of the third tracked camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
public float m_Weight2
```

Field Value  
Type: Single

**See Also**

Reference  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera_Cameram_Weight

Field

Weight of the fourth tracked camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[TooltipAttribute("The weight of the fourth track")]
public float m_Weight3 |            |

Field Value

Type: Single

### See Also

**Reference**

- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera
Field

Weight of the fifth tracked camera

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[TooltipAttribute("The weight of the fifth tracked camera")]
public float m_Weight4
```

Field Value
Type: **Single**

### See Also

**Reference**
- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera\_Weight5 Field

Weight of the sixth tracked camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `[TooltipAttribute("The weight of the sixth tracked camera")]
public float m_Weight5` |            |

Field Value  
Type: Single

**See Also**

Reference
- CinemachineMixingCamera Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineMixingCamera.m_Weight6

Field

Weight of the seventh tracked camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="#" alt="Syntax" /></td>
<td><img src="#" alt="Syntax" /></td>
</tr>
</tbody>
</table>

Field Value  
Type: Single

### See Also

**Reference**  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
**CinemachineMixingCamera**

**Field**

Weight of the eighth tracked camera

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

---

**Syntax**

```csharp
[TooltipAttribute("The weight of the eighth tracked camera")]
public float m_Weight7
```

```javascript
// No equivalent in JavaScript
```

---

**Field Value**

Type: Single

---

**See Also**

**Reference**

- CinemachineMixingCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineMixingCameraMaxCameras Field

The maximum number of tracked cameras. If you want to add more cameras, do it here in the source code, and be sure to add the extra member variables and to make the appropriate changes in GetWeight() and SetWeight(). The inspector will figure itself out based on this value.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public const int MaxCameras = 8</code></td>
<td></td>
</tr>
</tbody>
</table>

Field Value  
Type: Int32

**See Also**

Reference  
CinemachineMixingCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineOrbitalTransposer
Class

This is a CinemachineComponent in the Body section of the component pipeline. Its job is to position the camera in a variable relationship to a the vcam's Follow target object, with offsets and damping. This component is typically used to implement a camera that follows its target. It can accept player input from an input device, which allows the player to dynamically control the relationship between the camera and the target, for example with a joystick. The OrbitalTransposer introduces the concept of __Heading__, which is the direction in which the target is moving, and the OrbitalTransposer will attempt to position the camera in relationship to the heading, which is by default directly behind the target. You can control the default relationship by adjusting the Heading Bias setting. If you attach an input controller to the OrbitalTransposer, then the player can also control the way the camera positions itself in relation to the target heading. This allows the camera to move to any spot on an orbit around the target.

▶ Inheritance Hierarchy

- SystemObject  Object
  - Component
    - Behaviour
      - MonoBehaviour
        - CinemachineCinemachineComponentBase
          - CinemachineCinemachineTransposer
            - CinemachineCinemachineOrbitalTransposer

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)
▶ Syntax
The **CinemachineOrbitalTransposer** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AngularDamping</strong></td>
<td>Damping speeds for each of the 3 axes of the target's rotation (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>Damping</strong></td>
<td>Damping speeds for each of the 3 axes of the offset from target (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>EffectiveOffset</strong></td>
<td>Get the target offset, with sanitization (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a valid Follow target (Inherited from CinemachineTransposer.)</td>
</tr>
</tbody>
</table>
### LookAtTarget
Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)

### Stage
Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage (Inherited from CinemachineTransposer.)

### VcamState
Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)

### VirtualCamera
Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)

---

#### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTargetCameraPosition</td>
<td>Internal API for the Inspector Ed marker at the target (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>GetReferenceOrientation</td>
<td>Internal API for the Inspector Ed marker at the target (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>InitPrevFrameStateInfo</td>
<td>Initializes the state for previous 1...</td>
</tr>
</tbody>
</table>
MutateCameraState

Positions the virtual camera according to the transposer rules.
(Overrides CinemachineTransposerMutateCameraState(Single).)

OnPositionDragged

API for the editor, to process a position drag from the user.
This implementation adds the delta to the follow offset, after zeroing out local x.
(Overrides CinemachineTransposerOnPositionDragged().)

OnValidate

(Overrides CinemachineTransposerOnValidate().)

TrackTarget

Positions the virtual camera according to the transposer rules.
(Inherited from CinemachineTransposer.TrackTarget.)

UpdateHeading

When in slave mode, this should be called once every frame to update the heading.
In slave mode, this is called automatically.

Top

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BindingMode</td>
<td>The coordinate space to use when interpreting the offset from the target (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>m_FollowOffset</td>
<td>The distance which the transposer will attempt to maintain from the target</td>
</tr>
</tbody>
</table>
transposer subject
(Inherited from CinemachineTransposer.)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Heading</td>
<td>The definition of Forward. Camera will follow behind.</td>
</tr>
<tr>
<td>m_HeadingIsSlave</td>
<td>Drive the x-axis setting programmatically. Automatic heading updating will be disabled.</td>
</tr>
<tr>
<td>m_PitchDamping</td>
<td>How aggressively the camera tries to track the target rotation's X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>m_RecenterToTargetHeading</td>
<td>Parameters that control Automating Heading Recentering</td>
</tr>
<tr>
<td>m_RollDamping</td>
<td>How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>m_XAxis</strong></td>
<td>Axis representing the current heading. Value is in degrees and represents a rotation about the up vector.</td>
</tr>
<tr>
<td><strong>m_XDamping</strong></td>
<td>How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>m_YawDamping</strong></td>
<td>How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>m_YDamping</strong></td>
<td>How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly</td>
</tr>
</tbody>
</table>
translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors (Inherited from CinemachineTransposer.)

<table>
<thead>
<tr>
<th>m_ZDamping</th>
</tr>
</thead>
<tbody>
<tr>
<td>How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors (Inherited from CinemachineTransposer.)</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineOrbitalTransposer Properties

The CinemachineOrbitalTransposer type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AngularDamping</td>
<td>Damping speeds for each of the 3 axes of the target's rotation (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>Damping</td>
<td>Damping speeds for each of the 3 axes of the offset from target (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>EffectiveOffset</td>
<td>Get the target offset, with sanitization (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a valid Follow target (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>VirtualCamera</td>
<td>Get the associated CinemachineVirtualCameraBase. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**
- CinemachineOrbitalTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The `CinemachineOrbitalTransposer` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GetTargetCameraPosition</code></td>
<td>Internal API for the Inspector Editor, so it can draw a marker at the target (Inherited from <code>CinemachineTransposer</code>).</td>
</tr>
<tr>
<td><code>GetReferenceOrientation</code></td>
<td>Internal API for the Inspector Editor, so it can draw a marker at the target (Inherited from <code>CinemachineTransposer</code>).</td>
</tr>
<tr>
<td><code>InitPrevFrameStateInfo</code></td>
<td>Initializes the state for previous frame if appropriate. (Inherited from <code>CinemachineTransposer</code>).</td>
</tr>
<tr>
<td><code>MutateCameraState</code></td>
<td>Positions the virtual camera according to the transposer rules. (Overrides <code>CinemachineTransposerMutateCameraState</code>.)</td>
</tr>
<tr>
<td><code>OnPositionDragged</code></td>
<td>API for the editor, to process a position dragged from the user. This implementation adds the delta to the follow offset, after zeroing out local x. (Overrides <code>CinemachineTransposerOnPositionDragged</code>.)</td>
</tr>
<tr>
<td><code>OnValidate</code></td>
<td>(Overrides <code>CinemachineTransposerOnValidate</code>).</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>TrackTarget</strong></td>
<td>Positions the virtual camera according to the transposer rules. (Inherited from CinemachineTransposer)</td>
</tr>
<tr>
<td><strong>UpdateHeading</strong></td>
<td>When in slave mode, this should be called once every frame to update the heading. In slave mode, this is called automatically.</td>
</tr>
</tbody>
</table>

## See Also

- **Reference**
  - CinemachineOrbitalTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineOrbitalTransposerMutateCameraState Method

Positions the virtual camera according to the transposer rules.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public override void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)
```

## Parameters

- **curState**
  - Type: CinemachineCameraState
  - The current camera state
- **deltaTime**
  - Type: System.Single
  - Used for damping. If less than 0, no damping is done.

## See Also

- **Reference**
  - CinemachineOrbitalTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerOnPositionDragged Method

API for the editor, to process a position drag from the user. This implementation adds the delta to the follow offset, after zeroing out local x.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

_C#_  
```csharp
public override void OnPositionDragged(
    Vector3 delta
)
```

### Parameters

*delta*  
Type: _Vector3_

The amount dragged this frame

### See Also

- Reference  
  CinemachineOrbitalTransposer Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerOnValidate Method


**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected override void OnValidate()</code></td>
<td></td>
</tr>
</tbody>
</table>

### See Also

**Reference**  
CinemachineOrbitalTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerUpdate Method

When in slave mode, this should be called once and only once every frame to update the heading. When not in slave mode, this is called automatically.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public void UpdateHeading(
    float deltaTime,
    Vector3 up
)
```

### Parameters

**deltaTime**  
Type: `System.Single`  

**up**  
Type: `Vector3`  
[Missing <param name="up"/> documentation for "M:Cinemachine.CinemachineOrbitalTransposer.UpdateHeading(System.Single,Vector3)"

### See Also

Reference
CinemachineOrbitalTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The `CinemachineOrbitalTransposer` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BindingMode</td>
<td>The coordinate space to use when interpreting the offset from the target. (Inherited from <code>CinemachineTransposer</code>.)</td>
</tr>
<tr>
<td>m_FollowOffset</td>
<td>The distance which the transposer will attempt to maintain from the transposer subject. (Inherited from <code>CinemachineTransposer</code>.)</td>
</tr>
<tr>
<td>m_Heading</td>
<td>The definition of Forward. Camera will follow behind.</td>
</tr>
<tr>
<td>m_HeadingIsSlave</td>
<td>Drive the x-axis setting programmatically. Automatic heading updating will be disabled.</td>
</tr>
<tr>
<td>m_PitchDamping</td>
<td>How aggressively the camera tries to track the target rotation's X angle.</td>
</tr>
</tbody>
</table>
Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. (Inherited from CinemachineTransposer.)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_RecenterToTargetHeading</strong></td>
<td>Parameters that control Automating Heading Recentering</td>
</tr>
<tr>
<td><strong>m_RollDamping</strong></td>
<td>How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td><strong>m_XAxis</strong></td>
<td>Axis representing the current heading. Value is in degrees and represents a rotation about the up vector</td>
</tr>
<tr>
<td><strong>m_XDamping</strong></td>
<td>How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
</tbody>
</table>
Using different settings per axis can yield a wide range of camera behaviors (Inherited from CinemachineTransposer.)

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_YawDamping</td>
<td>How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>m_YDamping</td>
<td>How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors (Inherited from CinemachineTransposer.)</td>
</tr>
<tr>
<td>m_ZDamping</td>
<td>How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more</td>
</tr>
</tbody>
</table>
responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors (Inherited from CinemachineTransposer.)

See Also

Reference
CinemachineOrbitalTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineOrbitalTransposerm_Heading Field

The definition of Forward. Camera will follow behind.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**  
```csharp
[SpaceAttribute]
[TooltipAttribute("The definition of Forward. Camera will follow behind.")]
public CinemachineOrbitalTransposerHeading m_Heading
```

**JavaScript**

### Field Value

Type: CinemachineOrbitalTransposerHeading

### See Also

**Reference**  
- CinemachineOrbitalTransposer Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposer.m_HeadingIsSlave Field

Drive the x-axis setting programmatically. Automatic heading updating will be disabled.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[HideInInspector]
[NoSaveDuringPlayAttribute]
public bool m_HeadingIsSlave
```  

Field Value  
Type: Boolean

**See Also**

Reference  
CinemachineOrbitalTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposer\_Recentering

Field

Parameters that control Automating Heading Recentering

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Automatic heading recentering.
public CinemachineOrbitalTransposerRecentering m_
``` |

Field Value  
Type: CinemachineOrbitalTransposerRecentering

**See Also**

**Reference**  
CinemachineOrbitalTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineOrbitalTransposer.m_XAxis

Field

Axis representing the current heading. Value is in degrees and represents a rotation about the up vector

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Heading Control. The settings
public AxisState m_XAxis
``` | ![Copy](https://unity3d.com/legal/terms-of-service) |

### Field Value

Type: **AxisState**

### See Also

#### Reference

- CinemachineOrbitalTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerHeading Structure

How the "forward" direction is defined. Orbital offset is in relation to the forward direction.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▲ Syntax

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(6.2f, DocumentationSortingAttribute.Incremental)]
public struct Heading
```

The CinemachineOrbitalTransposerHeading type exposes the following members.

▲ Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚡ CinemachineOrbitalTransposerHeading</td>
<td>Constructor</td>
</tr>
</tbody>
</table>

▲ Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚡ m_HeadingBias</td>
<td>Additional Y rotation</td>
</tr>
</tbody>
</table>
applied to the target heading. When this value is 0, the camera will be placed behind the target.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_HeadingDefinition</td>
<td>The method by which the 'default heading' is calculated if recentering to target heading is enabled.</td>
</tr>
<tr>
<td>m_VelocityFilterStrength</td>
<td>Size of the velocity sampling window for target heading filter. Used only if deriving heading from target's movement.</td>
</tr>
</tbody>
</table>

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerHeading Constructor

Constructor

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
<tbody>
<tr>
<td>public Heading(</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CinemachineOrbitalTransposerHeadingHeadingDefinition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>int filterStrength,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>float bias</td>
<td></td>
<td></td>
</tr>
<tr>
<td>)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Parameters

**def**

Type: Cinemachine.CinemachineOrbitalTransposerHeadingHeadingDefinition  
[Missing <param name="def"/> documentation for  
"M:Cinemachine.CinemachineOrbitalTransposer.Heading.#ctor(Cinemachine.Cine]}

**filterStrength**

Type: System.Int32  
[Missing <param name="filterStrength"/> documentation for  
"M:Cinemachine.CinemachineOrbitalTransposer.Heading.#ctor(Cinemachine.Cine]}

**bias**

Type: System.Single  
[Missing <param name="bias"/> documentation for  
See Also

Reference
CinemachineOrbitalTransposerHeading Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# Heading Fields

The `CinemachineOrbitalTransposerHeading` type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_HeadingBias</td>
<td>Additional Y rotation applied to the target heading. When this value is 0, the camera will be placed behind the target.</td>
</tr>
<tr>
<td>m_HeadingDefinition</td>
<td>The method by which the 'default heading' is calculated if recentering to target heading is enabled.</td>
</tr>
<tr>
<td>m_VelocityFilterStrength</td>
<td>Size of the velocity sampling window for target heading filter. Used only if deriving heading from target's movement.</td>
</tr>
</tbody>
</table>

## See Also

Reference
- `CinemachineOrbitalTransposerHeading Structure`
- `Cinemachine Namespace`
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerHeading Field

Additional Y rotation applied to the target heading. When this value is 0, the camera will be placed behind the target.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
[RangeAttribute(-180f, 180f)]
[TooltipAttribute("Where the camera is placed when the X-axis value is zero. This is a rotation in degrees around the target.")]
public float m_HeadingBias
```

### Field Value

Type: Single

### See Also

**Reference**

CinemachineOrbitalTransposerHeading Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineOrbitalTransposerHeading Field

The method by which the 'default heading' is calculated if recentering to target heading is enabled

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;How 'forward' is defined. The camera will be placed by default behind the target. PositionDelta will consider 'forward' to be the direction in which the target is moving. &quot;)]</td>
<td></td>
</tr>
</tbody>
</table>

Field Value
Type: CinemachineOrbitalTransposerHeadingHeadingDefinition

See Also

Reference
- CinemachineOrbitalTransposerHeading Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerHeading Field

Size of the velocity sampling window for target heading filter. Used only if deriving heading from target's movement

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[RangeAttribute(0f, 10f)]
[TooltipAttribute("Size of the velocity sampling window for target heading filter. This filters out irregularities in the target's movement. Used only if deriving heading from target's movement (PositionDelta or Velocity")]
public int m_VelocityFilterStrength
```

**JavaScript**

```javascript
// (JavaScript code)
```

### Field Value

Type: **Int32**

### See Also

**Reference**

- CinemachineOrbitalTransposerHeading Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerHeader

Enumeration

Sets the algorithm for determining the target’s heading for purposes of re-centering the camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[DocumentationSortingAttribute(6.21f, DocumentationSortingAttribute)]
public enum HeadingDefinition
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PositionDelta</td>
<td>0</td>
<td>Target heading calculated from the difference between its position on last update and current frame.</td>
</tr>
<tr>
<td>Velocity</td>
<td>1</td>
<td>Target heading calculated from its Rigidbody’s velocity. If no Rigidbody exists, it will fall back to HeadingDerivationMode.Position.</td>
</tr>
<tr>
<td>TargetForward</td>
<td>2</td>
<td>Target heading calculated from the Target Transform’s euler Y angle.</td>
</tr>
</tbody>
</table>
Default heading is a constant workspace heading.

See Also

Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerRecentering Structure

Controls how automatic orbit recentering occurs

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(6.5f, Documentatic
public struct Recentering
```

The **CinemachineOrbitalTransposerRecentering** type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔄 <img src="image" alt="icon" /> Validate</td>
<td>Call this from OnValidate()</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔄 <img src="image" alt="icon" /> m_enabled</td>
<td>If checked, will enable automatic recentering of</td>
</tr>
</tbody>
</table>
the camera based on the heading calculation mode. If FALSE, recenting is disabled.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_RecenteringTime</td>
<td>Maximum angular speed of recenting. Will accelerate into and decelerate out of this</td>
</tr>
<tr>
<td>m_RecenterWaitTime</td>
<td>If no input has been detected, the camera will wait this long in seconds before moving its heading to the default heading.</td>
</tr>
</tbody>
</table>

See Also

Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
Recentering Methods

The CinemachineOrbitalTransposerRecentering type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validate</td>
<td>Call this from OnValidate()</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineOrbitalTransposerRecentering Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineOrbitalTransposerRecentering Method

Call this from OnValidate()

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠ Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
<th>Copy</th>
</tr>
</thead>
<tbody>
<tr>
<td>public void Validate()</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

⚠ See Also

Reference
CinemachineOrbitalTransposerRecentering Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
# Recentering Fields

The `CinemachineOrbitalTransposer`Recentering type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>m_enabled</code></td>
<td>If checked, will enable automatic recentering of the camera based on the heading calculation mode. If FALSE, recenting is disabled.</td>
</tr>
<tr>
<td><code>m_RecenteringTime</code></td>
<td>Maximum angular speed of recentering. Will accelerate into and decelerate out of this</td>
</tr>
<tr>
<td><code>m_RecenterWaitTime</code></td>
<td>If no input has been detected, the camera will wait this long in seconds before moving its heading to the default heading.</td>
</tr>
</tbody>
</table>

## See Also

Reference

- `CinemachineOrbitalTransposerRecentering` Structure
- `Cinemachine` Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerRecentering Field

If checked, will enable automatic recentering of the camera based on the heading calculation mode. If FALSE, recentering is disabled.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

![TooltipAttribute("If checked, will enable automatic recentering of the camera based on the heading calculation mode. If unchecked, recentering is disabled.

```csharp
public bool m_enabled
```

**Field Value**  
Type: **Boolean**

**See Also**

**Reference**  
CinemachineOrbitalTransposerRecentering Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerRecentering Field

Maximum angular speed of recentering. Will accelerate into and decelerate out of this

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("Maximum angular speed of recentering. Will accelerate into and
decelerate out of this.")]
public float m_RecenteringTime
```

Field Value  
Type: Single

### See Also

Reference  
CinemachineOrbitalTransposerRecentering Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineOrbitalTransposerRecentering Field

If no input has been detected, the camera will wait this long in seconds before moving its heading to the default heading.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

**C#**

```csharp
[TooltipAttribute("If no input has been detected, the camera will wait this long in seconds before moving its heading to the default heading.")]
public float m_RecenterWaitTime
```

Field Value

Type: Single

See Also

Reference

CinemachineOrbitalTransposerRecentering Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePath Class

Defines a world-space path, consisting of an array of waypoints, each of which has position, tangent, and roll settings. Bezier interpolation is performed between the waypoints, to get a smooth and continuous path.

Inheritance Hierarchy

```
SystemObject  Object
  Component
    Behaviour
      Cinemachine
        CinemachinePathBase
        CinemachinePath
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(18f, DocumentationSortingAttribute
[AddComponentMenu("Cinemachine/CinemachinePath")]
[SaveDuringPlayAttribute]
public class CinemachinePath : CinemachinePathBase
```

The CinemachinePath type exposes the following members.

Constructors

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
```

Copy
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DistanceCacheSampleStepsPerSegment</strong></td>
<td>When calculating the distance cache, sample the path this many times between points. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>Looped</strong></td>
<td>True if the path ends are joined to form a continuous loop. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>MaxPos</strong></td>
<td>The maximum value for the path position. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>MinPos</strong></td>
<td>The minimum value for the path position. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>PathLength</strong></td>
<td>Get the length of the path in distance units. If it is not valid, then calling this will potentially cost the regeneration of the distance cache. (Inherited from CinemachinePathBase)</td>
</tr>
</tbody>
</table>

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DistanceCacheIsValid</strong></td>
<td>See whether the distance cache is valid, then any call to GetPathLength, GetPathPositionFromDistance will potentially costly regenerate the cache. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>EvaluateOrientation</td>
<td>Get the orientation the curve at a point along the path. (Overrides CinemachinePathBaseEvaluateOrientation)</td>
</tr>
<tr>
<td>EvaluateOrientationAtUnit</td>
<td>Get the orientation the curve at a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluatePosition</td>
<td>Get a worldspace position of a point along the path. (Overrides CinemachinePathBaseEvaluatePosition)</td>
</tr>
<tr>
<td>EvaluatePositionAtUnit</td>
<td>Get a worldspace position of a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluateTangent</td>
<td>Get the tangent of the curve at a point along the path. (Overrides CinemachinePathBaseEvaluateTangent)</td>
</tr>
<tr>
<td>EvaluateTangentAtUnit</td>
<td>Get the tangent of the curve at a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>FindClosestPoint</td>
<td>Find the closest point on the path to a given worldspace target point. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>GetPathDistanceFromPosition</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the path is not valid, then calling this will result in costly regeneration of the path distance cache. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>GetPathPositionFromDistance</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the path is not valid, then calling this will result in costly regeneration of the path distance cache. (Inherited from CinemachinePathBase)</td>
</tr>
</tbody>
</table>
### Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>InvalidateDistanceCache</strong></td>
<td>Call this if the path changes in such a way as to affect distances or other cached path elements.</td>
</tr>
<tr>
<td><strong>MaxUnit</strong></td>
<td>Get the maximum value, for the given unity type.</td>
</tr>
<tr>
<td><strong>MinUnit</strong></td>
<td>Get the minimum value, for the given unity type.</td>
</tr>
<tr>
<td><strong>NormalizePathDistance</strong></td>
<td>Normalize a distance along the path length. If the distance cache is not valid, calling this will trigger a potentially costly regeneration.</td>
</tr>
<tr>
<td><strong>NormalizePos</strong></td>
<td>Get a normalized path position, taking into account if looped.</td>
</tr>
<tr>
<td><strong>NormalizeUnit</strong></td>
<td>Normalize the unit, so that it lies between MinUnit and MaxUnit.</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_Appearance</strong></td>
<td>The settings that control how the path will appear in the editor scene view. (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td><strong>m_Looped</strong></td>
<td>If checked, then the path ends.</td>
</tr>
</tbody>
</table>
are joined to form a continuous loop

<table>
<thead>
<tr>
<th></th>
<th>m_Resolution</th>
<th>Path samples per waypoint (Inherited from CinemachinePathBase.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>m_Waypoints</td>
<td>The waypoints that define the path. They will be interpolated using a bezier curve</td>
</tr>
</tbody>
</table>

Top

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePath Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachinePath.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#

```csharp
public CinemachinePath()
```

JavaScript

```javascript
// Not applicable
```

See Also

Reference

CinemachinePath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachinePath Properties

The `CinemachinePath` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="Image" /> DistanceCacheSampleStepsPerSegment</td>
<td>When calculating the distance cache, sample the path this many times between points. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><img src="image.png" alt="Image" /> Looped</td>
<td>True if the path ends are joined to form a continuous loop. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><img src="image.png" alt="Image" /> MaxPos</td>
<td>The maximum value for the path position. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><img src="image.png" alt="Image" /> MinPos</td>
<td>The minimum value for the path position. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td><img src="image.png" alt="Image" /> PathLength</td>
<td>Get the length of the path in distance units. If the path is not valid, then calling this will regenerate the path distance cache. (Inherited from CinemachinePathBase)</td>
</tr>
</tbody>
</table>

### See Also

- Reference
  - `CinemachinePath Class`
  - `Cinemachine Namespace`
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathDistanceCacheSampleStepsPerSegment Property

When calculating the distance cache, sample the path this many times between points

Namespace:  Cinemachine  
Assembly:  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public override int DistanceCacheSampleStepsPerSegment</code></td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type:  Int32

See Also

Reference
CinemachinePath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathLooped Property

True if the path ends are joined to form a continuous loop

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public override bool Looped { get; }
```

## Property Value

Type: Boolean

## See Also

- Reference
  - CinemachinePath Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathMaxPos Property

The maximum value for the path position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override float MaxPos { get; }
```

**Property Value**

Type: Single

**See Also**

Reference
- CinemachinePath Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathMinPos Property

The minimum value for the path position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override float MinPos { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: **Single**

**See Also**

Reference
- CinemachinePath Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePath Methods

The **CinemachinePath** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheIsValid</td>
<td>See whether the distance cache is valid, then any call to GetPathLength() or GetPathPositionFromDistance() will potentially costy regenerate cache (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluateOrientation</td>
<td>Get the orientation the curve at a point along the path. (Overrides CinemachinePathBase.EvaluateOrientation(Single))</td>
</tr>
<tr>
<td>EvaluateOrientationAtUnit</td>
<td>Get the orientation the curve at a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluatePosition</td>
<td>Get a worldspace position of a point along the path. (Overrides CinemachinePathBase.EvaluatePosition(Single))</td>
</tr>
<tr>
<td>EvaluatePositionAtUnit</td>
<td>Get a worldspace position of a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluateTangent</td>
<td>Get the tangent of the curve at a point along the path. (Overrides CinemachinePathBase.EvaluateTangent(Single))</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>EvaluateTangentAtUnit</td>
<td>Get the tangent of the curve path.</td>
</tr>
<tr>
<td>FindClosestPoint</td>
<td>Find the closest point on the worldspace target point.</td>
</tr>
<tr>
<td>GetPathDistanceFromPosition</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>GetPathPositionFromDistance</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>InvalidateDistanceCache</td>
<td>Call this if the path changes in such a way as to affect distances or other cached path elements.</td>
</tr>
<tr>
<td>MaxUnit</td>
<td>Get the maximum value, for the given unity type.</td>
</tr>
<tr>
<td>MinUnit</td>
<td>Get the minimum value, for the given unity type.</td>
</tr>
<tr>
<td>NormalizePathDistance</td>
<td>Normalize a distance along path length. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>NormalizePos</td>
<td>Get a normalized path position, taking spins into account if looped.</td>
</tr>
<tr>
<td>NormalizeUnit</td>
<td>Normalize the unit, so that and MaxUnit (Inherited from Cinemachine)</td>
</tr>
</tbody>
</table>

See Also

- Reference
  - CinemachinePath Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathEvaluateOrientation Method

Get the orientation the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override Quaternion EvaluateOrientation(float pos)
```

**Parameters**

- `pos`  
  Type: System.Single  
  Position along the path. Need not be normalized.

**Return Value**  
Type: Quaternion  
World-space orientation of the path, as defined by tangent, up, and roll.

**See Also**

- Reference  
  CinemachinePath Class  
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePath\EvaluatePosition Method

Get a worldspace position of a point along the path

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
public override Vector3 EvaluatePosition(float pos)
```

**Parameters**

- **pos**
  Type: `System.Single`
  Position along the path. Need not be normalized.

**Return Value**

Type: `Vector3`
World-space position of the point along at path at `pos`

**See Also**

- Reference
  - CinemachinePath Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathEvaluateTangent Method

Get the tangent of the curve at a point along the path.

**Namespace**: Cinemachine  
**Assembly**: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public override Vector3 EvaluateTangent(
    float pos
)
```

**JavaScript**

```javascript
// Not applicable
```

### Parameters

**pos**

Type: `System.Single`  
Position along the path. Need not be normalized.

