Reality Connect User Guide

Version 1.2

Disclaimer

Vizrt provides this publication “as is” without warranty of any kind, either expressed or implied. This publication may contain technical inaccuracies or typographical errors. While every precaution has been taken in the preparation of this document to ensure that it contains accurate and up-to-date information, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained in this document.

Vizrt’s policy is one of continual development, so the content of this document is periodically subject to be modified without notice. These changes will be incorporated in new editions of the publication. Vizrt may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time. Vizrt may have patents or pending patent applications covering subject matters in this document. The furnishing of this document does not give you any license to these patents.

Antivirus Considerations

Vizrt advises customers to use an AV solution that allows for custom exclusions and granular performance tuning to prevent unnecessary interference with our products. If interference is encountered:

- **Real-Time Scanning:** Keep it enabled, but exclude any performance-sensitive operations involving Vizrt-specific folders, files, and processes. For example:
  - C:\Program Files\[Product Name]
  - C:\ProgramData\[Product Name]
  - Any custom directory where [Product Name] stores data, and any specific process related to [Product Name].
- **Risk Acknowledgment:** Excluding certain folders/processes may improve performance, but also create an attack vector.
- **Scan Scheduling:** Run full system scans during off-peak hours.
- **False Positives:** If behavior-based detection flags a false positive, mark that executable as a trusted application.

Technical Support

For technical support and the latest news of upgrades, documentation, and related products, visit the Vizrt web site at www.vizrt.com.

Created on

2026/03/23

Contents

1 Introduction

Reality Connect enhances the virtual studio experience with more realism using immersive real-time talent shadow and reflection capabilities.

1.1 Related Documents

- Viz Engine
- AI Terminal

For more information about all of the Vizrt products, visit:

- www.vizrt.com
- Vizrt Documentation Center
- Vizrt Training Center
- Vizrt Forum

1.2 Feedback and Suggestions

We encourage feedback on our products and documentation. Please contact your local Vizrt customer support team at www.vizrt.com.

1.3 System Requirements

1.3.1 General

OS	Windows 10 (64-bit)
	Windows 11
Browser	Google Chrome
	Firefox
	Microsoft Edge

1.3.2 Hardware

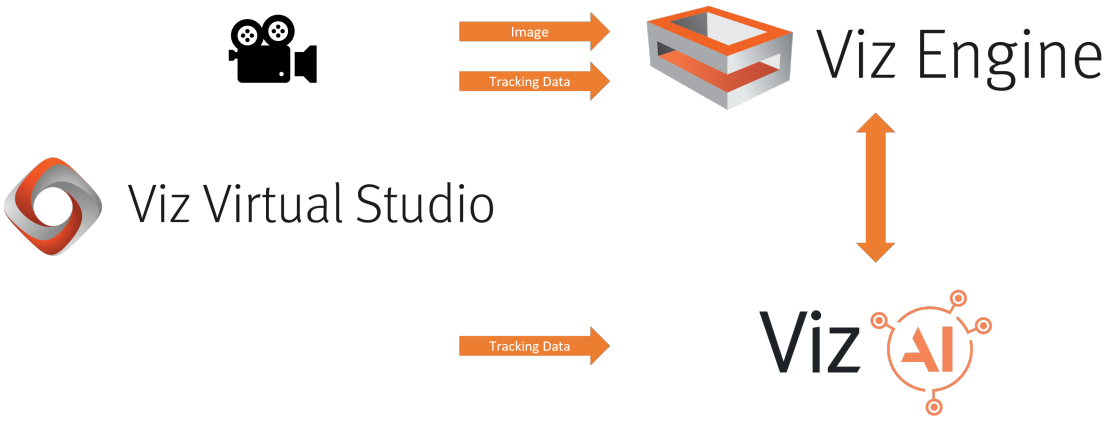
	Minimal	Recommended
Box	HP Z4 G5	HP Z8 G5
Graphics Card	NVIDIA RTX 5000 Ada	NVIDIA RTX 6000 Ada
		NVIDIA RTX 5000 Ada + NVIDIA RTX 5000 Ada

Note: The performance is strongly influenced by the complexity of the virtual studio scene. Reality Connect does offer different modes to optimize for best performance or best quality. A strong graphics card such as the NVIDIA RTX 6000 Ada is strongly recommended to achieve optimal quality. Alternatively, the load can be distributed among two graphic cards.

1.3.3 Software

- Viz Engine (5.4 or later)
- Viz Virtual Studio / Tracking Hub (1.8 or later)

1.4 System Overview



Reality Connect is a feature configured and controlled by the *Vizrt AI Terminal*. It extends the *Viz Virtual Studio* solution and integrates both with *Viz Engine* and *Viz Virtual Studio (Tracking Hub)* to track one or multiple talents using AI. Viz Engine can then be configured to use the talent tracking data to render effects such as talent shadows or reflections.

2 Installation

2.1 Branded Installer

1. Double click *Viz_AI_Tools-*.exe*.
2. Select **Requirements**, **AI Terminal**, and **Reality Connect**.
3. Click **INSTALL** and follow the instructions.

2.2 Preset Models

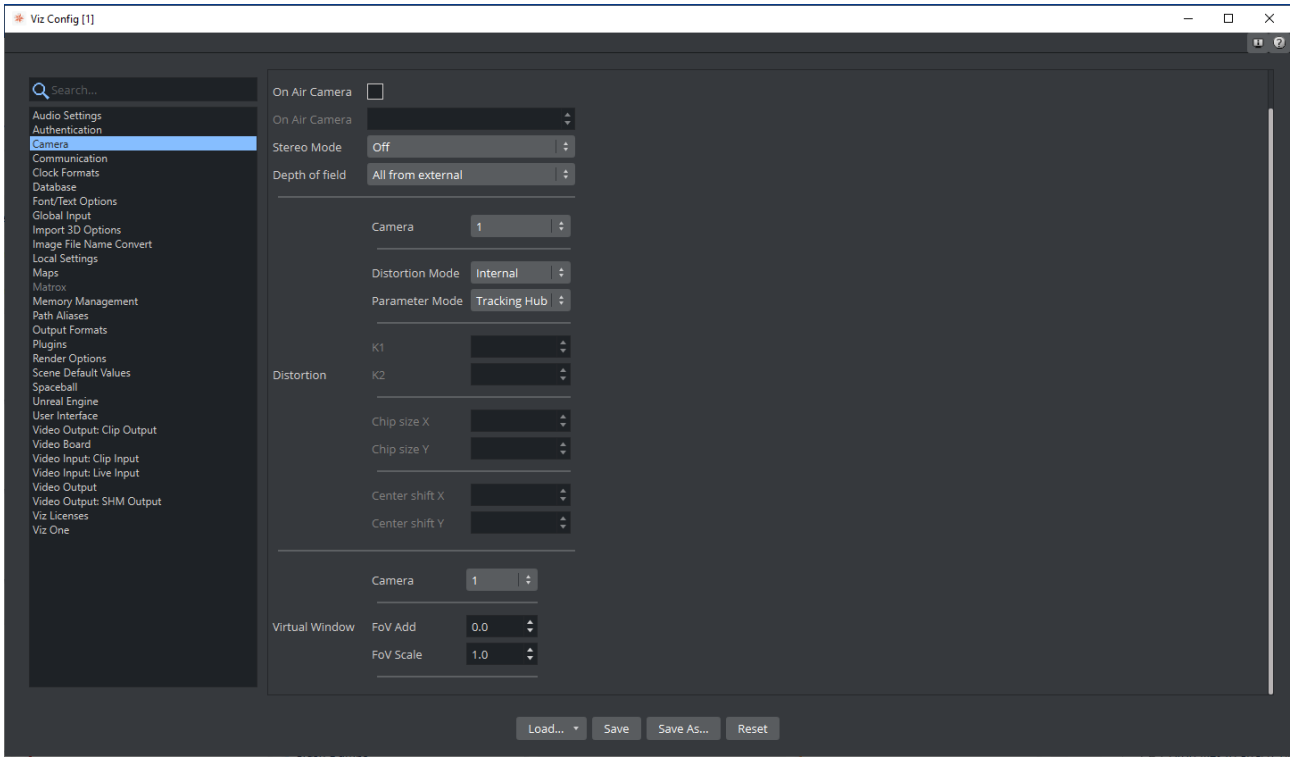
Download *RealityConnect_TalentModels.via* from the Vizrt FTP and import it into Graphic Hub.

3 Viz Engine Configuration

The Viz Engine configuration needs slight adaptions for Reality Connect. These are described below:

- Camera
- Communication
- Unreal Engine

3.1 Camera



3.1.1 Distortion Mode

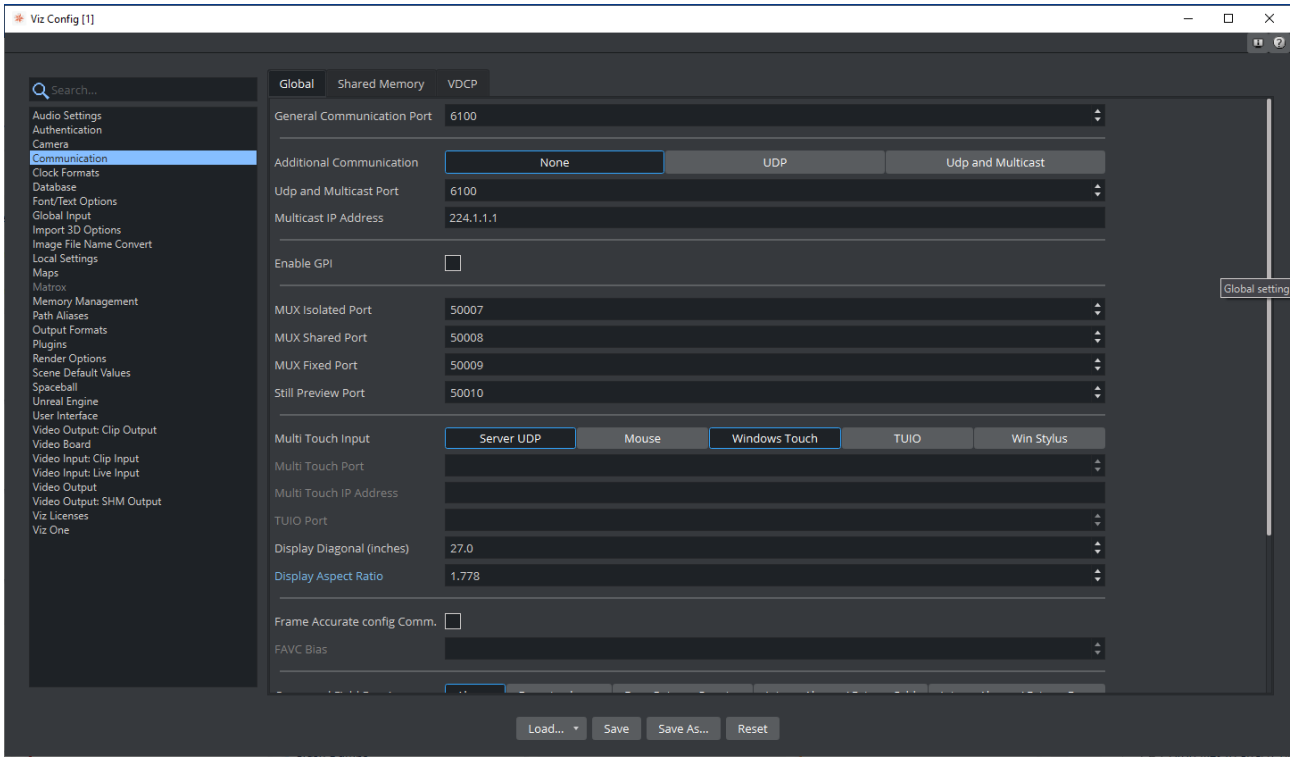
Set to *Internal*.

3.1.2 Parameter Mode

Set to *Tracking Hub*.

3.2 Communication

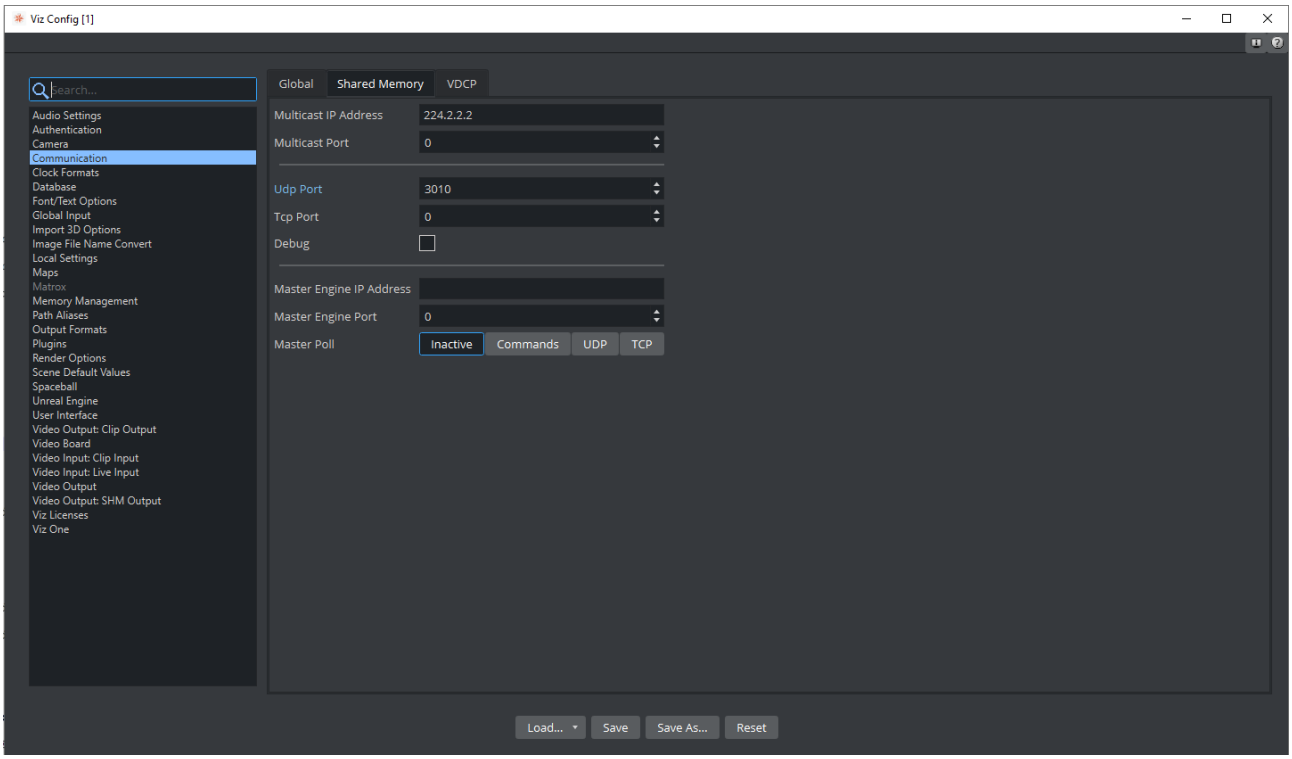
3.2.1 Global



General Communication Port

Set to an available port other than `0`. This port is required during the Tool Configuration.

