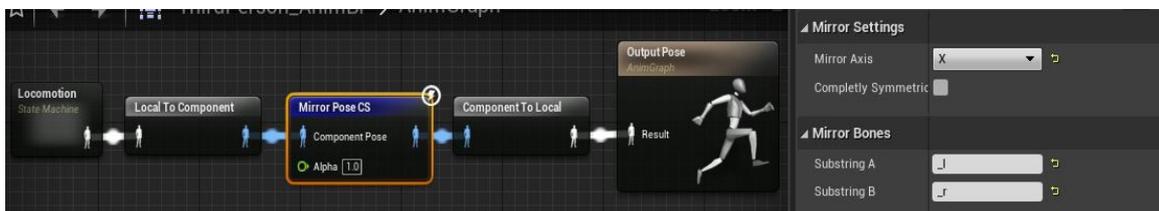


Mirror Animation System V2:

Quick Start:

1. Enable the plugin for your project.
2. Open the AnimBlueprint of one of your characters.
3. Add a “Mirror Pose CS” node right before the Output Pose.
4. Inside this node, set the Mirror Axis to the axis you wish to mirror animations across (“Mirror Axis = X” for the Mannequin).
5. Inside this node, set the Substring A, and Substring B parameters to the naming convention you use to identify bones with a twin (“_l” and “_r” for the Mannequin).
6. Compile and watch the results, this simple setup should be enough to mirror most skeleton types (like the UE 4 Mannequin or Mixamo characters).



Key Concepts:

To use all of the plugin's features you'll have to familiarize yourself with the following terms:

-Twin Bone:

In a skeleton hierarchy there are pairs of bones (like the hands, shoulders, legs), whose transform you want to mirror with the one from the opposite side. These bones are the ones referred to as Twin Bones.

-Non-Twin Bone:

Other bones you want mirrored, but lack a twin (like the head, spine, pelvis), will be called Non-Twin Bones.

-Mirror Bone:

This is the structure that holds the parameters needed to mirror a bone's transform in Bone Space:

Bone Name: Specifies the name of the bone you wish to mirror.

Mirror Axis: Specifies the plane that this bone's transform will be mirrored across.

Flip Axis: Specifies the axis of the bone's Transform that will be flipped.

Rotation Offset: Skeletons like the UE 4 Mannequin's skeleton are not as well tuned for mirroring as others like Mixamo's skeletons. Thus this parameter exists as a final tweak to add or subtract rotation to the bone's transform.

Is Twin Bone: Boolean parameter to specify whether this bone is a twin or not.