### Return Value

Type: `Vector3`  
World-space direction of the path tangent. Length of the vector represents the tangent strength

### See Also

Reference

- CinemachinePath Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePath Fields

The CinemachinePath type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Appearance</td>
<td>The settings that control how the path will appear in the editor scene view. (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td>m_Looped</td>
<td>If checked, then the path ends are joined to form a continuous loop</td>
</tr>
<tr>
<td>m_Resolution</td>
<td>Path samples per waypoint                                                  (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td>m_Waypoints</td>
<td>The waypoints that define the path. They will be interpolated using a bezier curve</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachinePath Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathm_Looped Field

If checked, then the path ends are joined to form a continuous loop

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("If checked, then the path ends are joined to form a continuous loop.")]
public bool m_Looped |

Field Value
Type: Boolean

See Also

Reference
CinemachinePath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathm_Waypoints

Field

The waypoints that define the path. They will be interpolated using a bezier curve

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[TooltipAttribute("The waypoints that define the path.	They will be interpolated using a bezier curve.")]
public CinemachinePathWaypoint[] m_Waypoints
```

Field Value

Type: CinemachinePathWaypoint

See Also

Reference

- CinemachinePath Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathWaypoint Structure

A waypoint along the path

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(18.2f, DocumentationSortingAttribute)
public struct Waypoint
```

The CinemachinePathWaypoint type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔷️ position</td>
<td>Position in path-local space</td>
</tr>
<tr>
<td>🔷️ roll</td>
<td>Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.</td>
</tr>
<tr>
<td>🔷️ tangent</td>
<td>Offset from the position, which defines the tangent of the curve at the waypoint. The length of the tangent encodes the strength of the bezier handle. The same handle is used symmetrically on both</td>
</tr>
</tbody>
</table>
sides of the waypoint, to ensure smoothness.
Waypoint Fields

The `CinemachinePathWaypoint` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>position</td>
<td>Position in path-local space</td>
</tr>
<tr>
<td>roll</td>
<td>Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.</td>
</tr>
<tr>
<td>tangent</td>
<td>Offset from the position, which defines the tangent of the curve at the waypoint. The length of the tangent encodes the strength of the bezier handle. The same handle is used symmetrically on both sides of the waypoint, to ensure smoothness.</td>
</tr>
</tbody>
</table>

See Also

Reference

- `CinemachinePathWaypoint Structure`
- `Cinemachine Namespace`

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathWaypointposition Field

Position in path-local space

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
[TooltipAttribute("Position in path-local space")]
public Vector3 position
```

### Field Value

**Type:** Vector3

### See Also

**Reference**

CinemachinePathWaypoint Structure

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachinePathWaypointrollof Field

Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.

Namespace: Cinemachine  
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
```csharp
[TooltipAttribute("Defines the roll of the path at this waypoint.
The other orientation axes are inferred from the tangent and world up.")]
public float roll
```

Field Value  
Type: Single

See Also

Reference  
- CinemachinePathWaypoint Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathWaypoint.tangent Field

Offset from the position, which defines the tangent of the curve at the waypoint. The length of the tangent encodes the strength of the bezier handle. The same handle is used symmetrically on both sides of the waypoint, to ensure smoothness.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Offset from the position, which defines the tangent of the curve at the waypoint. The length of the tangent encodes the strength of the bezier handle. The same handle is used symmetrically on both sides of the waypoint, to ensure smoothness.")]
``` | ```
public Vector3 tangent
``` |

Field Value

Type: **Vector3**

### See Also

**Reference**

- CinemachinePathWaypoint Structure
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachinePathBase Class

Abstract base class for a world-space path, suitable for a camera dolly track.

Inheritance Hierarchy

- System
  - Object
    - Component
      - Behaviour
        -MonoBehaviour
          - CinemachineCinemachinePathBase
            - CinemachineCinemachinePath
            - CinemachineCinemachineSmoothPath

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```csharp
public abstract class CinemachinePathBase : MonoBehaviour
```

The CinemachinePathBase type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachinePathBase</td>
<td></td>
</tr>
</tbody>
</table>

Top
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DistanceCacheSampleStepsPerSegment</strong></td>
<td>When calculating the distance cache, sample the path this many times between points</td>
</tr>
<tr>
<td><strong>Looped</strong></td>
<td>True if the path ends are joined to form a continuous loop</td>
</tr>
<tr>
<td><strong>MaxPos</strong></td>
<td>The maximum value for the path position</td>
</tr>
<tr>
<td><strong>MinPos</strong></td>
<td>The minimum value for the path position</td>
</tr>
<tr>
<td><strong>PathLength</strong></td>
<td>Get the length of the path in distance units. If the</td>
</tr>
</tbody>
</table>
distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheIsValid</td>
<td>See whether the distance cache is valid. If it's not valid, then any call to GetPathLength() or GetPathPositionFromDistance() will trigger a potentially costly regeneration of the path distance cache</td>
</tr>
<tr>
<td>EvaluateOrientation</td>
<td>Get the orientation the curve at a point along the path.</td>
</tr>
<tr>
<td>EvaluateOrientationAtUnit</td>
<td>Get the orientation the curve at a point along the path.</td>
</tr>
<tr>
<td>EvaluatePosition</td>
<td>Get a worldspace position of a point along the path</td>
</tr>
<tr>
<td>EvaluatePositionAtUnit</td>
<td>Get a worldspace position of a point along the path</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>EvaluateTangent</td>
<td>Get the tangent of the curve at a point along the path.</td>
</tr>
<tr>
<td>EvaluateTangentAtUnit</td>
<td>Get the tangent of the curve at a point along the path.</td>
</tr>
<tr>
<td>FindClosestPoint</td>
<td>Find the closest point on the path to a given worldspace target point.</td>
</tr>
<tr>
<td>GetPathDistanceFromPosition</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>GetPathPositionFromDistance</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>InvalidateDistanceCache</td>
<td>Call this if the path changes in such a way as to affect distances or other cached elements.</td>
</tr>
<tr>
<td>MaxUnit</td>
<td>Get the maximum value, for a given unity type.</td>
</tr>
<tr>
<td>MinUnit</td>
<td>Get the minimum value, for a given unity type.</td>
</tr>
<tr>
<td>NormalizePathDistance</td>
<td>Normalize a distance along the path based on the path length.</td>
</tr>
</tbody>
</table>
If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.

- **NormalizePos**
  - Get a normalized path position, taking spins into account if looped.

- **NormalizeUnit**
  - Normalize the unit, so that it lies between MinUnit and MaxUnit.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Appearance</td>
<td>The settings that control how the path will appear in the editor scene view.</td>
</tr>
<tr>
<td>m_Resolution</td>
<td>Path samples per waypoint</td>
</tr>
</tbody>
</table>

### See Also

- Reference
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachinePathBase.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
protected CinemachinePathBase()
```

See Also

Reference
- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
The `CinemachinePathBase` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>DistanceCacheSampleStepsPerSegment</code></td>
<td>When calculating the distance cache, sample the path this many times between points.</td>
</tr>
<tr>
<td><code>Looped</code></td>
<td>True if the path ends are joined to form a continuous loop.</td>
</tr>
<tr>
<td><code>MaxPos</code></td>
<td>The maximum value for the path position.</td>
</tr>
<tr>
<td><code>MinPos</code></td>
<td>The minimum value for the path position.</td>
</tr>
<tr>
<td>PathLength</td>
<td>Get the length of the path in distance units. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache</td>
</tr>
</tbody>
</table>

Top

See Also

Reference
CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.DistanceCacheSampleStepsPerSegment Property

When calculating the distance cache, sample the path this many times between points

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public abstract int DistanceCacheSampleStepsPerSegment
```

### Property Value

Type: Int32

### See Also

**Reference**
- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseLooped Property

True if the path ends are joined to form a continuous loop

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public abstract bool Looped { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: **Boolean**

**See Also**

- Reference  
  CinemachinePathBase Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseMaxPos Property

The maximum value for the path position

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public abstract float MaxPos { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Single

See Also

Reference
CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathBase.MinPos

Property

The minimum value for the path position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public abstract float MinPos { get; }
```

**C#**  |  **JavaScript**  |  **Copy**
---|---|---

**Property Value**  
Type: Single

**See Also**

**Reference**  
CinemachinePathBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.PathLength Property

Get the length of the path in distance units. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public float PathLength { get; }
```

### Return Value

Type: **Single**

The length of the path in distance units, when sampled at this rate.

### See Also

Reference

- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachinePathBase Methods

The CinemachinePathBase type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheIsValid</td>
<td>See whether the distance cache is valid. If it's not valid, then any call to GetPathLength() or GetPathPositionFromDistance() will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>EvaluateOrientation</td>
<td>Get the orientation the curve at a point along the path.</td>
</tr>
<tr>
<td>EvaluateOrientationAtUnit</td>
<td>Get the orientation the curve at a point along the path.</td>
</tr>
<tr>
<td>EvaluatePosition</td>
<td>Get a worldspace position of a point along the path.</td>
</tr>
<tr>
<td>EvaluatePositionAtUnit</td>
<td>Get a worldspace position of a point along the path.</td>
</tr>
<tr>
<td>EvaluateTangent</td>
<td>Get the tangent of the curve at a point along the path.</td>
</tr>
<tr>
<td>EvaluateTangentAtUnit</td>
<td>Get the tangent of the curve at a point along the path.</td>
</tr>
<tr>
<td>FindClosestPoint</td>
<td>Find the closest point on the path.</td>
</tr>
</tbody>
</table>
path to a given worldspace target point.

**GetPathDistanceFromPosition**

Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger potentially costly regeneration of the path distance cache.

**GetPathPositionFromDistance**

Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger potentially costly regeneration of the path distance cache.

**InvalidateDistanceCache**

Call this if the path changes in such a way as to affect distances or other cached path elements.

**MaxUnit**

Get the maximum value, for the given unity type.

**MinUnit**

Get the minimum value, for the given unity type.

**NormalizePathDistance**

Normalize a distance along path based on the path length. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.

**NormalizePos**

Get a normalized path position.
taking spins into account if looped

NormalizeUnit

Normalize the unit, so that lies between MinUnit and MaxUnit

See Also

Reference
CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseDistanceCacheIsValid Method

See whether the distance cache is valid. If it’s not valid, then any call to GetPathLength() or GetPathPositionFromDistance() will trigger a potentially costly regeneration of the path distance cache.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public bool DistanceCacheIsValid()
```

**Return Value**

Type: Boolean

Whether the cache is valid for this sampling rate

### See Also

**Reference**

- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachinePathBaseEvaluateOrientation Method

Get the orientation the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

#### C#  
```csharp
public abstract Quaternion EvaluateOrientation(
    float pos
)
```

#### JavaScript

`public abstract Quaternion EvaluateOrientation(pos)`

### Parameters

**pos**

Type: `System.Single`  
Position along the path. Need not be normalized.

### Return Value

Type: `Quaternion`  
World-space orientation of the path

### See Also

**Reference**  
CinemachinePathBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
# CinemachinePathBase.EvaluateOrientationAtUnit Method

Get the orientation the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public Quaternion EvaluateOrientationAtUnit(  
|   float pos,            |                                       |
|   CinemachinePathBasePositionUnits units    |                                       |
| )                                     |                                       |

**Parameters**

- **pos**  
  Type: System.Single  
  Postion along the path. Need not be normalized.

- **units**  
  Type: CinemachinePathBasePositionUnits  
  The unit to use when interpreting the value of pos.

**Return Value**  
Type: Quaternion  
World-space orientation of the path

## See Also

Reference  
CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseEvaluatePosition Method

Get a worldspace position of a point along the path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public abstract Vector3 EvaluatePosition(float pos)
```

**Parameters**

*pos*  
Type: `System.Single`  
Position along the path. Need not be normalized.

**Return Value**

Type: `Vector3`  
World-space position of the point along at path at pos

**See Also**

**Reference**  
CinemachinePathBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseEvaluatePositionAtUnit Method

Get a worldspace position of a point along the path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public Vector3 EvaluatePositionAtUnit(  
    float pos,  
    CinemachinePathBasePositionUnits units  
) |  |

### Parameters

- **pos**
  - Type: **System.Single**  
  - Position along the path. Need not be normalized.

- **units**
  - Type: **CinemachinePathBasePositionUnits**  
  - The unit to use when interpreting the value of pos.

### Return Value

- **Type:** **Vector3**  
  - World-space position of the point along at path at pos

### See Also

- **Reference**  
  - CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.EvaluateTangent Method

Get the tangent of the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public abstract Vector3 EvaluateTangent(
    float pos
)
```

**JavaScript**

```javascript
public abstract Vector3 EvaluateTangent(
    float pos
)
```

### Parameters

**pos**
Type: `System.Single`
Position along the path. Need not be normalized.

### Return Value
Type: `Vector3`
World-space direction of the path tangent. Length of the vector represents the tangent strength

### See Also

**Reference**
- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseEvaluateTangentAtUnit Method

Get the tangent of the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public Vector3 EvaluateTangentAtUnit(
    float pos,
    CinemachinePathBasePositionUnits units
)
```

**JavaScript**

```javascript
public Vector3 EvaluateTangentAtUnit(
    float pos,
    CinemachinePathBasePositionUnits units
)
```

### Parameters

- **pos**
  - Type: `System.Single`  
  - Position along the path. Need not be normalized.

- **units**
  - Type: `CinemachineCinemachinePathBasePositionUnits`  
  - The unit to use when interpreting the value of `pos`.

### Return Value

- **Type:** `Vector3`  
  - World-space direction of the path tangent. Length of the vector represents the tangent strength.

### See Also

Reference
<table>
<thead>
<tr>
<th>CinemachinePathBase Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cinemachine Namespace</td>
</tr>
</tbody>
</table>

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase::FindClosestPoint Method

Find the closest point on the path to a given worldspace target point.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public virtual float FindClosestPoint(
    Vector3 p,
    int startSegment,
    int searchRadius,
    int stepsPerSegment
)
```

**Parameters**

- **p**
  Type: **Vector3**  
  Worldspace target that we want to approach

- **startSegment**
  Type: **SystemInt32**  
  In what segment of the path to start the search. A Segment is a section of path between 2 waypoints.

- **searchRadius**
  Type: **SystemInt32**  
  How many segments on either side of the startSegment to search. -1 means no limit, i.e. search the entire path

- **stepsPerSegment**
  Type: **SystemInt32**
We search a segment by dividing it into this many straight pieces. The higher the number, the more accurate the result, but performance is proportionally slower for higher numbers.

Return Value
Type: Single
The position along the path that is closest to the target point. The value is in Path Units, not Distance units.

Remarks
Performance could be improved by checking the bounding polygon of each segment, and only entering the best segment(s).

See Also
Reference
CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.GetPathDistanceFromPosition Method

Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public float GetPathDistanceFromPosition(float pos)
```

Parameters

`pos`
Type: System.Single

[Missing <param name="pos"/> documentation for "M:Cinemachine.CinemachinePathBase.GetPathDistanceFromPosition(System.Single)"

Return Value

Type: Single
The length of the path in distance units, when sampled at this rate

See Also

Reference
CinemachinePathBase Class
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.GetPathPositionFromDistance Method

Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public float GetPathPositionFromDistance(
    float distance
)
```

### Parameters

- **distance**
  
  Type: System.Single

```
[Missing <param name="distance"/> documentation for "M:Cinemachine.CinemachinePathBase.GetPathPositionFromDistance(System.Single)"
```

### Return Value

Type: Single

The length of the path in distance units, when sampled at this rate.

### See Also

**Reference**

CinemachinePathBase Class

Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathBase::InvalidateDistanceCache Method

Call this if the path changes in such a way as to affect distances or other cached path elements

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public virtual void InvalidateDistanceCache()
```

### See Also

**Reference**  
CinemachinePathBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathBaseMaxUnit Method

Get the maximum value, for the given unity type

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public float MaxUnit(
    CinemachinePathBasePositionUnits units
)
```

### Parameters

**units**  
Type: `CinemachinePathBasePositionUnits`  
The uniot type

### Return Value

Type: `Single`  
The maximum allowable value for this path

### See Also

**Reference**  
CinemachinePathBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.MinUnit Method

Get the minimum value, for the given unity type

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public float MinUnit(
    CinemachinePathBasePositionUnits units
)
```

### Parameters

- **units**
  - Type: `CinemachinePathBasePositionUnits`
  - The unit type

### Return Value

- Type: `Single`
- The minimum allowable value for this path

### See Also

**Reference**

- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathBase.NormalizePathDistance Method

Normalize a distance along the path based on the path length. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public float NormalizePathDistance(float distance)
```

### Parameters

- **distance**
  - Type: System.Single
  - The distance to normalize

### Return Value

- Type: Single
  - The normalized distance, ranging from 0 to path length

### See Also

- Reference
  - CinemachinePathBase Class
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.NormalizePos Method

Get a normalized path position, taking spins into account if looped

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
public virtual float NormalizePos(
    float pos
)
``` | |

### Parameters

**pos**

Type: `SystemSingle`
Position along the path

### Return Value

Type: `Single`
Normalized position, between MinPos and MaxPos

## See Also

### Reference

- CinemachinePathBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseNormalizeUnMethod

Normalize the unit, so that it lies between MinUnit and MaxUnit

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public virtual float NormalizeUnit(
    float pos,
    CinemachinePathBasePositionUnits units
)
```

### Parameters

*pos*
- Type: `SystemSingle`
- The value to be normalized

*units*
- Type: `CinemachinePathBasePositionUnits`
- The unit type

### Return Value

- Type: `Single`
- The normalized value of pos, between MinUnit and MaxUnit

#### See Also

Reference  
**CinemachinePathBase Class**
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase Fields

The CinemachinePathBase type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Appearance</td>
<td>The settings that control how the path will appear in the editor scene view.</td>
</tr>
<tr>
<td>m_Resolution</td>
<td>Path samples per waypoint</td>
</tr>
</tbody>
</table>

### See Also

- Reference
  - CinemachinePathBase Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePathBase.m_Appearance Field

The settings that control how the path will appear in the editor scene view.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public CinemachinePathBaseAppearance m_Appearance</code></td>
<td></td>
</tr>
</tbody>
</table>

Field Value
Type: `CinemachinePathBaseAppearance`

### See Also

- Reference  
  - CinemachinePathBase Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathBase m_Resolution

Field
Path samples per waypoint

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[TooltipAttribute("Path samples per waypoint. This is used for calculating path distances.")]
[RangeAttribute(1f, 100f)]
public int m_Resolution
```

Field Value
Type: Int32

See Also

Reference
CinemachinePathBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBaseAppearance Class

This class holds the settings that control how the path will appear in the editor scene view. The path is not visible in the game view.

## Inheritance Hierarchy

- System.Object
  - Cinemachine
  - CinemachinePathBaseAppearance

### Namespace:

- Cinemachine

### Assembly:

- Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(18.1f, DocumentationSortingAttribute
)]
public class Appearance
```

The `CinemachinePathBaseAppearance` type exposes the following members.

## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachinePathBaseAppearance</td>
<td></td>
</tr>
</tbody>
</table>

## Fields
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inactivePathColor</td>
<td></td>
</tr>
<tr>
<td>pathColor</td>
<td></td>
</tr>
<tr>
<td>width</td>
<td></td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.Appearance Constructor


**Namespace:**  Cinemachine
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public Appearance()
```

**JavaScript**

```javascript

```

### See Also

**Reference**

CinemachinePathBase.Appearance Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Appearance Fields

The `CinemachinePathBaseAppearance` type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inactivePathColor</td>
<td></td>
</tr>
<tr>
<td>pathColor</td>
<td></td>
</tr>
<tr>
<td>width</td>
<td></td>
</tr>
</tbody>
</table>

## See Also

Reference
- `CinemachinePathBaseAppearance Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.Appearance.inactivePathColor Field

[Missing <summary> documentation for "F:Cinemachine.CinemachinePathBase.Appearance.inactivePathColor"]

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

C#  
```csharp
[TooltipAttribute("The color of the path itself when inactive in the editor")]
public Color inactivePathColor
```

### Field Value

Type: **Color**

### See Also

**Reference**  
CinemachinePathBase.Appearance Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.Appearance.Field

[Missing <summary> documentation for "F:Cinemachine.CinemachinePathBase.Appearance.pathColor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#     JavaScript

```csharp
[TooltipAttribute("The color of the path itself when it is active in the editor")]
public Color pathColor
```

Field Value
Type: Color

See Also

Reference
CinemachinePathBase.Appearance Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBase.Appearance.Field

[Missing <summary> documentation for "F:Cinemachine.CinemachinePathBase.Appearance.width"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  |  JavaScript
---|---

```csharp
[TooltipAttribute("The width of the railroad-tracks that are drawn to represent the path")
[RangeAttribute(0f, 10f)]

public float width
```

Field Value
Type: Single

See Also

Reference
CinemachinePathBase.Appearance Class
Cinemachine Namespace

Visit the Cinematic Forum
https://unity3d.com/legal/terms-of-service
CinemachinePathBasePositionUnits Enumeration

How to interpret the Path Position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public enum PositionUnits
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PathUnits</td>
<td>0</td>
<td>Use PathPosition units, where 0 is first waypoint, 1 is second waypoint, etc</td>
</tr>
<tr>
<td>Distance</td>
<td>1</td>
<td>Use Distance Along Path. Path will be sampled according to its Resolution setting, and a distance lookup table will be cached internally</td>
</tr>
</tbody>
</table>

### See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePipeline Class

Internal container for CinemachineComponentBase. Does nothing but contain.

▲ Inheritance Hierarchy

- System
  - Object
    - Component
      - Behaviour
        - MonoBehaviour
          - CinemachinePipeline

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▲ Syntax

```csharp
[AddComponentMenu("" )]
public sealed class CinemachinePipeline : MonoBehaviour
```

The CinemachinePipeline type exposes the following members.

▲ Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachinePipeline</td>
<td>Top</td>
</tr>
</tbody>
</table>

▲ See Also
Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePipeline Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachinePipeline.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
class CinemachinePipeline
{
    public CinemachinePipeline();
}
```

See Also

Reference
- CinemachinePipeline Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePOV Class

This is a CinemachineComponent in the Aim section of the component pipeline. Its job is to aim the camera in response to the user's mouse or joystick input. The composer does not change the camera's position. It will only pan and tilt the camera where it is, in order to get the desired framing. To move the camera, you have to use the virtual camera's Body section.

Inheritance Hierarchy

```
System
   Object
   Component
      Behaviour
         MonoBehaviour
            CinemachineComponentBase
                  CinemachinePOV
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
C#  

[DocumentationSortingAttribute(23f, DocumentationOrderAttribute)]
[AddComponentMenu("")]
[RequireComponent(typeof(CinemachinePipeline))]
[SaveDuringPlayAttribute]
public class CinemachinePOV : CinemachineComponent
```

The CinemachinePOV type exposes the following members.

Constructors
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CinemachinePOV</strong></td>
<td></td>
</tr>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a LookAt defined (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage</td>
</tr>
<tr>
<td></td>
<td>(Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>VirtualCamera</strong></td>
<td>Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
CinemachineComponentBase.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies the axis values and orients the camera accordingly. (Overrides CinemachineComponentBase.MutateCameraState(CameraState Single).)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_HorizontalAxis</td>
<td>The Horizontal axis. Value is -180..180. Controls the horizontal orientation</td>
</tr>
<tr>
<td>m_VerticalAxis</td>
<td>The Vertical axis. Value is -90..90. Controls the vertical orientation</td>
</tr>
</tbody>
</table>

### See Also

Reference
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePOV Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachinePOV.#ctor"]

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public CinemachinePOV()
```

**See Also**

Reference

- CinemachinePOV Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePOV Properties

The `CinemachinePOV` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a LookAt defined (Overrides <code>CinemachineComponentBaseIsValid</code>.)</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage (Overrides <code>CinemachineComponentBaseStage</code>.)</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam's CameraState. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>VirtualCamera</strong></td>
<td>Get the associated CinemachineVirtualCameraBase</td>
</tr>
</tbody>
</table>
(Inherited from CinemachineComponentBase.)

## See Also

**Reference**
- CinemachinePOV Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePOVIsValid

Property

True if component is enabled and has a LookAt defined

**Namespace:**  Cinemachine

**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public override bool IsValid { get; }</code></td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type:  Boolean

### See Also

**Reference**

CinemachinePOV Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
**CinemachinePOVStage Property**

Get the Cinemachine Pipeline stage that this component implements. Always returns the Aim stage.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public override CinemachineCoreStage Stage { get; }
```

**Type:** CinemachineCoreStage

### See Also

**Reference**

CinemachinePOV Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachinePOV Methods

The `CinemachinePOV` type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Applies the axis values and orients the camera accordingly (Overrides CinematicComponentBaseMutateCameraState).</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. (Inherited from CinematicComponentBase.)</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinematicPOV Class
- Cinematic Namespace

Visit the Cinematic Forum
https://unity3d.com/legal/terms-of-service
CinemachinePOVMutateCameraState Method

Applies the axis values and orients the camera accordingly

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
public override void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)

Parameters

- **curState**
  - Type: CinemachineCameraState
  - The current camera state
- **deltaTime**
  - Type: SystemSingle
  - Used for calculating damping. Not used.

See Also

Reference
- CinemachinePOV Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePOV Fields

The CinemachinePOV type exposes the following members.

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_HorizontalAxis</td>
<td>The Horizontal axis. Value is -180..180. Controls the horizontal orientation</td>
</tr>
<tr>
<td>m_VerticalAxis</td>
<td>The Vertical axis. Value is -90..90. Controls the vertical orientation</td>
</tr>
</tbody>
</table>

**See Also**

Reference
CinemachinePOV Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachinePOV\_HorizontalAxis Field

The Horizontal axis. Value is -180..180. Controls the horizontal orientation

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```c#
[TooltipAttribute("The Horizontal axis. Value is -180..180. Controls the horizontal orientation")]

public AxisState m_HorizontalAxis
```

**JavaScript**

Field Value  
Type: AxisState

### See Also

**Reference**  
CinemachinePOV Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachinePOV::m_VerticalAxis Field

The Vertical axis. Value is -90..90. Controls the vertical orientation

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[TooltipAttribute("The Vertical axis. Value is -90..90. Controls the vertical orientation")]
public AxisState m_VerticalAxis
```

**JavaScript**

```javascript
// Not applicable
```

### See Also

**Reference**

- CinemachinePOV Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPath Class

Defines a world-space path, consisting of an array of waypoints, each of which has position and roll settings. Bezier interpolation is performed between the waypoints, to get a smooth and continuous path. The path will pass through all waypoints, and (unlike CinemachinePath) first and second order continuity is guaranteed.

Inheritance Hierarchy

- SystemObject
  - Object
  - Component
    - Behaviour
      - MonoBehaviour
        - Cinemachine
          - CinemachinePathBase
            - CinemachineSmoothPath

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public class CinemachineSmoothPath : Cinemachine
```

The `CinemachineSmoothPath` type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheSampleStepsPerSegment</td>
<td>When calculating the distance cache, sample the path this many times between points. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td>Looped</td>
<td>True if the path ends are joined to form a continuous loop. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td>MaxPos</td>
<td>The maximum value for the path position. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td>MinPos</td>
<td>The minimum value for the path position. (Overrides CinemachinePathBase)</td>
</tr>
<tr>
<td>PathLength</td>
<td>Get the length of the path in distance units.</td>
</tr>
</tbody>
</table>

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheIsValid</td>
<td>See whether the distance cache is valid. If it is not valid, then any call to GetPathPositionFromDistance will potentially costy regenerating the cache. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>EvaluateOrientation</strong></td>
<td>Get the orientation the curve path. (Overrides CinemachinePathBaseEvaluateOrientation)</td>
</tr>
<tr>
<td><strong>EvaluateOrientationAtUnit</strong></td>
<td>Get the orientation the curve path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>EvaluatePosition</strong></td>
<td>Get a worldspace position of a point along the path. (Overrides CinemachinePathBaseEvaluatePosition)</td>
</tr>
<tr>
<td><strong>EvaluatePositionAtUnit</strong></td>
<td>Get a worldspace position of a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>EvaluateTangent</strong></td>
<td>Get the tangent of the curve path. (Overrides CinemachinePathBaseEvaluateTangent)</td>
</tr>
<tr>
<td><strong>EvaluateTangentAtUnit</strong></td>
<td>Get the tangent of the curve path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>FindClosestPoint</strong></td>
<td>Find the closest point on the path to a given worldspace target point. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>GetPathDistanceFromPosition</strong></td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the path is not valid, then calling this will trigger costly regeneration of the path distance cache. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td><strong>GetPathPositionFromDistance</strong></td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the path is not valid, then calling this will trigger costly regeneration of the path distance cache. (Inherited from CinemachinePathBase)</td>
</tr>
</tbody>
</table>
InvalidateDistanceCache
Call this if the path changes in such a way as to affect distances or other cached path elements.