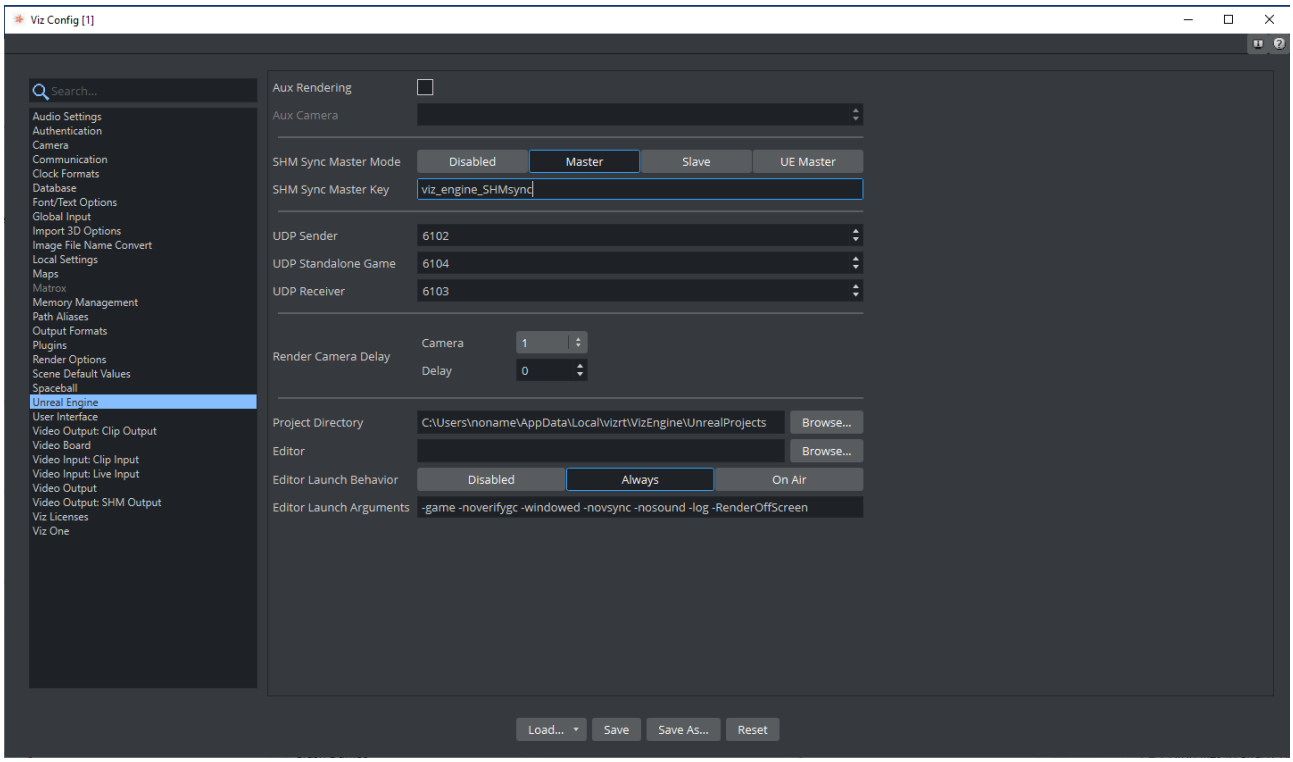
3.2.2 Shared Memory



UDP Port

Set to an available port other than `0`.

3.3 Unreal Engine



3.3.1 SHM Sync Master Mode

Set to *Master*.

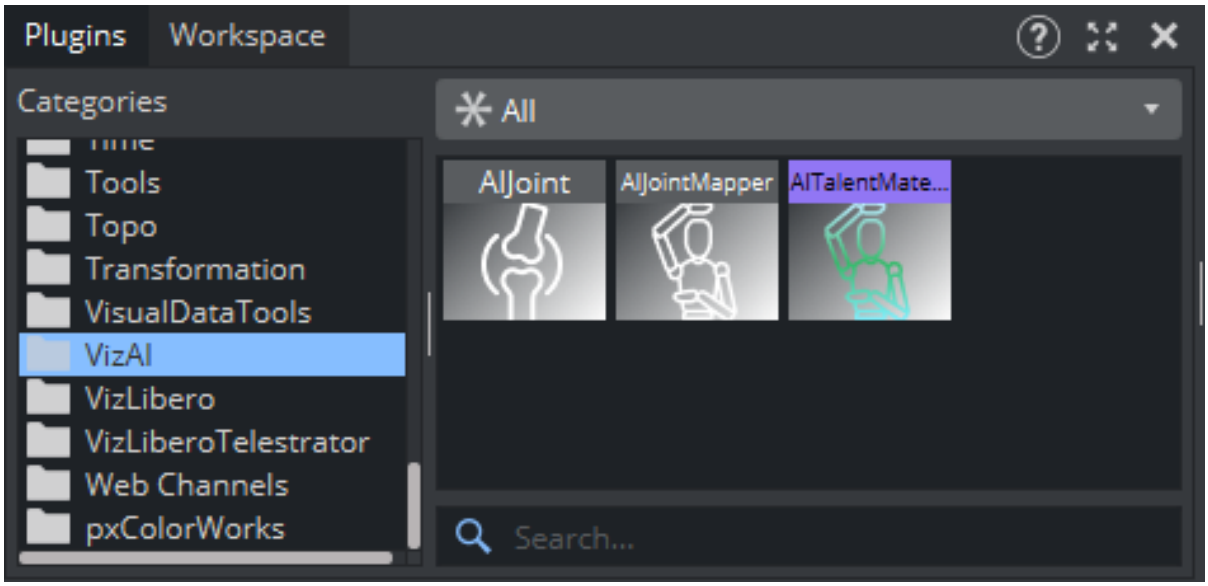
3.3.2 SHM Sync Master Key

Specify a non-empty name. This name is required during the Tool Configuration.

4 Viz Engine Scene Design

4.1 Viz AI Plug-ins

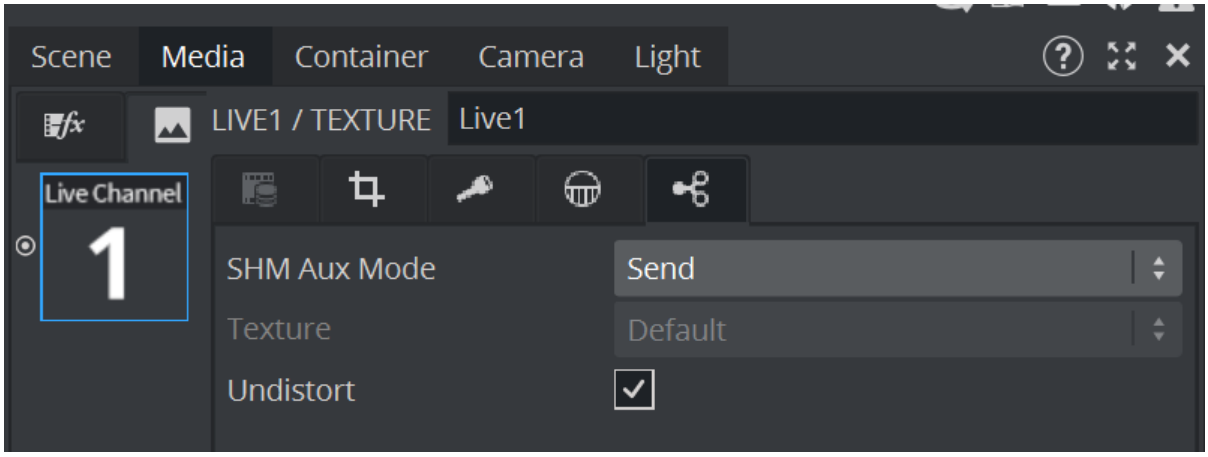
This section requires the Viz AI plug-ins to be installed on the system. Refer to the *AI Terminal User Guide* for installation instructions.



- **AI Joint:** Specifies and controls a skeleton joint.
- **AI Joint Mapper:** Specifies and controls the root container of a talent and maps it with Reality Connect.
- **AI Talent Material:** Use the input video as texture for the talent models to enable reflection effects.

4.2 Texture Sharing

Share the video input with Reality Connect with these settings in the **Media** tab.



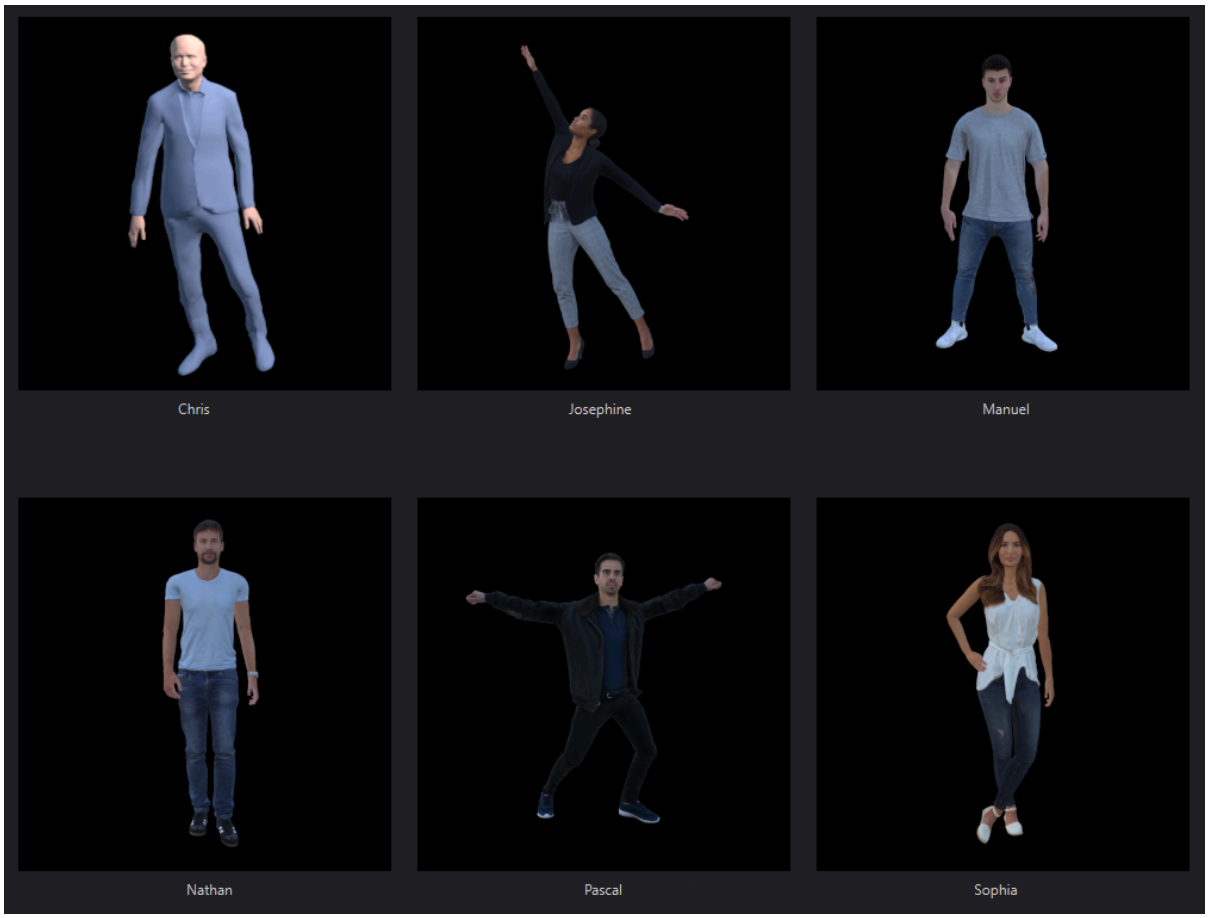
- Set **SHM Aux Mode** to *Send*.
- Check the **Undistort** box.

4.3 Preset Models

Preset models are rigged models that have already been enhanced and configured with the Reality Connect joint plug-ins, **AI Joint**, **AI Joint Mapper** and **AI Talent Material**. The preset models available with this release are shown in the image below.