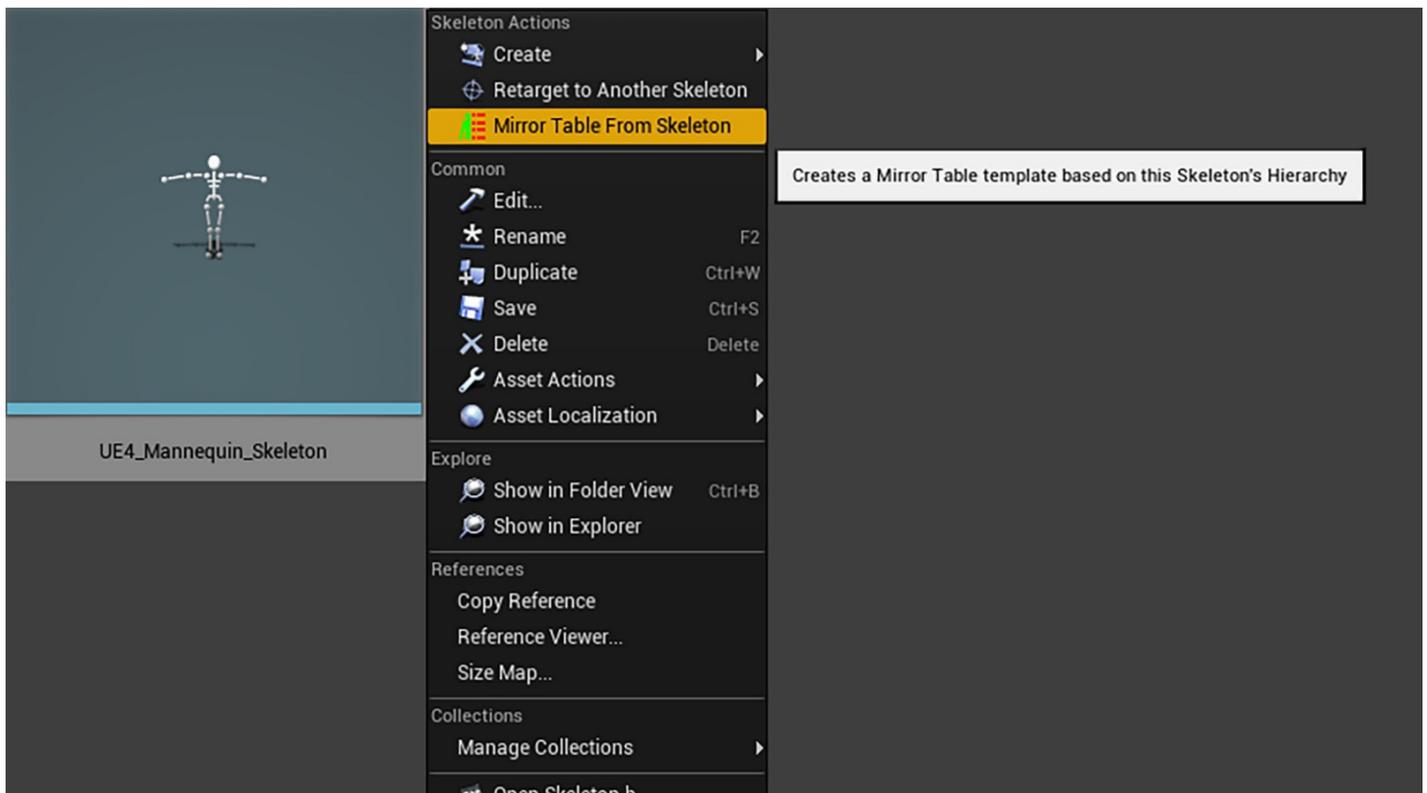
Twin Bone Name: Specifies the name of the twin bone that will be mirrored with.

Mirror Translation: Boolean parameter to specify whether you wish to mirror the translation between the twin bones or not.

-Mirror Table:

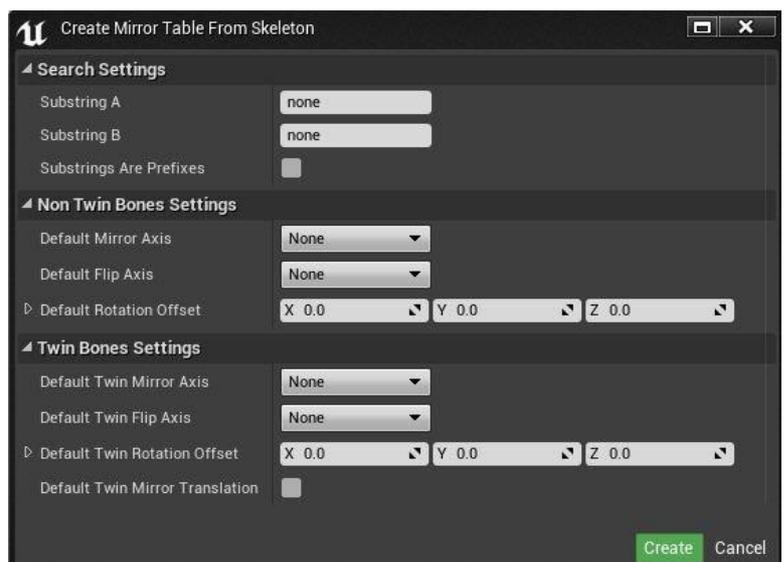
To mirror animations using the Bone Space method, you want to first make a Mirror Table for the character's skeleton.

You can easily do this by right clicking the character's skeleton and using the *Mirror Table From Skeleton* tool.



Clicking this will open the following window. Here you input the naming convention you use for the twin bones inside your skeleton (*_l* and *_r* for example).

Then you setup some default values. this serves as preliminary setup before you start adjusting the Mirror Table.

