**动画序列**

**本页面的内容：**

* [编辑](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/index.html#%E7%BC%96%E8%BE%91)
* [播放](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/index.html#%E6%92%AD%E6%94%BE)
* [通知](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/index.html#%E9%80%9A%E7%9F%A5)
* [Curves](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/index.html#curves)
* [Compression](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/index.html#compression)
* [Anim Asset Details](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/index.html#animassetdetails)

**动画序列（AnimationSequence）** 是一个可以在骨架网格体上播放的独立的动画资源。它们包含了一些关键帧，这些关键帧可以及时地指出某个特定点处的一个骨骼的位置、旋转度及缩放比例。通过回放序列中的这些关键帧，并在将它们互相混合，使得骨架网格体可以产生平滑的动画效果。

每个 *动画序列* 资源指向一个特定的骨架，且仅可以在那个骨架上进行播放。这意味着，为了在多个骨架网格物体间共享动画，所有网格物体都必须使用同样的骨架资源。

An example of a Skeleton using an Animation Sequence can also be found on the [Animation Content Examples](http://api.unrealengine.com/latest/CHN/Resources/ContentExamples/Animation/index.html) page under section 1.1.

**编辑**

**动画序列编辑器** 提供了预览及编辑单独的 *动画序列* 资源的功能。在其属性中可以设置 *动画序列* 的压缩机制，可以添加动画通知事件(也称为通知)来基于动画触发相机特效、粒子特效、声效等。

请参照 [编辑动画序列](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/Editor/index.html) 页面获得更多信息。

**播放**

*AnimationSequence(动画序列)* 的播放尽管可以通过代码来执行，但一般都通过应用到 *骨架网格物体组件* 的 *动画蓝图* 来处理。在 **动画图表** 中，可以对用于创建 *动画蓝图* 的骨架的所有 *动画序列* 进行采样。这些动画序列可以作为 序列播放器节点进行放置，其输出是对该 *动画序列* 采样后而生成的姿势。

请参照 [动画序列用户指南](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/UserGuide/index.html) 获得关于在 *动画蓝图* 中应用 *动画序列* 的指南。

**通知**

动画通知(简称AnimNotifies或通知)使得动画相关的程序员可以设置在动画序列的特定点处发生的事件。通知通常用于这样的特效，比如走动时的脚步声、跑动动画或在动画中产生一个粒子特效。然而，它有很多种不同的用途，因为您可以使用自定义的通知类型来扩展该系统，从而满足任何类型游戏的需求。

请参照 [动画通知 (通知)](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/Notifies/index.html) 页面获得关于各种可用的通知的描述及应用信息。

通过使用角色编辑器中的 [动画序列编辑器](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/Editor/index.html) ，您可以创建及编辑通知。

**Curves**

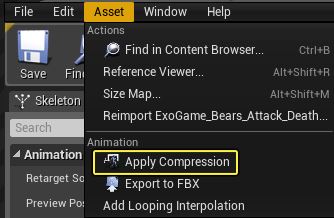
曲线提供了在动画正在播放过程中改变材质参数或顶点变形目标的方法。 其工作流程非常简单，只需要您简单地指定您要修改的资源(一个材质或顶点变形目标)，相应地命名该曲线，然后调整动画播放期间的关键帧的值。

See [动画曲线](http://api.unrealengine.com/latest/CHN/Engine/Animation/Sequences/Curves/index.html) for descriptions and usage information on how to set up and apply curves in Persona.

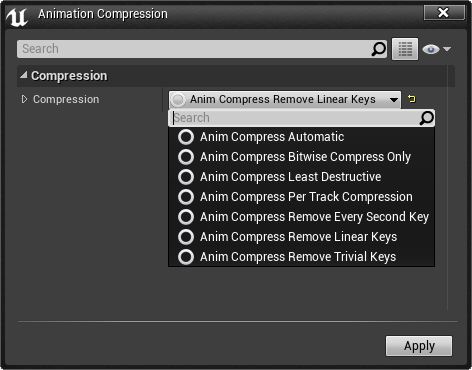
**Compression**

Animation Sequence assets when opened up in **Persona** can have compression applied to them.

To do this, from the **Menu Bar** select **Asset** then select **Apply Compression** (as seen below).



When choosing to apply compression, the **Animation Compression** window will open.