> **Note:** The preset models are available from the Vizrt FTP.

Read through the Model Adjustments and Create a New Model sections to learn how to configure your own models for Reality Connect.

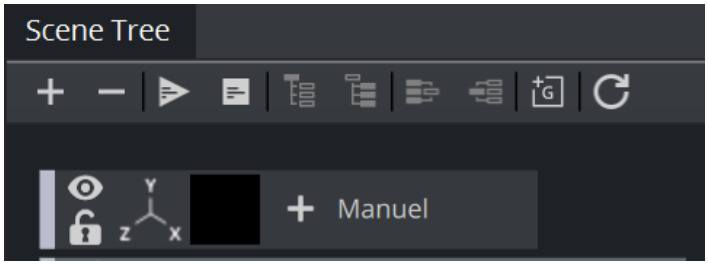


The following table lists the origin of the preset models before porting them to the Viz Engine and applying the steps listed in Model Adjustments.

Name	Link	Author	License
Chris		Vizrt	

Name	Link	Author	License
Nathan	https://sketchfab.com/3d-models/nathan-animated-003-walking-3d-man-143a2b1ea5eb4385ae90a73657aca3bc	https://renderpeople.com/	CC Attribution
Sophia	https://sketchfab.com/3d-models/sophia-animated-003-animated-3d-woman-dc448c3be0e74f96a55fb475a13433cf	https://renderpeople.com/	CC Attribution
Josephine	https://sketchfab.com/3d-models/carla-rigged-001-rigged-3d-business-women-acf520f450d14dd799f98a6fede3edf5	https://renderpeople.com/	CC Attribution
Manuel	https://sketchfab.com/3d-models/manuel-animated-001-3d-dancing-man-e65e0fef4e0743868c8d5bff36d61116	https://renderpeople.com/	CC Attribution
Pascal	https://sketchfab.com/3d-models/facial-body-animated-party-m-0001-actorcore-aecb1b0c293a4185a91a532e635f3e6d	https://www.reallusion.com/	CC Attribution

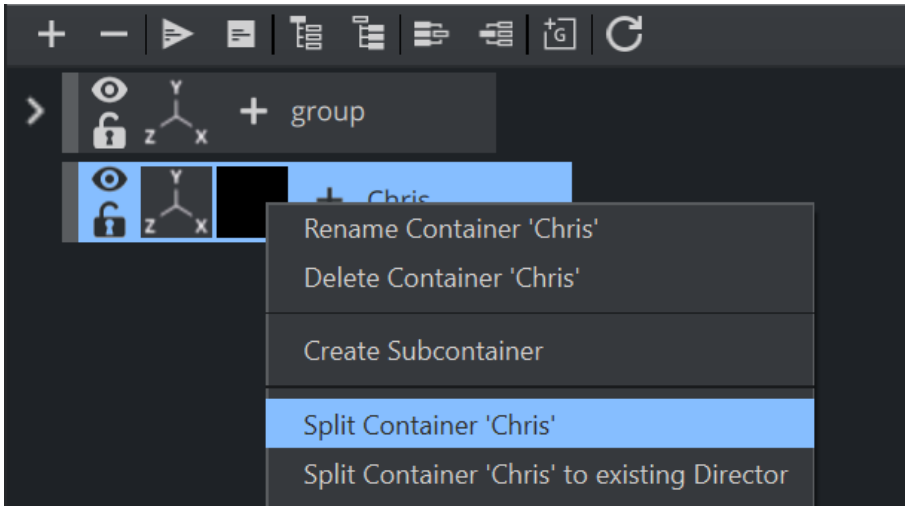
The preset models are provided as Geometries and can be added to a scene by a drag and drop operation.



Make sure that the position and the rotation of the new container are set to *(0,0,0)*. At this point, the imported geometry is automatically animated from a real talent detected by a running Reality Connect application.

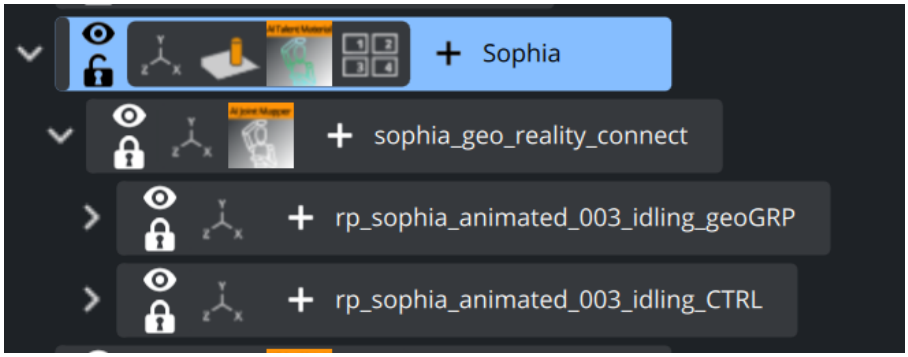
4.4 Model Configuration

Split (right-click) the container to obtain the full tree including all subcontainers.

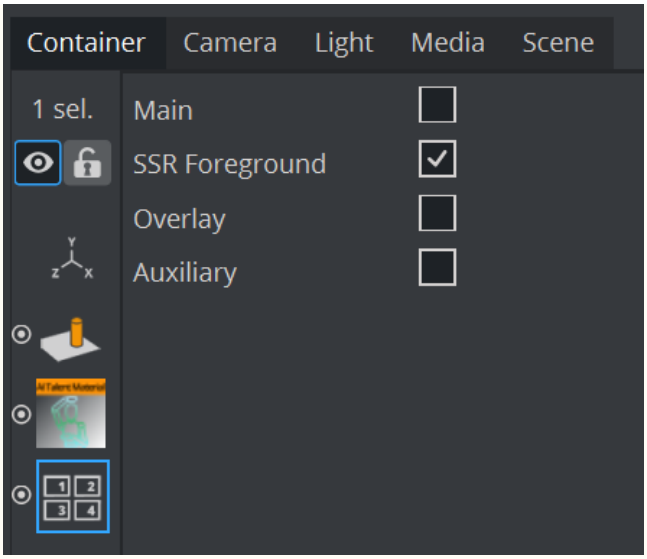


This creates access to a *RootNode* container with the following plug-ins:

- Expert.
- Shadow Caster.
- AI Joint Mapper.
- AI Talent Material.
- Layers.

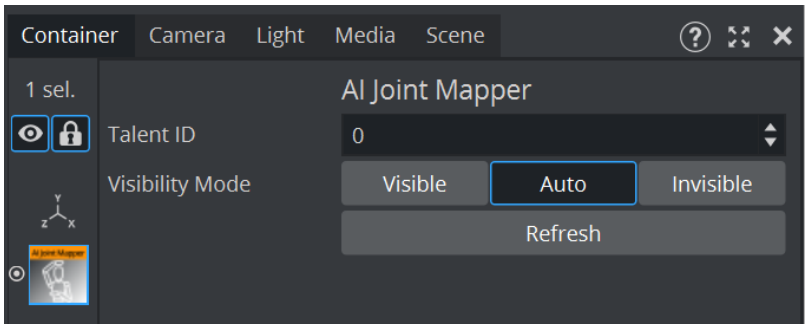


Note: For improved rendering quality, switch to **SSR Foreground** in the Layers plug-in.



4.4.1 Talent Mapping

The **AI Joint Mapper** plug-in comes with the following options:



- **Talent ID:** If multiple talents are tracked, each model needs a different Talent ID (0 - 7).
- **Visibility Mode:** During a production this should always be set to *Auto*.
  - **Visible:** The model is always visible (also when Reality Connect is not running).
  - **Auto:** Reality Connect automatically sets the visibility of the model when a talent appears or leaves the scene.
  - **Invisible:** The model is always invisible.
- **Refresh:** Recalculates the internal tree structure. This is required on manual change or creation of a talent model.

⚠ **Note:** Each talent detected by Reality Connect keeps an internal Talent ID between 0 and 7. This ID is lost when the talent leaves the scene and there is no guarantee the same ID is reapplied to the same talent after the talent re-enters the scene.

4.4.2 Shadow Configuration

Enable/disable the **Shadow Caster** plug-in to turn on/off shadows projected from the talent.

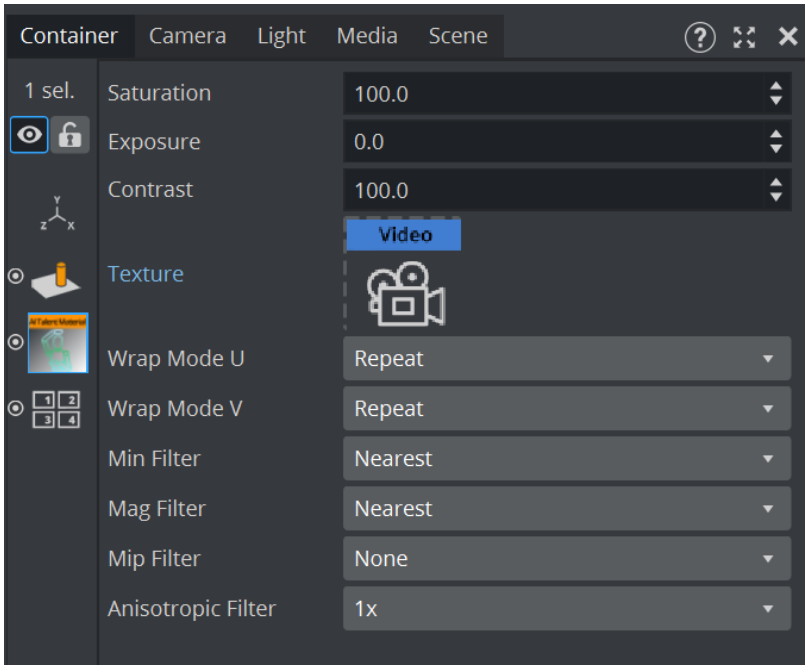
> **Note:** This requires the **Shadow Receiver** plug-in to be enabled on surfaces in the virtual scene where shadows are to be projected.

4.4.3 Reflection Configuration

Enable/disable the **AI Talent Material** plug-in to turn on/off reflections from the talent.

> **Note:** This requires the materials in the virtual studio to be configured as reflective.

Use the following options of the **AI Talent Material** plug-in to configure the reflection effect.



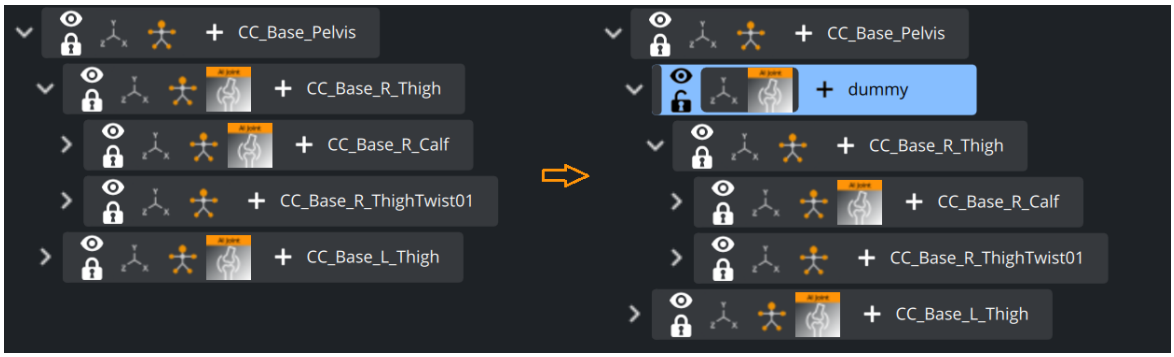
- **Saturation:** Controls the reflection's color saturation.
- **Exposure:** Controls the reflection's color exposure.
- **Contrast:** Controls the reflection's color contrast.
- **Texture:** Specify the video source (for example, *LIVE1*) to texture the talent with. This should be set to the same video source configured as input for Reality Connect.

4.5 Model Adjustments

4.5.1 Bone Offset

For advanced use cases, models can be customized to better fit the real talent. Expand the model container to visualize the full tree.

The bones controlled by Reality Connect have the **AI Joint** plug-in attached to it, and the bones that animate the geometry model have the **Bone&Skin** plug-in attached to it. If the two plug-ins are on the same bone, the bone is directly controlled by Reality Connect. Separate the two to introduce a position or rotation offset.



To add an offset to a bone:

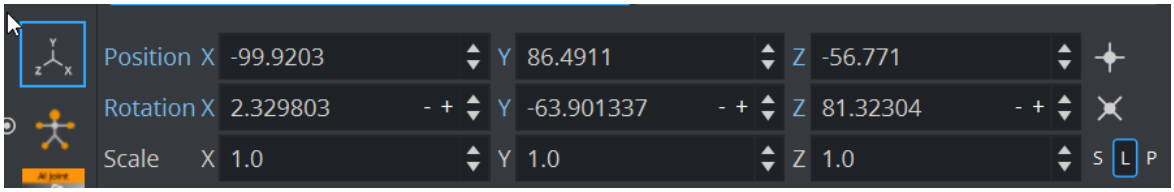
- Create a new parent container.
- Move the **AI Joint** plug-in to the parent container.
- Click the **Refresh** button of the **AI Joint Mapper** plug-in.
- Set the desired position and rotation offset in the child container.

4.5.2 Bone Scaling

To fit the size of a model better to a specific talent, the **Scale** parameter of containers holding an AI Joint plug-in can be used to adjust individual bones.