MaxUnit
Get the maximum value, for the given unity type.

MinUnit
Get the minimum value, for the given unity type.

NormalizePathDistance
Normalize a distance along the path length. If the distance cache is not valid, calling this will trigger a potentially costly regeneration of the path distance cache.

NormalizePos
Get a normalized path position, taking spins into account if looped.

NormalizeUnit
Normalize the unit, so that it lies between MinUnit and MaxUnit.

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Appearance</td>
<td>The settings that control how the path will appear in the editor scene view. (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td><strong>m_Looped</strong></td>
<td>If checked, then the path ends are joined to form a continuous loop</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_Resolution</strong></td>
<td>Path samples per waypoint (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td><strong>m_Waypoints</strong></td>
<td>The waypoints that define the path. They will be interpolated using a bezier curve</td>
</tr>
</tbody>
</table>

**Top**

**See Also**

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPath Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineSmoothPath.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    JavaScript

```csharp
public CinemachineSmoothPath()
```

See Also

Reference
CinemachineSmoothPath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPath Properties

The `CinemachineSmoothPath` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheSampleStepsPerSegment</td>
<td>When calculating the distance cache, sample the path this many times between points. (Overrides <code>CinemachinePathBase</code>)</td>
</tr>
<tr>
<td>Looped</td>
<td>True if the path ends are joined to form a continuous loop. (Overrides <code>CinemachinePathBase</code>)</td>
</tr>
<tr>
<td>MaxPos</td>
<td>The maximum value for the path position. (Overrides <code>CinemachinePathBase</code>)</td>
</tr>
<tr>
<td>MinPos</td>
<td>The minimum value for the path position. (Overrides <code>CinemachinePathBase</code>)</td>
</tr>
<tr>
<td>PathLength</td>
<td>Get the length of the path. If the path is not valid, then regeneration of the path distance cache is not valid, then calling this will cause regeneration of the path distance cache. (Inherited from <code>CinemachinePathBase</code>)</td>
</tr>
</tbody>
</table>

See Also

Reference

`CinemachineSmoothPath Class`
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathDistanceCacheSampleStepsPerSegment

When calculating the distance cache, sample the path this many times between points

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

**C#**

```csharp
public override int DistanceCacheSampleStepsPerSegment
```

**JavaScript**

Invalid syntax as JavaScript does not use keywords like `public` or `override`.

**Property Value**

Type: Int32

**See Also**

**Reference**

CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPathLooped Property

True if the path ends are joined to form a continuous loop

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```
public override bool LoopedListed { get; }
```

**Property Value**

Type: Boolean

**See Also**

Reference
CinemachineSmoothPath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathMaxPos Property

The maximum value for the path position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override float MaxPos { get; }
```

**Property Value**

Type: Single

**See Also**

Reference  
CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathMinPos

Property

The minimum value for the path position

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override float MinPos { get; }
```

**Property Value**  
Type: Single

**See Also**

Reference  
CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPath Methods

The CinemachineSmoothPath type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DistanceCacheIsValid</td>
<td>See whether the distance cache is valid, then any call to GetPathLength() or GetPathPositionFromDistance() will potentially costly regenerate cache. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluateOrientation</td>
<td>Get the orientation the curve at a point along the path. (Overrides CinemachinePathBase.EvaluateOrientation(Single))</td>
</tr>
<tr>
<td>EvaluateOrientationAtUnit</td>
<td>Get the orientation the curve at a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluatePosition</td>
<td>Get a worldspace position of a point along the path. (Overrides CinemachinePathBase.EvaluatePosition(Single))</td>
</tr>
<tr>
<td>EvaluatePositionAtUnit</td>
<td>Get a worldspace position of a point along the path. (Inherited from CinemachinePathBase)</td>
</tr>
<tr>
<td>EvaluateTangent</td>
<td>Get the tangent of the curve at a point along the path. (Overrides CinemachinePathBase.EvaluateTangent())</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>EvaluateTangentAtUnit</td>
<td>Get the tangent of the curve along the path. (Inherited from CinemachinePathBaseEv)</td>
</tr>
<tr>
<td>FindClosestPoint</td>
<td>Find the closest point on the worldspace target point. (Inherited from CinemachinePathBaseEv)</td>
</tr>
<tr>
<td>GetPathDistanceFromPosition</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>GetPathPositionFromDistance</td>
<td>Get the path position (in path units) corresponding to this distance along the path. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
<tr>
<td>InvalidateDistanceCache</td>
<td>Call this if the path changes in such a way as to affect distances or other cached path elements. (Overrides CinemachinePathBaseInvalidateDistanceCache)</td>
</tr>
<tr>
<td>MaxUnit</td>
<td>Get the maximum value, for the given unity type. (Inherited from CinemachinePathBaseEv)</td>
</tr>
<tr>
<td>MinUnit</td>
<td>Get the minimum value, for the given unity type. (Inherited from CinemachinePathBaseEv)</td>
</tr>
<tr>
<td>NormalizePathDistance</td>
<td>Normalize a distance along the path length. If the distance cache is not valid, then calling this will trigger a potentially costly regeneration of the path distance cache.</td>
</tr>
</tbody>
</table>

CinemachinePathBaseEv
<table>
<thead>
<tr>
<th>NormalizePos</th>
<th>Get a normalized path position, taking loops into account if looped (Inherited from CinemachinePathBase)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NormalizeUnit</td>
<td>Normalize the unit, so that it lies between MinUnit and MaxUnit (Inherited from CinemachinePathBase)</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineSmoothPath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPathEvaluateOrientation Method

Get the orientation the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public override Quaternion EvaluateOrientation(
    float pos
)
```

### Parameters

**pos**  
Type: System.Single  
Postion along the path. Need not be normalized.

### Return Value

Type: Quaternion  
World-space orientation of the path, as defined by tangent, up, and roll.

## See Also

**Reference**  
CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPathEvaluatePosition Method

Get a worldspace position of a point along the path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

### C#  

```csharp
public override Vector3 EvaluatePosition(float pos)
```

### JavaScript

```javascript
public override Vector3 EvaluatePosition(pos)
```

## Parameters

**pos**

- **Type:** System.Single  
  Position along the path. Need not be normalized.

## Return Value

- **Type:** Vector3  
  World-space position of the point along at path at pos

## See Also

- **Reference**  
  CinemachineSmoothPath Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathEvaluateTangent Method

Get the tangent of the curve at a point along the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

### Syntax

**C#**

```csharp
public override Vector3 EvaluateTangent(
    float pos
)
```

**JavaScript**

```javascript
// Not available
```

### Parameters

**pos**  
Type: `System.Single`  
Postion along the path. Need not be normalized.

### Return Value

Type: `Vector3`  
World-space direction of the path tangent. Length of the vector represents the tangent strength

### See Also

**Reference**  
CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathInvalidateDistanceCache Method

Call this if the path changes in such a way as to affect distances or other cached path elements

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

C#    | JavaScript
---|---
```csharp
public override void InvalidateDistanceCache()
```  

**See Also**

Reference  
CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPath Fields

The CinemachineSmoothPath type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Appearance</td>
<td>The settings that control how the path will appear in the editor scene view. (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td>m_Looped</td>
<td>If checked, then the path ends are joined to form a continuous loop</td>
</tr>
<tr>
<td>m_Resolution</td>
<td>Path samples per waypoint (Inherited from CinemachinePathBase.)</td>
</tr>
<tr>
<td>m_Waypoints</td>
<td>The waypoints that define the path. They will be interpolated using a bezier curve</td>
</tr>
</tbody>
</table>

See Also

Reference
CinemachineSmoothPath Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathm_Looped Field

If checked, then the path ends are joined to form a continuous loop.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```c#```  
```public bool m_Looped```  

Field Value  
Type: **Boolean**

### See Also

**Reference**  
CinemachineSmoothPath Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathm_Waypo Field

The waypoints that define the path. They will be interpolated using a bezier curve

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

#### C#

```csharp
[TooltipAttribute("The waypoints that define the path. They will be interpolated using a bezier curve.")] public CinemachineSmoothPathWaypoint[] m_Waypoint
```

#### JavaScript

```javascript
m_Waypoint
```

### Field Value

**Type:** CinemachineSmoothPathWaypoint

### See Also

**Reference**

- CinemachineSmoothPath Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPathWaypoint Structure

A waypoint along the path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="CinemachineSmoothPathWaypointWaypoint" alt="Code" /></td>
<td><img src="CinemachineSmoothPathWaypointWaypoint" alt="Code" /></td>
</tr>
</tbody>
</table>

The CinemachineSmoothPathWaypoint type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>position</td>
<td>Position in path-local space</td>
</tr>
<tr>
<td>roll</td>
<td>Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.</td>
</tr>
</tbody>
</table>

## See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Waypoint Fields

The CinemachineSmoothPathWaypoint type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>position</td>
<td>Position in path-local space</td>
</tr>
<tr>
<td>roll</td>
<td>Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.</td>
</tr>
</tbody>
</table>

## See Also

### Reference

- CinemachineSmoothPathWaypoint Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineSmoothPathWaypoint Field

Position in path-local space

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
[TooltipAttribute("Position in path-local space")]
public Vector3 position
```

Field Value  
Type: **Vector3**

### See Also

**Reference**  
CinemachineSmoothPathWaypoint Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineSmoothPathWaypointintro Field

Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.

**Namespace:**  Cinemachine

**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("Defines the roll of the path at this waypoint. The other orientation axes are inferred from the tangent and world up.")]
| public float roll |

### Field Value

Type: **Single**

### See Also

**Reference**

- CinemachineSmoothPathWaypoint Structure
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCamera Class

This is a virtual camera "manager" that owns and manages a collection of child Virtual Cameras. These child vcams are mapped to individual states in an animation state machine, allowing you to associate specific vcams to specific animation states. When that state is active in the state machine, then the associated camera will be activated. You can define custom blends and transitions between child cameras. In order to use this behaviour, you must have an animated target (i.e. an object animated with a state machine) to drive the behaviour.

Inheritance Hierarchy

- System
- Object
- Component
- Behaviour
- MonoBehaviour
- Cinemachine
- CinemachineVirtualCameraBase
- CinemachineStateDrivenCamera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[DocumentationSortingAttribute(13f, DocumentationSortingAttribute.ExecuteInEditMode, DocumentationSortingAttribute.DisallowMultipleComponent)
[AddComponentMenu("Cinemachine/CinemachineStateDrivenCamera")] public class CinemachineStateDrivenCamera : CinemachineStateDrivenCamera
```
The **CinemachineStateDrivenCamera** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ChildCameras</strong></td>
<td>The list of child cameras. These are immediate children in the hierarchy.</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug information.</td>
</tr>
<tr>
<td><strong>Follow</strong></td>
<td>Get the current Follow target. If parent is non-null and no specific Follow defined for this camera.</td>
</tr>
<tr>
<td><strong>IsBlending</strong></td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td><strong>LiveChild</strong></td>
<td>Get the current &quot;best&quot; child virtual camera, that would be chosen if the State Driven Camera were active.</td>
</tr>
<tr>
<td><strong>LiveChildOrSelf</strong></td>
<td>Return the live child.</td>
</tr>
<tr>
<td><strong>LookAt</strong></td>
<td>Get the current LookAt target. LookAt if parent is non-null and no specific LookAt defined for this camera.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Get the name of the Virtual Camera. Implementation returns the owner GameObject.</td>
</tr>
</tbody>
</table>
ParentCamera

Support for meta-virtual-camera situation where a virtual camera is the public face of a private army of which it manages on its own. It is the VirtualCamera owner, if an are implemented as Transform Transform parent vcam.

(Inherited from CinemachineVirtualCameraBase)

PreviousStateIsValid

Set this to force the next update deltaTime and reset itself

(Inherited from CinemachineVirtualCameraBase)

Priority

Get the Priority of the virtual camera which determines its placement in the CinemachineCore's queue of eligible shots.

(Inherited from CinemachineVirtualCameraBase)

State

The State of the current live child

(Overides CinemachineVirtualCameraBase)

ValidatingStreamVersion

Version that was last streamed for legacy

(Inherited from CinemachineVirtualCameraBase)

VirtualCameraGameObject

The GameObject owner of the behaviour.

(Inherited from CinemachineVirtualCameraBase)

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Method Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>CreateFakeHashName</td>
<td>API for the inspector editor. Implement nested state.</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is a live child of this camera. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is a live camera if it shares the highest priority with its peers and the most recent one goes to the top of the priority subqueue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Makes sure the internal child cache is up to date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransformChildrenChanged</td>
<td>Makes sure the internal child cache is up to date.</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields when changed in the inspector. ValidationStreamVersion will be valid. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
PreUpdateChildCameras

This is called prior to the vcams being updated on FixedUpdate if the child cameras are updating on FixedUpdate. If they are updating on LateUpdate instead, Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)

RemovePostPipelineStageHook

Remove a Pipeline stage hook callback. (Overrides CinemachineVirtualCameraBase)

ResolveFollow

Returns this vcam's Follow target, or if that is null, will return null. (Inherited from CinemachineVirtualCameraBase)

ResolveLookAt

Returns this vcam's LookAt target, or if that is null, will return null. (Inherited from CinemachineVirtualCameraBase)

Start

Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)

Update

Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)

UpdateCameraState

Called by CinemachineCore. Updates all the children, chooses the best one, and implements any required blending. (Overrides CinemachineVirtualCameraBase)

ValidateInstructions

Internal API for the inspector editor.

---

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is there to support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>m_AnimatedTarget</strong></td>
<td>The state machine whose state changes will drive this camera's choice of active child.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_CustomBlends</strong></td>
<td>This is the asset which contains custom settings for specific child blends.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_DefaultBlend</strong></td>
<td>The blend which is used if you don't explicitly define a blend between two Virtual Camera children.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_EnableAllChildCameras</strong></td>
<td>Force all child cameras to be enabled. This is useful if animating them in Timeline, but consumes extra resources.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_ExcludedPropertiesInInspector</strong></td>
<td>Inspector control - Use for hiding sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_Follow</strong></td>
<td>Default object for the child (the target object the camera children want to move with, e.g., the body target), if not specified in a child trigger. May be empty.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_Instructions</strong></td>
<td>The set of instructions associating virtual cameras with specific states. These instructions are used to choose the live child at any given moment.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_LayerIndex</strong></td>
<td>Which layer in the target FSM to observe.</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>m_LockStageInInspector</strong></td>
<td>Inspector control - Use for enabling sections of the Inspector.</td>
</tr>
</tbody>
</table>
UI. (Inherited from CinemachineVirtualCameraBase)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_LookAt</td>
<td>Default object for the children to look at (the target), if not specified in rig. May be empty.</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>m_ShowDebugText</td>
<td>When enabled, the current camera and blend will be indicated in the game window, for debugging.</td>
</tr>
<tr>
<td>OnPostPipelineStage</td>
<td>A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The **CinemachineStateDrivenCamera** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChildCameras</td>
<td>The list of child cameras. These are just the immediate children in the hierarchy.</td>
</tr>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info. (Overrides CinemachineVirtualCameraBase.Description)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the current Follow target. If Follow if parent is non-null and defined for this camera (Overrides CinemachineVirtualCameraBase.Follow)</td>
</tr>
<tr>
<td>IsBlending</td>
<td>Is there a blend in progress?</td>
</tr>
<tr>
<td>LiveChild</td>
<td>Get the current &quot;best&quot; child virtual camera. This would be chosen if the State Drive Camera is active.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Return the live child. (Overrides CinemachineVirtualCameraBase.LiveChildOrSelf)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the current LookAt target.</td>
</tr>
</tbody>
</table>
LookAt if parent is non-null and LookAt defined for this camera
(Overrides CinemachineVirtualCameraBase)

<table>
<thead>
<tr>
<th>Name</th>
<th>Get the name of the Virtual Camera. Implementation returns the owner GameObject's name. (Inherited from CinemachineVirtualCameraBase)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-camera situation where a virtual camera acts as a public face of a private army of child cameras, which it manages on its own. This method gets the VirtualCamera owner, if any, which are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to use deltaTime and reset itself. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. Determines its placement in the CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>State</td>
<td>The State of the current live child. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed for legacy. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
See Also

Reference
CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraChildCameras

Property

The list of child cameras. These are just the immediate children in the hierarchy.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public CinemachineVirtualCameraBase[] ChildCameras
```

**Property Value**

Type: CinemachineVirtualCameraBase

**See Also**

Reference

CinemachineStateDrivenCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraDescription Property

Gets a brief debug description of this virtual camera, for use when displaying debug info

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

```csharp
public override string Description { get; }
```

**Property Value**

Type: `String`

Implements

`ICinemachineCameraDescription`

⚠️ See Also

**Reference**

CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraFollow Property

Get the current Follow target. Returns parent's Follow if parent is non-null and no specific Follow defined for this camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override Transform Follow { get; set; }
```

Property Value  
Type: `Transform`

Implements  
`ICinemachineCameraFollow`

### See Also

- Reference  
  * CinemachineStateDrivenCamera Class  
  * Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera.IsBlending Property

Is there a blend in progress?

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public bool IsBlending { get; }
```

**Property Value**
Type: Boolean

**See Also**

Reference
CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCameraLiveChild Property

Get the current "best" child virtual camera, that would be chosen if the State Driven Camera were active.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

```csharp
public ICinemachineCamera LiveChild { get; set; }
```

### Property Value

Type: ICinemachineCamera

### See Also

Reference
- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraLiveChildOrSelf Property

Return the live child.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

## Syntax

```csharp
public override ICinemachineCamera LiveChildOrSelf
```

## Property Value

Type: `ICinemachineCamera`

**Implements**

`ICinemachineCameraLiveChildOrSelf`

## See Also

**Reference**

[CinemachineStateDrivenCamera Class](#)

[Cinemachine Namespace](#)

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCamera

Property

Get the current LookAt target. Returns parent's LookAt if parent is non-null and no specific LookAt defined for this camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

C#         JavaScript

```csharp
public override Transform LookAt { get; set; }
```

Property Value
Type: Transform
Implements ICinemachineCameraLookAt

See Also

Reference
CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCameraState

Property

The State of the current live child

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override CameraState State { get; }
```

**Property Value**

Type: CameraState  
Implements ICinemachineCameraState

**See Also**

Reference  
CinemachineStateDrivenCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
## CinemachineStateDrivenCamera Methods

The `CinemachineStateDrivenCamera` type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="AddPostPipelineStageHook" /></td>
<td>A delegate to hook into the state calculation pipeline. See <code>CinemachineCore.Stage</code>. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><img src="image" alt="CreateFakeHashName" /></td>
<td>API for the inspector editor.</td>
</tr>
<tr>
<td><img src="image" alt="InvokePostPipelineStageCallback" /></td>
<td>Invokes the <code>PostPipelineStageDelegate</code> for this camera, and up the hierarchy for all cameras. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><img src="image" alt="IsLiveChild" /></td>
<td>Check whether the vcam is Live.</td>
</tr>
<tr>
<td><img src="image" alt="MoveToTopOfPrioritySubqueue" /></td>
<td>When multiple virtual cameras share the highest priority in the queue with its peers, the most recent one becomes the Live camera if it shares the highest priority with its peers. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><img src="image" alt="OnDestroy" /></td>
<td>Base class implementation removes the virtual camera from the priority queue. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><img src="image" alt="OnDisable" /></td>
<td>Base class implementation ensures the priority queue remains up-to-date. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>OnEnable</strong></td>
<td>Makes sure the internal child cache is up to date. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnTransformChildrenChanged</strong></td>
<td>Makes sure the internal child cache is up to date.</td>
</tr>
<tr>
<td><strong>OnTransformParentChanged</strong></td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnTransitionFromCamera</strong></td>
<td>Notification that this virtual camera is going live. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnValidate</strong></td>
<td>Enforce bounds for fields when the base method is called, \ValidatingStreamVersion\ will be valid. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>PreUpdateChildCameras</strong></td>
<td>This is called prior to the child cameras being updated on FixedUpdate or LateUpdate instead. Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>RemovePostPipelineStageHook</strong></td>
<td>Remove a Pipeline stage hook callback. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ResolveFollow</strong></td>
<td>Returns this vcam's Follow target, or if that is null, will return (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>ResolveLookAt</strong></td>
<td>Returns this vcam's LookAt target, or if that is null, will return (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Base class implementation does nothing. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>Update</strong></td>
<td>Base class implementation makes sure the priority queue remains up-to-date. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>UpdateCameraState</strong></td>
<td>Called by CinemachineCore to update all the children, updates all the children, (Overrides CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>
See Also

Reference
- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraCreateFakeHashName Method

API for the inspector editor. Animation module does not have hashes for state parents, so we have to invent them in order to implement nested state handling.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public static string CreateFakeHashName(
    int parentHash,
    string stateName
)
```

Parameters

`parentHash`
Type: `SystemInt32`
[Missing <param name="parentHash"/> documentation for "M:Cinemachine.CinemachineStateDrivenCamera.CreateFakeHashName(System.Int32," stateName"

`stateName`
Type: `SystemString`
[Missing <param name="stateName"/> documentation for "M:Cinemachine.CinemachineStateDrivenCamera.CreateFakeHashName(System.Int32,"

Return Value
Type: `String`
See Also

Reference

CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera.IsLiveChild Method

Check whether the vcam a live child of this camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```
public override bool IsLiveChild(
    ICinemachineCamera vcam
)
```

**Parameters**

`vcam`
- Type: CinemachineICinemachineCamera  
The Virtual Camera to check

**Return Value**
- Type: Boolean  
True if the vcam is currently actively influencing the state of this vcam

**Implements**
- ICinemachineCamera.IsLiveChild(ICinemachineCamera)

**See Also**

**Reference**
- CinemachineStateDrivenCamera Class  
- Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraOnEnable Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>protected override void OnEnable()</code></td>
<td></td>
</tr>
</tbody>
</table>

### See Also

Reference  
CinemachineStateDrivenCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraOnTransformChildrenChanged Method

Makes sure the internal child cache is up to date

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public void OnTransformChildrenChanged()</td>
<td></td>
</tr>
</tbody>
</table>

### See Also

- Reference  
  - CinemachineStateDrivenCamera Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraRemovePostPipelineStageHook Method

Remove a Pipeline stage hook callback. Make sure it is removed from all the children.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

C# | JavaScript
---|---
```csharp
public override void RemovePostPipelineStageHook(
    CinemachineVirtualCameraBaseOnPostPipelineStageDelegate d
)
```

### Parameters

- `d`  
  Type: `CinemachineVirtualCameraBaseOnPostPipelineStageDelegate`  
  The delegate to remove.

### See Also

- Reference  
  - CinemachineStateDrivenCamera Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraUpdateCameraState Method

Called by CinemachineCore at designated update time so the vcam can position itself and track its targets. This implementation updates all the children, chooses the best one, and implements any required blending.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public override void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

### Parameters

- **worldUp**
  - Type: **Vector3**
  - Default world Up, set by the CinemachineBrain

- **deltaTime**
  - Type: **System.Single**
  - Delta time for time-based effects (ignore if less than or equal to 0)

### Implements

ICinemachineCameraUpdateCameraState(Vector3, Single)

### See Also
Reference

CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCamera Val Method

Internal API for the inspector editor.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public void ValidateInstructions()
```

### See Also

**Reference**  
CinemachineStateDrivenCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera

Fields

The `CinemachineStateDrivenCamera` type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>m_AnimatedTarget</td>
<td>The state machine whose state changes will drive this camera's choice of active child</td>
</tr>
<tr>
<td>m_CustomBlends</td>
<td>This is the asset which contains custom settings for specific child blends.</td>
</tr>
<tr>
<td>m_DefaultBlend</td>
<td>The blend which is used if you don't explicitly define a blend between two Virtual Camera children.</td>
</tr>
<tr>
<td>m_EnableAllChildCameras</td>
<td>Force all child cameras to be enabled. This is useful when animating them in Timeline, but consumes extra resources.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>m_ExcludedPropertiesInInspector</code></td>
<td>Inspector control - Use for hiding sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>m_Follow</code></td>
<td>Default object for the children wants to move with (the body target), if not specified in a child rig. May be empty.</td>
</tr>
<tr>
<td><code>m_Instructions</code></td>
<td>The set of instructions associating virtual cameras with states. These instructions are used to choose the live child at any given moment.</td>
</tr>
<tr>
<td><code>m_LayerIndex</code></td>
<td>Which layer in the target FSM to observe.</td>
</tr>
<tr>
<td><code>m_LockStageInInspector</code></td>
<td>Inspector control - Use for enabling sections of the UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>m_LookAt</code></td>
<td>Default object for the children to look at (the aim target), if not specified in a child rig. May be empty.</td>
</tr>
<tr>
<td><code>m_Priority</code></td>
<td>The priority will determine which camera becomes active based on the state of other cameras. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><code>m_ShowDebugText</code></td>
<td>When enabled, the current virtual camera's name is shown in the Inspector.</td>
</tr>
</tbody>
</table>
camera and blend will be indicated in the game window, for debugging

<table>
<thead>
<tr>
<th>OnPostPipelineStage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A delegate to hook into the state calculation pipeline. Implementation must call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

Top

See Also

Reference

- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera Field

The state machine whose state changes will drive this camera's choice of active child

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[SpaceAttribute]
[TooltipAttribute("The state machine whose state changes will drive this camera's choice of active child")]
public Animator m_AnimatedTarget
```

**JavaScript**

```javascript
// No JavaScript syntax available.
```

### Field Value

Type: **Animator**

### See Also

**Reference**
- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCamera
Field

This is the asset which contains custom settings for specific child blends.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

```csharp
[TooltipAttribute("This is the asset which contains custom settings for specific child blends")]
public CinemachineBlenderSettings m_CustomBlends
```

### Field Value

Type: CinemachineBlenderSettings

### See Also

**Reference**
- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera Field

The blend which is used if you don't explicitly define a blend between two Virtual Camera children.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
[CinemachineBlendDefinitionPropertyAttribute]
[TooltipAttribute("The blend which is used if you
don't explicitly define a blend between two Virtual Camera children")]
public CinemachineBlendDefinition m_DefaultBlend
``` |

### Field Value

Type: `CinemachineBlendDefinition`

### See Also

**Reference**  
CinemachineStateDrivenCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera

Field

Force all child cameras to be enabled. This is useful if animating them in Timeline, but consumes extra resources.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://unity3d.com/images/copy.png" alt="Copy" /></td>
<td><img src="https://unity3d.com/images/copy.png" alt="Copy" /></td>
</tr>
</tbody>
</table>

[TooltipAttribute("Force all child cameras to be enabled. This is useful if animating them in Timeline, but consumes extra resources")]

```csharp
public bool m_EnableAllChildCameras
```

Field Value

Type: Boolean

**See Also**

Reference

CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera
Field

Default object for the camera children wants to move with (the body target), if not specified in a child rig. May be empty

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("Default object for the camera
 [NoSaveDuringPlayAttribute]
public Transform m_Follow

Field Value
Type: Transform

See Also

Reference
CinemachineStateDrivenCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Field

The set of instructions associating virtual cameras with states. These instructions are used to choose the live child at any given moment.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The set of instructions associating virtual cameras with states. These instructions are used to choose the live child at any given moment")]
public CinemachineStateDrivenCameraInstruction[] m_Instructions
```

**Field Value**  
**Type:** CinemachineStateDrivenCameraInstruction

### See Also

**Reference**  
- CinemachineStateDrivenCamera Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera Field

Which layer in the target FSM to observe

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[TooltipAttribute("Which layer in the target state machine to observe")]
public int m_LayerIndex
```

Field Value

Type: **Int32**

**See Also**

Reference

CinemachineStateDrivenCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCamera Field

Default object for the camera children to look at (the aim target), if not specified in a child rig. May be empty

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[TooltipAttribute("Default object for the camera aim target, if not specified in a child camera.
May be empty if all the children define targets of their own.")]
[NoSaveDuringPlayAttribute]
public Transform m_LookAt
```

Field Value
Type: **Transform**

**See Also**

Reference
- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineStateDrivenCamera Field

When enabled, the current camera and blend will be indicated in the game window, for debugging

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[TooltipAttribute("When enabled, the current child camera and blend will be indicated in the game window, for debugging")]
public bool m_ShowDebugText
```

**Field Value**

Type: **Boolean**

### See Also

**Reference**

- CinemachineStateDrivenCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraInstruction

Structure

This represents a single instruction to the StateDrivenCamera. It associates an state from the state machine with a child Virtual Camera, and also holds activation tuning parameters.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
```csharp
[SerializableAttribute]
public struct Instruction
```

JavaScript

```
```

The **CinemachineStateDrivenCameraInstruction** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_ActivateAfter</td>
<td>How long to wait (in seconds) before activating the virtual camera. This filters out very short state durations</td>
</tr>
<tr>
<td>m_FullHash</td>
<td>The full hash of the animation state</td>
</tr>
<tr>
<td>m_MinDuration</td>
<td>The minimum length of time (in seconds) to keep a virtual</td>
</tr>
</tbody>
</table>
camera active

|   | m_VirtualCamera | The virtual camera to activate when the animation state becomes active |

See Also

Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
### Instruction Fields

The `CinemachineStateDrivenCameraInstruction` type exposes the following members.

#### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>m_ActivateAfter</code></td>
<td>How long to wait (in seconds) before activating the virtual camera. This filters out very short state durations</td>
</tr>
<tr>
<td><code>m_FullHash</code></td>
<td>The full hash of the animation state</td>
</tr>
<tr>
<td><code>m_MinDuration</code></td>
<td>The minimum length of time (in seconds) to keep a virtual camera active</td>
</tr>
<tr>
<td><code>m_VirtualCamera</code></td>
<td>The virtual camera to activate when the animation state becomes active</td>
</tr>
</tbody>
</table>

#### See Also

Reference
- `CinemachineStateDrivenCameraInstruction Structure`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraField

How long to wait (in seconds) before activating the virtual camera. This filters out very short state durations

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("How long to wait (in seconds)
public float m_ActivateAfter |

<table>
<thead>
<tr>
<th>Field Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Single</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

CinemachineStateDrivenCameraInstruction Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCameraField

The full hash of the animation state

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
[TooltipAttribute("The full hash of the animation state")]
public int m_FullHash
```

### Field Value

**Type:** Int32

### See Also

**Reference**  
CinemachineStateDrivenCameraInstruction Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Field

The minimum length of time (in seconds) to keep a virtual camera active

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[TooltipAttribute("The minimum length of time (in seconds) to keep a virtual camera active")]
public float m_MinDuration
```

**JavaScript**

```javascript
// No equivalent in JavaScript
```

**Field Value**

**Type:** Single

### See Also

**Reference**

- CinemachineStateDrivenCameraInstruction Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineStateDrivenCamera INS

Field

The virtual camera to activate when the animation state becomes active

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
[TooltipAttribute("The virtual camera to activate when the animation state becomes active")]
public CinemachineVirtualCameraBase m_VirtualCamera
```

**Field Value**

Type: CinemachineVirtualCameraBase

**See Also**

Reference
- CinemachineStateDrivenCameraInstruction Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroup Class

Defines a group of target objects, each with a radius and a weight. The weight is used when calculating the average position of the target group. Higher-weighted members of the group will count more. The bounding box is calculated by taking the member positions, weight, and radii into account.

Inheritance Hierarchy

- SystemObject
  - Object
  - Component
  - Behaviour
  - MonoBehaviour
  - Cinemachine
  - CinemachineTargetGroup

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(19f, DocumentationSortingAttribute)]
[AddComponentMenu("Cinemachine/CinemachineTargetGroup")]
[SaveDuringPlayAttribute]
[ExecuteInEditMode]
public class CinemachineTargetGroup : MonoBehaviour
```

The CinemachineTargetGroup type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CinemachineTargetGroup

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoundingBox</td>
<td>The axis-aligned bounding box of the group, computed using the targets positions and radii</td>
</tr>
<tr>
<td>IsEmpty</td>
<td>Return true if there are no members with weight &gt; 0</td>
</tr>
</tbody>
</table>

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetViewSpaceBoundingBox</td>
<td>The axis-aligned bounding box of the group, in a specific reference frame</td>
</tr>
</tbody>
</table>

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_PositionMode</td>
<td>How the group's position is calculated</td>
</tr>
<tr>
<td>m_RotationMode</td>
<td>How the group's orientation is calculated</td>
</tr>
<tr>
<td>m_Targets</td>
<td>The target objects, together</td>
</tr>
</tbody>
</table>
with their weights and radii, that will contribute to the group's average position, orientation, and size.

| m_UpdateMethod | When to update the group's transform based on the position of the group members |

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTargetGroup Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineTargetGroup.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript

```csharp
public CinemachineTargetGroup()
```

See Also

Reference
CinemachineTargetGroup Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The `CinemachineTargetGroup` type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoundingBox</td>
<td>The axis-aligned bounding box of the group, computed using the targets positions and radii</td>
</tr>
<tr>
<td>IsEmpty</td>
<td>Return true if there are no members with weight &gt; 0</td>
</tr>
</tbody>
</table>

### See Also

**Reference**
- `CinemachineTargetGroup Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupBoundingE
Property

The axis-aligned bounding box of the group, computed using the
targets positions and radii

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```csharp
public Bounds BoundingBox { get; }
```

Property Value
Type: Bounds

See Also

Reference
CinemachineTargetGroup Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupIsEmpty Property

Return true if there are no members with weight > 0

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public bool IsEmpty { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**  
Type: Boolean

**See Also**

Reference  
CinemachineTargetGroup Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
The CinemachineTargetGroup type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetViewSpaceBoundingBox</td>
<td>The axis-aligned bounding box of the group, in a specific reference frame</td>
</tr>
</tbody>
</table>

## See Also

Reference
- CinemachineTargetGroup Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTargetGroup.GetViewSpaceBoundingBox Method

The axis-aligned bounding box of the group, in a specific reference frame

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public Bounds GetViewSpaceBoundingBox(
    Matrix4x4 mView
)
```

Parameters

mView
Type: Matrix4x4
The frame of reference in which to compute the bounding box

Return Value
Type: Bounds
The axis-aligned bounding box of the group, in the desired frame of reference

See Also

Reference
CinemachineTargetGroup Class
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroup Fields

The CinemachineTargetGroup type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_PositionMode</td>
<td>How the group's position is calculated</td>
</tr>
<tr>
<td>m_RotationMode</td>
<td>How the group's orientation is calculated</td>
</tr>
<tr>
<td>m_Targets</td>
<td>The target objects, together with their weights and radii, that will contribute to the group's average position, orientation, and size</td>
</tr>
<tr>
<td>m_UpdateMethod</td>
<td>When to update the group's transform based on the position of the group members</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineTargetGroup Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupm_PositionMode Field

How the group's position is calculated

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[TooltipAttribute(&quot;How the group's position is calculated.	Select GroupCenter for the center of the bounding box,	and GroupAverage for a weighted average of the positions of the members.&quot;)]</code></td>
<td></td>
</tr>
<tr>
<td><code>public CinemachineTargetGroupPositionMode m_PositionMode</code></td>
<td></td>
</tr>
</tbody>
</table>

Field Value  
Type: CinemachineTargetGroupPositionMode

### See Also

**Reference**  
CinemachineTargetGroup Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupm_RotationField

How the group's orientation is calculated

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[TooltipAttribute(&quot;How the group's rotation is calculated.	Select Manual to use the value in the group's transform, and GroupAverage for a weighted average of the orientations of the members.&quot;)]</td>
<td>public CinemachineTargetGroupRotationMode m_RotationMode</td>
</tr>
</tbody>
</table>

Field Value  
Type: CinemachineTargetGroupRotationMode

### See Also

**Reference**  
CinemachineTargetGroup Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTargetGroupm_Targets Field

The target objects, together with their weights and radii, that will contribute to the group's average position, orientation, and size

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**  
```csharp
[NoSaveDuringPlayAttribute]
[TooltipAttribute("The target objects, together with their weights and radii, that will contribute to the group's average position, orientation, and size.")]
public CinemachineTargetGroupTarget[] m_Targets
```

**Field Value**  
Type: CinemachineTargetGroupTarget

### See Also

**Reference**  
CinemachineTargetGroup Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Field

When to update the group's transform based on the position of the group members

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```
[TooltipAttribute("When to update the group's transform based on the position of the group members")]
public CinemachineTargetGroupUpdateMethod m_UpdateMethod
```

**Field Value**

Type: `CinemachineTargetGroupUpdateMethod`

**See Also**

Reference

- CinemachineTargetGroup Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupPositionMode Enumeration

How the group's position is calculated

Namespace: Cinemachine  
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[DocumentationSortingAttribute(19.2f, DocumentationSortingAttribute)] public enum PositionMode</td>
<td></td>
</tr>
</tbody>
</table>

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupCenter</td>
<td>0</td>
<td>Group position will be the center of the group's axis-aligned bounding box</td>
</tr>
<tr>
<td>GroupAverage</td>
<td>1</td>
<td>Group position will be the weighted average of the positions of the members</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTargetGroupRotationMode

How the group's orientation is calculated

**Namespace**: Cinemachine

**Assembly**: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

C#  

```csharp
public enum RotationMode
```

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual</td>
<td>0</td>
<td>Manually set in the group's transform</td>
</tr>
<tr>
<td>GroupAverage</td>
<td>1</td>
<td>Weighted average of the orientation of its members.</td>
</tr>
</tbody>
</table>

### See Also

Reference

Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTargetGroupTarget Structure

Holds the information that represents a member of the group

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

C#  
```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(19.1f, DocumentationSortingAttribute)
public struct Target
```

The **CinemachineTargetGroupTarget** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius</td>
<td>The radius of the target, used for calculating the bounding box. Cannot be negative</td>
</tr>
<tr>
<td>target</td>
<td>The target objects. This object's position and orientation will contribute to the group's average position and orientation, in accordance with its weight</td>
</tr>
<tr>
<td>weight</td>
<td>How much weight to give the target</td>
</tr>
</tbody>
</table>
when averaging. Cannot be negative

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## Target Fields

The `CinemachineTargetGroupTarget` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>radius</code></td>
<td>The radius of the target, used for calculating the bounding box. Cannot be negative</td>
</tr>
<tr>
<td><code>target</code></td>
<td>The target objects. This object's position and orientation will contribute to the group's average position and orientation, in accordance with its weight</td>
</tr>
<tr>
<td><code>weight</code></td>
<td>How much weight to give the target when averaging. Cannot be negative</td>
</tr>
</tbody>
</table>

### See Also

- Reference: [CinemachineTargetGroupTarget Structure](#)
- [Cinemachine Namespace](#)

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupTargetRadius

Field

The radius of the target, used for calculating the bounding box. Cannot be negative.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
[TooltipAttribute("The radius of the target, used for calculating the bounding box. Cannot be negative")]
public float radius
```

### Field Value

Type: **Single**

### See Also

**Reference**

- [CinemachineTargetGroupTarget Structure](#)
- [Cinemachine Namespace](#)

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTargetGroupTargetTarget Target Field

The target objects. This object's position and orientation will contribute to the group's average position and orientation, in accordance with its weight.

**Namespace**: Cinemachine
**Assembly**: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The target objects. This object's position and orientation will contribute to the group's average position and orientation, in accordance with its weight")]
public Transform target
```

**Field Value**
**Type**: `Transform`

### See Also

**Reference**
- CinemachineTargetGroupTarget Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupTargetWeight Field

How much weight to give the target when averaging. Cannot be negative

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public float weight</td>
<td></td>
</tr>
</tbody>
</table>

[TooltipAttribute("How much weight to give the target when averaging. Cannot be negative")]

### Field Value

Type: **Single**

### See Also

Reference  
- CinemachineTargetGroupTarget Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTargetGroupUpdateMethod Enumeration

This enum defines the options available for the update method.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public enum UpdateMethod</td>
<td></td>
</tr>
</tbody>
</table>

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update</td>
<td>0</td>
<td>Updated in normal MonoBehaviour Update.</td>
</tr>
<tr>
<td>FixedUpdate</td>
<td>1</td>
<td>Updated in sync with the Physics module, in FixedUpdate</td>
</tr>
<tr>
<td>LateUpdate</td>
<td>2</td>
<td>Updated in MonoBehaviour LateUpdate.</td>
</tr>
</tbody>
</table>

See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDolly Class

A Cinemachine Virtual Camera Body component that constrains camera motion to a CinemachinePath. The camera can move along the path. This behaviour can operate in two modes: manual positioning, and Auto-Dolly positioning. In Manual mode, the camera's position is specified by animating the Path Position field. In Auto-Dolly mode, the Path Position field is animated automatically every frame by finding the position on the path that's closest to the virtual camera's Follow target.

Inheritance Hierarchy

```
SystemObject  Object
  Component
    Behaviour
      MonoBehaviour
        CinemachineCinemachineComponentBase
          CinemachineCinemachineTrackedDolly
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(7f, DocumentationSortingAttribute.
[AddComponentMenu("")]
[RequireComponent(typeof(CinemachinePipeline))]
[SaveDuringPlayAttribute]
public class CinemachineTrackedDolly : Cinemachin
```

The CinemachineTrackedDolly type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineTrackedDolly</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>

## Top

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a path (Overrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage (Overrides CinemachineComponentBaseStage.)</td>
</tr>
<tr>
<td>VcamState</td>
<td>Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)</td>
</tr>
</tbody>
</table>
**VirtualCamera**

Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Positions the virtual camera according to the transposer rules. (Overrides CinemachineComponentBaseMutateCameraState(CameraState Single).)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position dragged. Implementation adds the delta to the follow offset. (Overrides CinemachineComponentBaseOnPositionDragged(Vector3).)</td>
</tr>
</tbody>
</table>

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AutoDolly</td>
<td>Controls how automatic dollying occurs.</td>
</tr>
<tr>
<td>m_CameraUp</td>
<td>How to set the virtual camera's Up vector. This will affect the screen composition.</td>
</tr>
<tr>
<td>m_Path</td>
<td>The path to which the camera will be constrained. This must be non-null.</td>
</tr>
<tr>
<td>m_PathOffset</td>
<td>Where to put the camera</td>
</tr>
</tbody>
</table>
relative to the path position. X is perpendicular to the path, Y is up, and Z is parallel to the path.

- **m_PathPosition**
  The position along the path at which the camera will be placed. This can be animated directly, or set automatically by the Auto-Dolly feature to get as close as possible to the Follow target.

- **m_PitchDamping**
  "How aggressively the camera tries to track the target rotation's X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera."

- **m_PositionUnits**
  How to interpret the Path Position

- **m_RollDamping**
  How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

- **m_XDamping**
  How aggressively the camera tries to maintain the offset perpendicular to the path. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers
give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

| m_YawDamping | How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera. |
| m_YDamping | How aggressively the camera tries to maintain the offset in the path-local up direction. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors |
| m_ZDamping | How aggressively the camera tries to maintain the offset parallel to the path. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors |
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDolly Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineTrackedDolly.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#        JavaScript

```csharp
public CinemachineTrackedDolly()
```

See Also

Reference
CinemachineTrackedDolly Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDolly Properties

The `CinemachineTrackedDolly` type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a path (Overrides <code>CinemachineComponentBaseIsValid</code>).</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements.  Always returns the Body stage (Overrides <code>CinemachineComponentBaseStage</code>).</td>
</tr>
<tr>
<td><strong>VcamState</strong></td>
<td>Returns the owner vcam's CameraState. (Inherited from <code>CinemachineComponentBase</code>.)</td>
</tr>
</tbody>
</table>
VirtualCamera
Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)

See Also
Reference
CinemachineTrackedDolly Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollyIsValid Property

True if component is enabled and has a path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public override bool IsValid { get; }
```

Property Value  
Type: Boolean

**See Also**

Reference  
CinemachineTrackedDolly Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollyStage Property

Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```
public override CinemachineCoreStage Stage {
    get;
}
```

**Property Value**

Type: `CinemachineCoreStage`

**See Also**

Reference

- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineTrackedDolly Methods

The CinemachineTrackedDolly type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MutateCameraState</td>
<td>Positions the virtual camera according to the transposer rules. (Overrides CinemachineComponentBaseMutateCameraState (Single).)</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user. (Overrides CinemachineComponentBaseOnPositionDragged (Single).)</td>
</tr>
</tbody>
</table>

See Also

Reference
- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDollyMutateCameraState Method

Positions the virtual camera according to the transposer rules.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public override void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)
```

**Parameters**

- **curState**  
  Type: CinemachineCameraState  
  The current camera state

- **deltaTime**  
  Type: System.Single  
  Used for damping. If less that 0, no damping is done.

**See Also**

Reference  
CinemachineTrackedDolly Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollyOnPositionDragged Method

API for the editor, to process a position drag from the user. This implementation adds the delta to the follow offset.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public override void OnPositionDragged(
    Vector3 delta
)
```

#### Parameters

- **delta**  
  Type: `Vector3`  
  The amount dragged this frame

### See Also

**Reference**  
CinemachineTrackedDolly Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDolly Fields

The **CinemachineTrackedDolly** type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_AutoDolly</td>
<td>Controls how automatic dollying occurs</td>
</tr>
<tr>
<td>m_CameraUp</td>
<td>How to set the virtual camera's Up vector. This will affect the screen composition.</td>
</tr>
<tr>
<td>m_Path</td>
<td>The path to which the camera will be constrained. This must be non-null.</td>
</tr>
<tr>
<td>m_PathOffset</td>
<td>Where to put the camera relative to the path position. X is perpendicular to the path, Y is up, and Z is parallel to the path.</td>
</tr>
<tr>
<td>m_PathPosition</td>
<td>The position along the path at which the camera will be placed. This can be animated directly, or set automatically by the Auto-Dolly feature to get as close as possible to the Follow target.</td>
</tr>
<tr>
<td>m_PitchDamping</td>
<td>&quot;How aggressively the camera tries to track the target rotation's X angle. Small</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>m_PositionUnits</td>
<td>How to interpret the Path Position</td>
</tr>
<tr>
<td>m_RollDamping</td>
<td>How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_XDamping</td>
<td>How aggressively the camera tries to maintain the offset perpendicular to the path. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.</td>
</tr>
<tr>
<td>m_YawDamping</td>
<td>How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_YDamping</td>
<td>How aggressively the camera tries to maintain the offset in</td>
</tr>
</tbody>
</table>
the path-local up direction. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

| m_ZDamping | How aggressively the camera tries to maintain the offset parallel to the path. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors |

See Also

Reference
- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDollym_AutoDolly

Field

Controls how automatic dollying occurs

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("Controls how automatic dollying occurs.
A Follow target is necessary to use this feature.")]
public CinemachineTrackedDollyAutoDolly m_AutoDolly
``` |

Field Value

Type: CinemachineTrackedDollyAutoDolly

### See Also

**Reference**

- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_Camer Field

How to set the virtual camera's Up vector. This will affect the screen composition.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("How to set the virtual camera's Up vector. This will affect the screen composition, because the camera Aim behaviours will always try to respect the Up direction.")]
public CinemachineTrackedDollyCameraUpMode m_CameraUp
```

### Field Value

**Type:** CinemachineTrackedDollyCameraUpMode

### See Also

**Reference**  
CinemachineTrackedDolly Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDolly.m_Path Field

The path to which the camera will be constrained. This must be non-null.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("The path to which the camera will be constrained. This must be non-null.")]
public CinemachinePathBase m_Path |

#### Field Value

Type: CinemachinePathBase

### See Also

**Reference**
- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDolly m_PathOffset Field

Where to put the camera relative to the path position. X is perpendicular to the path, Y is up, and Z is parallel to the path.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[TooltipAttribute("Where to put the camera relative to the path position. X is perpendicular to the path, Y is up, and Z is parallel to the path.
This allows the camera to be offset from the path itself (as if on a tripod, for example).")]
public Vector3 m_PathOffset
```

**Field Value**  
**Type:** `Vector3`

### See Also

**Reference**  
- CinemachineTrackedDolly Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_PathPosition Field

The position along the path at which the camera will be placed. This can be animated directly, or set automatically by the Auto-Dolly feature to get as close as possible to the Follow target.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[TooltipAttribute("The position along the path at which the camera will be placed.
This can be animated directly, or set automatically by the Auto-Dolly feature to get as close as possible to the Follow target.
The value is interpreted according to the Position Units setting.")]
public float m_PathPosition
```

Field Value
Type: Single

See Also

Reference
CinemachineTrackedDolly Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_PitchDamping Field

"How aggressively the camera tries to track the target rotation's X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#  
```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to track the target rotation's X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.")]
public float m_PitchDamping
```

Field Value
Type: Single

See Also

Reference
CinemachineTrackedDolly Class
Cinemachine Namespace
Cinemachine
CinemachineTrackedDollym_PositionField

How to interpret the Path Position

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("How to interpret Path Position.
public CinemachinePathBasePositionUnits m_PositionUnits
``` |

Field Value

Type: CinemachinePathBasePositionUnits

### See Also

**Reference**

CinemachineTrackedDolly Class

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_RollDamping Field

How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[RangeAttribute(0f, 20f)]</td>
<td></td>
</tr>
<tr>
<td>[TooltipAttribute(&quot;How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.&quot;)]</td>
<td></td>
</tr>
</tbody>
</table>

```csharp
class CinemachineTrackedDolly
{
    public float m_RollDamping
}
```

**Field Value**  
**Type:** Single

### See Also

**Reference**  
- CinemachineTrackedDolly Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_XDamping Field

How aggressively the camera tries to maintain the offset perpendicular to the path. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain its position in a direction perpendicular to the path.
Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.")]
public float m_XDamping
```

Field Value

Type: Single

See Also

Reference
CinemachineTrackedDolly Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Field

How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://unity3d.com/legal/terms-of-service" alt="Code" /></td>
<td></td>
</tr>
</tbody>
</table>

Field Value

Type: Single

See Also

Reference

CinemachineTrackedDolly Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_YDamping

Field

How aggressively the camera tries to maintain the offset in the path-local up direction. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain its position in the path-local up direction. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.")]
public float m_YDamping
```

Field Value

Type: Single

See Also

Reference
- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollym_ZDamping Field

How aggressively the camera tries to maintain the offset parallel to the path. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[RangedAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain its position in a direction parallel to the path.
Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.")]
public float m_ZDamping
```

### Field Value

Type: **Single**

### See Also

**Reference**
- CinemachineTrackedDolly Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollyAutoDolly Structure

Controls how automatic dollying occurs

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Syntax" /></td>
<td><img src="image" alt="Syntax" /></td>
</tr>
</tbody>
</table>

The CinemachineTrackedDollyAutoDolly type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="CinemachineTrackeDollyAutoDolly" /></td>
<td>Constructor with specific field values</td>
</tr>
</tbody>
</table>

### Fields
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Enabled</td>
<td>If checked, will enable automatic dolly, which chooses a path position that is as close as possible to the Follow target.</td>
</tr>
<tr>
<td>m_PositionOffset</td>
<td>Offset, in current position units, from the closest point on the path to the follow target.</td>
</tr>
<tr>
<td>m_SearchRadius</td>
<td>Search up to how many waypoints on either side of the current position. Use 0 for Entire path</td>
</tr>
<tr>
<td>m_SearchResolution</td>
<td>We search between waypoints by dividing the segment into this many straight pieces. The higher the number, the more accurate the result, but performance is proportionally slower for higher numbers</td>
</tr>
</tbody>
</table>

**See Also**

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDollyAutoDolly Constructor

Constructor with specific field values

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
public AutoDolly(
    bool enabled,
    float positionOffset,
    int searchRadius,
    int stepsPerSegment
)
```

**Parameters**

- **enabled**
  Type: SystemBoolean

- **positionOffset**
  Type: SystemSingle

- **searchRadius**
  Type: SystemInt32

- **stepsPerSegment**
Type: `SystemInt32`


See Also

Reference

CinemachineTrackedDollyAutoDolly Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
## AutoDolly Fields

The **CinemachineTrackedDollyAutoDolly** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_Enabled</td>
<td>If checked, will enable automatic dolly, which chooses a path position that is as close as possible to the Follow target.</td>
</tr>
<tr>
<td>m_PositionOffset</td>
<td>Offset, in current position units, from the closest point on the path to the follow target.</td>
</tr>
<tr>
<td>m_SearchRadius</td>
<td>Search up to how many waypoints on either side of the current position. Use 0 for Entire path.</td>
</tr>
<tr>
<td>m_SearchResolution</td>
<td>We search between waypoints by dividing the segment into this many straight pieces. The higher the number, the more accurate the result, but performance is proportionally slower for higher numbers</td>
</tr>
</tbody>
</table>
See Also

Reference
CinemachineTrackedDollyAutoDolly Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDollyAutoDolly Field

If checked, will enable automatic dolly, which chooses a path position that is as close as possible to the Follow target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="C# Code" /></td>
<td><img src="image" alt="JavaScript Code" /></td>
</tr>
</tbody>
</table>

Field Value  
Type: Boolean

### See Also

- **Reference**  
  - CinemachineTrackedDollyAutoDolly Structure  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTrackedDollyAutoDolly

Field

Offset, in current position units, from the closest point on the path to the follow target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public float m_PositionOffset
```

**JavaScript**

```javascript
// TODO: Implement JavaScript code
```

### Field Value

Type: **Single**

### See Also

**Reference**

- [CinemachineTrackedDollyAutoDolly Structure](#)
- [Cinemachine Namespace](#)

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
**CinemachineTrackedDollyAutoDolly Field**

Search up to how many waypoints on either side of the current position. Use 0 for Entire path

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
[TooltipAttribute("Search up to how many waypoint
public int m_SearchRadius
```

**Field Value**  
Type: Int32

### See Also

**Reference**  
CinemachineTrackedDollyAutoDolly Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollyAutoDolly Field

We search between waypoints by dividing the segment into this many straight pieces. The higher the number, the more accurate the result, but performance is proportionally slower for higher numbers.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[FormerlySerializedAsAttribute("m_StepsPerSegment")]
[TooltipAttribute("We search between waypoints by dividing the segment into this many straight pieces. The higher the number, the more accurate the result, but performance is proportionally slower for higher numbers")]
public int m_SearchResolution
```

Field Value

Type: Int32

See Also

Reference
- CinemachineTrackedDollyAutoDolly Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTrackedDollyCameraUp Mode

Different ways to set the camera's up vector

**namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
[DocumentationSortingAttribute(7.1f, DocumentationSortingAttribute)
public enum CameraUpMode
```

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>0</td>
<td>Leave the camera's up vector alone. It will be set according to the Brain's WorldUp.</td>
</tr>
<tr>
<td>Path</td>
<td>1</td>
<td>Take the up vector from the path's up vector at the current point</td>
</tr>
<tr>
<td>PathNoRoll</td>
<td>2</td>
<td>Take the up vector from the path's up</td>
</tr>
</tbody>
</table>
vector at the current point, but with the roll zeroed out

<table>
<thead>
<tr>
<th>FollowTarget</th>
<th>3</th>
<th>Take the up vector from the Follow target's up vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>FollowTargetNoRoll</td>
<td>4</td>
<td>Take the up vector from the Follow target's up vector, but with the roll zeroed out</td>
</tr>
</tbody>
</table>

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTransposer Class

This is a CinemachineComponent in the Body section of the component pipeline. Its job is to position the camera in a fixed relationship to the vcam's Follow target object, with offsets and damping. The Transposer will only change the camera's position in space. It will not re-orient or otherwise aim the camera. To do that, you need to instruct the vcam in the Aim section of its pipeline.