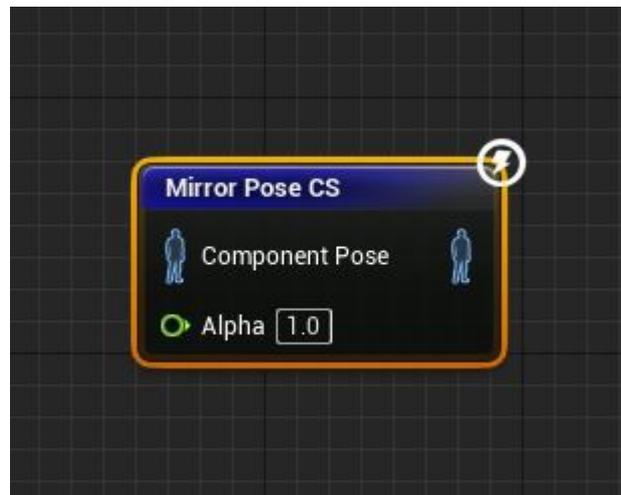


Note: in case you want to create a new Mirror Table from scratch you would go to the *Miscellaneous* section, click on *Data Asset*, and select the *MirrorTable* class.

AnimGraph Nodes:

Mirror Pose CS:

This Node takes a component space pose and mirrors it according to a Mirror Axis parameter. It is the most intuitive approach in the plugin to mirroring animations.

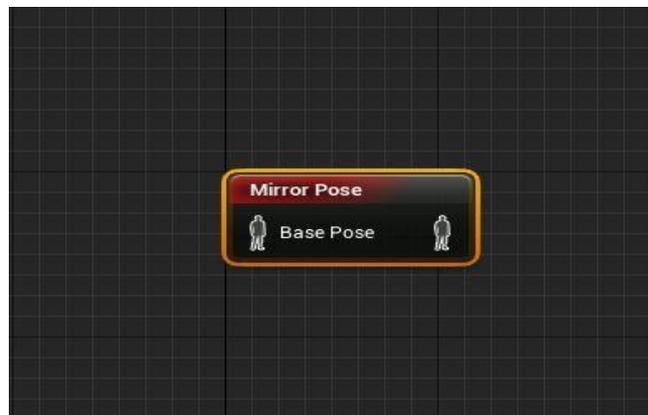


Mirror Pose:

This node takes a bone space pose and mirrors the bones that have been mapped in the Mirror Table.

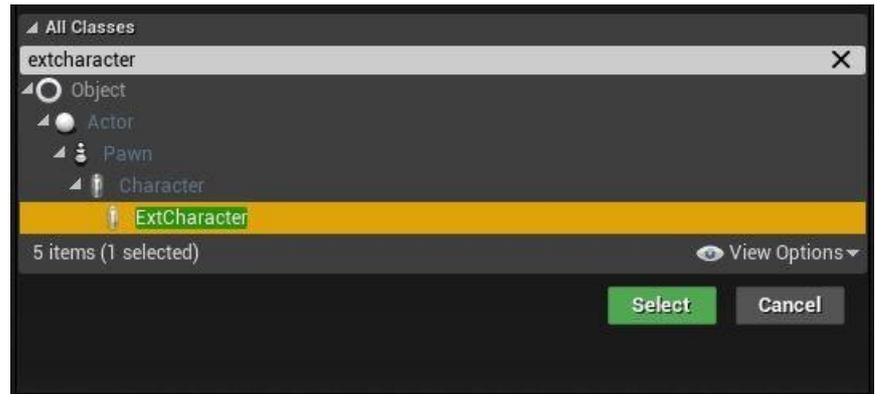
When setting up the Mirror Axis, Flip Axis and Rotation Offset variables keep in mind that the bone transforms are in bone space, which means that the axis and rotation are mirrored relative to the parent bone.

Because of this, if your skeleton is setup similarly to the UE4 Mannequin, some bones like *upperarm_l* and *upperarm_r* only require to set *IsTwin Bone = true* along with writing the name of its twin bone, leaving the other options as is.