> **Note:** Scaling a certain bone scales all of its children equally (for example scaling the left hip scales the entire left leg accordingly while scaling the pelvis scales the entire model).

The Reality Connect only supports the Scaling Modes **Locked (L)** and **Proportional (P)**.



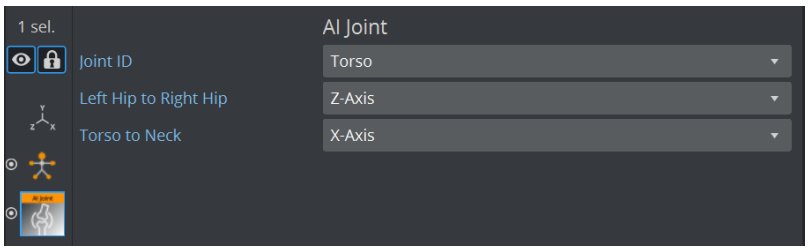
4.6 Create a New Model

This section contains additional information for advanced users that create their own models.

4.6.1 Talent Mapping

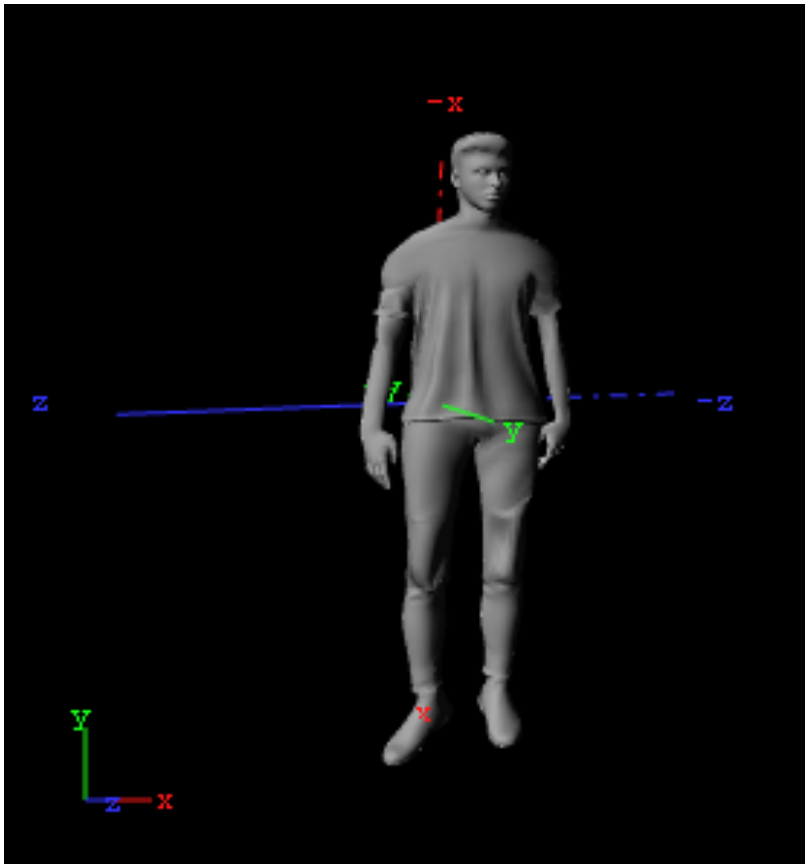
Add the **AI Joint Mapper** plug-in to the root container. Add the **AI Joint** plug-in to each bone container that should be animated. Remember to click the **Refresh** button of the **AI Joint Mapper** plug-in to apply changes of the tree structure or **AI Joint** plug-ins.

4.6.2 Joint Mapping



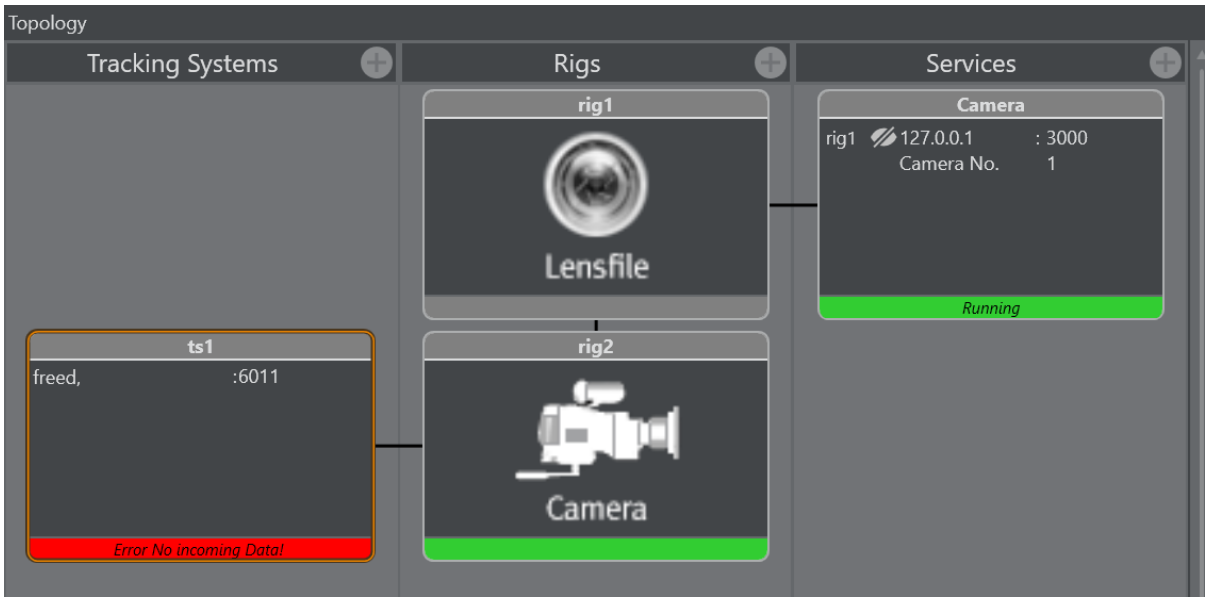
- **Joint ID:** The joint the container / bone is referring to.
- **Axes:** The Joint axes used to calculate the rotation of the joint.

Note: To identify the correct axes, check the local coordinate system of the container and set the axes for the specified directions accordingly. For the right hip joint in the talent below the the z-axis points into **Left Hip to Right Hip** direction and the x-axis into **Right Hip to Knee** direction.



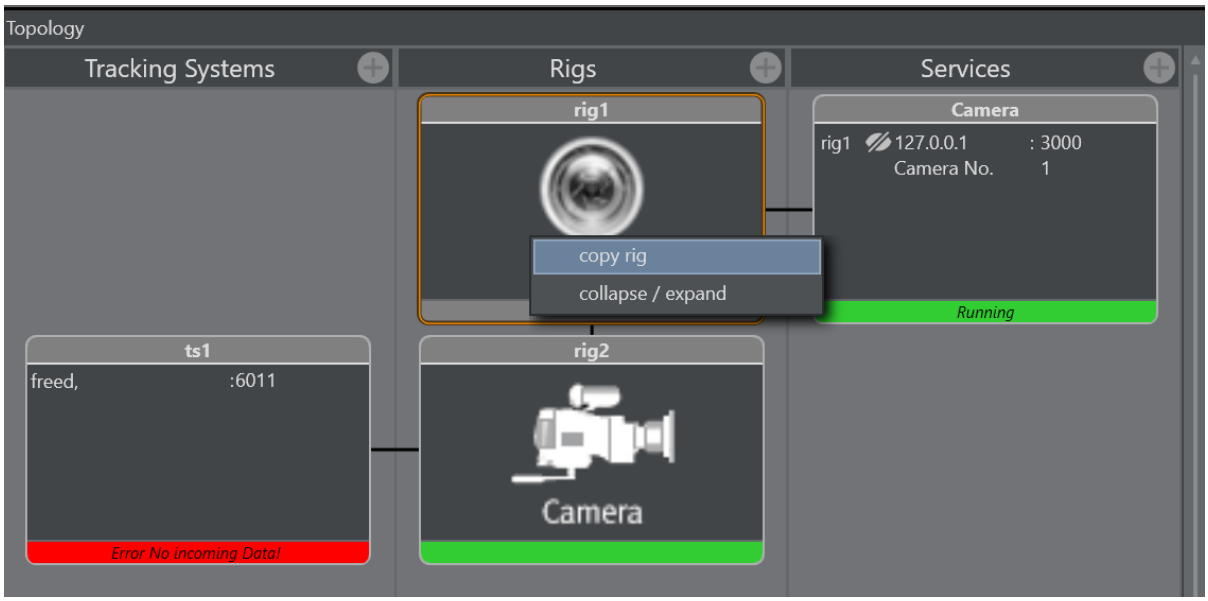
5 Tracking Hub Configuration

Reality Connect requires camera data to function properly. This implies a few changes to the regular Tracking Hub configuration. For this guide, let us assume the following configuration.



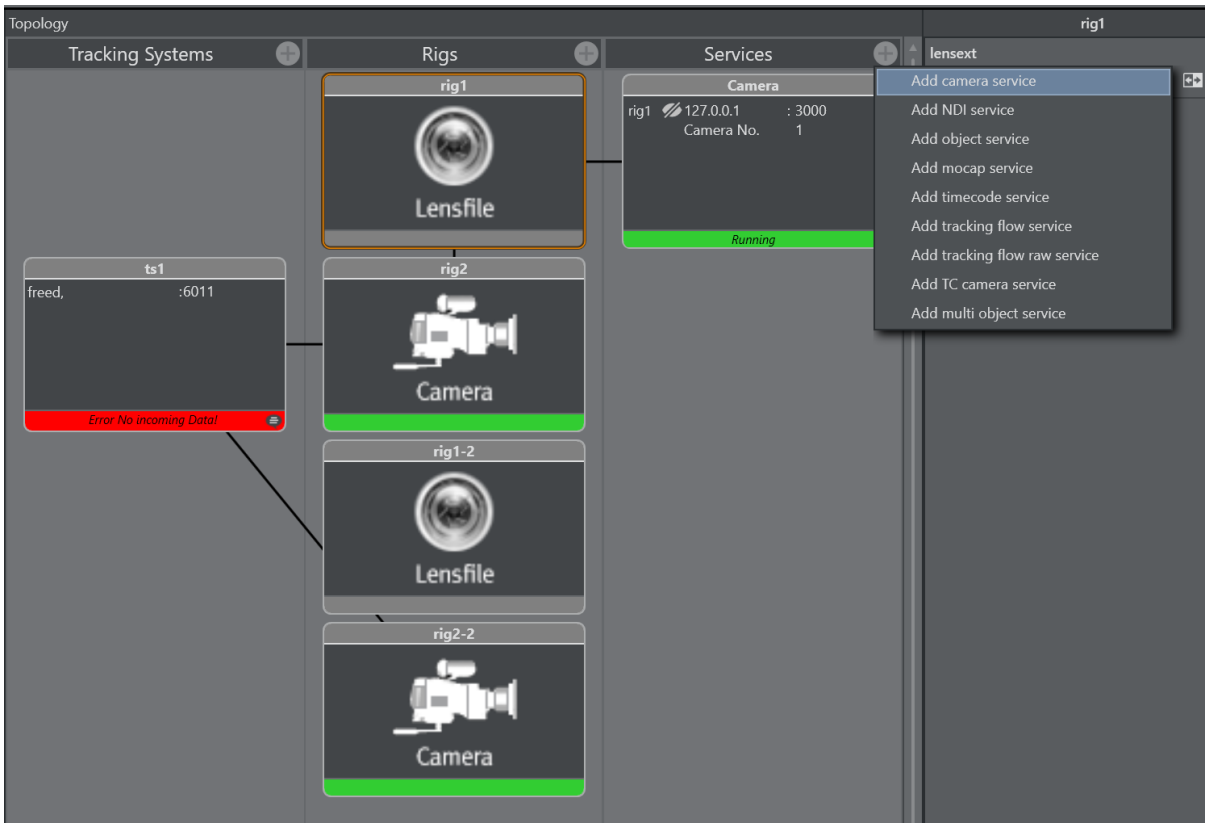
5.1 Copy Rig

Copy the last rig in the chain, connected to the camera service, which is usually the *Lensfile*.

