Animation Compression Codec Reference

Reference the settings and properties of the Animation Compression Codecs.
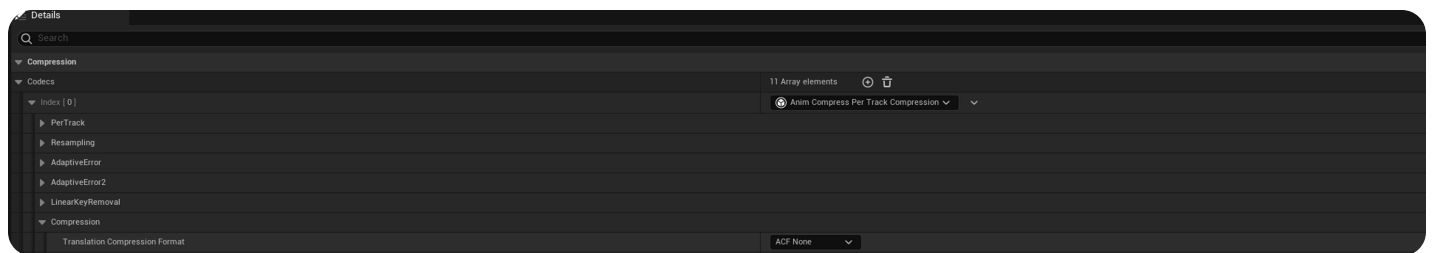
Compression Codecs are compression algorithms and methods you can use to drive **Animation Compression Settings** assets, to compress animation data. You can use the following document to reference a list and description of the available **Bone** and **Curve Compression Codecs** available in **Unreal Engine**.

Bone Compression Codec Reference

Here you can reference each of the bone compression codecs packaged with Unreal Engine, as well as a description of their functionality and their available properties.

Anim Compress Bitwise Compress Only

The Anim Compress Bitwise Compress Only codec focuses on Bitwise compression and retains all animation keys.

Here you can reference a list of the **Anim Compress Bitwise Compress Only** codec's properties and a description of their functionality:

Property	Description
Compression	Set the following properties to control how the compression codec formats the compressed animation data: • **Translation Compression Format**: Set the format for bitwise compression of **translation data**. See the [Compression Format Reference](#) for more information. • **Rotation Compression Format**: Set the format for bitwise compression of **rotation** data. See the [Compression Format Reference](#) for more information. • **Scale Compression Format**: Set the format for bitwise compression of **scale** data. See the [Compression Format Reference](#) for more information. • **Description**: Set a description for this codec.

Anim Compress Least Destructive

The Anim Compress Least Destructive codec restores any present animation compression to raw data.

Here you can reference a list of the **Anim Compress Least Destructive** codec's properties and a description of their functionality:

Property	Description
Compression	Set the following properties to control how the compression codec formats the compressed animation data: • **Translation Compression Format**: Set the format for bitwise compression of **translation data**. See the [Compression Format Reference](#) for more information. • **Rotation Compression Format**: Set the format for bitwise compression of **rotation** data. See the [Compression Format Reference](#) for more information. • **Scale Compression Format**: Set the format for bitwise compression of **scale** data. See the [Compression Format Reference](#) for more

Property	Description
	information. • **Description**: Set a description for this codec.

Anim Compress Per Track Compression

The Anim Compress Per Track Compression codec focuses compression to a per-track basis, and compresses each track individually.

Here you can reference a list of the **Anim Compress Per track Compression** codec's properties and a description of their functionality:

Property	Description
PerTrack	Set the following properties to control how the compression codec operates for each track: • **Max Zeroing Threshold**: Set the maximum threshold to use when replacing a component with zero. Lower values retain more keys, but yield less compression. • **Max Pos Diff Bitwise**: Set the maximum **position** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Angle Diff Bitwise**: Set the maximum **angle** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Scale Diff Bitwise**: Set the maximum **scale** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Allowed Rotation Formats**: Set which encoding format the per-track compressor is allowed to try on **rotation** keys. For available options, see the [Compression Format Reference](#). • **Allowed Translation Formats**: Which encoding formats the per-track compressor is allowed to try on **translation** keys. For available options, see the [Compression Format Reference](#) • **Allowed Scale Formats**: Which encoding formats the per-track compressor is allowed to try on **scale** keys. For available options, see the [Compression Format Reference](#)