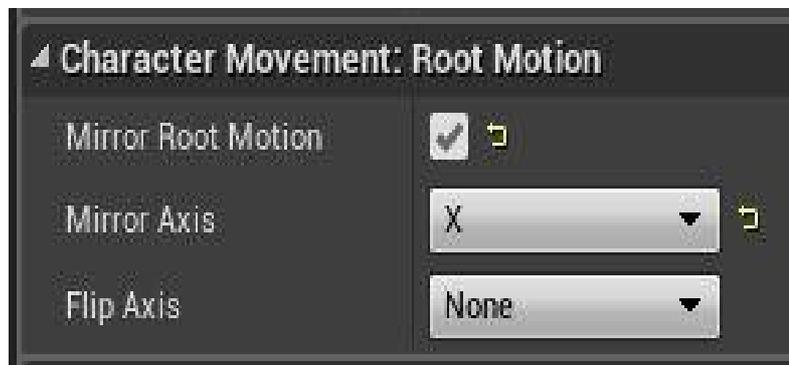


Mirroring Root Motion:

To mirror the character's root motion, you create a new actor that derives from the *ExtCharacter* class.

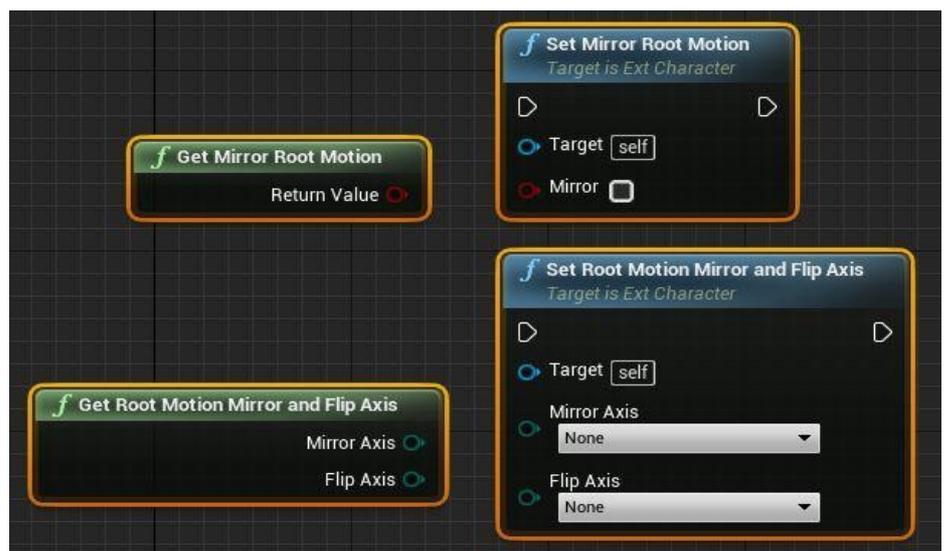


This new character will have a custom UCharacterMovement Component that has been tweaked to mirror the Root Motion that receives based on the following settings:



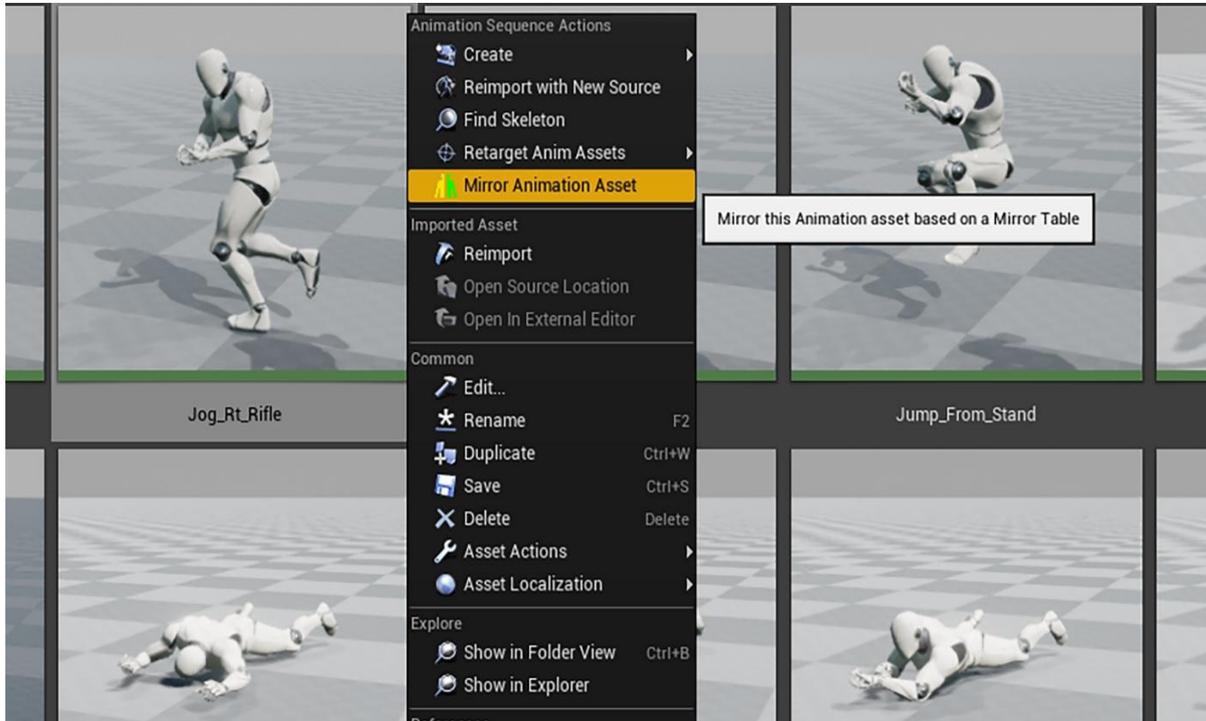
Typically, you will edit the Mirror Axis and Flip Axis variables inside the movement component's settings.