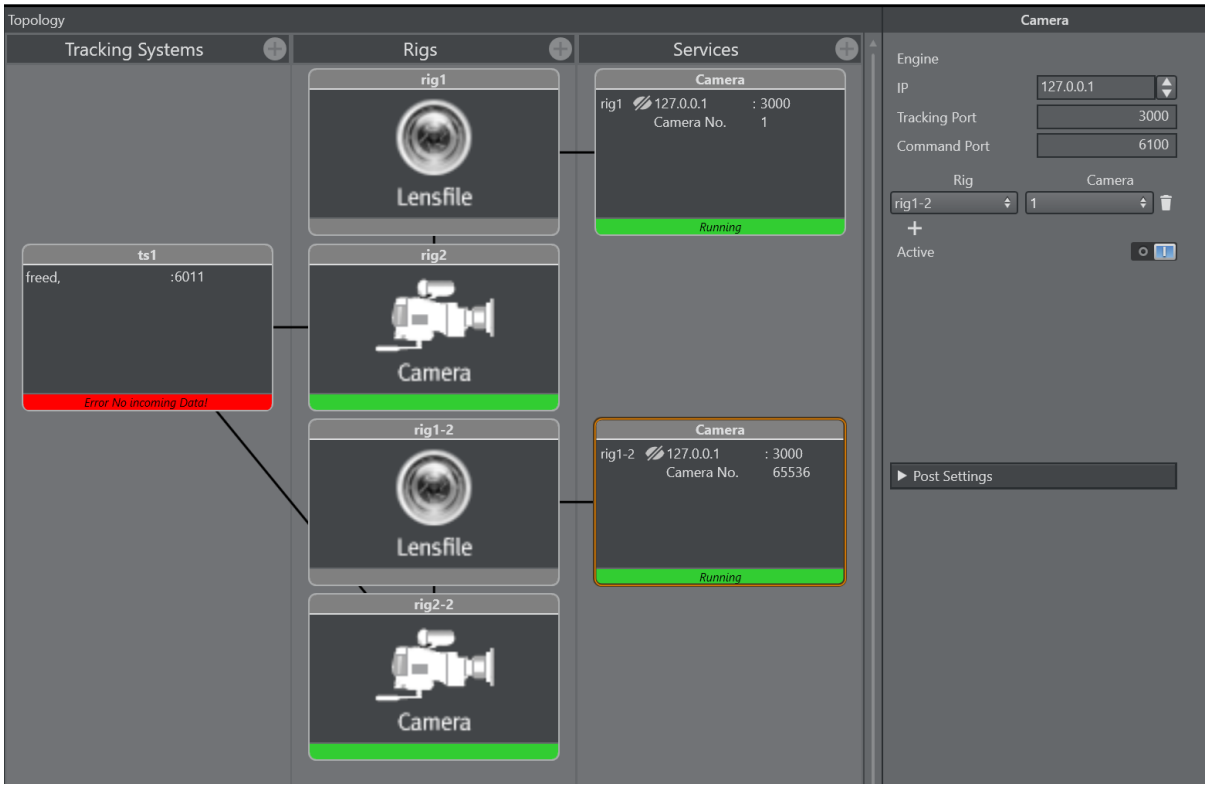


5.2 Add Camera Service

Add a second camera service.



Connect the second camera service with the previously created copy of the rig.

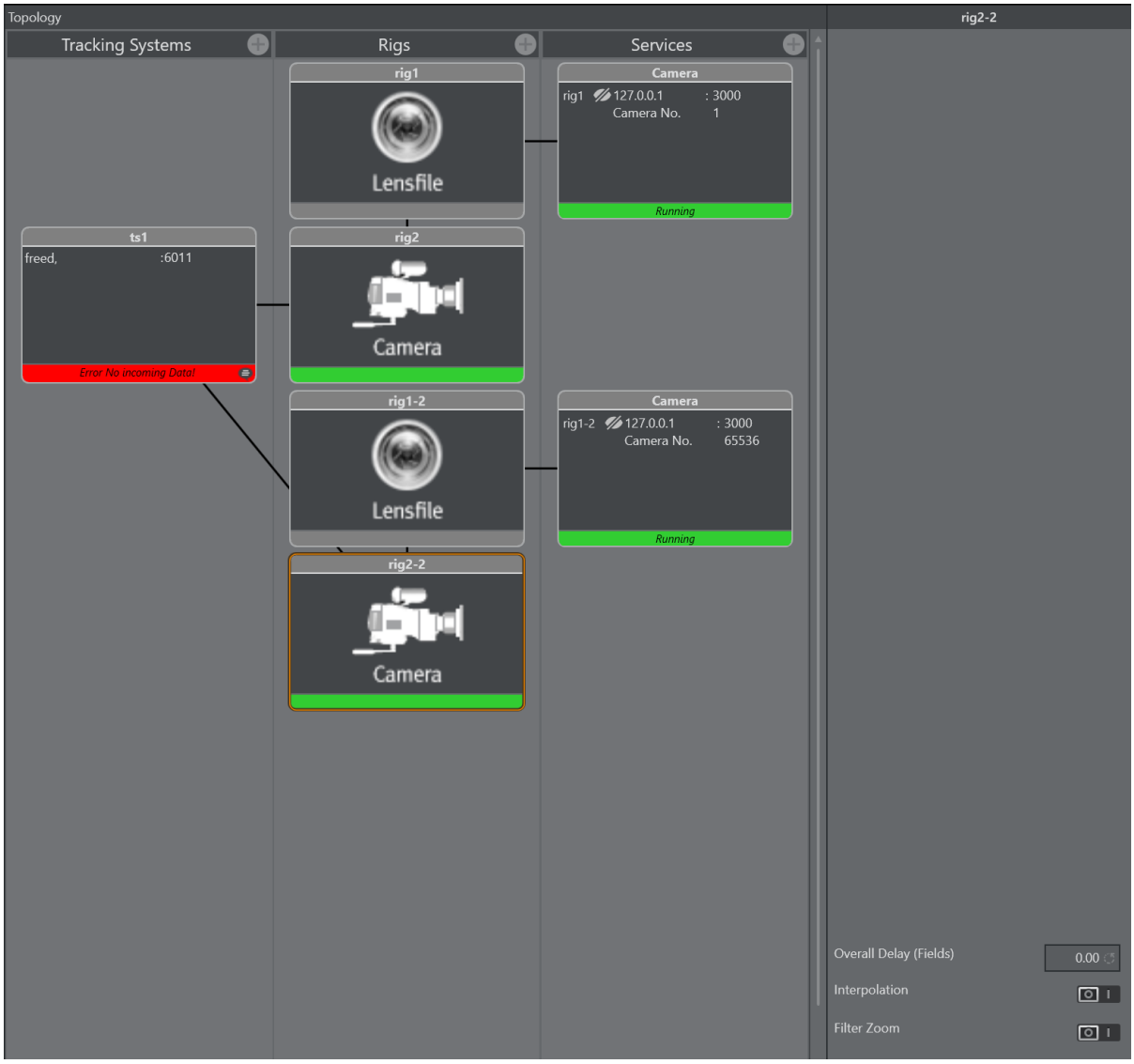


Specify the following values:

- **IP:** IP of the machine where Reality Connect is running.
- **Tracking Port:** Choose a free port other than for the first camera service. Reuse this port for the Tool Configuration.
- **Command Port:** Same value as for the first camera service.

5.3 Delay Configuration

Both camera rigs contain a field *Overall Delay (Fields)*. This allows to specify different delays for sending the camera tracking data to Viz Engine and Reality Connect. Follow the instructions in Delay Configuration to configure the optimal delays.



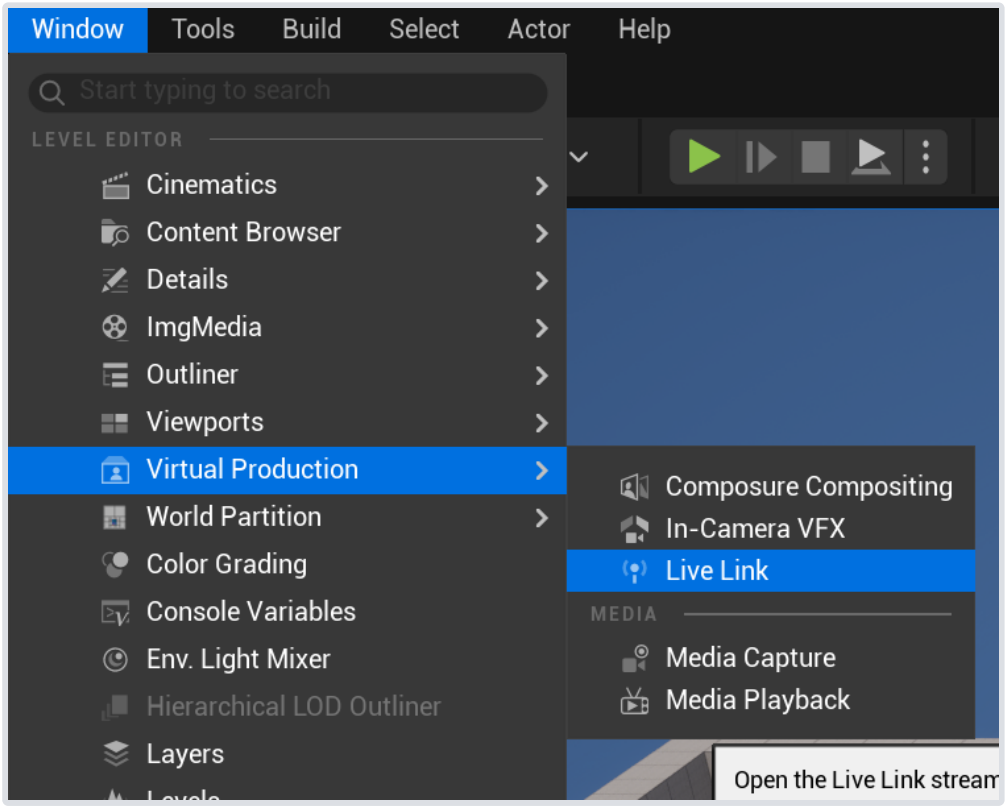
6 Unreal Engine Configuration

Reality Connect data can be used directly in Unreal Engine. This part of the guide assumes a working Reality Connect setup with Viz Engine already. For the guide on that please follow this page: Reality Connect User Guide.

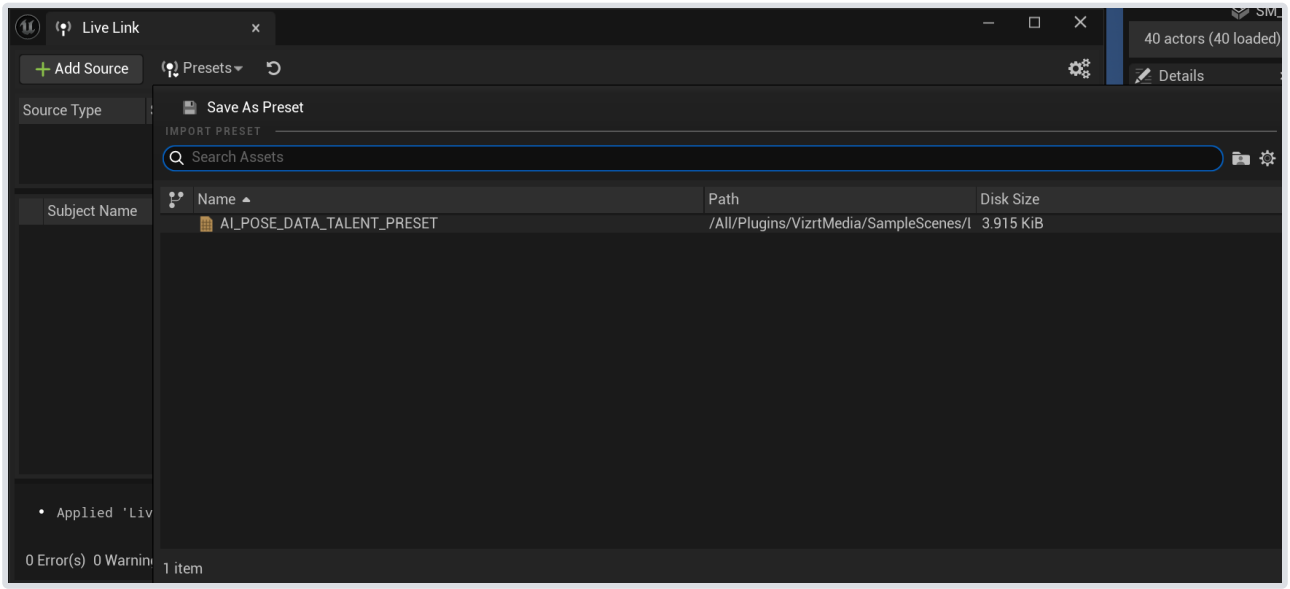
6.1 To Setup Reality Connect in Unreal Engine

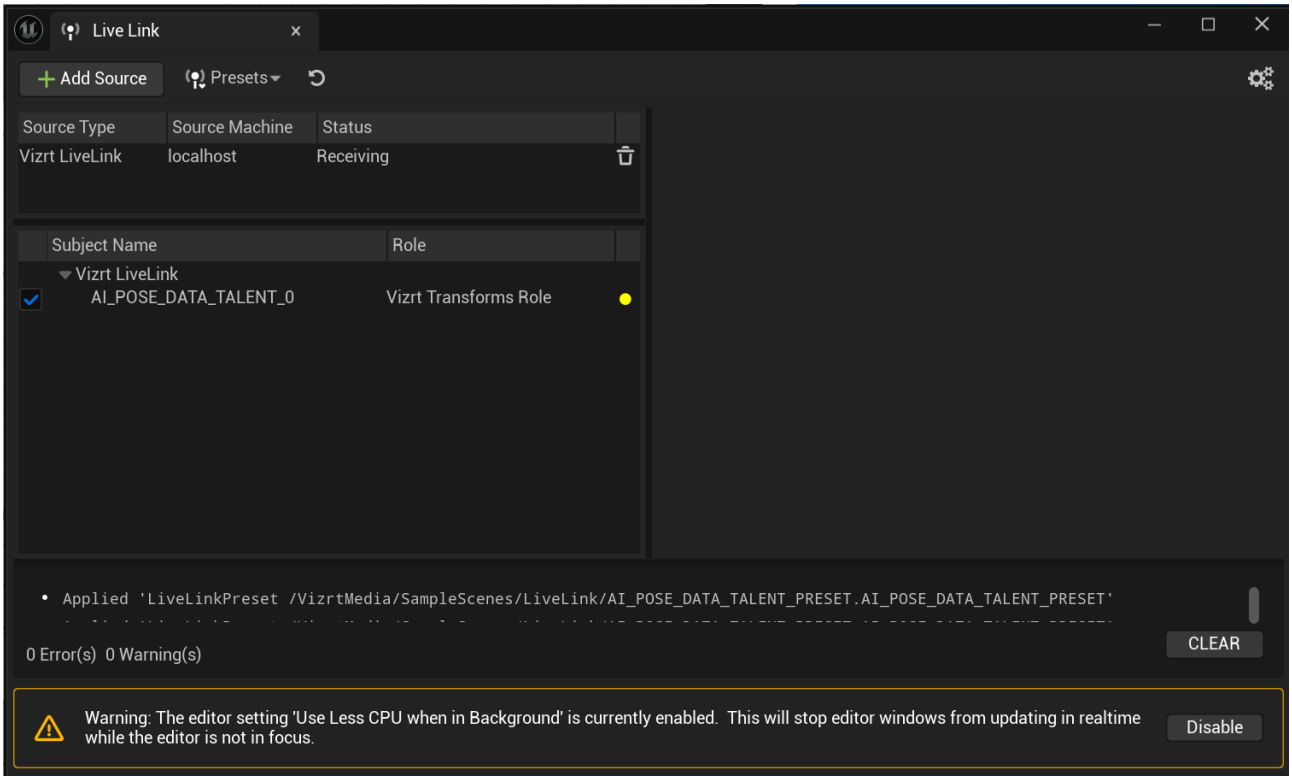
Reality Connect sends 34 transforms to the Unreal Engine that receives that data via LiveLink. Those transforms can then be assigned to bones on a skeleton to match the pose. There are two example scenes in the SampleScenes folder that use the base Unreal Engine skeleton and character (SKM_Manny - has to be added manually and belongs to the Third Person pack from base Unreal Engine or can be downloaded here - *X*) that can be used as a reference for other characters - if they use the same skeleton, the pose will match as well - for different skeletons, adjustments will have to be made.