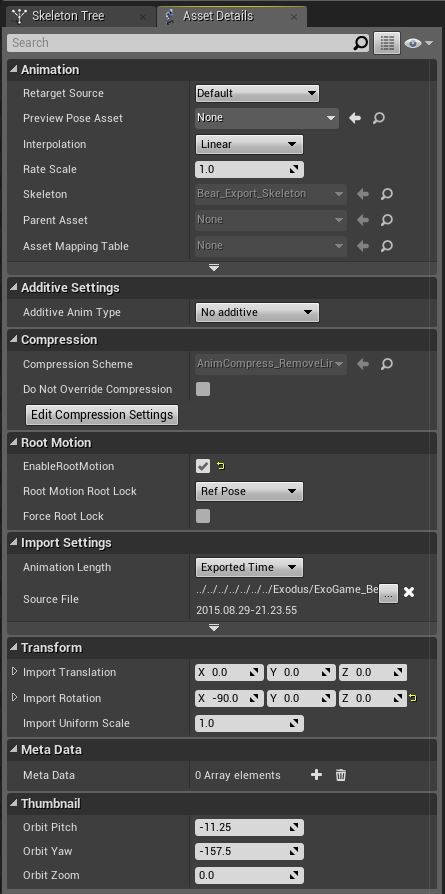


Each option provides a different compression method as outlined in the table below:

| **Compression Type** | **Description** |
| --- | --- |
| **Automatic** | Animation compression algorithm that is just a shell for trying the range of other compression schemes and picking the smallest result within a configurable error threshold. |
| **Bitwise Compress Only** | Bitwise animation compression only; performs no key reduction. |
| **Least Destructive** | Reverts any animation compression, restoring the animation to the raw data. |
| **Remove Every Second Key** | Keyframe reduction algorithm that simply removes every second key. |
| **Remove Linear Keys** | Keyframe reduction algorithm that simply removes keys which are linear interpolations of surrounding keys. |
| **Compress each track independently** | Keyframe reduction algorithm that removes keys which are linear interpolations of surrounding keys, as well as choosing the best bitwise compression for each track independently. |
| **Remove Trivial Keys** | Removes trivial frames of tracks when position or orientation is constant over the entire animation from the raw animation data. |

**Anim Asset Details**

When an Animation Sequence asset is opened in **Persona**, there are several properties that can be adjusted inside the **Anim Asset Details** panel which can determine how the asset is handled. These properties include things such as tweaking playback speed, enabling/disabling Root Motion, assigning any Meta Data to the asset and others which are outlined in the tables below.



**Linear Key Removal**

|  |  |
| --- | --- |
| **Max Pos Diff** | Max 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** | Max 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** | Max 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 changes 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-effectors parent. |
| **Min Effector Diff** | Similar to Max Effector Diff, however this value is used for the end-effectors parent, allowing tighter restrictions near the end of a skeletal chain. |
| **Effector Diff Socket** | Error threshold for End Effectors with Sockets attached to them. Typically for more important bones where we want to be less aggressive with compression. |
| **Parent Key Scale** | 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** | If true, as the animation is compressed, adjust animated nodes to compensate for compression error. If false, do not adjust animated nodes. |
| **Actually Filter Linear Keys** | Controls whether the final filtering step will occur, or only the retargeting after bitwise compression. If both this and **Retarget** are false, then the linear compressor will do no better than the underlying bitwise compressor, extremely slowly. |

**Animation**

|  |  |
| --- | --- |
| **Retarget Source** | When retargeting, this pose will be used as a base of animation. |
| **Rate Scale** | Number for tweaking playback rate of this animation globally. |
| **Skeleton** | Contains the skeleton associated with this Montage. Cannot be changed in the editor. |

**Compression**

|  |  |
| --- | --- |
| **Description** | Name of Compression Scheme used for this asset. |
| **Do Not Override Compression** | Do not attempt to override compression scheme when running Compress Animations commandlet. Some high frequency animations are too sensitive and shouldn't be changed. |

**Additive Settings**

|  |  |
| --- | --- |
| **Additive Anim Type** | The type of additive animation type to use: No Additive, Local Space or Mesh Space. Local space is additive and delta is calculated by local space. Mesh Space is additive and delta will be applied in component space. |
| **Base Pose Type** | Decides how the additive delta is to be calculated: None (No Base Pose), Reference Pose (The mesh's default reference pose), Selected Animation Scaled (choose another animation and scale the animation), Selected Animation Frame (choose another animation and only based on the frame number). Base Pose Type is only available if Additive Anim Type is set to Local or Mesh Space. |

**Root Motion**

|  |  |
| --- | --- |
| **Enable Root Motion** | If this is on, it will allow extracting of [Root Motion](http://api.unrealengine.com/latest/CHN/Engine/Animation/RootMotion/index.html) . |
| **RootMotion Root Lock** | Root Bone will be locked to that position specified when extracting root motion. |

**Import Settings**

|  |  |
| --- | --- |
| **Asset Import Data** | The importing data and options used for the mesh. |

**Meta Data**

|  |  |
| --- | --- |
| **Meta Data** | This is Meta Data that can be saved with the asset. The meta data is a Blueprintable class derived from the Anim Meta Data class. This allows you to add custom meta data to animation assets (Anim Sequence, Anim Montage, Anim Composite, and Blendspace are all supported). You can query the data from an animation asset in C++ by calling the GetMetaData() method, or by using GetSectionMetaData() on Anim Montages. |

**Thumbnail**

|  |  |
| --- | --- |
| **Orbit Pitch** | The pitch of the orbit camera around the asset. |
| **Orbit Yaw** | The yaw of the orbit camera around the asset. |
| **Orbit Zoom** | The offset from the bounds sphere distance from the asset. |