Inheritance Hierarchy

```
System
  Object
  Component
    Behaviour
      MonoBehaviour
        Cinemachine
          CinemachineComponentBase
            CinemachineTransposer
            CinemachineOrbitalTransposer
```

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(5f, Documentation$]
[AddComponentMenu('')]
[RequireComponent(typeof(CinemachinePipeline))]
[SaveDuringPlayAttribute]
public class CinemachineTransposer : Cinemachine(    
```

The CinemachineTransposer type exposes the following members.
## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineTransposer</td>
<td></td>
</tr>
</tbody>
</table>

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AngularDamping</td>
<td>Damping speeds for each of the 3 axes of the target's rotation</td>
</tr>
<tr>
<td>Damping</td>
<td>Damping speeds for each of the 3 axes of the offset from target</td>
</tr>
<tr>
<td>EffectiveOffset</td>
<td>Get the target offset, with sanitization</td>
</tr>
<tr>
<td>FollowTarget</td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>IsValid</td>
<td>True if component is enabled and has a valid Follow target</td>
</tr>
<tr>
<td>LookAtTarget</td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td>Stage</td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage</td>
</tr>
</tbody>
</table>
VcamState
Returns the owner vcam's CameraState.
(Inherited from CinemachineComponentBase.)

VirtualCamera
Get the associated CinemachineVirtualCameraBase
(Inherited from CinemachineComponentBase.)

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTargetCameraPosition</td>
<td>Internal API for the Inspector Editor, to draw a marker at the target.</td>
</tr>
<tr>
<td>GetReferenceOrientation</td>
<td>Internal API for the Inspector Editor, to draw a marker at the target.</td>
</tr>
<tr>
<td>InitPrevFrameStateInfo</td>
<td>Initializes the state for previous frame if appropriate.</td>
</tr>
<tr>
<td>MutateCameraState</td>
<td>Positions the virtual camera according to the transposer rules.</td>
</tr>
<tr>
<td>(Overrides CinemachineComponentBase.MutateCameraState(CameraState Single)).</td>
<td></td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position dragged from the user.</td>
</tr>
<tr>
<td>(Overrides CinemachineComponentBase.OnPositionDragged(Vector3)).</td>
<td></td>
</tr>
<tr>
<td>OnValidate</td>
<td></td>
</tr>
</tbody>
</table>
**TrackTarget** Positions the virtual camera according to the transposer rules.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BindingMode</td>
<td>The coordinate space to use when interpreting the offset from the target</td>
</tr>
<tr>
<td>m_FollowOffset</td>
<td>The distance which the transposer will attempt to maintain from the transposer subject</td>
</tr>
<tr>
<td>m_PitchDamping</td>
<td>How aggressively the camera tries to track the target rotation’s X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_RollDamping</td>
<td>How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_XDamping</td>
<td>How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset.</td>
</tr>
</tbody>
</table>
Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_YawDamping</strong></td>
<td>How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td><strong>m_YDamping</strong></td>
<td>How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors</td>
</tr>
<tr>
<td><strong>m_ZDamping</strong></td>
<td>How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposer Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineTransposer.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
public CinemachineTransposer()
```

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The **CinemachineTransposer** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AngularDamping</strong></td>
<td>Damping speeds for each of the 3 axes of the target's rotation</td>
</tr>
<tr>
<td><strong>Damping</strong></td>
<td>Damping speeds for each of the 3 axes of the offset from target</td>
</tr>
<tr>
<td><strong>EffectiveOffset</strong></td>
<td>Get the target offset, with sanitization</td>
</tr>
<tr>
<td><strong>FollowTarget</strong></td>
<td>Returns the owner vcam's Follow target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>IsValid</strong></td>
<td>True if component is enabled and has a valid Follow target. (Overrrides CinemachineComponentBaseIsValid.)</td>
</tr>
<tr>
<td><strong>LookAtTarget</strong></td>
<td>Returns the owner vcam's LookAt target. (Inherited from CinemachineComponentBase.)</td>
</tr>
<tr>
<td><strong>Stage</strong></td>
<td>Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage</td>
</tr>
</tbody>
</table>
VcamState

Returns the owner vcam's CameraState. (Inherited from CinemachineComponentBase.)

VirtualCamera

Get the associated CinemachineVirtualCameraBase (Inherited from CinemachineComponentBase.)

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerAngularDamping Property

Damping speeds for each of the 3 axes of the target's rotation

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C# JavaScript

protected Vector3 AngularDamping { get; }

Property Value
Type: Vector3

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerDamping Property

Damping speeds for each of the 3 axes of the offset from target

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
protected Vector3 Damping { get; }
```

### Property Value

**Type:** Vector3

### See Also

**Reference**  
CinemachineTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTransposerEffectiveOffset Property

Get the target offset, with sanitization

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected Vector3 EffectiveOffset { get; }
```

**Property Value**  
Type: **Vector3**

### See Also

**Reference**  
CinemachineTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTransposerIsValid Property

True if component is enabled and has a valid Follow target.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public override</td>
<td>bool IsValid { get; }</td>
</tr>
</tbody>
</table>

**Property Value**

Type: Boolean

**See Also**

Reference

CinemachineTransposer Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineTransposerStage Property

Get the Cinemachine Pipeline stage that this component implements. Always returns the Body stage

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

**C#**

```csharp
public override CinemachineCoreStage Stage { get; }
```

**JavaScript**

### See Also

- **Reference**  
  - CinemachineTransposer Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTransposer
Methods

The **CinemachineTransposer** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetTargetCameraPosition</td>
<td>Internal API for the Inspector Editor, so it can draw a marker at the target</td>
</tr>
<tr>
<td>GetReferenceOrientation</td>
<td>Internal API for the Inspector Editor, so it can draw a marker at the target</td>
</tr>
<tr>
<td>InitPrevFrameStateInfo</td>
<td>Initializes the state for previous frame if appropriate.</td>
</tr>
<tr>
<td>MutateCameraState</td>
<td>Positions the virtual camera according to the transposer rules.</td>
</tr>
<tr>
<td>(Overrides CinemachineComponentBase.MutateCameraState(CameraState Single))</td>
<td></td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user.</td>
</tr>
<tr>
<td>implementation adds the delta to the follow offset.</td>
<td></td>
</tr>
<tr>
<td>(Overrides CinemachineComponentBase.OnPositionDragged(Vector3))</td>
<td></td>
</tr>
<tr>
<td>OnValidate</td>
<td></td>
</tr>
<tr>
<td>TrackTarget</td>
<td>Positions the virtual camera according to the transposer rules.</td>
</tr>
</tbody>
</table>

---

**See Also**
Reference

CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerGeTargetCameraPosition Method

Internal API for the Inspector Editor, so it can draw a marker at the target

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public Vector3 GeTargetCameraPosition(
    Vector3 worldUp
)
```

### Parameters

**worldUp**  
Type: Vector3  

### Return Value

Type: Vector3  

### See Also

**Reference**  
CinemachineTransposer Class  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposer.GetReferenceOrientation Method

Internal API for the Inspector Editor, so it can draw a marker at the target

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

**C#**
```csharp
public Quaternion GetReferenceOrientation(
    Vector3 worldUp
)
```

### Parameters

**worldUp**
Type: Vector3


### Return Value

Type: Quaternion  


### See Also

Reference  
CinemachineTransposer Class  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposer.InitPrevFrameStateInfo Method

Initializes the state for previous frame if appropriate.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
protected void InitPrevFrameStateInfo(
    ref CameraState curState,
    float deltaTime
)
```

**Parameters**

- **curState**
  - Type: Cinemachine.CameraState

- **deltaTime**
  - Type: System.Single

**See Also**

Reference
CinemachineTransposer Class
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerMutateCamMethod

Positions the virtual camera according to the transposer rules.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

### C#  
```csharp
public override void MutateCameraState(
    ref CameraState curState,
    float deltaTime
)
```

### JavaScript

**Parameters**

- **curState**  
  - Type: `CinemachineCameraState`  
  - The current camera state

- **deltaTime**  
  - Type: `System.Single`  
  - Used for damping. If less than 0, no damping is done.

**See Also**

Reference

- CinemachineTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTransposerOnPositionDragged Method

API for the editor, to process a position drag from the user. This implementation adds the delta to the follow offset.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#

```csharp
public override void OnPositionDragged(  
    Vector3 delta  
)
```

Parameters

delta

Type: Vector3
The amount dragged this frame

See Also

Reference

CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerOnValidate Method


Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
protected virtual void OnValidate();

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerTrackTarget Method

Positions the virtual camera according to the transposer rules.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
protected void TrackTarget(
    float deltaTime,
    Vector3 up,
    Vector3 desiredCameraOffset,
    out Vector3 outTargetPosition,
    out Quaternion outTargetOrient
)
```

### Parameters

- **deltaTime**  
  Type: System.Single  
  Used for damping. If less than 0, no damping is done.

- **up**  
  Type: Vector3  
  Current camera up

- **desiredCameraOffset**  
  Type: Vector3  
  Where we want to put the camera relative to the follow target

- **outTargetPosition**  
  Type: Vector3  
  Resulting camera position
outTargetOrient
Type: Quaternion
Damped target orientation

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
## CinemachineTransposer Fields

The `CinemachineTransposer` type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_BindingMode</td>
<td>The coordinate space to use when interpreting the offset from the target.</td>
</tr>
<tr>
<td>m_FollowOffset</td>
<td>The distance which the transposer will attempt to maintain from the transposer subject.</td>
</tr>
<tr>
<td>m_PitchDamping</td>
<td>How aggressively the camera tries to track the target rotation's X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_RollDamping</td>
<td>How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
<tr>
<td>m_XDamping</td>
<td>How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
</tbody>
</table>
more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors

<table>
<thead>
<tr>
<th>m_YawDamping</th>
<th>How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_YDamping</td>
<td>How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors</td>
</tr>
<tr>
<td>m_ZDamping</td>
<td>How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera.</td>
</tr>
</tbody>
</table>
camera. Using different settings per axis can yield a wide range of camera behaviors.

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
Field

The coordinate space to use when interpreting the offset from the target

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public</td>
<td><code>CinemachineTransposerBindingMode m_BINDING</code></td>
</tr>
</tbody>
</table>

### Field Value

Type: `CinemachineTransposerBindingMode`

### See Also

- Reference
  - CinemachineTransposer Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerm_FollowOffset Field

The distance which the transposer will attempt to maintain from the transposer subject

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[TooltipAttribute("The distance vector that the transposer will attempt to maintain from the Follow target")]
public Vector3 m_FollowOffset
```

Field Value
Type: Vector3

See Also

Reference
CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerm_PitchDamping Field

How aggressively the camera tries to track the target rotation’s X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tracks the target rotation’s X angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.")] public float m_PitchDamping
```

Field Value  
Type: Single

See Also

Reference  
CinemachineTransposer Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTransposerm_RollDamp Field

How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to track the target rotation's Z angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.")]
public float m_RollDamping
```

**Field Value**

**Type:** Single

### See Also

**Reference**
- CinemachineTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineTransposerm_XDamping Field

How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain the offset in the X-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's x-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors."])]
public float m_XDamping
```

Field Value

Type: **Single**

### See Also

**Reference**
- CinemachineTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineTransposer.

Field

How aggressively the camera tries to track the target rotation's Y angle. Small numbers are more responsive. Larger numbers give a more heavy slowly responding camera.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```
public float m_YawDamping
```

Field Value

Type: Single

See Also

Reference

CinemachineTransposer Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Field

How aggressively the camera tries to maintain the offset in the Y-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's y-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[RangeAttribute(0f, 20f)]</code></td>
<td><code>[RangeAttribute(0f, 20f)]</code></td>
</tr>
<tr>
<td><code>[TooltipAttribute(&quot;How aggressively the camera...&quot;)</code></td>
<td><code>[TooltipAttribute(&quot;How aggressively the camera...&quot;)</code></td>
</tr>
<tr>
<td><code>public float m_YDamping</code></td>
<td><code>public float m_YDamping</code></td>
</tr>
</tbody>
</table>

Field Value

Type: **Single**

### See Also

**Reference**
- CinemachineTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTransposerm_ZDamping Field

How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
[RangeAttribute(0f, 20f)]
[TooltipAttribute("How aggressively the camera tries to maintain the offset in the Z-axis. Small numbers are more responsive, rapidly translating the camera to keep the target's z-axis offset. Larger numbers give a more heavy slowly responding camera. Using different settings per axis can yield a wide range of camera behaviors.")]
public float m_ZDamping
```

Field Value
Type: Single

See Also

Reference
- CinemachineTransposer Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineTransposerBindingMode Enumeration

The coordinate space to use when interpreting the offset from the target

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

## Syntax

```csharp
[DocumentationSortingAttribute(5.01f, Documentatio
public enum BindingMode
```

## Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LockToTargetOnAssign</td>
<td>0</td>
<td>Camera will be bound to the Follow target using a frame of reference consisting of the target's local frame at the moment when the camera is bound.</td>
</tr>
</tbody>
</table>
virtual camera was enabled, or when the target was assigned.

<table>
<thead>
<tr>
<th>Function</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LockToTargetWithWorldUp</td>
<td>1</td>
</tr>
<tr>
<td>Camera will be bound to the Follow target using a frame of reference consisting of the target's local frame, with the tilt and roll zeroed out.</td>
<td></td>
</tr>
<tr>
<td>LockToTargetNoRoll</td>
<td>2</td>
</tr>
<tr>
<td>Camera will be bound to the Follow target using a frame of reference consisting of the target's local frame, with the roll zeroed out.</td>
<td></td>
</tr>
<tr>
<td>LockToTarget</td>
<td>3</td>
</tr>
<tr>
<td>Camera will be bound to the Follow target using the target's</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>WorldSpace</td>
<td>4</td>
</tr>
<tr>
<td>-------------</td>
<td>---</td>
</tr>
<tr>
<td>SimpleFollowWithWorldUp</td>
<td>5</td>
</tr>
</tbody>
</table>

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCamera Class

This behaviour is intended to be attached to an empty Transform GameObject, and it represents a Virtual Camera within the Unity scene. The Virtual Camera will animate its Transform according to the rules contained in its CinemachineComponent pipeline (Aim, Body, and Noise). When the virtual camera is Live, the Unity camera will assume the position and orientation of the virtual camera. A virtual camera is not a camera. Instead, it can be thought of as a camera controller, not unlike a cameraman. It can drive the Unity Camera and control its position, orientation, lens settings, and PostProcessing effects. Each Virtual Camera owns its own Cinemachine Component Pipeline, through which you provide the instructions for dynamically tracking specific game objects. A virtual camera is very lightweight, and does no rendering of its own. It merely tracks interesting GameObjects, and positions itself accordingly. A typical game can have dozens of virtual cameras, each set up to follow a particular character or capture a particular event. A Virtual Camera can be in any of three states: *
**Live**: The virtual camera is actively controlling the Unity Camera. The virtual camera is tracking its targets and being updated every frame. **Standby**: The virtual camera is tracking its targets and being updated every frame, but no Unity Camera is actively being controlled by it. This is the state of a virtual camera that is enabled in the scene but perhaps at a lower priority than the Live virtual camera. **Disabled**: The virtual camera is present but disabled in the scene. It is not actively tracking its targets and so consumes no processing power. However, the virtual camera can be made live from the Timeline. The Unity Camera can be driven by any virtual camera in the scene. The game logic can choose the virtual camera to make live by manipulating the virtual cameras' enabled flags and their priorities, based on game logic. In order to be driven by a virtual camera, the Unity Camera must have a CinemachineBrain behaviour, which will select the most eligible virtual camera based on its priority or on other criteria, and will manage blending.
Inheritance Hierarchy

SystemObject  Object
  Component
    Behaviour
      MonoBehaviour
        Cinemachine
          CinemachineVirtualCameraBase
            CinemachineVirtualCamera

Namespace:  Cinemachine
Assembly:  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript

```csharp
[DocumentationSortingAttribute(1f, DocumentationSortingAttribute.ExecuteInEditMode, DocumentationSortingAttribute.DisallowMultipleComponent)]
[AddComponentMenu("Cinemachine/CinemachineVirtualCamera")]
public class CinemachineVirtualCamera : CinemachineVirtualCameraBase
```

The CinemachineVirtualCamera type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineVirtualCamera</td>
<td>Top</td>
</tr>
</tbody>
</table>

Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description of the virtual camera, for use when displaying debug info. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the Follow target for the Body component in the Cinemachine Pipeline. If this vcam is a part of a meta-camera collection, then the owner's target is used if the local target is null. (Overrides <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Just returns self. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the LookAt target for the Aim component in the Cinemachine Pipeline. If this vcam is a part of a meta-camera collection, then the owner's target is used if the local target is null. (Overrides <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Base implementation returns the GameObject's name. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-camera support. In the situation where a virtual camera is the public face of a private meta-camera collection of virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update ignore deltaTime and reset itself. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. This determines its placement in CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>State</td>
<td>The CameraState object holds information necessary to position the Unity camera. It is the output of this class. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>UserIsDragging</td>
<td>API for the editor, to make the behavior of position handles behave better.</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed, for upgrading legacy. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>The GameObject owner of the Camera behaviour. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AddCinemachineComponentT</td>
<td>Add a component to the cinemachine pipeline.</td>
</tr>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into the state calculation pipeline.</td>
</tr>
<tr>
<td></td>
<td>be called after each pipeline stage, to allow others to hook into the</td>
</tr>
<tr>
<td></td>
<td>pipeline.</td>
</tr>
<tr>
<td></td>
<td><em>(Inherited from)</em></td>
</tr>
<tr>
<td>DestroyCinemachineComponentT</td>
<td>Remove a component from the cinemachine pipeline.</td>
</tr>
<tr>
<td></td>
<td><em>(Inherited from)</em></td>
</tr>
<tr>
<td>GetCinemachineComponent(CinemachineCoreStage)</td>
<td>Get the component set for a specific stage.</td>
</tr>
<tr>
<td>GetCinemachineComponentT</td>
<td>Get an existing component of a specific type from the cinemachine pipeline.</td>
</tr>
<tr>
<td>GetComponentOwner</td>
<td>Get the hidden CinemachinePipeline child object.</td>
</tr>
<tr>
<td>GetComponentPipeline</td>
<td>Get the component pipeline owned by the hidden child pipeline container.</td>
</tr>
<tr>
<td></td>
<td><em>(GetCinemachineComponentT method.)</em></td>
</tr>
<tr>
<td>InvalidateComponentPipeline</td>
<td>Editor API: Call this when changing the pipeline from the editor.</td>
</tr>
<tr>
<td></td>
<td><em>(Inherited from)</em></td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy</td>
</tr>
<tr>
<td></td>
<td>for all.</td>
</tr>
<tr>
<td></td>
<td><em>(Inherited from)</em></td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is a live child of this camera.</td>
</tr>
<tr>
<td></td>
<td>class implementation always returns false.</td>
</tr>
<tr>
<td></td>
<td><em>(Inherited from)</em></td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is sometimes</td>
</tr>
<tr>
<td></td>
<td>the need to push one to the top, making it the current Live camera with its</td>
</tr>
<tr>
<td></td>
<td>peers. Vcam is enabled: the most recent one goes to the top of the priority</td>
</tr>
<tr>
<td></td>
<td>queue with its peers.</td>
</tr>
<tr>
<td></td>
<td>Vcam is enabled: the most recent one goes to the top of the priority peers.</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OnDestroy</td>
<td>Calls DestroyPipelineDelegate for destroying the hidden child object, to support undo.</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Make sure that the pipeline cache is up-to-date.</td>
</tr>
<tr>
<td>OnPositionDragged</td>
<td>API for the editor, to process a position drag from the user.</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live.</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields, when changed in inspector.</td>
</tr>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of the vcam’s child cameras, the children are updating on FixedUpdate, then this will not necessarily be called.</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook callback.</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcam’s Follow target, or if that is null, will return the parent vcam’s Follow target.</td>
</tr>
</tbody>
</table>
ResolveLookAt

Start

Update

UpdateCameraState

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is there to support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>CreatePipelineOverride</td>
<td>Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do the same thing as the CreatePipeline method in this class.</td>
</tr>
<tr>
<td>s</td>
<td>DestroyPipelineOverride</td>
</tr>
<tr>
<td>---</td>
<td>--------------------------</td>
</tr>
<tr>
<td>m</td>
<td>m_ExcludedPropertiesInInspector</td>
</tr>
<tr>
<td>m</td>
<td>m_Follow</td>
</tr>
<tr>
<td>m</td>
<td>m_Lens</td>
</tr>
<tr>
<td>m</td>
<td>m_LockStageInInspector</td>
</tr>
<tr>
<td>m</td>
<td>m_LookAt</td>
</tr>
</tbody>
</table>
**m_Priority**

The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)

**OnPostPipelineStage**

A delegate to hook into the state calculation pipeline. Implementation must be sure to call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)

**PipelineName**

This is the name of the GameObject that will be created as a child object of the virtual camera. This hidden GameObject acts as a container for the polymorphic CinemachineComponent. The Inspector UI for the Camera provides access to the pipeline, as do the CinemachineComponent public methods in this class. The lifecycle of the pipeline GameObject is managed automatically.
See Also

Reference
Cinemachine Namespace
CinemachineCinemachineVirtualCameraBase
CinemachineLensSettings
CinemachineCinemachineComposer
CinemachineCinemachineTransposer
CinemachineCinemachineBasicMultiChannelPerlin

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCamera Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineVirtualCamera.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public CinemachineVirtualCamera()
```

See Also

Reference
CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCamera
Properties

The **CinemachineVirtualCamera** type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the Follow target for the Body component in the CinemachinePipeline. If this vcam is a part of a meta-camera collection, then the owner's target is used if the local target is null. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Just returns self. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the LookAt target for the Aim component in the CinemachinePipeline. If this vcam is a part of a meta-camera collection, then the owner's target is used if the local target is null. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Base implementation returns the owner GameObject's name. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-virtual-cameras is the situation where a virtual camera is the public face of a private army of virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform children of the parent vcam. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to ignore deltaTime and reset itself. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. This determines its placement in CinemachineCore's queue of eligible shots. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>State</td>
<td>The CameraState object holds all the information necessary to position the Unity camera. It is the output of this class. (Overrides CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td>UserIsDragging</td>
<td>API for the editor, to make the behavior of position handles behave better.</td>
</tr>
</tbody>
</table>
| **ValidatingStreamVersion** | Version that was last streamed
upgrading legacy
(Inherited from CinemachineVirtualCameraBase) |
|---------------------------|---------------------------------------------------------------------------------
| **VirtualCameraGameObject** | The GameObject owner of the Camera behaviour.
(Inherited from CinemachineVirtualCameraBase) |

See Also

Reference
CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraFollow Property

Get the Follow target for the Body component in the CinemachinePipeline. If this vcam is a part of a meta-camera collection, then the owner's target will be used if the local target is null.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C# | JavaScript
---|---

```csharp
public override Transform Follow { get; set; }
```

Property Value
Type: Transform
Implements ICinemachineCameraFollow

See Also

Reference
CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCamera

Property

Get the LookAt target for the Aim component in the CinemachinePipeline. If this vcam is a part of a meta-camera collection, then the owner's target will be used if the local target is null.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

public override Transform LookAt { get; set; }

Property Value
Type: Transform
Implements ICinemachineCameraLookAt

See Also

Reference
CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraState Property

The CameraState object holds all of the information necessary to position the Unity camera. It is the output of this class.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public override CameraState State { get; }
```

**JavaScript**

```javascript
// JavaScript
```

### Property Value

- **Type:** CameraState
- **Implements:** ICinemachineCameraState

### See Also

- **Reference**
  - CinemachineVirtualCamera Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraUserIsDragging Property

API for the editor, to make the dragging of position handles behave better.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public bool UserIsDragging { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Boolean

See Also

Reference
CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCamera
Methods

The `CinemachineVirtualCamera` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AddCinemachineComponentT</code></td>
<td>Add a component to the cinemachine pipeline.</td>
</tr>
<tr>
<td><code>AddPostPipelineStageHook</code></td>
<td>A delegate to hook into the state calculation pipeline. Be called after each pipeline stage, to allow others to hook into the pipeline.</td>
</tr>
<tr>
<td><code>DestroyCinemachineComponentT</code></td>
<td>Remove a component from the cinemachine pipeline.</td>
</tr>
<tr>
<td><code>GetCinemachineComponent(CinemachineCoreStage)</code></td>
<td>Get the component set for a specific stage.</td>
</tr>
<tr>
<td><code>GetCinemachineComponentT</code></td>
<td>Get an existing component of a specific type from the cinemachine pipeline.</td>
</tr>
<tr>
<td><code>GetComponentOwner</code></td>
<td>Get the hidden CinemachinePipeline child object.</td>
</tr>
<tr>
<td><code>GetComponentPipeline</code></td>
<td>Get the component pipeline owned by the hidden child pipeline container.</td>
</tr>
<tr>
<td><code>InvalidateComponentPipeline</code></td>
<td>Editor API: Call this when changing the pipeline from the editor.</td>
</tr>
<tr>
<td><code>InvokePostPipelineStageCallback</code></td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all (Inherited)</td>
</tr>
</tbody>
</table>

(Notes: (Inherited) indicates that the method is inherited from another class.)
IsLiveChild

Check whether the vcam is a live child of this camera. Class implementation always returns false.

MoveToTopOfPrioritySubqueue

When multiple virtual cameras have the highest priority, there is sometimes the need to push one to the top, making it the current Live camera if vcam is enabled: the most recent one goes to the top of the priority subqueue.

OnDestroy

Calls the DestroyPipelineDelegate for destroying the hidden child object, to support undo.

OnDisable

Base class implementation makes sure the priority queue remains up-to-date.

OnEnable

Make sure that the pipeline cache is up-to-date.

OnPositionDragged

API for the editor, to process a position drag from the user.

OnTransformParentChanged

Base class implementation makes sure the priority queue remains up-to-date.

OnTransitionFromCamera

Notification that this virtual camera is going live. The implementation must be called by any overridden method.

OnValidate

Enforce bounds for fields, when changed in inspector.

PreUpdateChildCameras

This is called prior to the updating of the vcam's child
cameras, the children are updating on `FixedUpdate`, then this will not necessarily be called on `LateUpdate` instead.

Item
- **RemovePostPipelineStageHook**
- **ResolveFollow**
- **ResolveLookAt**
- **Start**
- **Update**
- **UpdateCameraState**

**See Also**

Reference
- CinemachineVirtualCamera Class
- Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraAddCinemachineComponent Method

Add a component to the cinemachine pipeline.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C# | JavaScript
---|---

```csharp
public T AddCinemachineComponent<T>()
where T : CinemachineComponentBase
```

Type Parameters

\( T \)

Return Value

Type: \( T \)

[Missing <returns> documentation for "M:Cinemachine.CinemachineVirtualCamera.AddCinemachineComponent`1"]

See Also

Reference

CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraDestroyCinemachineComponent Method

Remove a component from the cinemachine pipeline.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public void DestroyCinemachineComponent&lt;T&gt;()</td>
<td>where T : CinemachineComponentBase</td>
</tr>
</tbody>
</table>

**Type Parameters**

*T*

### See Also

**Reference**
- CinemachineVirtualCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# CinemachineVirtualCamera.GetCinemachineComponent Method

## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GetCinemachineComponentT</code></td>
<td>Get an existing component of a specific type from the cinemachine pipeline.</td>
</tr>
<tr>
<td><code>GetCinemachineComponent(CinemachineCoreStage)</code></td>
<td>Get the component set for a specific stage.</td>
</tr>
</tbody>
</table>

## See Also

**Reference**
- CinemachineVirtualCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraGetCinemachineComponent Method

Get an existing component of a specific type from the cinemachine pipeline.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public T GetCinemachineComponent<T>()
where T : CinemachineComponentBase
```

**Type Parameters**

- **T**

**Return Value**

Type: **T**

[Missing <returns> documentation for "M:Cinemachine.CinemachineVirtualCamera.GetCinemachineComponent`1"]

**See Also**

Reference
- CinemachineVirtualCamera Class
- GetCinemachineComponent Overload
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCamera:GetCinemachineComponent Method (CinemachineCoreStage)

Get the component set for a specific stage.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public CinemachineComponentBase GetCinemachineComponent(CinemachineCoreStage stage)
```

## Parameters

- **stage**
  - Type: `CinemachineCoreStage`
  - The stage for which we want the component

## Return Value

- Type: `CinemachineComponentBase`
- The Cinemachine component for that stage, or null if not defined

## See Also

- Reference
  - CinemachineVirtualCamera Class
  - GetCinemachineComponent Overload
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraGetComponentOwner Method

Get the hidden CinemachinePipeline child object.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>public</strong> Transform GetComponentOwner()</td>
<td></td>
</tr>
</tbody>
</table>

**Return Value**

Type: **Transform**

[Missing <returns> documentation for "M:Cinemachine.CinemachineVirtualCamera.GetComponentOwner"]

### See Also

**Reference**

- CinemachineVirtualCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraGetComponentPipeline Method

Get the component pipeline owned by the hidden child pipeline container. For most purposes, it is preferable to use the GetCinemachineComponent method.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public CinemachineComponentBase[] GetComponentPipeline</td>
<td></td>
</tr>
</tbody>
</table>

Return Value

Type: CinemachineComponentBase

[Missing <returns> documentation for "M:Cinemachine.CinemachineVirtualCamera.GetComponentPipeline"]

See Also

Reference
- CinemachineVirtualCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraInvalidateComponentPipeline Method

Editor API: Call this when changing the pipeline from the editor. Will force a rebuild of the pipeline cache.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public void InvalidateComponentPipeline()</td>
<td></td>
</tr>
</tbody>
</table>

### See Also

- Reference  
  - CinemachineVirtualCamera Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraOnDestroy Method

Calls the DestroyPipelineDelegate for destroying the hidden child object, to support undo.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

#### Syntax