To receive the data, go to the top menu and select **Window > Virtual Production > LiveLink**.

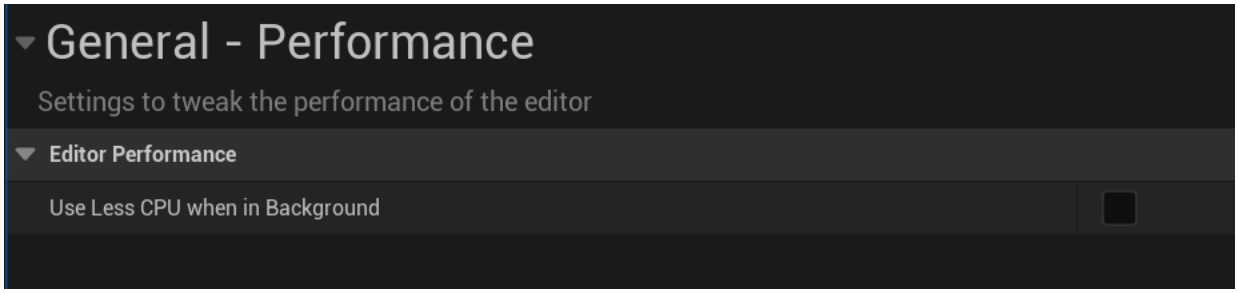


From there, go to the Presets and select the already available VIZRT preset (**AI_POSE_DATA_TALENT_PRESET**).

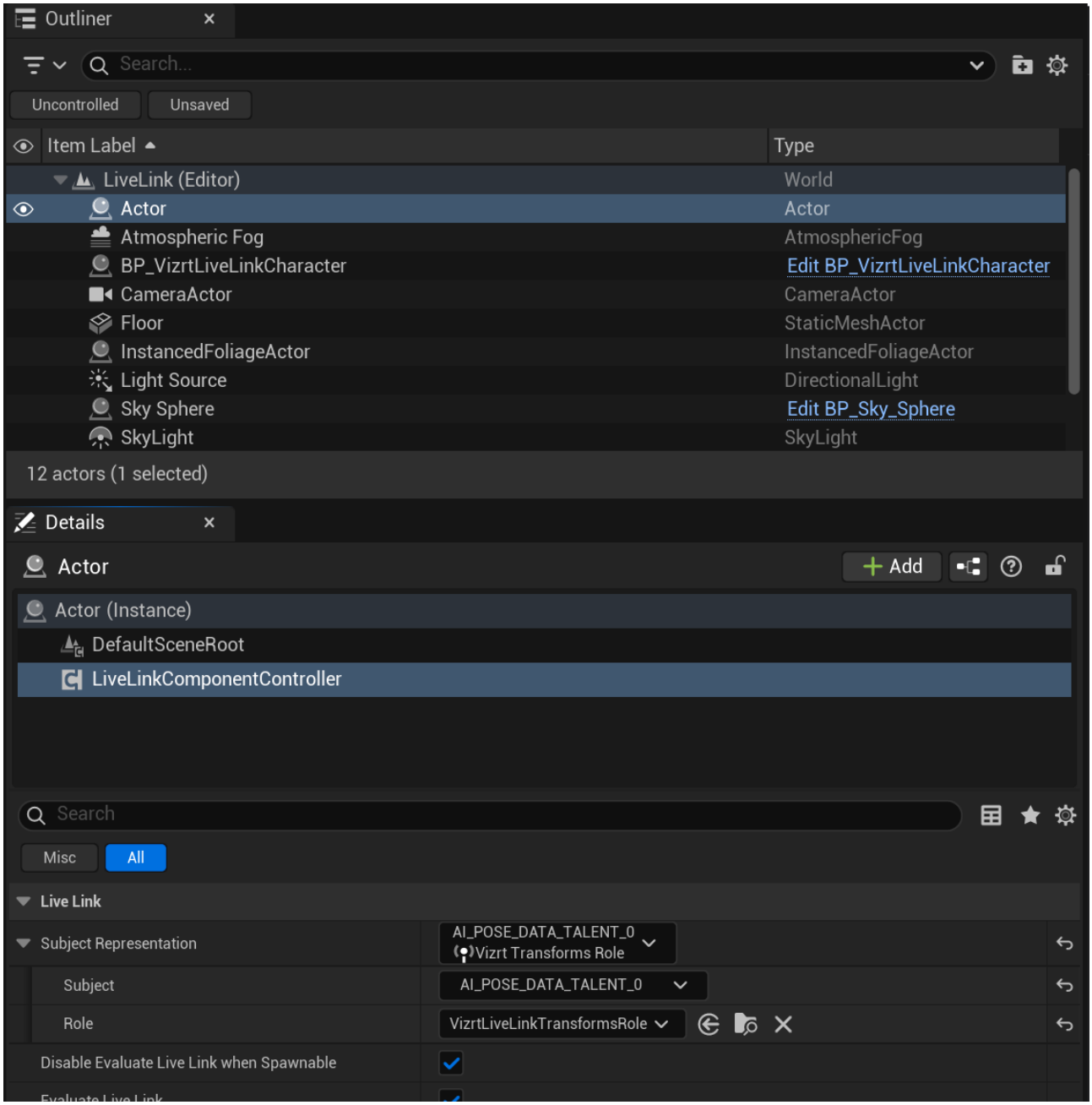


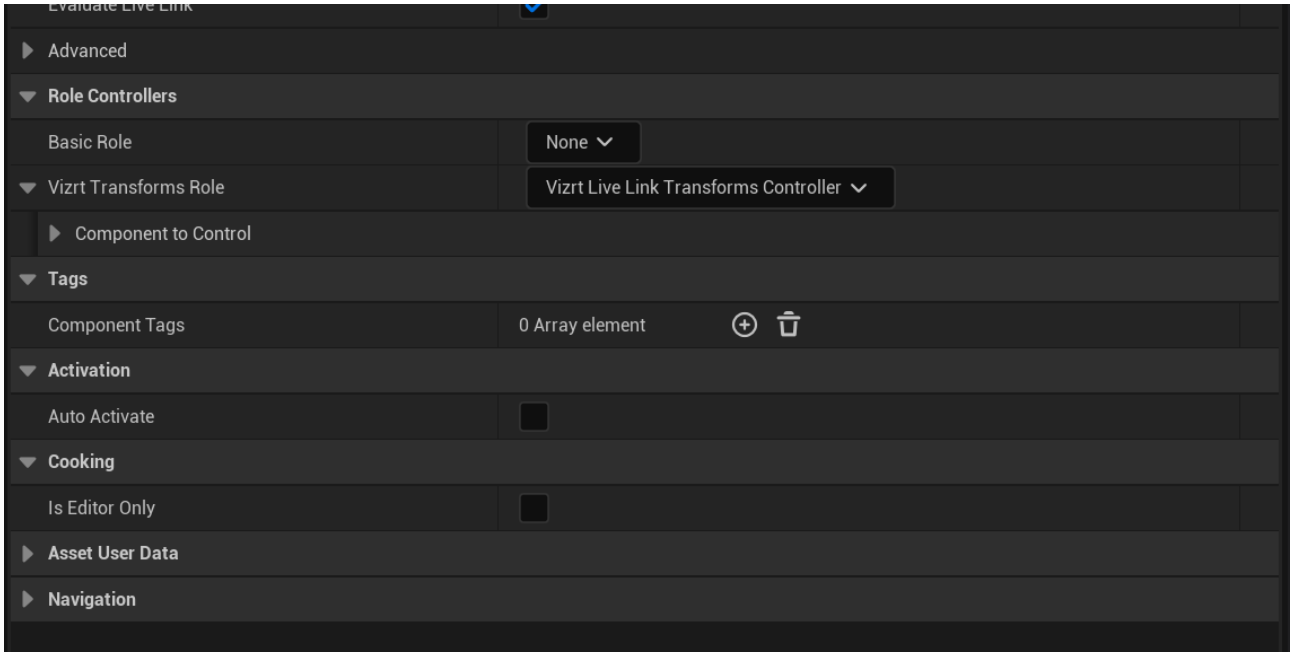


If the **Use Less CPU when in Background** setting is enabled, make sure to disable it in the **Disable** button or in **Editor Preferences > General > Performance**.

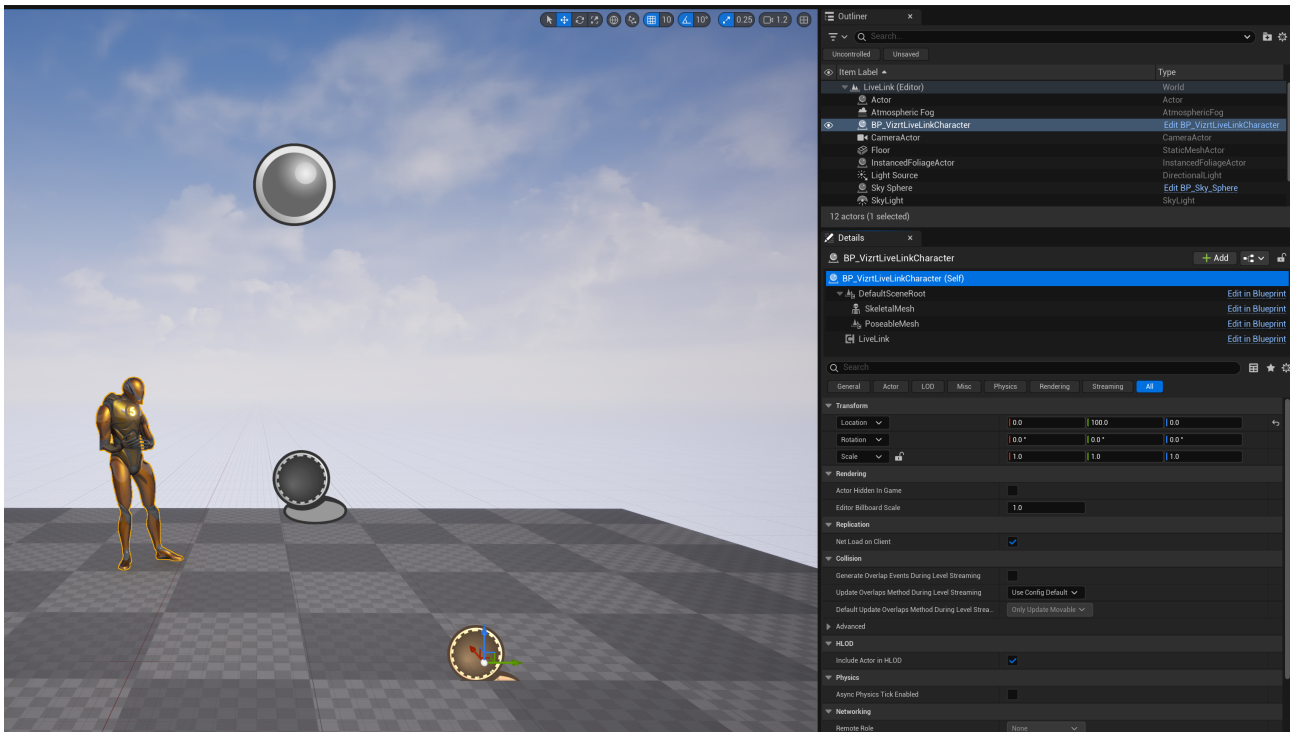


Inside your level, you need to have an **Actor** with a **LiveLinkComponentController** component. Inside the component, The **Subject Representation** needs to be set to the same preset as what exists in the LiveLink window and the **Vizrt Transforms Role** should be set to **Vizrt Live Link Transforms Controller**.