Then inside the character's blueprint use the *Set Mirror Root Motion* node to true or false at runtime.



Mirror Animation Asset Tool:

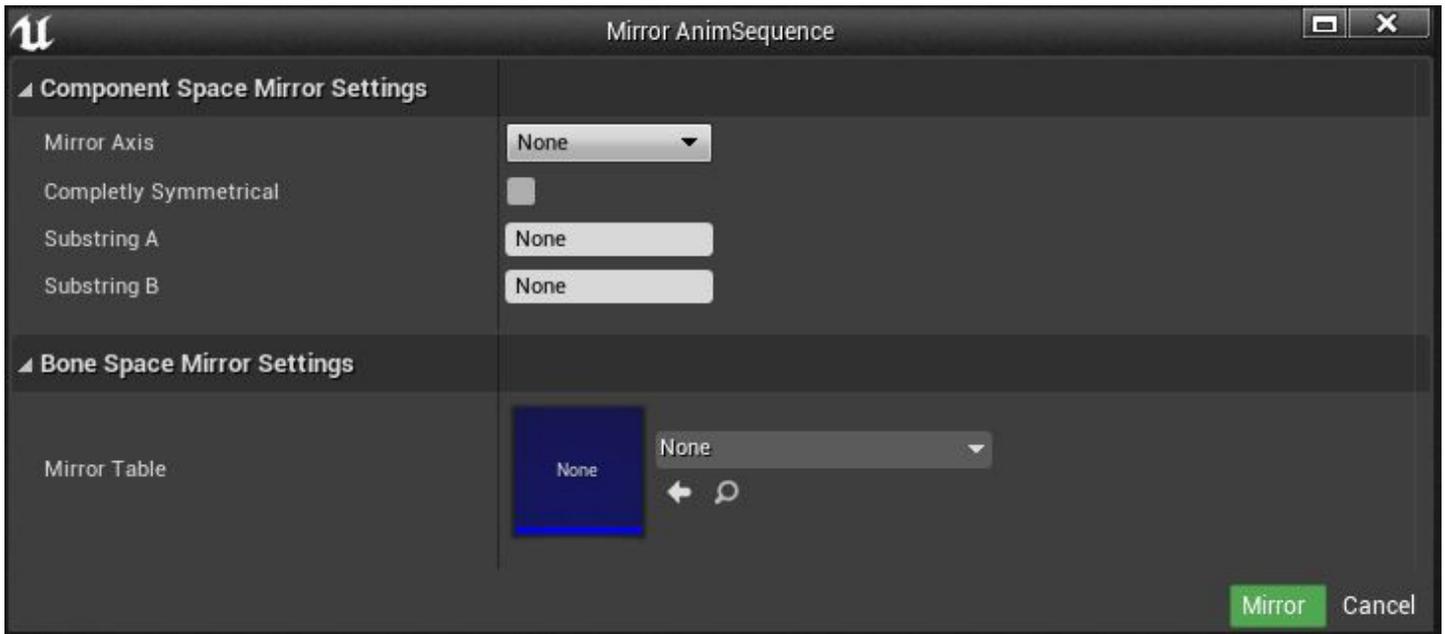
To use this tool simply right click an existing animation asset and select Mirror Animation Asset.