```csharp
protected override void OnDestroy()
```

#### See Also

**Reference**  
CinemachineVirtualCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCamera.OnEnable Method

Make sure that the pipeline cache is up-to-date.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected override void OnEnable()
```

### See Also

- **Reference**  
  - CinemachineVirtualCamera Class  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraOnPositionDragged Method

API for the editor, to process a position drag from the user.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public void OnPositionDragged(  
  Vector3 delta
) |

**Parameters**

*delta*

Type: `Vector3`


### See Also

Reference  
CinemachineVirtualCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraOnValidate Method

Enforce bounds for fields, when changed in inspector.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▼ **Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
protected override void OnValidate()
``` | Copy |

▼ **See Also**

Reference  
CinemachineVirtualCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraUpdateCameraState Method

Called by CinemachineCore at LateUpdate time so the vcam can position itself and track its targets. This class will invoke its pipeline and generate a CameraState for this frame.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public override void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

### Parameters

- **worldUp**  
  Type: **Vector3**  

- **deltaTime**  
  Type: **System.Single**  

### Implements

ICinemachineCameraUpdateCameraState(Vector3, Single)
See Also

Reference

CinemachineVirtualCamera Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# CinemachineVirtualCamera Fields

The `CinemachineVirtualCamera` type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://via.placeholder.com/15" alt="Image" /> CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is there to support the soon-to-be-removed Cinemachine Debugger in the Editor. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/15" alt="Image" /> CreatePipelineOverride</td>
<td>Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do the same thing as the CreatePipeline method in this class.</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/15" alt="Image" /> DestroyPipelineOverride</td>
<td>Override component pipeline destruction. This needs to be done by the editor to support Undo.</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/15" alt="Image" /> m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use sections of the Inspector. (Inherited from <code>CinemachineVirtualCameraBase</code>)</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/15" alt="Image" /> m_Follow</td>
<td>The object that the camera wants to move with (the Body)</td>
</tr>
</tbody>
</table>
The Body component of the CinemachineComponent pipeline will refer to this target position the vcam in accordance with rules and settings provided to it. If this is null, then the vcam's Transform position will be used.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>m_Lens</strong></td>
<td>Specifies the LensSettings of this Virtual Camera. These settings will be transferred to the camera when the vcam is live.</td>
</tr>
<tr>
<td><strong>m_LockStageInInspector</strong></td>
<td>Inspector control - Use for enabling sections of the Inspector UI. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>m_LookAt</strong></td>
<td>The object that the camera wants to look at (the Aim target). Aim component of the CinemachineComponent will refer to this target the vcam in accordance with rules and settings that are provided to it. If this is null, then the vcam's Transform orientation will be used.</td>
</tr>
<tr>
<td><strong>m_Priority</strong></td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
<tr>
<td><strong>OnPostPipelineStage</strong></td>
<td>A delegate to hook into the state calculation pipeline. Implementation must be called after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage. (Inherited from CinemachineVirtualCameraBase)</td>
</tr>
</tbody>
</table>

| **PipelineName** | This is the name of the GameObject that will act as a child object of the virtual camera. This hidden GameObject acts as a container for the polymorphic CinemachineComponent pipeline. The Inspector UI for the Virtual Camera provides access to this pipeline, as do the CinemachineComponent public methods in this class. The lifecycle of the pipeline GameObject is managed automatically. |

**Top**

**See Also**

**Reference**
- CinemachineVirtualCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraCreatePipelineOverride Field

Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do exactly the same thing as the CreatePipeline method in this class.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public static CinemachineVirtualCameraCreatePipelineDelegate</code></td>
<td></td>
</tr>
</tbody>
</table>

### Field Value

Type: `CinemachineVirtualCameraCreatePipelineDelegate`

### See Also

- Reference  
  - `CinemachineVirtualCamera Class`  
  - `Cinemachine Namespace`

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraDestroyPipeline

Field

Override component pipeline destruction. This needs to be done by the editor to support Undo.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public static</code> CinemachineVirtualCameraDestroyPipelineDelegate</td>
<td></td>
</tr>
</tbody>
</table>

### Field Value

**Type:** CinemachineVirtualCameraDestroyPipelineDelegate

### See Also

**Reference**
- CinemachineVirtualCamera Class  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCamera.m_Follow Field

The object that the camera wants to move with (the Body target). The Body component of the CinemachineComponent pipeline will refer to this target and position the vcam in accordance with rules and settings that are provided to it. If this is null, then the vcam's Transform position will be used.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The object that the camera wants to move with (the Body target). If this is null, then the vcam's Transform position will be used.

- [NoSaveDuringPlayAttribute]
```

```javascript
public Transform m_Follow
```

### Field Value

**Type:** Transform

### See Also

- Reference  
  CinemachineVirtualCamera Class  
  Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCamera.m_Lens

Field

Specifies the LensSettings of this Virtual Camera. These settings will be transferred to the Unity camera when the vcam is live.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#  JavaScript

```csharp
[FormerlySerializedAsAttribute("m_LensAttributes")]
[TooltipAttribute("Specifies the lens properties of this Virtual Camera. This generally mirrors the Unity Camera's lens settings, and will be used to drive the Unity camera when the vcam is active.")]
[LensSettingsPropertyAttribute]
public LensSettings m_Lens
```

Field Value

Type: LensSettings

See Also

Reference
- CinemachineVirtualCamera Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The object that the camera wants to look at (the Aim target). The Aim component of the CinemachineComponent pipeline will refer to this target and orient the vcam in accordance with rules and settings that are provided to it. If this is null, then the vcam's Transform orientation will be used.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The object that the camera wants to look at (the Aim target).
If this is null, then the vcam's Transform orientation will define the camera's orientation")]
[NoSaveDuringPlayAttribute]
public Transform m_LookAt
```

### Field Value

**Type:** Transform

### See Also

**Reference**  
CinemachineVirtualCamera Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraPipeline

Field

This is the name of the hidden GameObject that will be created as a child object of the virtual camera. This hidden game object acts as a container for the polymorphic CinemachineComponent pipeline. The Inspector UI for the Virtual Camera provides access to this pipeline, as do the CinemachineComponent-family of public methods in this class. The lifecycle of the pipeline GameObject is managed automatically.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

C#  
```csharp
public const string PipelineName = "cm"
```

Field Value

Type: String

### See Also

- Reference
  - CinemachineVirtualCamera Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraCreatePipelineDelegate

Override component pipeline creation. This needs to be done by the editor to support Undo. The override must do exactly the same thing as the CreatePipeline method in the CinemachineVirtualCamera class.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

| C#                          |  
|-----------------------------|-----------------------------|  
| public delegate Transform CreatePipelineDelegate(  
| CinemachineVirtualCamera vcam,  
| string name,  
| CinemachineComponentBase[] copyFrom  
| )                           |  

**Parameters**

- **vcam**  
  Type: CinemachineVirtualCamera  
- **name**  
  Type: System.String  
- **copyFrom**  
  Type: CinemachineComponentBase[]

**Return Value**  
Type: Transform

### See Also
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraDestroyFDelegate

Override component pipeline destruction. This needs to be done by the editor to support Undo.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public delegate void DestroyPipelineDelegate(
    GameObject pipeline
)
```

**JavaScript**

**Parameters**

- **pipeline**
  - Type: **GameObject**

### See Also

**Reference**

- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBase
Class

Base class for a Monobehaviour that represents a Virtual Camera within the Unity scene. This is intended to be attached to an empty Transform GameObject. Inherited classes can be either standalone virtual cameras such as CinemachineVirtualCamera, or meta-cameras such as CinemachineClearShot or CinemachineFreeLook. A CinemachineVirtualCameraBase exposes a Priority property. When the behaviour is enabled in the game, the Virtual Camera is automatically placed in a queue maintained by the static CinemachineCore singleton. The queue is sorted by priority. When a Unity camera is equipped with a CinemachineBrain behaviour, the brain will choose the camera at the head of the queue. If you have multiple Unity cameras with CinemachineBrain behaviours (say in a split-screen context), then you can filter the queue by setting the culling flags on the virtual cameras. The culling mask of the Unity Camera will then act as a filter for the brain. Apart from this, there is nothing that prevents a virtual camera from controlling multiple Unity cameras simultaneously.

▲ Inheritance Hierarchy

- SystemObject
  - Object
    - Component
      - Behaviour
        - Monobehaviour
          - Cinemachine
            - CinemachineVirtualCameraBase
              - More...

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)
The `CinemachineVirtualCameraBase` type exposes the following members.

### Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CinemachineVirtualCameraBase</code></td>
<td></td>
</tr>
</tbody>
</table>

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Description</code></td>
<td>Gets a brief debug description of this virtual camera, for use when displayiong debug info</td>
</tr>
<tr>
<td><code>Follow</code></td>
<td>Get the Follow target for the Body component in the CinemachinePipeline.</td>
</tr>
<tr>
<td><code>LiveChildOrSelf</code></td>
<td>Just returns self.</td>
</tr>
<tr>
<td><code>LookAt</code></td>
<td>Get the LookAt target for the Aim component in the CinemachinePipeline.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Get the name of the Virtual Camera. Base implementation returns the owner GameObject's name.</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>ParentCamera</strong></td>
<td>Support for meta-virtual-cameras. This is the situation where a virtual camera is in fact the public face of a private army of virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform children of the parent vcam.</td>
</tr>
<tr>
<td><strong>PreviousStateIsValid</strong></td>
<td>Set this to force the next update to ignore deltaTime and reset itself</td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td>Get the Priority of the virtual camera. This determines its placement in the CinemachineCore's queue of eligible shots.</td>
</tr>
</tbody>
</table>
| **State** | The CameraState object holds all of the information necessary to position
the Unity camera. It is the output of this class.

### ValidatingStreamVersion
Version that was last streamed, for upgrading legacy

### VirtualCameraGameObject
The GameObject owner of the Virtual Camera behaviour.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into state calculation pipeline. This will be called after each pipeline stage, to allow others to hook into the pipeline. See CinemachineCore.Stage.</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all parent cameras (if any).</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcam is a live child of this camera. This base class implementation always returns false.</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual</td>
</tr>
</tbody>
</table>
Cameras have the highest priority, there is sometimes the need to push one to top, making it the current Live camera if it shares highest priority in the queue with its peers. This happens automatically when a new vcam is enabled: the most recent one goes to the top of the priority subqueue. Use this method to push a vcam to the top of its priority peers. If it and its peers share the highest priority, then this vcam will become Live.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Base class implementation adds the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live. Base class implementation must be called by any overridden method.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>OnValidate</strong></td>
<td>Enforce bounds for fields when changed in inspector. Call base class</td>
</tr>
<tr>
<td><strong>PreUpdateChildCameras</strong></td>
<td>This is called prior to the updating of the vcam's child cameras, in order</td>
</tr>
<tr>
<td><strong>RemovePostPipelineStageHook</strong></td>
<td>Remove a Pipeline stage hook callback.</td>
</tr>
<tr>
<td><strong>ResolveFollow</strong></td>
<td>Returns this vcam's Follow target, or if that is null, will return the</td>
</tr>
<tr>
<td><strong>ResolveLookAt</strong></td>
<td>Returns this vcam's LookAt target, or if that is null, will return the</td>
</tr>
<tr>
<td><strong>Start</strong></td>
<td>Base class implementation does nothing.</td>
</tr>
</tbody>
</table>
**Update**  
Base class implementation makes sure the priority queue remains up-to-date.

**UpdateCameraState**  
Called by CinemachineCore at designated update time the vcam can position it and track its targets. Do not call this method. Let the framework do it at the appropriate time.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is here to support the soon-to-be-removed Cinemachine Debugger in the Editor.</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use for hiding sections of the Inspector UI.</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Use for enabling sections of the Inspector UI.</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this</td>
</tr>
<tr>
<td><strong>OnPostPipelineStage</strong></td>
<td>A delegate to hook into the state calculation pipeline. Implementation must be sure to call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage.</td>
</tr>
</tbody>
</table>

---

**See Also**

Reference

Cinemachine Namespace

**Inheritance Hierarchy**

```
System
  Object
  Component
  Behaviour
    MonoBehaviour
      Cinemachine
        CinemachineVirtualCameraBase
      Cinemachine
        CinemachineBlendListCamera
        CinemachineClearShot
        CinemachineExternalCamera
        CinemachineFreeLook
        CinemachineMixingCamera
        CinemachineStateDrivenCamera
        CinemachineVirtualCamera
```

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBase Constructor

[Missing <summary> documentation for "M:Cinemachine.CinemachineVirtualCameraBase.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    JavaScript

protected CinemachineVirtualCameraBase()  

See Also

Reference
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## CinemachineVirtualCameraBase Properties

The **CinemachineVirtualCameraBase** type exposes the following members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info</td>
</tr>
<tr>
<td>Follow</td>
<td>Get the Follow target for the Body component in the CinemachinePipeline.</td>
</tr>
<tr>
<td>LiveChildOrSelf</td>
<td>Just returns self.</td>
</tr>
<tr>
<td>LookAt</td>
<td>Get the LookAt target for the Aim component in the CinemachinePipeline.</td>
</tr>
<tr>
<td>Name</td>
<td>Get the name of the Virtual Camera. Base implementation returns the owner GameObject's name.</td>
</tr>
<tr>
<td>ParentCamera</td>
<td>Support for meta-</td>
</tr>
</tbody>
</table>
virtual-cameras. This is the situation where a virtual camera is in fact the public face of a private army of virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform children of the parent vcam.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreviousStateIsValid</td>
<td>Set this to force the next update to ignore deltaTime and reset itself</td>
</tr>
<tr>
<td>Priority</td>
<td>Get the Priority of the virtual camera. This determines its placement in the CinemachineCore's queue of eligible shots.</td>
</tr>
<tr>
<td>State</td>
<td>The CameraState object holds all of the information necessary to position the Unity camera. It is the output of this class.</td>
</tr>
<tr>
<td>ValidatingStreamVersion</td>
<td>Version that was last streamed, for upgrading legacy</td>
</tr>
</tbody>
</table>
VirtualCameraGameObject

The GameObject owner of the Virtual Camera behaviour.

See Also

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBaseDescription

Property

Gets a brief debug description of this virtual camera, for use when displaying debug info

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public virtual string Description { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: **String**

Implements **ICinemachineCameraDescription**

### See Also

**Reference**

CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseFollow Property

Get the Follow target for the Body component in the CinemachinePipeline.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public abstract Transform Follow { get; set; }
```

**Property Value**

Type: **Transform**  
Implements  
ICinemachineCameraFollow

### See Also

**Reference**

CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
**CinemachineVirtualCameraBaseLiveChildOrSelf Property**

Just returns self.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public virtual ICinemachineCamera LiveChildOrSelf</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: ICinemachineCamera

Implements

ICinemachineCameraLiveChildOrSelf

## See Also

**Reference**

CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseLoc
Property

Get the LookAt target for the Aim component in the CinemachinePipeline.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

C#  
```csharp
public abstract Transform LookAt { get; set; }
```

**Property Value**

Type: Transform

Implements ICinemachineCameraLookAt

**See Also**

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
**CinemachineVirtualCameraBaseName Property**

Get the name of the Virtual Camera. Base implementation returns the owner GameObject's name.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
public string Name { get; }
```

**Property Value**

Type: `String`

Implements `ICinemachineCameraName`

**See Also**

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBasePar

Property

Support for meta-virtual-cameras. This is the situation where a virtual camera is in fact the public face of a private army of virtual cameras, which it manages on its own. This method gets the VirtualCamera owner, if any. Private armies are implemented as Transform children of the parent vcam.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public ICinemachineCamera ParentCamera { get; }
```

**Property Value**

**Type:** ICinemachineCamera

**Implements**

ICinemachineCameraParentCamera

**See Also**

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseProperty

Set this to force the next update to ignore deltaTime and reset itself

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
public bool PreviousStateIsValid { get; set; }
``` | |

**Property Value**

Type: Boolean

### See Also

**Reference**

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBasePriority Property

Get the Priority of the virtual camera. This determines its placement in the CinemachineCore's queue of eligible shots.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public int Priority { get; set; }
```

Property Value
Type: Int32
Implements ICinemachineCameraPriority

See Also

Reference
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBaseState Property

The CameraState object holds all of the information necessary to position the Unity camera. It is the output of this class.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public abstract CameraState State { get; }
```

**Property Value**
- **Type:** CameraState
- **Implements**
  - ICinemachineCameraState

### See Also

**Reference**
- CinemachineVirtualCameraBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseValidatingStreamVersion

Property

Version that was last streamed, for upgrading legacy

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C# | JavaScript
---|---
```csharp
public int ValidatingStreamVersion { get; }
```

Property Value

Type: Int32

See Also

Reference
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBase

Property

The GameObject owner of the Virtual Camera behaviour.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public GameObject VirtualCameraGameObject { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: **GameObject**

Implements  
ICinemachineCameraVirtualCameraGameObject

**See Also**

Reference  
CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
The `CinemachineVirtualCameraBase` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddPostPipelineStageHook</td>
<td>A delegate to hook into state calculation pipeline. This will be called after each pipeline stage, to allow others to hook into the pipeline. See CinemachineCore.Stage.</td>
</tr>
<tr>
<td>InvokePostPipelineStageCallback</td>
<td>Invokes the <code>PostPipelineStageDelegate</code> for this camera, and up hierarchy for all parent cameras (if any).</td>
</tr>
<tr>
<td>IsLiveChild</td>
<td>Check whether the vcamera live child of this camera. This base class implementation always returns false.</td>
</tr>
<tr>
<td>MoveToTopOfPrioritySubqueue</td>
<td>When multiple virtual cameras have the highest priority, there is sometimes the need to push one to the top, making it the current camera.</td>
</tr>
</tbody>
</table>
Live camera if it shares highest priority in the queue with its peers. This happens automatically when a new vcam is enabled: the most recent one goes to the top of the priority subqueue. Use this method to push a vcam to the top of its priority peers. If it and its peers share highest priority, then this vcam will become Live.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OnDestroy</td>
<td>Base class implementation removes the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>OnDisable</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnEnable</td>
<td>Base class implementation adds the virtual camera from the priority queue.</td>
</tr>
<tr>
<td>OnTransformParentChanged</td>
<td>Base class implementation makes sure the priority queue remains up-to-date.</td>
</tr>
<tr>
<td>OnTransitionFromCamera</td>
<td>Notification that this virtual camera is going live. Base class implementation must be called by any overridden method.</td>
</tr>
<tr>
<td>OnValidate</td>
<td>Enforce bounds for fields when changed in inspector. Call base class method.</td>
</tr>
</tbody>
</table>

OnValidate
implementation at the beginning of overridden method. After base method is called, ValidatingStreamVersion will be valid.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PreUpdateChildCameras</td>
<td>This is called prior to the updating of the vcam's child cameras, in order to allow the parent to prepare its children. If the children are updating on FixedUpdate, then this will not necessarily be called prior to every FixedUpdate, but it might be called on LateUpdate instead. Base class implementation does nothing.</td>
</tr>
<tr>
<td>RemovePostPipelineStageHook</td>
<td>Remove a Pipeline stage hook callback.</td>
</tr>
<tr>
<td>ResolveFollow</td>
<td>Returns this vcam's Follow target, or if that is null, will return the parent vcam's Follow target.</td>
</tr>
<tr>
<td>ResolveLookAt</td>
<td>Returns this vcam's LookAt target, or if that is null, will return the parent vcam's LookAt target.</td>
</tr>
<tr>
<td>Start</td>
<td>Base class implementation does nothing.</td>
</tr>
</tbody>
</table>
| Update                                       | Base class implementation makes sure the priority }
queue remains up-to-date.

UpdateCameraState

Called by CinemachineCore at designated update time the vcam can position it and track its targets. Do not call this method. Let the framework do it at the appropriate time.

Top

See Also

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseAddPostPipelineStageHook Method

A delegate to hook into the state calculation pipeline. This will be called after each pipeline stage, to allow others to hook into the pipeline. See CinemachineCore.Stage.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public virtual void AddPostPipelineStageHook(
    CinemachineVirtualCameraBaseOnPostPipeline
)
```

**JavaScript**

```
public virtual void AddPostPipelineStageHook(
    CinemachineVirtualCameraBaseOnPostPipeline
)
```

### Parameters

- `d`  
  Type: CinemachineVirtualCameraBaseOnPostPipeline  
  The delegate to call.

### See Also

- Reference
  - CinemachineVirtualCameraBase Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseInvokePostPipelineStageCallback

Method

Invokes the PostPipelineStageDelegate for this camera, and up the hierarchy for all parent cameras (if any).

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
protected void InvokePostPipelineStageCallback(
    CinemachineVirtualCameraBase vcam,
    CinemachineCore.Stage stage,
    ref CameraState newState,
    float deltaTime
)
```

### Parameters

**vcam**

Type: *Cinemachine.CinemachineVirtualCameraBase*

[Missing <param name="vcam"/> documentation for
"M:Cinemachine.CinemachineVirtualCameraBase.InvokePostPipelineStageCallba

**stage**

Type: *Cinemachine.CinemachineCore.Stage*

[Missing <param name="stage"/> documentation for
"M:Cinemachine.CinemachineVirtualCameraBase.InvokePostPipelineStageCallba

**newState**

Type: *Cinemachine.CameraState*

[Missing <param name="newState"/> documentation for
"M:Cinemachine.CinemachineVirtualCameraBase.InvokePostPipelineStageCallba
**deltaTime**

Type: **System.Single**


**See Also**

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
**CinemachineVirtualCameraBase.IsLiveChild** Method

Check whether the vcam a live child of this camera. This base class implementation always returns false.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public virtual bool IsLiveChild(
    ICinemachineCamera vcam
)
```

**Parameters**

`vcam`  
Type: `ICinemachineCamera`  
The Virtual Camera to check

**Return Value**

Type: `Boolean`  
True if the vcam is currently actively influencing the state of this vcam

**Implements**

`ICinemachineCamera.IsLiveChild(ICinemachineCamera)`

**See Also**

Reference  
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseMoveToTopOfPrioritySubqueue Method

When multiple virtual cameras have the highest priority, there is sometimes the need to push one to the top, making it the current Live camera if it shares the highest priority in the queue with its peers. This happens automatically when a new vcam is enabled: the most recent one goes to the top of the priority subqueue. Use this method to push a vcam to the top of its priority peers. If it and its peers share the highest priority, then this vcam will become Live.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public void MoveToTopOfPrioritySubqueue()
```

### See Also

**Reference**  
[CinemachineVirtualCameraBase Class](#)  
[Cinemachine Namespace](#)

Visit the Cinemachine Forum  
CinemachineVirtualCameraBaseOnDestroy Method

Base class implementation removes the virtual camera from the priority queue.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
protected virtual void OnDestroy()
```

**See Also**

**Reference**
- CinemachineVirtualCameraBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseOnI
Method

Base class implementation makes sure the priority queue remains up-
to-date.

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

### Syntax

```csharp
protected virtual void OnDisable()
```

### See Also

**Reference**

CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseOnEnable Method

Base class implementation adds the virtual camera from the priority queue.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

C#  
```csharp
protected virtual void OnEnable()
```

### See Also

Reference  
CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseOnTransformParentChanged Method

Base class implementation makes sure the priority queue remains up-to-date.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
protected virtual void OnTransformParentChanged();
``` |

### See Also

- Reference
  - CinemachineVirtualCameraBase Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBaseOnTransitionFromCamera Method

Notification that this virtual camera is going live. Base class implementation must be called by any overridden method.

Namespace: Cinemachine  
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public virtual void OnTransitionFromCamera(
    ICinemachineCamera fromCam
)
```

Parameters

fromCam
Type: `CinemachineICinemachineCamera`  
The camera being deactivated. May be null.

Implements

`ICinemachineCameraOnTransitionFromCamera` (ICinemachineCamera)

See Also

Reference

CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBaseOnValidate Method

Enforce bounds for fields, when changed in inspector. Call base class implementation at the beginning of overridden method. After base method is called, ValidatingStreamVersion will be valid.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected virtual void OnValidate()
```

### See Also

**Reference**  
CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBasePreUpdateChildCameras Method

This is called prior to the updating of the vcam's child cameras, in order to allow the parent to prepare its children. If the children are updating on FixedUpdate, then this will not necessarily be called prior to every FixedUpdate, but it might be called on LateUpdate instead. Base class implementation does nothing.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public virtual void PreUpdateChildCameras(
    Vector3 worldUp,
    float deltaTime
)
```

### Parameters

- **worldUp**
  - Type: `Vector3`
  - Default world Up, set by the CinemachineBrain

- **deltaTime**
  - Type: `System.Single`
  - Delta time for time-based effects (ignore if less than 0)

### Implements

`ICinemachineCameraPreUpdateChildCameras(Vector3, Single)`

### See Also
Reference
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseRer Method

Remove a Pipeline stage hook callback.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public virtual void RemovePostPipelineStageHook(
    CinemachineVirtualCameraBaseOnPostPipeline
)
```

### Parameters

- **d**
  - Type: `CinemachineVirtualCameraBaseOnPostPipeline`  
The delegate to remove.

## See Also

**Reference**
- `CinemachineVirtualCameraBase Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseResolveFollow Method

Returns this vcam's Follow target, or if that is null, will retrun the parent vcam's Follow target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
protected Transform ResolveFollow(
    Transform localFollow
)
```

### Parameters

- **localFollow**
  - Type: `Transform`
  - This vcam's Follow value.

### Return Value

- Type: `Transform`
  - The same value, or the parent's if null and a parent exists.

### See Also

- **Reference**
  - CinemachineVirtualCameraBase Class
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBaseResolveLookAt Method

Returns this vcam's LookAt target, or if that is null, will return the parent vcam's LookAt target.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

```csharp
protected Transform ResolveLookAt(
    Transform localLookAt
)
```

### Parameters

- **localLookAt**  
  Type: `Transform`  
  This vcam's LookAt value.

### Return Value

- **Type:** `Transform`  
  The same value, or the parent's if null and a parent exists.

### See Also

- **Reference**  
  [CinemachineVirtualCameraBase Class](#)  
  [Cinemachine Namespace](#)

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBase

Method

Base class implementation does nothing.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
protected virtual void Start()
```

**See Also**

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseUpdate Method

Base class implementation makes sure the priority queue remains up-to-date.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
protected virtual void Update()
```

**See Also**

- Reference
  - CinemachineVirtualCameraBase Class
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBaseUpdateCameraState Method

Called by CinemachineCore at designated update time so the vcam can position itself and track its targets. Do not call this method. Let the framework do it at the appropriate time.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll)  
**Version:** 2.0.0.0

**Syntax**

```c#
public abstract void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

**Parameters**

- **worldUp**
  - Type: `Vector3`
  - Default world Up, set by the CinemachineBrain

- **deltaTime**
  - Type: `System.Single`
  - Delta time for time-based effects (ignore if less than 0)

**Implements**

`ICinemachineCameraUpdateCameraState(Vector3, Single)`

**See Also**

Reference
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBase Fields

The `CinemachineVirtualCameraBase` type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGUIDebuggerCallback</td>
<td>This is deprecated. It is here to support the soon-to-be-removed Cinemachine Debugger in the Editor.</td>
</tr>
<tr>
<td>m_ExcludedPropertiesInInspector</td>
<td>Inspector control - Use for hiding sections of the Inspector UI.</td>
</tr>
<tr>
<td>m_LockStageInInspector</td>
<td>Inspector control - Use for enabling sections of the Inspector UI.</td>
</tr>
<tr>
<td>m_Priority</td>
<td>The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority.</td>
</tr>
<tr>
<td>OnPostPipelineStage</td>
<td>A delegate to hook into the state calculation pipeline. Implementations are provided in CinemachineBase class.</td>
</tr>
</tbody>
</table>
must be sure to call this after each pipeline stage to allow other services to hook into the pipeline. See CinemachineCore.Stage.