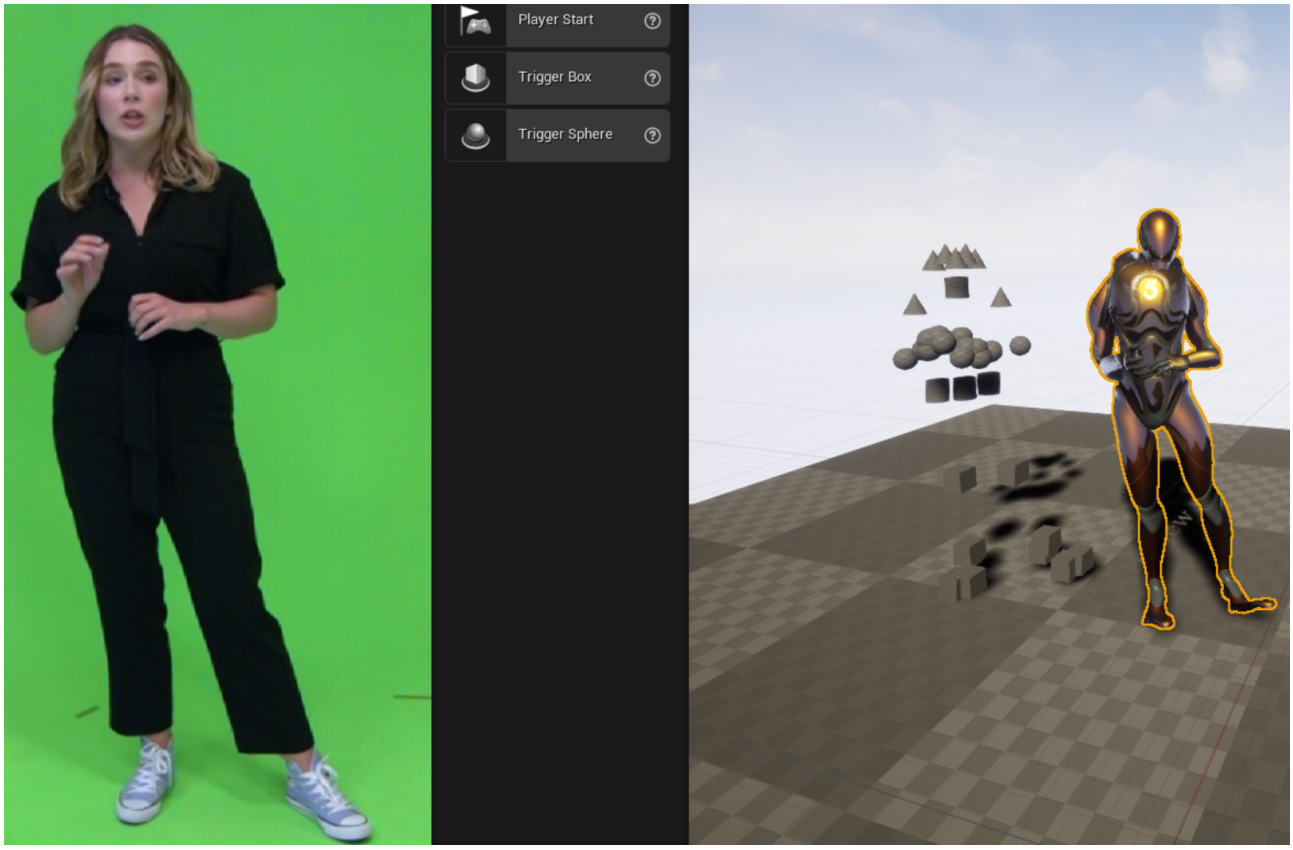


There also needs to be a Character with a **LiveLink** component attached (in our example case, that is the **BP_VizrtLiveLinkCharacter** present on the plugin).



When pressing play, this should be enough to receive the data, check if there is a green circle where the yellow circle was no warnings on the LiveLink window.

If the data is received correctly, the character should have a similar pose to the reference video or live feed (incoherence in position normally means the wrong transform is assigned to the bone):



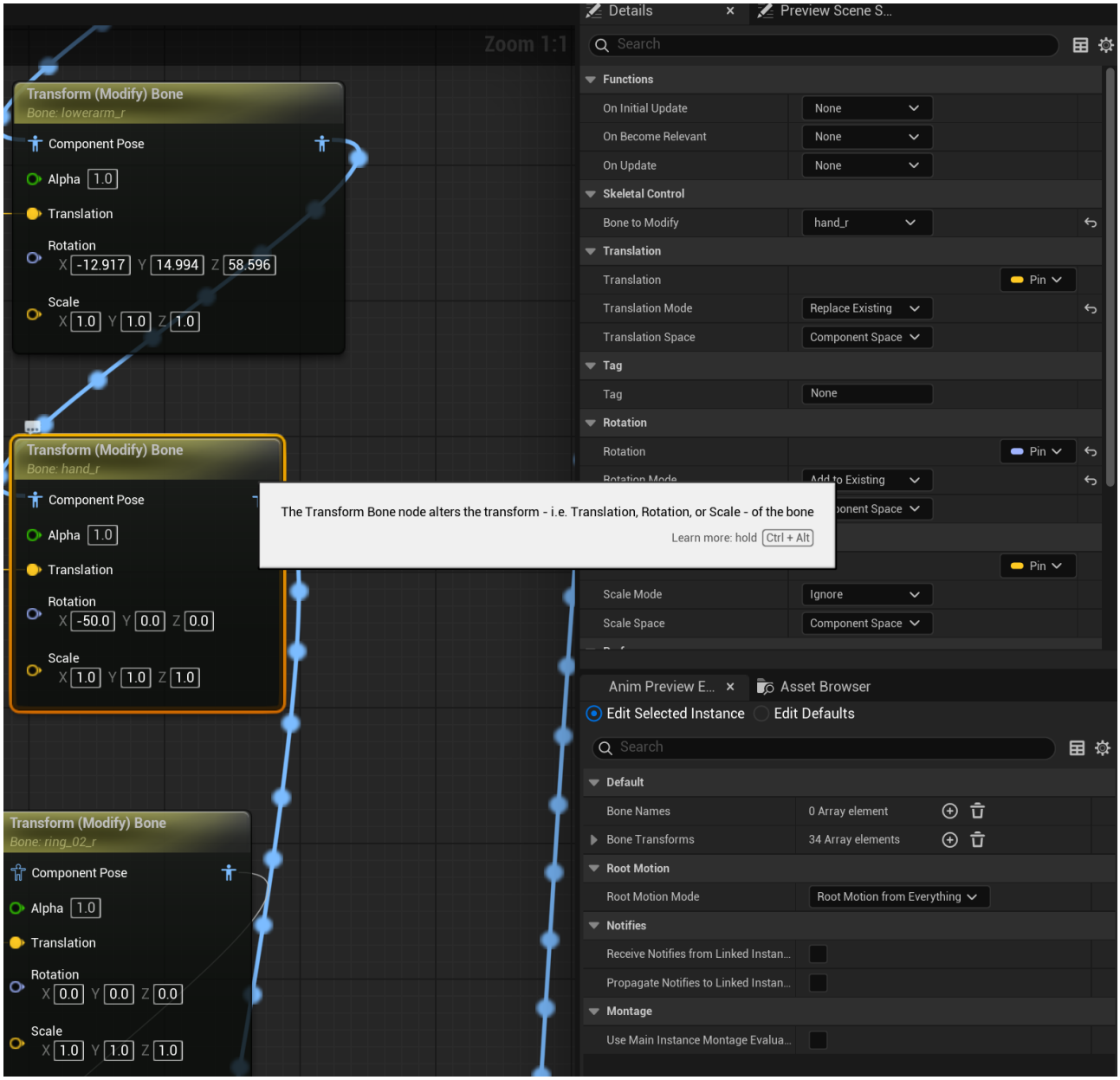
On the default skeleton, the current transform index to bone number on the Animation Graph is:

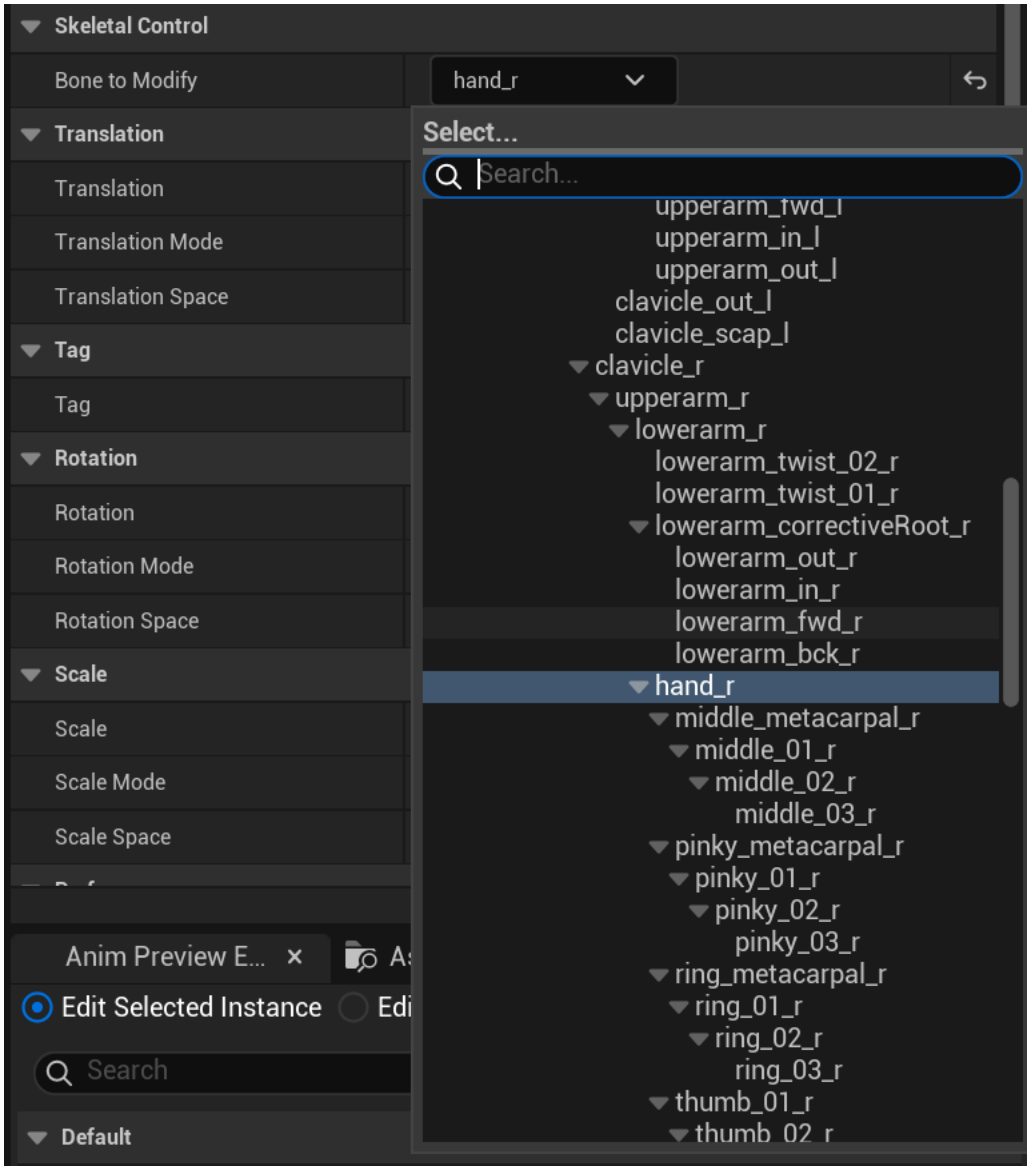
Bone name	Transform #
pelvis	0
spine_03	3
thigh_l	1
calf_l	4
foot_l	13
ball_l	9
thigh_r	2
calf_r	5
foot_r	14

Bone name	Transform #
ball_r	10
neck_01	6
upperarm_l	20
lowerarm_l	22
hand_l	24
upperarm_r	21
lowerarm_r	23
hand_r	25
head	16

More transforms and bones are available but these are the base ones for the example skeleton to work. Different models require different bone combinations but this reference should give a clear picture of what to look for.

To add a specific character with a different model, the advised method would be to copy the current BP_VizrtLiveLinkCharacter and AnimBP_LiveLink assets and modify those copies to suit the needed layout. Refer to the above table for references in terms of hands, feet, legs, arms and so on. To change the bone specific to each transform, select the **Transform (Modify) Bone** nodes on the Animation Graph and put it in the desired bone.





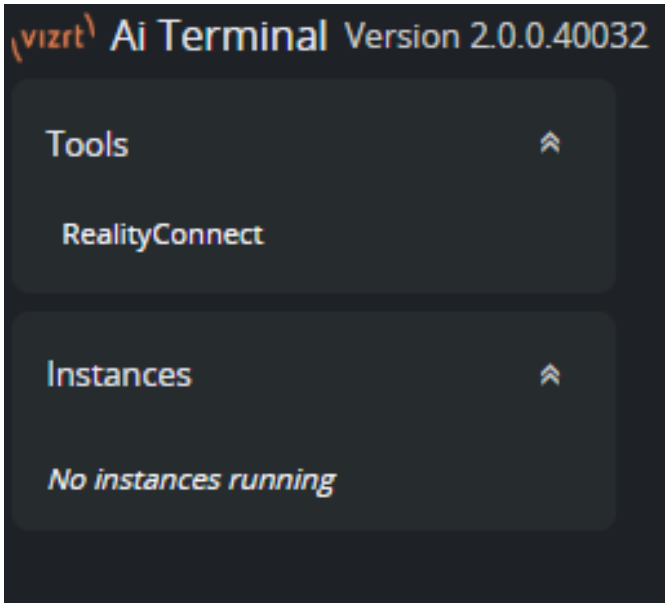
Some rotation offsets can be applied on the same node if the pose is not quite correct.

For an example on possible interactions, check the LiveLink_Interaction level which has a sphere that changes color when the transform animated character collides with the Interaction Sphere.