Property	Description
Resampling	Set the following properties to control how the compression codec operates when resampling animation data: • **Resample Animation**: When enabled, the codec will resample the animation's framerate with the value set as the **Resample Framerate** property's value. • **Resampled Framerate**: When the **Resampling** property is enabled, set the framerate to resample animations with during compression. • **Min Keys for Resampling**: Set the **minimum** number of keys an animation should have for a resample to be performed. All animations with fewer keys than this property, will not be resampled when compressing.
AdaptiveError	Set the following properties to control how the compression codec's error threshold values animation data: • **Use Adaptive Error**: When enabled, the error threshold will be adjusted based on the **height** value within the skeleton. • **Use Override for End Effectors**: When enabled, use the **Min Effector Diff** property's value as the threshold for end effectors. • **Track Height Bias**: Set a value that is used to determine a bias that is added to the track height before using it to calculate the adaptive error. • **Parenting Divisor**: When the **UseAdpative Error** property is enabled, set a value to reduce the error tolerance the further up the tree that a key occurs. The following is the formula used: `EffectiveErrorTolerance = Max(BaseErrorTolerance / p[ower( parentingDiviosor, Max(Heigh + Bias,0) * ParentingDivisiorExponent), ZeroingThreshold)` • **Parenting Divisor Exponent**: When the **UseAdpative Error** property is enabled, set a value to exponentially reduce the error tolerance the further up the tree that a key occurs. The following is the formula used: `EffectiveErrorTolerance = Max(BaseErrorTolerance / p[ower( parentingDiviosor, Max(Heigh + Bias,0) * ParentingDivisiorExponent), ZeroingThreshold)`

Property	Description
AdaptiveError2	Set the following properties to apply additional control how the compression codec's error threshold values animation data: • **Use Additive Error 2**: When enabled, the adaptive error system will determine how much error to allow for each track, based on the error introduced in end effectors due to errors in the track. • **Rotation Error Source Ratio**: Set a value to determine the ratio of how much error in an end effector **rotation** can come from a given track's rotation error or translation error. If a value of `1` is used, all of it must come from rotation error. If a value of `0.5` is used, half of the error ratio can come from each. If a value of `0.0` is used, all of it must come from translation error. • **Translation Error Source Ratio**: Set a value to determine the ratio of how much error in an end effector **translation** can come from a given track's rotation error or translation error. If a value of `1` is used, all of it must come from translation error. If a value of `0.5` is used, half of the error ratio can come from each. If a value of `0.0` is used, all of it must come from the rotation error. • **Scale Error Source Ratio**: Set a value to determine the ratio of how much end effector scale error can come from a given track's rotation error or scale error. If a value of `1` is used, all of it must come from rotation error. If a value of `0.5` is used, half of the error ratio can come from each. If a value of `0.0` is used, all of it must come from the scale error. • **Max Error Per Track Ratio**: Set a fraction value that determines how much of the total error budget can be introduced by any particular track.
LinearKeyRemoval	Set the following properties to control how the compression codec removes linear keys from the animation data: • **Max Pos Diff**: Set a maximum **position** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Angle Diff**: Set a maximum **angle** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Scale Diff**: Set a maximum **scale** difference to use when testing if an animation key may be removed. Lower values retain