See Also

Reference
CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBase

Field

This is deprecated. It is here to support the soon-to-be-removed Cinemachine Debugger in the Editor.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[HideInInspector] [NoSaveDuringPlayAttribute]</td>
<td>public Action CinemachineGUIDebuggerCallback</td>
</tr>
</tbody>
</table>

Field Value

Type: Action

See Also

Reference

CinemachineVirtualCameraBase Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBase

Field

Inspector control - Use for hiding sections of the Inspector UI.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>[HideInInspector]</code></td>
<td></td>
</tr>
<tr>
<td><code>[SerializeField]</code></td>
<td></td>
</tr>
<tr>
<td><code>[NoSaveDuringPlayAttribute]</code></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public string[] m_ExcludedPropertiesInInspector
```

### Field Value

**Type:** String

## See Also

### Reference

- CinemachineVirtualCameraBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Field

Inspector control - Use for enabling sections of the Inspector UI.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>[HideInInspector]</td>
<td></td>
</tr>
<tr>
<td>[SerializeField]</td>
<td></td>
</tr>
<tr>
<td>[NoSaveDuringPlayAttribute]</td>
<td></td>
</tr>
<tr>
<td>public CinemachineCoreStage[] m_LockStageInInspector</td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**

Type: CinemachineCoreStage

### See Also

**Reference**

CinemachineVirtualCameraBase Class  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineVirtualCameraBase
Field

The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
[NoSaveDuringPlayAttribute]
[TooltipAttribute("The priority will determine which camera becomes active based on the state of other cameras and this camera. Higher numbers have greater priority.")]

public int m_Priority
```

### Field Value

Type: Int32

### See Also

**Reference**
- CinemachineVirtualCameraBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseOnPostPipelineStageDelegate

Field

A delegate to hook into the state calculation pipeline. Implementation must be sure to call this after each pipeline stage, to allow other services to hook into the pipeline. See CinemachineCore.Stage.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
C#       JavaScript
protected CinemachineVirtualCameraBaseOnPostPipelineStageDelegate
```

**Field Value**

Type: `CinemachineVirtualCameraBaseOnPostPipelineStageDelegate`

### See Also

**Reference**

- CinemachineVirtualCameraBase Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineVirtualCameraBaseOnPipelineStageDelegate

A delegate to hook into the state calculation pipeline. This will be called after each pipeline stage, to allow other services to hook into the pipeline. See CinemachineCore.Stage. Parameters:
- CinemachineVirtualCameraBase vcam: the virtual camera being updated
- CinemachineCore.Stage stage: what stage in the pipeline has just been updated
- ref CameraState newState: the current state of the vcam
- float deltaTime: the frame timestep. Less than 0 means "don't consider the previous frame"

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    JavaScript

```csharp
public delegate void OnPostPipelineStageDelegate(
    CinemachineVirtualCameraBase vcam,
    CinemachineCore.Stage stage,
    ref CameraState newState,
    float deltaTime
)
```

Parameters

- vcam
  Type: CinemachineVirtualCameraBase
- stage
  Type: CinemachineCore.Stage
- newState

Copy
Type: CinemachineCameraState

deltaTime
  Type: SystemSingle

See Also

Reference
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
DocumentationSortingAttribute

Class

Attribute to control the automatic generation of documentation.

Inheritance Hierarchy

- System
- System.Object
- System.Attribute
- CinemachineDocumentationSortingAttribute

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  |  JavaScript
---|---
```csharp
[DocumentationSortingAttribute(of, DocumentationSortingAttribute)]
public sealed class DocumentationSortingAttribute
```

The `DocumentationSortingAttribute` type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DocumentationSortingAttribute</td>
<td>Constructor with specific values</td>
</tr>
</tbody>
</table>

Properties
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Refinement level of the documentation. The more refined, the more is excluded.</td>
</tr>
<tr>
<td>SortOrder</td>
<td>Where this type appears in the manual. Smaller number sort earlier.</td>
</tr>
</tbody>
</table>

**See Also**

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
DocumentationSortingAttribute
Constructor

Constructor with specific values

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public DocumentationSortingAttribute(
    float sortOrder,
    DocumentationSortingAttributeLevel category)
```

**JavaScript**

### Parameters

**sortOrder**

Type: `System.Single`

[Missing <param name="sortOrder"/> documentation for "M:Cinemachine.DocumentationSortingAttribute.#ctor(System.Single,Cinemachin..."

**category**

Type: `CinemachineDocumentationSortingAttributeLevel`

[Missing <param name="category"/> documentation for "M:Cinemachine.DocumentationSortingAttribute.#ctor(System.Single,Cinemachin..."

### See Also

Reference

DocumentationSortingAttribute Class  
Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The DocumentationSortingAttribute type exposes the following members.

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>Refinement level of the documentation. The more refined, the more is excluded.</td>
</tr>
<tr>
<td>SortOrder</td>
<td>Where this type appears in the manual. Smaller number sort earlier.</td>
</tr>
</tbody>
</table>

### See Also

Reference

DocumentationSortingAttribute Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
DocumentationSortingAttribute Category

Property

Refinement level of the documentation. The more refined, the more is excluded.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

C#  
```csharp
public DocumentationSortingAttributeLevel Category
```

**Property Value**

Type: `DocumentationSortingAttributeLevel`

See Also

Reference

- `DocumentationSortingAttribute Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
DocumentationSortingAttribute

Property

Where this type appears in the manual. Smaller number sort earlier.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
public float SortOrder { get; }
```

**Property Value**

Type: Single

**See Also**

Reference

DocumentationSortingAttribute Class

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
DocumentationSortingAttributeLevel Enumeration

Refinement level of the documentation

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public enum Level</code></td>
<td></td>
</tr>
</tbody>
</table>

### Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undoc</td>
<td>0</td>
<td>Type is excluded from documentation</td>
</tr>
<tr>
<td>API</td>
<td>1</td>
<td>Type is documented in the API reference</td>
</tr>
<tr>
<td>UserRef</td>
<td>2</td>
<td>Type is documented in the highly-refined User Manual</td>
</tr>
</tbody>
</table>

### See Also

Reference

Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCamera Interface

An abstract representation of a virtual camera which lives within the Unity scene

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public interface ICinemachineCamera
```

The **ICinemachineCamera** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="icinemachinemolecule.png" alt="" /> Description</td>
<td>Gets a brief debug description of this virtual camera, for use when displayiong debug info</td>
</tr>
<tr>
<td><img src="icinemachinemolecule.png" alt="" /> Follow</td>
<td>The thing the camera wants to follow (moving camera). May be null.</td>
</tr>
<tr>
<td><img src="icinemachinemolecule.png" alt="" /> LiveChildOrSelf</td>
<td>For cameras that implement child cameras, return the live child, otherwise, just returns self.</td>
</tr>
</tbody>
</table>
LookAt

The thing the camera wants to look at (aim). May be null.

Name

Gets the name of this virtual camera. For use when deciding how to blend to or from this camera.

ParentCamera

For cameras that implement child cameras, returns the parent vcam, otherwise null.

Priority

Gets the priority of this ICinemachineCamera. The virtual camera will be inserted into the global priority stack based on this value.

State

Camera state at the current time.

VirtualCameraGameObject

Gets the virtual camera game attached to this class.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🚀 IsLiveChild</td>
<td>Check whether the vcam is a live child</td>
</tr>
<tr>
<td>🚀 OnTransitionFromCamera</td>
<td>Notification that a new camera is entered</td>
</tr>
</tbody>
</table>
the currently active camera. Both may be active simultaneously for a while, if blending.

### PreUpdateChildCameras

This is called prior to the updating of the vcam's child cameras, in order to allow the parent to prepare its children. However, if the children are updating on FixedUpdate, this will not necessarily be called prior to every FixedUpdate, but it might be called on LateUpdate instead.

### UpdateCameraState

Updates this Cinemachine Camera. For an active camera this should be called once and only once each frame. If this is not the case, you should never call this method directly.

CinemachineCore.UpdateVirtualCamera(ICinemachineCamera, float), which has protection against multiple calls per frame.

---

**See Also**

Reference

Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
The **ICinemachineCamera** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>Gets a brief debug description of this virtual camera, for use when displaying debug info</td>
</tr>
<tr>
<td><strong>Follow</strong></td>
<td>The thing the camera wants to follow (moving camera). May be null.</td>
</tr>
<tr>
<td><strong>LiveChildOrSelf</strong></td>
<td>For cameras that implement child cameras, return the live child, otherwise, just returns self.</td>
</tr>
<tr>
<td><strong>LookAt</strong></td>
<td>The thing the camera wants to look at (aim). May be null.</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Gets the name of this virtual camera. For use when deciding how to blend to or from this camera</td>
</tr>
<tr>
<td><strong>ParentCamera</strong></td>
<td>For cameras that implement child cameras, returns the</td>
</tr>
</tbody>
</table>
parent vcam, otherwise null.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>Gets the priority of this <code>ICinemachineCamera</code>. The virtual camera will be inserted into the global priority stack based on this value.</td>
</tr>
<tr>
<td>State</td>
<td>Camera state at the current time.</td>
</tr>
<tr>
<td>VirtualCameraGameObject</td>
<td>Gets the virtual camera game attached to this class.</td>
</tr>
</tbody>
</table>

**See Also**

- Reference
  - `ICinemachineCamera Interface`
  - `Cinemachine Namespace`

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCameraDescription

Property

Gets a brief debug description of this virtual camera, for use when displaying debug info

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
string Description { get; }
```

**Property Value**

Type: String

**See Also**

**Reference**

ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
ICinemachineCameraFollow Property

The thing the camera wants to follow (moving camera). May be null.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
C# JavaScript
```

```
Transform Follow { get; set; }
```  

Property Value
Type: Transform

See Also

Reference
ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCameraLiveChildOrSelf Property

For cameras that implement child cameras, return the live child, otherwise, just returns self.

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICinemachineCamera LiveChildOrSelf { get; }</td>
<td></td>
</tr>
</tbody>
</table>

### Property Value

Type: ICinemachineCamera

### See Also

**Reference**
- ICinemachineCamera Interface
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
ICinemachineCameraLookAt Property

The thing the camera wants to look at (aim). May be null.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
Transform LookAt { get; set; }
```

**Property Value**  
**Type:** Transform

### See Also

**Reference**  
ICinemachineCamera Interface  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
ICinemachineCameraName Property

Gets the name of this virtual camera. For use when deciding how to blend to or from this camera

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

(Syntax)

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>string Name { get; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: String

See Also

Reference
ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCameraParentCamera

Property

For cameras that implement child cameras, returns the parent vcam, otherwise null.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICinemachineCamera ParentCamera { get; }</td>
<td></td>
</tr>
</tbody>
</table>

**Property Value**

Type: ICinemachineCamera

### See Also

**Reference**
- ICinemachineCamera Interface  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
ICinemachineCameraPriority Property

Gets the priority of this ICinemachineCamera. The virtual camera will be inserted into the global priority stack based on this value.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>int Priority { get; set; }</td>
<td></td>
</tr>
</tbody>
</table>

Property Value
Type: Int32

See Also

Reference
ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCameraState

Property

Camera state at the current time.

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#   JavaScript

```
CameraState State { get; }
```

Property Value

Type: CameraState

See Also

Reference

ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCameraVirtualCamera

Property

Gets the virtual camera game attached to this class.

**Namespace:**  Cinemachine  
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
GameObject VirtualCameraGameObject { get; }
```

Property Value

Type: **GameObject**

**See Also**

Reference

ICinemachineCamera Interface  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
ICinemachineCamera Methods

The **ICinemachineCamera** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="IsLiveChild" /></td>
<td>Check whether the vcam is a live child of this camera.</td>
</tr>
<tr>
<td><img src="image" alt="OnTransitionFromCamera" /></td>
<td>Notification that a new camera is being activated. The currently active camera. Both may be active simultaneously for a while, if blending.</td>
</tr>
<tr>
<td><img src="image" alt="PreUpdateChildCameras" /></td>
<td>This is called prior to the updating of the vcam's child cameras, in order to allow the parent to prepare its children. If children are updating on FixedUpdate, this will not necessarily be called prior to every FixedUpdate, but it might be called on LateUpdate instead.</td>
</tr>
<tr>
<td><img src="image" alt="UpdateCameraState" /></td>
<td>Updates this Cinemachine Camera. For an active camera this should be called once and only once per frame, you should never call this method directly. CinemachineCore.UpdateVirtualCamera(float), which has protection against multiple calls per frame.</td>
</tr>
</tbody>
</table>

### See Also

Reference

ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
ICinemachineCamera.IsLiveChild Method

Check whether the vcam is a live child of this camera.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

### C#

```csharp
bool IsLiveChild(
    ICinemachineCamera vcam
)
```

### JavaScript

```javascript
false
```  

### C#

```csharp
bool IsLiveChild(
    ICinemachineCamera vcam
)
```

### JavaScript

```javascript
false
```  

## Parameters

**vcam**

Type: `CinemachineICinemachineCamera`

The Virtual Camera to check

## Return Value

Type: `Boolean`

True if the vcam is currently actively influencing the state of this vcam

## See Also

Reference
- `ICinemachineCamera Interface`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
Cinemachine
ICinemachineCameraOnTransitionFromCamera Method

Notification that a new camera is being activated. This is sent to the currently active camera. Both may be active simultaneously for a while, if blending.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>void OnTransitionFromCamera(</code></td>
<td><code>void OnTransitionFromCamera(</code></td>
</tr>
<tr>
<td><code>ICinemachineCamera fromCam</code></td>
<td><code>ICinemachineCamera fromCam</code></td>
</tr>
<tr>
<td><code>)</code></td>
<td><code>)</code></td>
</tr>
</tbody>
</table>

### Parameters

*fromCam*

Type: **CinemachineICinemachineCamera**  
The camera being deactivated. May be null.

### See Also

**Reference**
- ICinemachineCamera Interface
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
ICinemachineCameraPreUpdateChildCameras

Method

This is called prior to the updating of the vcam's child cameras, in order to allow the parent to prepare its children. If the children are updating on FixedUpdate, then this will not necessarily be called prior to every FixedUpdate, but it might be called on LateUpdate instead.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

**C#**

```csharp
void PreUpdateChildCameras(
    Vector3 worldUp,
    float deltaTime
)
```

**JavaScript**

```javascript
void PreUpdateChildCameras(
    Vector3 worldUp,
    float deltaTime
)
```

## Parameters

**worldUp**
- Type: Vector3
- Default world Up, set by the CinemachineBrain

**deltaTime**
- Type: SystemSingle
- Delta time for time-based effects (ignore if less than 0)

## See Also

- Reference
  - ICinemachineCamera Interface
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ICinemachineCameraUpdateCameraState Method

Updates this Cinemachine Camera. For an active camera this should be called once and only once each frame. To guarantee this, you should never call this method directly. Always use CinemachineCore.UpdateVirtualCamera(ICinemachineCamera, float), which has protection against multiple calls per frame.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
void UpdateCameraState(
    Vector3 worldUp,
    float deltaTime
)
```

### Parameters

- **worldUp**  
  Type: **Vector3**  
  Default world Up, set by the CinemachineBrain  

- **deltaTime**  
  Type: **SystemSingle**  
  Delta time for time-based effects (ignore if less than 0)

## See Also

- Reference  
  ICinemachineCamera Interface
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
LensSettings Structure

Describes the FOV and clip planes for a camera. This generally mirrors the Unity Camera's lens settings, and will be used to drive the Unity camera when the vcam is active.

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0  
(2.0.0.0)

## Syntax

**C#**

```
[SerializableAttribute]
[DocumentationSortingAttribute(2f, DocumentationSortingAttribute)]
public struct LensSettings
```

**JavaScript**

```
C# | JavaScript
---|---

The `LensSettings` type exposes the following members.

## Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LensSettings</td>
<td>Explicit constructor for this LensSettings</td>
</tr>
</tbody>
</table>

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromCamera</td>
<td>Creates a new LensSettings, copying the values from the</td>
</tr>
</tbody>
</table>
Lerp

Scalar that linearly blends the fields of two LensSettings and returns the result.

Validate

Make sure lens settings are sane. Call this from OnValidate().

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>Default Lens Settings</td>
</tr>
<tr>
<td>Dutch</td>
<td>The dutch (tilt) to be applied to the camera. In degrees</td>
</tr>
<tr>
<td>FarClipPlane</td>
<td>The far clip plane for this LensSettings</td>
</tr>
<tr>
<td>FieldOfView</td>
<td>This is the camera view in vertical degrees. For Cinematic people, a 50mm lens on a super-35mm sensor would equal a 19.6 degree FOV</td>
</tr>
<tr>
<td>NearClipPlane</td>
<td>The near clip plane for this LensSettings</td>
</tr>
<tr>
<td>OrthographicSize</td>
<td>When using an orthographic camera, this defines the height, in world co-ordinates, of the camera view.</td>
</tr>
</tbody>
</table>

See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
LensSettings Constructor

Explicit constructor for this LensSettings

**Namespace:**  Cinemachine
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public LensSettings(
    float fov,
    float orthographicSize,
    float nearClip,
    float farClip,
    float dutch,
    bool ortho,
    float aspect
)
```

### Parameters

- **fov**
  - Type: System.Single
  - The Vertical field of view

- **orthographicSize**
  - Type: System.Single
  - If orthographic, this is the half-height of the screen

- **nearClip**
  - Type: System.Single
  - The near clip plane

- **farClip**
  - Type: System.Single
  - The far clip plane
**dutch**
Type: **SystemSingle**
Camera roll, in degrees. This is applied at the end Whether the lens is orthographic. The aspect ratio of the lens Width/height after shot composition.

**ortho**
Type: **SystemBoolean**

**aspect**
Type: **SystemSingle**

**See Also**

**Reference**
LensSettings Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
LensSettings Methods

The LensSettings type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>➕ FromCamera</td>
<td>Creates a new LensSettings, copying the values from the supplied Camera</td>
</tr>
<tr>
<td>➕ Lerp</td>
<td>Linearly blends the fields of two LensSettings and returns the result</td>
</tr>
<tr>
<td>➕ Validate</td>
<td>Make sure lens settings are sane. Call this from OnValidate()</td>
</tr>
</tbody>
</table>

See Also

Reference
LensSettings Structure
Cinemachine Namespace
Cinemachine
LensSettingsFromCamera Method

Creates a new LensSettings, copying the values from the supplied Camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

```csharp
public static LensSettings FromCamera(
    Camera fromCamera
)
```

### Parameters

*fromCamera*
  
  **Type:** Camera  
  The Camera from which the FoV, near and far clip planes will be copied.

### Return Value

**Type:** LensSettings  

[Missing <returns> documentation for "M:Cinemachine.LensSettings.FromCamera(UnityEngine.Camera)"]

⚠️ See Also

- Reference
  - LensSettings Structure
  - Cinemachine Namespace
Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
LensSettingsLerp Method

Linearly blends the fields of two LensSettings and returns the result

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public static LensSettings Lerp(
    LensSettings lensA,
    LensSettings lensB,
    float t
)
```

### Parameters

- **lensA**  
  Type: CinemachineLensSettings  
  The LensSettings to blend from

- **lensB**  
  Type: CinemachineLensSettings  
  The LensSettings to blend to

- **t**  
  Type: SystemSingle  
  The interpolation value. Internally clamped to the range [0,1]

### Return Value

Type: LensSettings  
Interpolated settings

### See Also
Reference

LensSettings Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
LensSettingsValidate Method

Make sure lens settings are sane. Call this from OnValidate().

**Namespace:** Cinemachine
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public void Validate()
```

**See Also**

Reference
- LensSettings Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
## LensSettings Fields

The **LensSettings** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>Default Lens Settings</td>
</tr>
<tr>
<td>Dutch</td>
<td>The dutch (tilt) to be applied to the camera. In degrees</td>
</tr>
<tr>
<td>FarClipPlane</td>
<td>The far clip plane for this LensSettings</td>
</tr>
<tr>
<td>FieldOfView</td>
<td>This is the camera view in vertical degrees. For cinematic people, a 50mm lens on a super-35mm sensor would equal a 19.6 degree FOV</td>
</tr>
<tr>
<td>NearClipPlane</td>
<td>The near clip plane for this LensSettings</td>
</tr>
<tr>
<td>OrthographicSize</td>
<td>When using an orthographic camera, this defines the height, in world co-ordinates, of the camera view.</td>
</tr>
</tbody>
</table>

### See Also

**Reference**

LensSettings Structure
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
LensSettingsDefault Field

Default Lens Settings

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public static</code></td>
<td>LensSettings Default</td>
</tr>
</tbody>
</table>

Field Value  
Type: LensSettings

### See Also

- Reference  
  - LensSettings Structure  
  - Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
LensSettingsDutch Field

The dutch (tilt) to be applied to the camera. In degrees

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
[RangeAttribute(-180f, 180f)]
[TooltipAttribute("Camera Z roll, or tilt, in degrees.")]
public float Dutch
```

**Field Value**

**Type:** Single

## See Also

**Reference**

- LensSettings Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
LensSettingsFarClipPlane Field

The far clip plane for this LensSettings

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
[TooltipAttribute("This defines the far region of the renderable range of the camera frustum.
Typically you want to set this value as low as possible without cutting off desired distant objects")]
public float FarClipPlane
```

**Field Value**

Type: Single

**See Also**

Reference

- LensSettings Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# LensSettingsFieldOfView Field

This is the camera view in vertical degrees. For cinematic people, a 50mm lens on a super-35mm sensor would equal a 19.6 degree FOV

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
[RangeAttribute(1f, 179f)]
[TooltipAttribute("This is the camera view in vertical degrees. For cinematic people, a 50mm lens on a super-35mm sensor would equal a 19.6 degree FOV")]
public float FieldOfView
``` | |

## Field Value

Type: **Single**

## See Also

- Reference  
  - [LensSettings Structure](#)  
  - [Cinemachine Namespace](#)

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
LensSettingsNearClipPlane Field

The near clip plane for this LensSettings

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public float NearClipPlane</td>
<td></td>
</tr>
</tbody>
</table>

[TooltipAttribute("This defines the near region in the renderable range of the camera frustum. Raising this value will stop camera distortion and allow for the camera, which can sometimes come in handy. Larger values will also increase your shadow resolution.")]

Field Value

Type: Single

## See Also

**Reference**
- LensSettings Structure
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
LensSettingsOrthographicSize Field

When using an orthographic camera, this defines the height, in world co-ordinates, of the camera view.

**Namespace:** Cinemachine

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

```csharp
[TooltipAttribute("When using an orthographic camera, this defines the height, in world co-ordinates, of the camera view.")]
public float OrthographicSize
```

Field Value

Type: **Single**

**See Also**

Reference

- LensSettings Structure
- Cinemachine Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
LensSettingsPropertyAttribute

Class

Property applied to LensSettings. Used for custom drawing in the inspector.

Inheritance Hierarchy

- System
- SystemObject
- SystemAttribute
- PropertyAttribute
  - Cinemachine
  - LensSettingsPropertyAttribute

Namespace: Cinemachine

Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public sealed class LensSettingsPropertyAttribute
```

The `LensSettingsPropertyAttribute` type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LensSettingsPropertyAttribute</td>
<td></td>
</tr>
</tbody>
</table>

See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
LensSettingsPropertyAttribute

Constructor

[Missing <summary> documentation for "M:Cinemachine.LensSettingsPropertyAttribute.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
public LensSettingsPropertyAttribute()
```

See Also

Reference
LensSettingsPropertyAttribute Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettings Class

This is an asset that defines a noise profile. A noise profile is the shape of the noise as a function of time. You can build arbitrarily complex shapes by combining different base perlin noise frequencies at different amplitudes. The frequencies and amplitudes should be chosen with care, to ensure an interesting noise quality that is not obviously repetitive. As a mathematical side-note, any arbitrary periodic curve can be broken down into a series of fixed-amplitude sine-waves added together. This is called fourier decomposition, and is the basis of much signal processing. It doesn't really have much to do with this asset, but it's super interesting!

Inheritance Hierarchy

- System
- Object
- ScriptableObject
- CinemachineNoiseSettings

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```csharp
[DocumentationSortingAttribute(9f, DocumentationSortingAttribute
public sealed class NoiseSettings : ScriptableObject
```

The NoiseSettings type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>

Copy
## NoiseSettings

### Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrientationNoise</td>
<td>Gets the array of orientation noise channels for this NoiseSettings</td>
</tr>
<tr>
<td>PositionNoise</td>
<td>Gets the array of positional noise channels for this NoiseSettings</td>
</tr>
</tbody>
</table>

### See Also

- Reference
  - Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettings Constructor

[Missing <summary> documentation for "M:Cinemachine.NoiseSettings.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```
public NoiseSettings()
```

See Also

Reference
  NoiseSettings Class
  Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettings Properties

The `NoiseSettings` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrientationNoise</td>
<td>Gets the array of orientation noise channels for this <code>NoiseSettings</code></td>
</tr>
<tr>
<td>PositionNoise</td>
<td>Gets the array of positional noise channels for this <code>NoiseSettings</code></td>
</tr>
</tbody>
</table>

## See Also

Reference
- `NoiseSettings Class`
- `Cinemachine Namespace`

Visit the Cinemachine Forum
NoiseSettingsOrientationNoise Property

Gets the array of orientation noise channels for this NoiseSettings

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#      JavaScript

```java
public NoiseSettingsTransformNoiseParams[] Orient
```

Property Value

Type: NoiseSettingsTransformNoiseParams

See Also

Reference
- NoiseSettings Class
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettingsGetPositionNoise Property

Gets the array of positional noise channels for this NoiseSettings

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#   JavaScript

```csharp
public NoiseSettingsTransformNoiseParams[] PositionNoise;
```

Property Value
Type: NoiseSettingsTransformNoiseParams

See Also
Reference
NoiseSettings Class
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettings
NoiseParams
Structure

Describes the behaviour for a channel of noise

**Namespace:**  Cinemachine
**Assembly:**  Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

C#  JavaScript

```
[SerializableAttribute]
[DocumentationSortingAttribute(9.1f, Documentatic
public struct NoiseParams
```

The *NoiseSettings*NoiseParams* type exposes the following members.

## Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amplitude</td>
<td>The amplitude of the noise for this channel. Larger numbers vibrate higher</td>
</tr>
<tr>
<td>Frequency</td>
<td>The frequency of noise for this channel. Higher magnitudes vibrate faster</td>
</tr>
</tbody>
</table>

## See Also
Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseParams Fields

The NoiseSettingsNoiseParams type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amplitude</td>
<td>The amplitude of the noise for this channel. Larger numbers vibrate higher</td>
</tr>
<tr>
<td>Frequency</td>
<td>The frequency of noise for this channel. Higher magnitudes vibrate faster</td>
</tr>
</tbody>
</table>

See Also

Reference

NoiseSettingsNoiseParams Structure
Cinemachine Namespace

Visit the Cinematicine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettingsNoiseParamsAmplitude Field

The amplitude of the noise for this channel. Larger numbers vibrate higher

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[TooltipAttribute("The amplitude of the noise for this channel.	Larger numbers vibrate higher.")] public float Amplitude
```

Field Value  
Type: Single

### See Also

**Reference**  
- NoiseSettingsNoiseParams Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
NoiseSettingsNoiseParamsFrequency Field

The frequency of noise for this channel. Higher magnitudes vibrate faster

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```
[TooltipAttribute("The frequency of noise for this channel. Higher magnitudes vibrate faster.")]
public float Frequency
``` |  |

**Field Value**

Type: *Single*

### See Also

**Reference**

- NoiseSettingsNoiseParams Structure  
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
NoiseSettingsTransformNoiseParams Structure

Contains the behaviour of noise for the noise module for all 3 cardinal axes of the camera

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
[SerializableAttribute]
[DocumentationSortingAttribute(9.2f, DocumentationSortingAttribute)]
public struct TransformNoiseParams
```

The **NoiseSettingsTransformNoiseParams** type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Noise definition for X-axis</td>
</tr>
<tr>
<td>Y</td>
<td>Noise definition for Y-axis</td>
</tr>
<tr>
<td>Z</td>
<td>Noise definition for Z-axis</td>
</tr>
</tbody>
</table>

### See Also
Reference

Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
TransformNoiseParams Fields

The NoiseSettingsTransformNoiseParams type exposes the following members.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Noise definition for X-axis</td>
</tr>
<tr>
<td>Y</td>
<td>Noise definition for Y-axis</td>
</tr>
<tr>
<td>Z</td>
<td>Noise definition for Z-axis</td>
</tr>
</tbody>
</table>

### See Also

Reference
- NoiseSettingsTransformNoiseParams Structure
- Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
**NoiseSettingsTransformNoiseParams Field**

Noise definition for X-axis

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("Noise definition for X-axis")]
public NoiseSettingsNoiseParams X |

Field Value  
Type: NoiseSettingsNoiseParams

**See Also**

Reference  
NoiseSettingsTransformNoiseParams Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
NoiseSettingsTransformNoiseParam Field

Noise definition for Y-axis

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| [TooltipAttribute("Noise definition for Y-axis")]
public NoiseSettingsNoiseParams Y |

**Field Value**  
Type: NoiseSettingsNoiseParams

**See Also**

**Reference**  
NoiseSettingsTransformNoiseParams Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Cinemachine
NoiseSettingsTransformNoiseParam

Field

Noise definition for Z-axis

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ **Syntax**

```csharp
[TooltipAttribute("Noise definition for Z-axis")]
public NoiseSettingsNoiseParams Z
```

Field Value  
Type: `NoiseSettingsNoiseParams`

⚠️ **See Also**

Reference  
NoiseSettingsTransformNoiseParams Structure  
Cinemachine Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
NoSaveDuringPlayAttribute

Class

Suppresses play-mode-save for a field. Use it if the calsee has [SaveDuringPlay] attribute but there are fields in the class that shouldn't be saved.

Inheritance Hierarchy

- System
  - SystemObject
  - SystemAttribute
  - PropertyAttribute
  - CinemachineNoSaveDuringPlayAttribute

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C# JavaScript

```csharp
public sealed class NoSaveDuringPlayAttribute : 
```

See Also

public sealed class NoSaveDuringPlayAttribute : 

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoSaveDuringPlayAttribute</td>
<td></td>
</tr>
</tbody>
</table>

Top
<table>
<thead>
<tr>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cinemachine Namespace</td>
</tr>
</tbody>
</table>

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
NoSaveDuringPlayAttribute
Constructor

[Missing <summary> documentation for "M:Cinemachine.NoSaveDuringPlayAttribute.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

⚠️ Syntax

**C#**
```csharp
public NoSaveDuringPlayAttribute()
```

**JavaScript**

See Also

Reference
NoSaveDuringPlayAttribute Class
Cinemachine Namespace

Visit the Cinematic Forum
https://unity3d.com/legal/terms-of-service
SaveDuringPlayAttribute Class

Invoke play-mode-save for a class. This class's fields will be scanned upon exiting play mode, and its property values will be applied to the scene object. This is a stopgap measure that will become obsolete once Unity implements play-mode-save in a more general way.

▷ Inheritance Hierarchy

```
System
  Object
  SystemAttribute
  Cinemachine
  SaveDuringPlayAttribute
```

**Namespace:** Cinemachine  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▷ Syntax

```
public sealed class SaveDuringPlayAttribute : Attribute
```

The **SaveDuringPlayAttribute** type exposes the following members.

▷ Constructors

```

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SaveDuringPlayAttribute</td>
<td></td>
</tr>
</tbody>
</table>
```

▷ See Also

Reference
Cinemachine Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
SaveDuringPlayAttribute
Constructor

[Missing <summary> documentation for "M:Cinemachine.SaveDuringPlayAttribute.#ctor"]

Namespace: Cinemachine
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#       JavaScript

public SaveDuringPlayAttribute()
# Cinemachine.Utility Namespace

Cinemachine non-domain utilities and Unity extensions and helpers

## Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![ ] CinemachineGameWindowDebug</td>
<td>Manages onscreen positions for Cinemachine debugging output</td>
</tr>
<tr>
<td>![ ] Damper</td>
<td>Utility to perform realistic damping of float or Vector3 values. The algorithm is based on exponentially decaying the delta until only a negligible amount remains.</td>
</tr>
<tr>
<td>![ ] ReflectionHelpers</td>
<td>An ad-hoc collection of helpers for reflection, used by</td>
</tr>
<tr>
<td>UnityQuaternionExtensions</td>
<td>Extentions to the Quaternion class, used in various places by Cinemachine</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UnityRectExtensions</td>
<td>Ad-hoc xxtentions to the Rect structure, used by Cinemachine</td>
</tr>
<tr>
<td>UnityVectorExtensions</td>
<td>Extensions to the Vector3 class, used by Cinemachine</td>
</tr>
</tbody>
</table>

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGameWindowDebug Class

Manages onscreen positions for Cinemachine debugging output

Inheritance Hierarchy

System
  Object
  Cinemachine.Utility
  CinemachineGameWindowDebug

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

```
public class CinemachineGameWindowDebug
```

The CinemachineGameWindowDebug type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CinemachineGameWindowDebug</td>
<td></td>
</tr>
</tbody>
</table>

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetScreenPos</td>
<td>Reserve an on-screen</td>
</tr>
</tbody>
</table>
ReleaseScreenPos

Release a screen rectangle previously obtained through GetScreenPos()

See Also

Reference
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGameWindowDebug Constructor


Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▶ Syntax

```csharp
public CinemachineGameWindowDebug()
```

▶ See Also

Reference
CinemachineGameWindowDebug Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
CinemachineGameWindowDebug

Methods

The `CinemachineGameWindowDebug` type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetScreenPos</td>
<td>Reserve an on-screen rectangle for debugging output.</td>
</tr>
<tr>
<td>ReleaseScreenPos</td>
<td>Release a screen rectangle previously obtained through GetScreenPos()</td>
</tr>
</tbody>
</table>

See Also

Reference

- `CinemachineGameWindowDebug Class`
- `Cinemachine.Utility Namespace`

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
CinemachineGameWindowDebugGetScreenPos

Method

Reserve an on-screen rectangle for debugging output.

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

(2.0.0.0)

**Syntax**

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public static Rect GetScreenPos(
  Object client,
  string text,
  GUIStyle style
) |

**Parameters**

*client*

Type: **Object**  
The client caller. This is used as a handle.

*text*

Type: **System.String**  
Sample text, for determining rectangle size

*style*

Type: **GUIStyle**  
What style will be used to draw, used here for determining rect size

**Return Value**

Type: **Rect**  
An area on the game screen large enough to print the text in the
See Also

Reference

Cinemachine.GameWindowDebug Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
CinemachineGameWindowDebug

Method

Release a screen rectangle previously obtained through GetScreenPos()

**Namespace:** Cinemachine.Utility

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

**Syntax**

```csharp
public static void ReleaseScreenPos(
    Object client
)
```

**Parameters**

`client`  
Type: **Object**  
The client caller. Used as a handle.

**See Also**

Reference
- CinemachineGameWindowDebug Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Damper Class

Utility to perform realistic damping of float or Vector3 values. The algorithm is based on exponentially decaying the delta until only a negligible amount remains.

Inheritance Hierarchy

- System
- Object
- Cinemachine.Utility
- Damper

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  
```csharp
public static class Damper
```

The Damper type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Damper](Damp(Single, Single, Single))</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
<tr>
<td>![Damper](Damp(Vector3, Single, Single))</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
</tbody>
</table>
**Damp(Vector3, Vector3, Single)**

Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.

### Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![s] kNegligibleResidual</td>
<td>Standard residual</td>
</tr>
</tbody>
</table>

### See Also

**Reference**

Cinemachine.Utility Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
## Damper Methods

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![s] Damp(Single, Single, Single)</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
<tr>
<td>![s] Damp(Vector3, Single, Single)</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
<tr>
<td>![s] Damp(Vector3, Vector3, Single)</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
</tbody>
</table>

#### See Also

**Reference**
- Damper Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service
Damper

Damp Method

Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌟 Damp(Single, Single, Single)</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
<tr>
<td>🌟 Damp(Vector3, Single, Single)</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
<tr>
<td>🌟 Damp(Vector3, Vector3, Single)</td>
<td>Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.</td>
</tr>
</tbody>
</table>

Top

See Also

Reference
- Damper Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
**DamperDamp Method (Single, Single, Single)**

Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```csharp
public static float Damp(
    float initial,
    float dampTime,
    float deltaTime
)
```

**Parameters**

- **initial**  
  Type: `System.Single`  
  The amount that will be damped

- **dampTime**  
  Type: `System.Single`  
  The rate of damping. This is the time it would take to reduce the original amount to a negligible percentage

- **deltaTime**  
  Type: `System.Single`  
  The time over which to damp

**Return Value**

Type: `Single`
The damped amount. This will be the original amount scaled by a value between 0 and 1.

See Also

Reference
Damper Class
Damp Overload
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
DamperDamp Method (Vector3, Single, Single)

Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public static Vector3 Damp(  
  Vector3 initial,  
  float dampTime,  
  float deltaTime  
) | |

### Parameters

- **initial**  
  Type: **Vector3**  
  The amount that will be damped  
- **dampTime**  
  Type: **System.Single**  
  The rate of damping. This is the time it would take to reduce the original amount to a negligible percentage  
- **deltaTime**  
  Type: **System.Single**  
  The time over which to damp

### Return Value

Type: **Vector3**
The damped amount. This will be the original amount scaled by a value between 0 and 1.

See Also

Reference
Damper Class
Damp Overload
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
DamperDamp Method (Vector3, Vector3, Single)

Get a damped version of a quantity. This is the portion of the quantity that will take effect over the given time.

**Namespace:** Cinemachine.Utility

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```csharp
public static Vector3 Damp(
    Vector3 initial,
    Vector3 dampTime,
    float deltaTime
)
```

### Parameters

- **initial**
  - Type: **Vector3**
  - The amount that will be damped

- **dampTime**
  - Type: **Vector3**
  - The rate of damping. This is the time it would take to reduce the original amount to a negligible percentage

- **deltaTime**
  - Type: **System.Single**
  - The time over which to damp

### Return Value

- Type: **Vector3**
The damped amount. This will be the original amount scaled by a value between 0 and 1.

See Also

Reference
Damper Class
Damp Overload
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
Damper Fields

The Damper type exposes the following members.

Fields

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kNegligibleResidual</td>
<td>Standard residual</td>
</tr>
</tbody>
</table>

See Also

Reference
- Damper Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Damper
kNegligibleResidual
Field

Standard residual

**Namespace:** Cinemachine.Utility
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>public const float kNegligibleResidual = 0.01f</code></td>
<td></td>
</tr>
</tbody>
</table>

**Field Value**
Type: **Single**

### See Also

**Reference**
- Damper Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ReflectionHelpers Class

An ad-hoc collection of helpers for reflection, used by Cinemachine or its editor tools in various places

Inheritance Hierarchy

- System
  - Object
  - Cinemachine.Utility
    - ReflectionHelpers

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#    | JavaScript
----- | ------------

```csharp
[DocumentationSortingAttribute(0f, DocumentationSortingAttribute.SortByFieldOrder)]
public static class ReflectionHelpers
```

The ReflectionHelpers type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AccessInternalFieldT</code></td>
<td>Cheater extension to access internal field of an object</td>
</tr>
<tr>
<td><code>CopyFields</code></td>
<td>Copy the fields from one object to another</td>
</tr>
<tr>
<td><code>GetFieldPathTTType, TValue</code></td>
<td>Returns a string path from an expression - mostly</td>
</tr>
</tbody>
</table>
used to retrieve serialized properties without hardcoding the field path. Safer, and allows for proper refactoring.


**S** GetParentObject

Get the object owner of a field. This method processes the "." separator to get from the object that owns the compound field to the object that owns the leaf field

---

**Top**

**See Also**

**Reference**

Cinemachine.Utility Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
# ReflectionHelpers Methods

The `ReflectionHelpers` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="ReflectionHelpers" /> <img src="image" alt="ReflectionHelpers" /> <img src="image" alt="ReflectionHelpers" /></td>
<td><strong>AccessInternalFieldT</strong></td>
</tr>
<tr>
<td><img src="image" alt="ReflectionHelpers" /></td>
<td><strong>CopyFields</strong></td>
</tr>
<tr>
<td><img src="image" alt="ReflectionHelpers" /> <img src="image" alt="ReflectionHelpers" /></td>
<td><strong>GetFieldPathT</strong>, <strong>TValue</strong></td>
</tr>
<tr>
<td><img src="image" alt="ReflectionHelpers" /></td>
<td><strong>GetParentObject</strong></td>
</tr>
</tbody>
</table>

Top

See Also
Reference
ReflectionHelpers Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ReflectionHelpers\n
AccessInternalField\n
Method

Cheater extension to access internal field of an object

**Namespace:** Cinemachine.Utility

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```csharp
public static T AccessInternalField<T>(
    this Type type,
    Object obj,
    string memberName
)
```

### Parameters

- **type**
  - Type: **System.Type**
  - The type of the field

- **obj**
  - Type: **System.Object**
  - The object to access

- **memberName**
  - Type: **System.String**
  - The string name of the field to access

### Type Parameters

- **T**
Return Value
Type: $T$
The value of the field in the objects

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type $Type$. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
ReflectionHelpers Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ReflectionHelpers

Copy the fields from one object to another

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>

```
public static void CopyFields(
    Object src,
    Object dst,
    BindingFlags bindingAttr = BindingFlags.
)
```

### Parameters

**src**

Type: System.Object  
The source object to copy from

**dst**

Type: System.Object  
The destination object to copy to

**bindingAttr (Optional)**

Type: System.ReflectionBindingFlags  
The mask to filter the attributes. Only those fields that get caught in the filter will be copied

## See Also
Reference

ReflectionHelpers Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
ReflectionHelpers

**GetFieldPath**<TTtype, TValue> Method

Returns a string path from an expression - mostly used to retrieve serialized properties without hardcoding the field path. Safer, and allows for proper refactoring.

**Namespace:** Cinemachine.Utility

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public static string GetFieldPath<TType, TValue>(
    Expression<Func<TType, TValue>> expr
)
```

### Parameters

**expr**

Type: `System.Linq.Expressions.Expression<Func<TType, TValue>>`


### Type Parameters

- **TType**
- **TValue**

### Return Value

Type: `String`


```c#```
See Also

Reference
ReflectionHelpers Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
ReflectionHelpers

GetParentObject

Method

Get the object owner of a field. This method processes the '.' separator to get from the object that owns the compound field to the object that owns the leaf field.

**Namespace:** Cinemachine.Utility

**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

```
public static Object GetParentObject(
  string path,
  Object obj
)
```

### Parameters

- **path**
  - **Type:** System.String
  - The name of the field, which may contain '.' separators

- **obj**
  - **Type:** System.Object
  - the owner of the compound field

### Return Value

**Type:** Object

See Also

Reference

ReflectionHelpers Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityQuaternionExtensions

Class

Extentions to the Quaternion class, usen in various places by Cinemachine

Inheritance Hierarchy

```
System
   Object
   Cinemachine.Utility
   UnityQuaternionExtensions
```

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

```
public static class UnityQuaternionExtensions
```

The UnityQuaternionExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>☀️ ApplyCameraRotation</td>
<td>Apply rotations, first about world up, then about (travelling) local right. rot.y is rotation about worldUp, and rot.x is second rotation, about local right.</td>
</tr>
</tbody>
</table>
GetCameraRotationToTarget

Get the rotations, first about world up, then about (travelling) local right, necessary to align the quaternion's forward with the target direction. This represents the tripod head movement needed to look at the target. This formulation makes it easy to interpolate without introducing spurious roll.

Normalized

Normalize a quaternion

SlerpWithReferenceUp

This is a slerp that mimics a camera operator's movement in that it chooses a path that avoids the lower hemisphere, as defined by the up param

See Also

Reference
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
The **UnityQuaternionExtensions** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="blue" alt="ApplyCameraRotation" /></td>
<td><strong>ApplyCameraRotation</strong> Apply rotations, first about world up, then about (travelling) local right. rot.y is rotation about worldUp, and rot.x is second rotation, about local right.</td>
</tr>
<tr>
<td><img src="blue" alt="GetCameraRotationToTarget" /></td>
<td><strong>GetCameraRotationToTarget</strong> Get the rotations, first about world up, then about (travelling) local right, necessary to align the quaternion's forward with the target direction. This represents the tripod head movement needed to look at the target. This formulation makes it easy to</td>
</tr>
</tbody>
</table>
interpolate without introducing spurious roll.

<table>
<thead>
<tr>
<th>💭</th>
<th>Normalized</th>
<th>Normalize a quaternion</th>
</tr>
</thead>
<tbody>
<tr>
<td>💭宋</td>
<td>SlerpWithReferenceUp</td>
<td>This is a slerp that mimics a camera operator's movement in that it chooses a path that avoids the lower hemisphere, as defined by the up param</td>
</tr>
</tbody>
</table>

See Also

Reference

UnityQuaternionExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityQuaternionExtensionsApplyCameraRotation

Method

Apply rotations, first about world up, then about (travelling) local right. rot.y is rotation about worldUp, and rot.x is second rotation, about local right.

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0

Syntax

```csharp
public static Quaternion ApplyCameraRotation(
    this Quaternion orient,
    Vector2 rot,
    Vector3 worldUp
)
```

Parameters

orient
Type: Quaternion

rot
Type: Vector2
Vector2.y is rotation about worldUp, and Vector2.x is second rotation, about local right.

worldUp
Type: Vector3
Which way is up
Return Value
Type: Quaternion


Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Quaternion. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
UnityQuaternionExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
UnityQuaternionExtensions.GetCameraRotationToTarget Method

Get the rotations, first about world up, then about (travelling) local right, necessary to align the quaternion's forward with the target direction. This represents the tripod head movement needed to look at the target. This formulation makes it easy to interpolate without introducing spurious roll.

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

**C#**

```csharp
public static Vector2 GetCameraRotationToTarget(
    this Quaternion orient,
    Vector3 lookAtDir,
    Vector3 worldUp
)
```

**JavaScript**

```javascript

```

## Parameters

**orient**  
Type: **Quaternion**  
[Missing <param name="orient"/> documentation for "M:Cinemachine.Utility.UnityQuaternionExtensions.GetCameraRotationToTarget(l**

**lookAtDir**  
Type: **Vector3**  
The worldspace target direction in which we want to look

**worldUp**  
Type: **Vector3**  
Which way is up
Return Value
Type: **Vector2**
Vector2.y is rotation about worldUp, and Vector2.x is second rotation, about local right.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type **Quaternion**. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
- UnityQuaternionExtensions Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityQuaternionExtensions.Normalize Method

Normalize a quaternion

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

**Syntax**

```csharp
public static Quaternion Normalized(  
    this Quaternion q
)
```

**Parameters**

- **q**  
  Type: Quaternion  

**Return Value**

Type: Quaternion  
The normalized quaternion. Unit length is 1.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Quaternion. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).
See Also

Reference
UnityQuaternionExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
UnityQuaternionExtensions

SlerpWithReferenceUp

Method

This is a slerp that mimics a camera operator's movement in that it
chooses a path that avoids the lower hemisphere, as defined by the up
param

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#       JavaScript

```csharp
public static Quaternion SlerpWithReferenceUp(
    Quaternion qA,
    Quaternion qB,
    float t,
    Vector3 up
)
```

Parameters

- **qA**
  - Type: Quaternion
  - First direction

- **qB**
  - Type: Quaternion
  - Second direction

- **t**
  - Type: System.Single
  - Interpolation amount

- **up**
  - Type: Vector3
Defines the up direction

Return Value
Type: Quaternion

See Also

Reference
UnityQuaternionExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
UnityRectExtensions Class

Ad-hoc extensions to the Rect structure, used by Cinemachine

Inheritance Hierarchy

- System
  - Object
  - Cinemachine.Utility
  - UnityRectExtensions

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#  JavaScript

```csharp
public static class UnityRectExtensions
```

The UnityRectExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inflated</td>
<td>Inflate a rect</td>
</tr>
</tbody>
</table>

Top

See Also

Reference
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
The **UnityRectExtensions** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inflated</td>
<td>Inflate a rect</td>
</tr>
</tbody>
</table>

## See Also

Reference

- UnityRectExtensions Class
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum

https://unity3d.com/legal/terms-of-service
Cinemachine
UnityRectExtensions.Inflated Method

Inflate a rect

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| public static Rect Inflated(  
   this Rect r,  
   Vector2 delta  
) | |

Parameters

r
Type: Rect
[Missing <param name="r"/> documentation for

delta
Type: Vector2
   x and y are added/subtracted from/to the edges of the rect, inflating it in all directions

Return Value
Type: Rect
The inflated rect

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Rect. When you use instance method
syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
UnityRectExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensions Class

Extensions to the Vector3 class, used by Cinemachine

Inheritance Hierarchy

- System
  - Object
  - Cinemachine.Utility
    - UnityVectorExtensions

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0
(2.0.0.0)

Syntax

C#  JavaScript

```csharp
public static class UnityVectorExtensions
```

The UnityVectorExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AlmostZero</td>
<td>Is the vector within Epsilon of zero length?</td>
</tr>
<tr>
<td>ClosestPointOnSegment(Vector2, Vector2, Vector2)</td>
<td>Get the closest point on a line segment.</td>
</tr>
<tr>
<td>ClosestPointOnSegment(Vector3, Vector3, Vector3)</td>
<td>Get the closest point</td>
</tr>
</tbody>
</table>
on a line segment.

<table>
<thead>
<tr>
<th></th>
<th>ProjectOntoPlane</th>
<th>Returns a non-normalized projection of the supplied vector onto a plane as described by its normal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SignedAngle</td>
<td>Get a signed angle between two vectors</td>
</tr>
<tr>
<td></td>
<td>SlerpWithReferenceUp</td>
<td>This is a slerp that mimics a camera operator's movement in that it chooses a path that avoids the lower hemisphere, as defined by the up param</td>
</tr>
</tbody>
</table>

**Fields**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epsilon</td>
<td>A useful Epsilon</td>
</tr>
</tbody>
</table>
See Also

Reference
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
# UnityVectorExtensions Methods

The **UnityVectorExtensions** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="MethodName.png" alt="MethodName" /></td>
<td><img src="Description.png" alt="Description" /></td>
</tr>
<tr>
<td><img src="MethodName.png" alt="MethodName" /></td>
<td><img src="Description.png" alt="Description" /></td>
</tr>
<tr>
<td><img src="MethodName.png" alt="MethodName" /></td>
<td><img src="Description.png" alt="Description" /></td>
</tr>
<tr>
<td><img src="MethodName.png" alt="MethodName" /></td>
<td><img src="Description.png" alt="Description" /></td>
</tr>
<tr>
<td><img src="MethodName.png" alt="MethodName" /></td>
<td><img src="Description.png" alt="Description" /></td>
</tr>
</tbody>
</table>

- **AlmostZero**: Is the vector within Epsilon of zero length?
- **ClosestPointOnSegment(Vector2, Vector2, Vector2)**: Get the closest point on a line segment.
- **ClosestPointOnSegment(Vector3, Vector3, Vector3)**: Get the closest point on a line segment.
- **ProjectOntoPlane**: Returns a non-normalized projection of the supplied vector onto a plane as described by its normal.
- **SignedAngle**: Get a signed angle between two vectors.
| SlerpWithReferenceUp | This is a slerp that mimics a camera operator's movement in that it chooses a path that avoids the lower hemisphere, as defined by the up param |

See Also

Reference
UnityVectorExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
UnityVectorExtensionsAlmostZero Method

Is the vector within Epsilon of zero length?

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

▲ Syntax

C#  
```
public static bool AlmostZero(
    this Vector3 v
)
```

Return Value  
Type: **Boolean**  
True if the square magnitude of the vector is within Epsilon of zero

Usage Note  
In Visual Basic and C#, you can call this method as an instance method on any object of type **Vector3**. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).
See Also

Reference
UnityVectorExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensions.ClosestPointOnSegment Method

## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ClosestPointOnSegment(Vector2, Vector2, Vector2)</code></td>
<td>Get the closest point on a line segment.</td>
</tr>
<tr>
<td><code>ClosestPointOnSegment(Vector3, Vector3, Vector3)</code></td>
<td>Get the closest point on a line segment.</td>
</tr>
</tbody>
</table>

## See Also

- **Reference**
  - UnityVectorExtensions Class
  - Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
UnityVectorExtensionsClosestPointOnSegment Method (Vector2, Vector2, Vector2)

Get the closest point on a line segment.

**Namespace:** Cinemachine.Utility
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

```csharp
public static float ClosestPointOnSegment(
    this Vector2 p,
    Vector2 s0,
    Vector2 s1
)
```

### Parameters

- **p**
  - Type: Vector2
  - A point in space

- **s0**
  - Type: Vector2
  - Start of line segment

- **s1**
  - Type: Vector2
  - End of line segment

### Return Value

- Type: Single
  - The interpolation parameter representing the point on the segment, with 0==s0, and 1==s1
Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type `Vector2`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
UnityVectorExtensions Class
ClosestPointOnSegment Overload
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensionsClosestPointOnSegment Method (Vector3, Vector3, Vector3)

Get the closest point on a line segment.

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
<tbody>
<tr>
<td>public static float ClosestPointOnSegment(</td>
<td></td>
</tr>
<tr>
<td>this Vector3 p,</td>
<td></td>
</tr>
<tr>
<td>Vector3 s0,</td>
<td></td>
</tr>
<tr>
<td>Vector3 s1</td>
<td></td>
</tr>
<tr>
<td>)</td>
<td></td>
</tr>
</tbody>
</table>

### Parameters

- **p**
  - Type: **Vector3**
  - A point in space

- **s0**
  - Type: **Vector3**
  - Start of line segment

- **s1**
  - Type: **Vector3**
  - End of line segment

### Return Value

- Type: **Single**
  - The interpolation parameter representing the point on the segment, with 0==s0, and 1==s1
Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type \texttt{Vector3}. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference

- UnityVectorExtensions Class
- ClosestPointOnSegment Overload
- Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensions.ProjectOntoPlane Method

Returns a non-normalized projection of the supplied vector onto a plane as described by its normal.

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

## Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| `public static Vector3 ProjectOntoPlane(
    this Vector3 vector,
    Vector3 planeNormal
)` |

### Parameters

- **vector**  
  Type: `Vector3`  

- **planeNormal**  
  Type: `Vector3`  
  The normal that defines the plane. Cannot be zero-length.

### Return Value

Type: `Vector3`  
The component of the vector that lies in the plane.

### Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type `Vector3`. When you use instance
method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
UnityVectorExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
Cinemachine
UnityVectorExtensionsSignedAngle Method

Get a signed angle between two vectors

**Namespace**: Cinemachine.Utility  
**Assembly**: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>JavaScript</th>
</tr>
</thead>
</table>
| ```csharp
public static float SignedAngle(
    Vector3 from,
    Vector3 to,
    Vector3 refNormal
)
``` |  |

### Parameters

**from**
- Type: `Vector3`  
  Start direction

**to**
- Type: `Vector3`  
  End direction

**refNormal**
- Type: `Vector3`  
  This is needed in order to determine the sign. For example, if from an to lie on the XZ plane, then this would be the Y unit vector, or indeed any vector which, when dotted with Y unit vector, would give a positive result.

### Return Value
Type: Single
The signed angle between the vectors

See Also

Reference
UnityVectorExtensions Class
Cinemachine.Utility Namespace

Visit the Cinematic Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensions

This is a slerp that mimics a camera operator's movement in that it chooses a path that avoids the lower hemisphere, as defined by the up param

Namespace: Cinemachine.Utility
Assembly: Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

Syntax

C#

```csharp
public static Vector3 SlerpWithReferenceUp(
    Vector3 vA,
    Vector3 vB,
    float t,
    Vector3 up
)
```

JavaScript

Parameters

vA
Type: Vector3
First direction

vB
Type: Vector3
Second direction

t
Type: System.Single
Interpolation amount

up
Type: Vector3
Defines the up direction

Return Value
Type: **Vector3**


See Also

Reference
UnityVectorExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensions Fields

The UnityVectorExtensions type exposes the following members.

FIELDS

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚪️ Epsilon</td>
<td>A useful Epsilon</td>
</tr>
</tbody>
</table>

See Also

Reference

UnityVectorExtensions Class
Cinemachine.Utility Namespace

Visit the Cinemachine Forum
https://unity3d.com/legal/terms-of-service
UnityVectorExtensionsEpsilon Field

A useful Epsilon

**Namespace:** Cinemachine.Utility  
**Assembly:** Cinemachine (in Cinemachine.dll) Version: 2.0.0.0 (2.0.0.0)

### Syntax

**C#**

```csharp
public const float Epsilon = 0.0001f
```

**JavaScript**

```javascript
//

```

### Field Value

Type: **Single**

### See Also

**Reference**

UnityVectorExtensions Class  
Cinemachine.Utility Namespace

Visit the Cinemachine Forum  
https://unity3d.com/legal/terms-of-service