The following window will show up and you'll have 2 options.

If you wish to mirror the animation in component space, simply set the appropriate component space mirror settings, and leave the Mirror Table parameter as None.

If you wish to mirror the animation in Bone Space, set the Mirror Table parameter to the Mirror Table you've created and leave the other options as is.

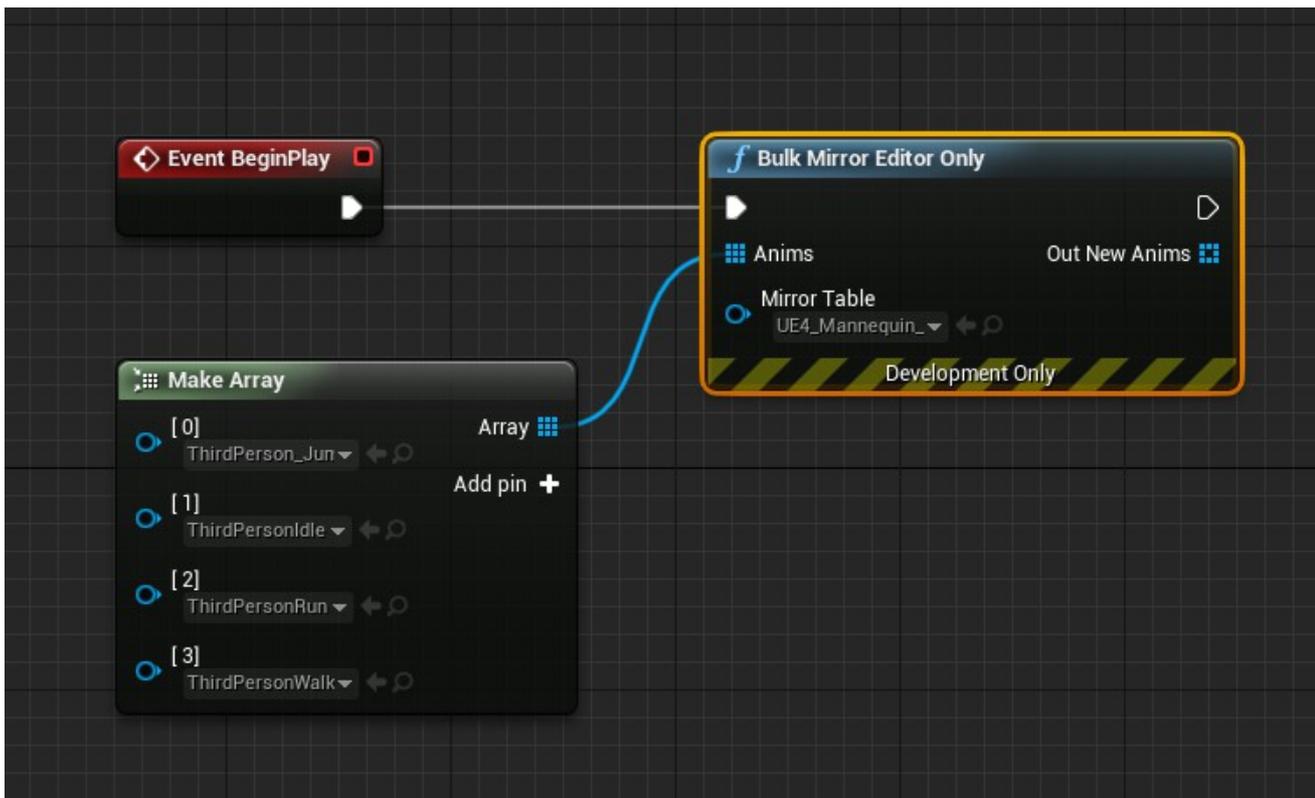
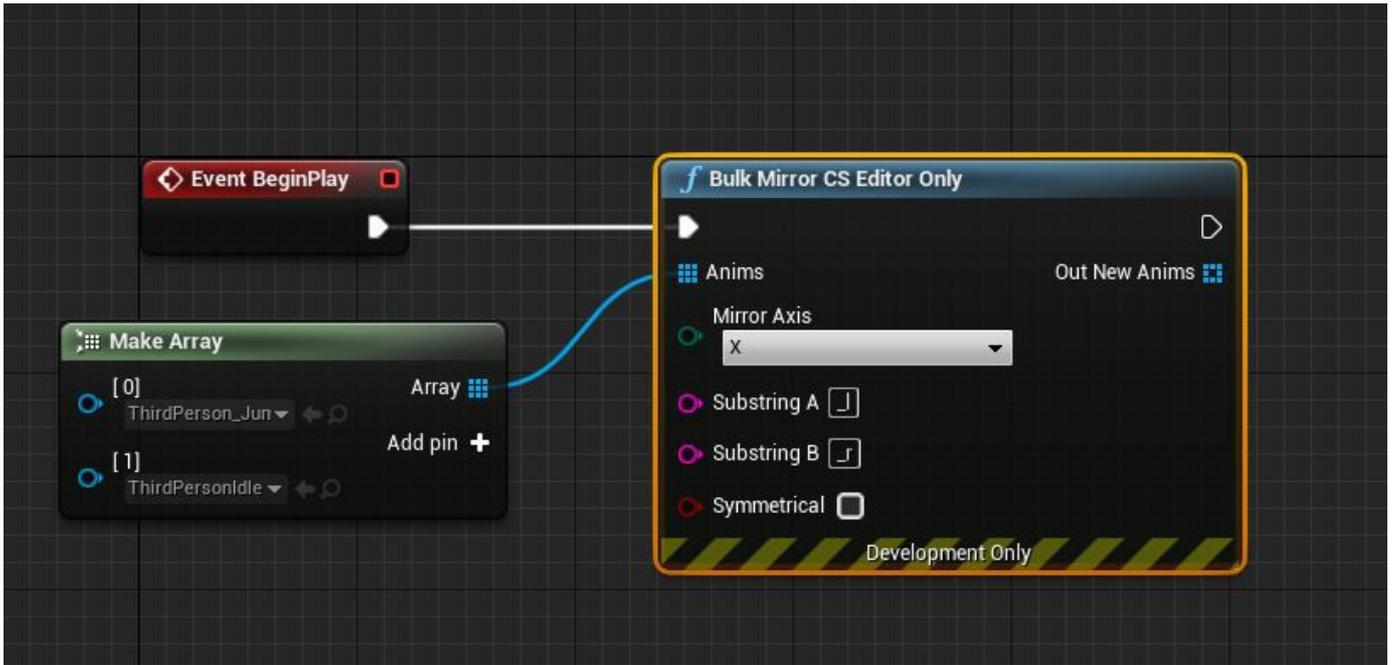