Property	Description
	more keys, but yield less compression. - **Max Effector Diff**: As keys are tested for removal, we monitor the effects all the way down to the end effectors. If their position changed by more than this amount as a result of removing a key, the key will be retained. This value is used for all bones except the end-effector's parent. - **Min Effector Diff**: As keys are tested for removal, we monitor the effects all the way down to the end effectors. If their position changed by less than this amount as a result of removing a key, the key will be retained. This value is used for all bones except the end-effector's parent. - **Effector Diff Socket**: Set an error threshold for End Effectors with Sockets attached to them. Typically End Effectors with Sockets are more important bones, where you may want to be less aggressive with compression to retain more detail. - **Parent Key Scale**: Set a **scale** value which increases the likelihood that a bone will retain a key if its parent also had a key at the same time position. Higher values can remove shaking artifacts from the animation, at the cost of compression. - **Retarget**: When enabled, nodes will be adjusted to compensate for compression errors, as the animation is compressed. When disabled, no node adjustment will occur. - **Actually Filter Linear Keys**: When enabled, the final filtering step will occur, or only the retargeting after Bitwise compression. If this property is enabled and the **Retarget** property is disabled, then the linear compressor will do no better than the underlying bitwise compressor, and be extremely slow.
Compression	Set the following properties to control how the compression codec formats the compressed animation data: - **Translation Compression Format**: Set the format for bitwise compression of **translation data**. See the [Compression Format Reference](#) for more information. - **Rotation Compression Format**: Set the format for bitwise compression of **rotation** data. See the [Compression Format Reference](#) for more information. - **Scale Compression Format**: Set the format for bitwise compression of **scale** data. See the [Compression Format Reference](#) for more information.

Property	Description
	• **Description**: Set a description for this codec.

Anim Compress Removes Every Second Key

Removes every second key present in the animation.

Here you can reference a list of the **Anim Compress Removes Every Second Key**, and a description of their function:

Property	Description
AnimationCompressionAlgorithm_RemoveEverySecondKey	Set the following properties to control how the compression codec removed keys: **Min Keys**: Animations with fewer than MInKeys will not lose any keys. **Start at Second Key**: When enabled, the codec will begin to remove keys from the second key. For example, keys `1`, `3`, `5` will be removed. When disabled, the codec will begin to remove keys from the first key. For example, keys `0`, `2`, `4` will be removed.
Compression	Set the following properties to control how the compression codec formats the compressed animation data: **Translation Compression Format**: Set the format for bitwise compression of **translation data**. See the [Compression Format Reference](#) for more information. **Rotation Compression Format**: Set the format for bitwise compression of **rotation** data. See the [Compression Format Reference](#) for more information. **Scale Compression Format**: Set the format for bitwise compression of **scale** data. See

Property	Description
	the [Compression Format Reference](#) for more information. • **Description**: Set a description for this codec.

Anim Compress Remove Linear Keys

Removes any keys that are unchanged from surrounding keys.

Here you can reference a list of the **Anim Compress Remove Linear Keys**, and a description of their function:

Property	Description
LinearKeyRemoval	Set the following properties to control how the compression codec removes linear keys from the animation data: • **Max Pos Diff**: Set a maximum **position** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Angle Diff**: Set a maximum **angle** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Scale Diff**: Set a maximum **scale** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Effector Diff**: As keys are tested for removal, we monitor the effects all the way down to the end effectors. If their position changed by more than this amount as a result of removing a key, the key will be retained. This value is used for all bones except the end-effector's parent. • **Min Effector Diff**: As keys are tested for removal, we monitor the effects all the way down to the end effectors. If their position changed by less than this amount as a result of removing a key, the key will be retained. This value is used for all bones except the end-effector's parent. • **Effector Diff Socket**: Set an error threshold for End Effectors with Sockets attached to them. Typically End Effectors with Sockets are

Property	Description
	more important bones, where you may want to be less aggressive with compression to retain more detail. • **Parent Key Scale**: Set a **scale** value which increases the likelihood that a bone will retain a key if its parent also had a key at the same time position. Higher values can remove shaking artifacts from the animation, at the cost of compression. • **Retarget**: When enabled, nodes will be adjusted to compensate for compression errors, as the animation is compressed. When disabled, no node adjustment will occur. • **Actually Filter Linear Keys**: When enabled, the final filtering step will occur, or only the retargeting after Bitwise compression. If this property is enabled and the **Retarget** property is disabled, then the linear compressor will do no better than the underlying bitwise compressor, and be extremely slow.
Compression	Set the following properties to control how the compression codec formats the compressed animation data: • **Translation Compression Format**: Set the format for bitwise compression of **translation data**. See the [Compression Format Reference](#) for more information. • **Rotation Compression Format**: Set the format for bitwise compression of **rotation** data. See the [Compression Format Reference](#) for more information. • **Scale Compression Format**: Set the format for bitwise compression of **scale** data. See the [Compression Format Reference](#) for more information. • **Description**: Set a description for this codec.