Disclaimer: Retargeted animations can fail to mirror depending on the skeleton type. In case you encounter this, a workaround is to simply mirror the source animation and then retarget that mirrored version.

In the case you don't have access to the original animation of the retargeted animation, you'll have to export the animation as preview mesh, this will create a new animation next to the exported animation inside the content browser that can now be mirrored using this tool.

Bulk Mirror Nodes:

In case you need to mirror a large number of animation assets in one go, you can simply use these nodes from blueprint.



Replication:

For replicating the functionality of the Mirror Pose nodes, you use the same method for regular Animation Blueprints, and just make sure the variables that the output pose depends on are replicated properly.

The Mirror Root Motion settings inside the *ExtCharacterMovementComponent* are also like the other movement variables (Max Walk Speed, Ground Friction, Max Step Height, etc.). Which means they are not replicated variables themselves and have to be set in both server and client.

Mirror Table Library:

You can find a collection of Mirror Tables already set up for generic skeleton templates like Epic's Skeleton or Mixamo's at the following Link:

<https://www.dropbox.com/sh/tyou1nwz9v2k6nm/AAB1-kIKbgFUNrCqzxsVuauBa?dl=0>

You can also submit your own character skeletons to make Mirror Tables for them by e-mailing me at: Hethger@Gmail.com.

Example Project:

For a better understanding of these features you can find the example project by clicking the following link:

[Mirror Example](#)

There you will see advanced uses of the Mirror Animation System's features.