Anim Compress Removes Trivial Keys

Removes any keys where position and orientation of Assets are unchanged from surrounding keys.

Here you can reference a list of the **Anim Compress Removes Trivial Keys**, and a description of their function:

Property	Description
AniamtionCompressionAlgorithm_RemoveTrivialKeys	Set the following properties to control how the compression codec removes trivial keys from the animation data: • **Max Pos Diff**: Set a maximum **position** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Angle Diff**: Set a maximum **angle** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression. • **Max Scale Diff**: Set a maximum **scale** difference to use when testing if an animation key may be removed. Lower values retain more keys, but yield less compression.
Compression	Set the following properties to control how the compression codec formats the compressed animation data: • **Translation Compression Format**: Set the format for bitwise compression of **translation data**. See the [Compression Format Reference](#) for more information. • **Rotation Compression Format**: Set the format for bitwise compression of **rotation** data. See the [Compression Format Reference](#) for more information. • **Scale Compression Format**: Set the format for bitwise compression of **scale** data. See the [Compression Format Reference](#) for more information. • **Description**: Set a description for this codec.

Bitwise Compression Format Reference

Here you can reference a list of the available bitwise compression data formats, and a description of their function:

Format	Description
ACF None	This formant retains all **32 floats**, performing no compression. This format essentially retains raw animation data.
ACF Float 96No W	This format only compresses rotation quaternion **X**, **Y**, and **Z** rotation data, as a **float32** and the **W** component is dropped and reconstructed during decompression. This format retains near raw animation data with minimal precision loss.
ACF Fixed 48No W	This format only compresses rotation quaternion **X**, **Y**, and **Z** rotation data, and is stored on **16 bits** each, as quantized data. The quaternion (**W**) component is also dropped.
ACF Interval Fixed 32 No W	This format stores the compressed values on **11-11-10 bits per component with range reduction**. For rotations, the quaternion (**W**) component is dropped.
ACF Fixed 32No W	This format stores the compressed values on **11-11-10 bits per component**. For rotations, the quaternion (**W**) component is dropped.
ACF Float 32No W	This format stores compressed values using a custom float format with 6 or 7 bits for the mantissa and 3 bits for the exponent. For rotations, the quaternion (W) component is dropped.
ACF Identity	This format always returns the quaternion (**W**) to the rotation track.

Curve Compression Codec Reference

Here you can reference each of the curve compression codecs packaged with Unreal Engine, as well as a description of their functionality and their available properties.

Compress Rich Curves

This codec compresses rich animation curves when compressing Animation Sequences. Here you can reference the Compress Rich Curve codec's properties and a description of their function:

Property	Description
Max Curve Error	Set the maximum error allowed when compressing the rich animation curves.
Use Anim Sequence Sample Rate	When enabled, the animation will be compressed using the Animation Sequence sample rate. When disabled, the animation will be sampled using the **Error Sample Rate** property's value.
Error Sample Rate	When the **Use Anim Sequence Sample Rate** property is disabled, set the value that will be used for the animation's sample rate, when compressing.

Uniform Indexable

This codec uniformly **indexes** animation curves when compressing Animation Sequences.

Uniformly Sampled

This codec uniformly **samples** animation curves when compressing Animation Sequences. Here you can reference the Uniformly Sampled codec's properties and a description of their function:

Property	Description
Use Anim Sequence Sample Rate	When enabled, the animation will be compressed using the Animation Sequence sample rate. When disabled, the animation will be sampled using the **Error Sample Rate** property's value.

Property	Description
Error Sample Rate	When the **Use Anim Sequence Sample Rate** property is disabled, set the value that will be used for the animation's sample rate, when compressing.