7 Tool Configuration

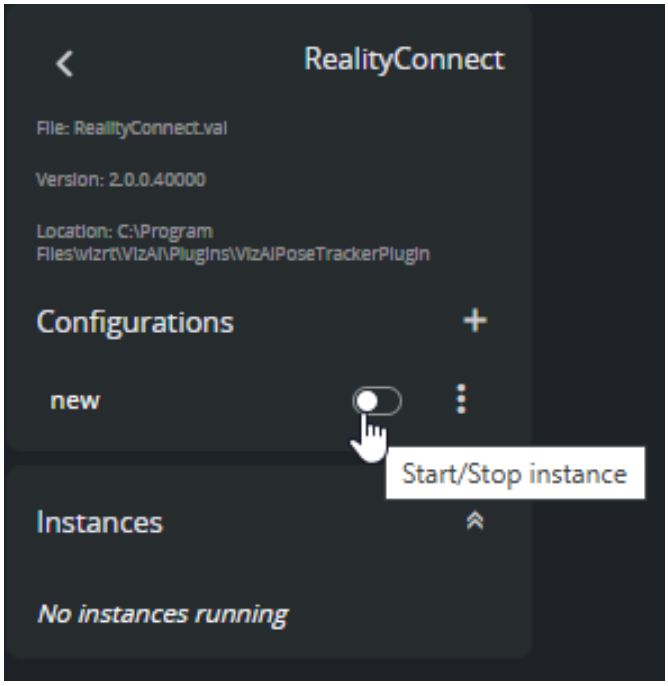
7.1 Create Instance

After installation, Reality Connect is automatically discovered by the AI Terminal. Select the Reality Connect tool.



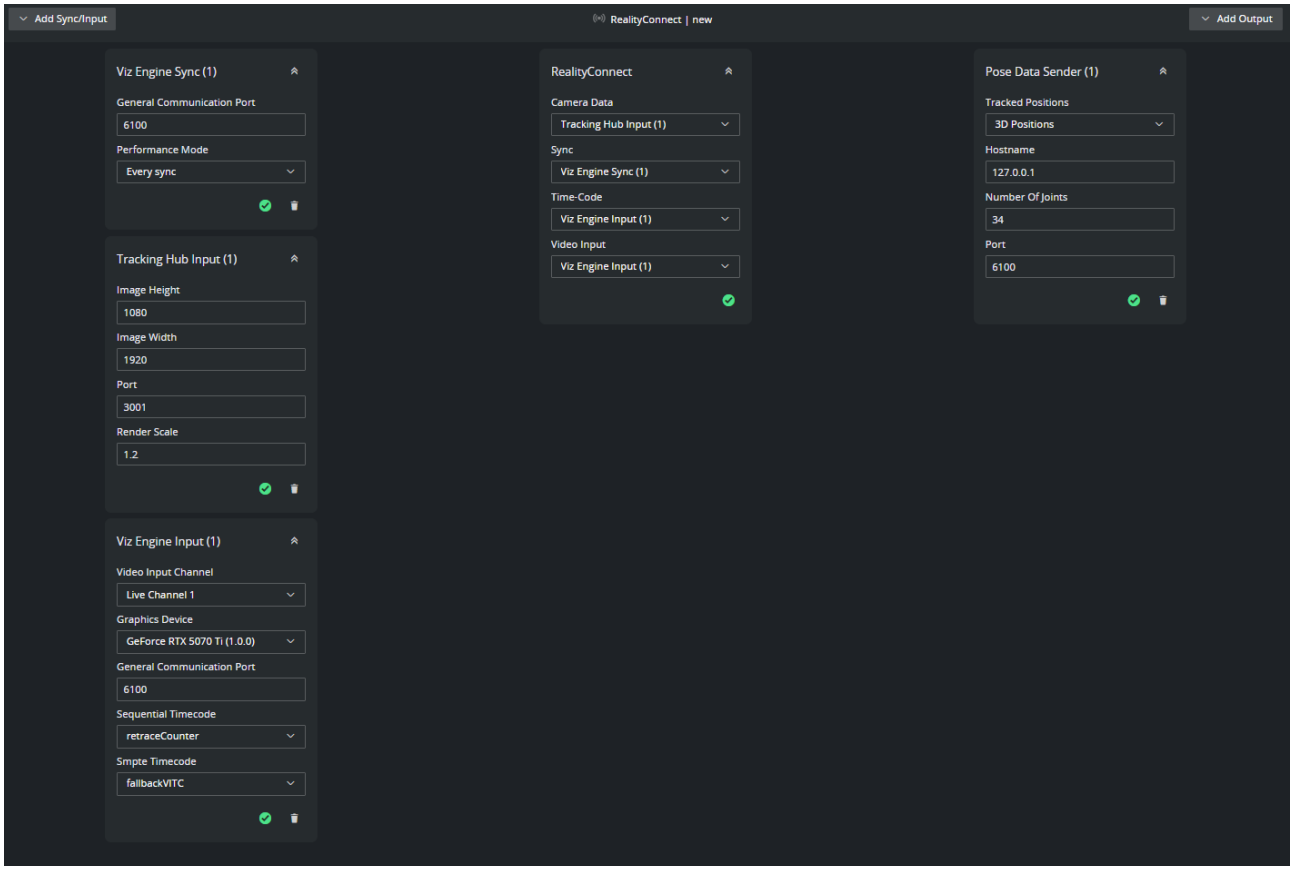
And:

- Create a configuration with the + button.
- Launch an instance with the power on/off switch.



7.2 Input/Output Configuration

Create and configure the following input/output modules.



7.2.1 Viz Engine Sync

- Specify the same value for **Port** as the **General Communication Port** configured earlier in Viz Engine Configuration.
- Choose a **Performance mode**. If performance problems appear later when starting the tool, this mode can be changed to improve performance.
- Select this sync as **Sync** in the Reality Connect box.

7.2.2 Tracking Hub Input

- Specify the same value for **Port** as the **Tracking Port** configured earlier in Tracking Hub Configuration for the camera service sending to Reality Connect.
- Specify the **Image Width**, **Image Height** and **Render Scale**, as configured in Viz Engine.
- Select this input as **Camera Data** in the Reality Connect box.

7.2.3 Viz Engine Input

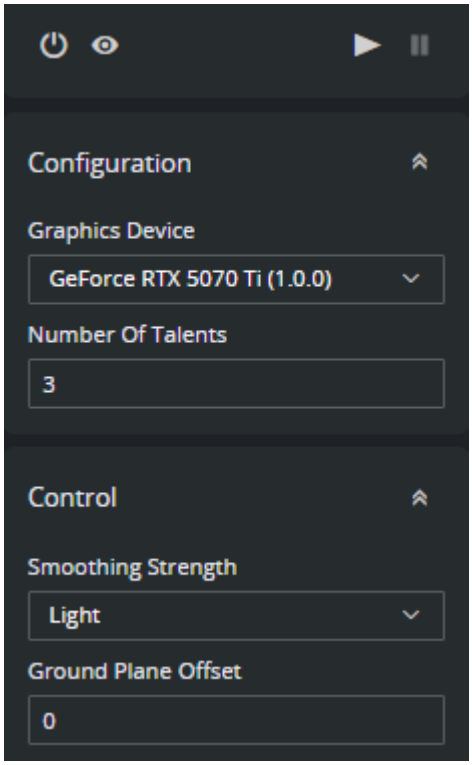
- Select the input channel from the drop box.
- Set **Graphics device** to the device Viz Engine is running on.
- Specify the same value for **Port** as the **General Communication Port** configured earlier in Viz Engine Configuration.

- Choose any timecodes (only required to fetch the video frequency).
- Select this input as **Time-Code** in the Reality Connect box.
- Select this input as **Video Input** in the Reality Connect box.

7.2.4 Pose Data Sender

- Select *Joint Positions* as the **3D Positions**.
- Set the **Hostname** to *127.0.0.1* (if the output Engine is run in same machine). Otherwise enter the IP of the machine that produced the **Viz Engine output**.
- Set the **Number of joints** to *34*.
- Specify the same value for **Port** as the **General Communication Port** configured earlier in Viz Engine Configuration.

7.3 Tool Configuration



7.3.1 Configuration

- Select the **Graphics device** that is used by Reality Connect.
- Specify the maximum number of talents to be tracked. This **must** match the number of models imported earlier in Viz Engine Scene Design.

Note: Despite multiple talents being tracked, this version of Reality Connect does not contain a control mechanism to assign talent models to the actual talents.

7.3.2 Control

- Choose the desired **Smoothing Strength** mode. This affects the quality of the tracked data and the overall system delay (check Delay Configuration for more information).
- Specify the *Z-Position* in **cm** of the virtual ground plane as **Ground Plane Offset**. This controls on which height the tracked talents are projected to.

Info: The control parameters can be changed dynamically during runtime.

7.4 Verify Configuration

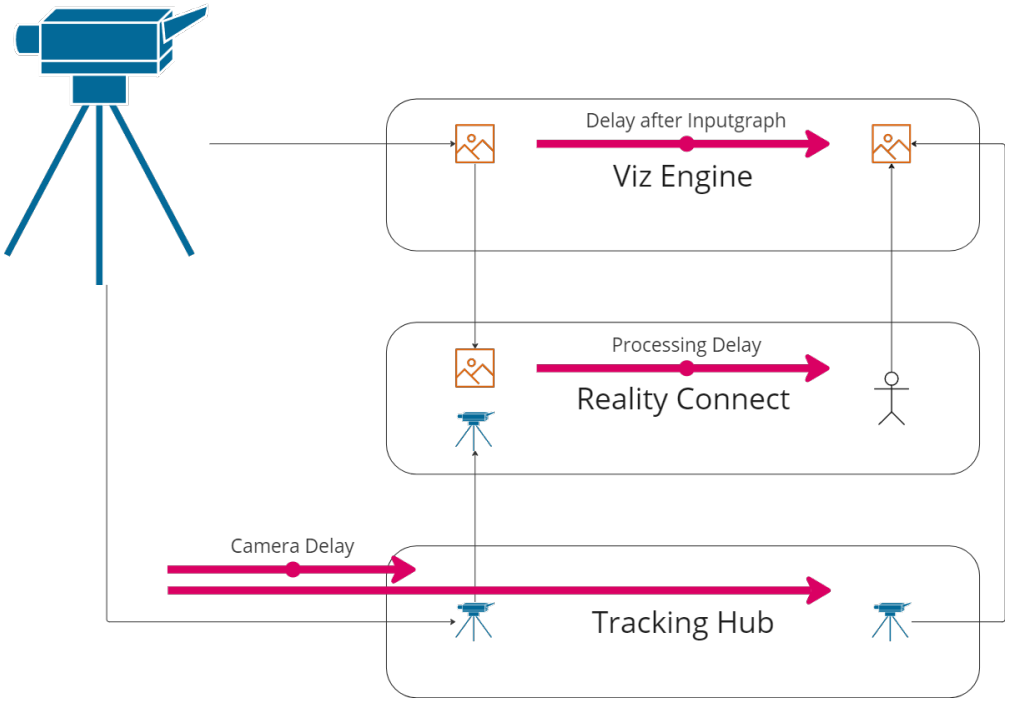
Start the tool and verify that no errors are shown. Check the *AI Terminal User Guide* for proposed actions in case of errors.

If all applications are setup as instructed, the models imported as described in Viz Engine Scene Design, should be animated by the tracked talent motion.

8 Delay Configuration

Camera tracking and video data needs to be synchronized both for Viz Engine and Reality Connect. The delays are affected by multiple factors such as hardware performance and parameters. Read through this chapter to learn how to correct these delays.

8.1 Overview



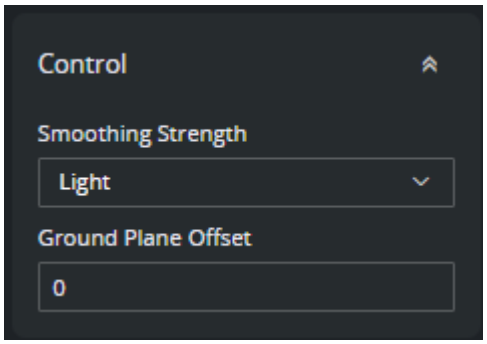
Reality Connect adds an additional delay to the virtual studio system. The image and camera data need to be delayed correspondingly for Viz Engine to render the scene. Details are explained in the following section.

8.2 Synchronization Guidelines

Configure the following parameters to synchronize the system.

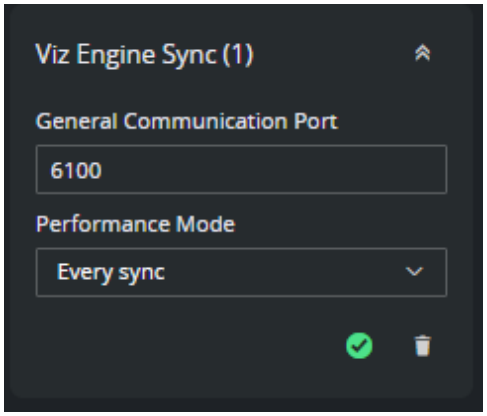
> **Note:** It is important to specify the following delays in a **top-down** order, as some parameters affect others.

8.2.1 Smoothing Strength



A stronger **Smoothing Strength** increases the overall delay of the system.

8.2.2 Performance Mode



The overall delay of the system is increased when not every field is processed.

> **Info:** This assumes that no performance problems are encountered. If a corresponding warning is displayed choose a different performance mode.

8.2.3 Tracking Hub Delay for Viz Engine

Synchronize the camera data with the Viz Engine using the **Overall Delay (Fields)** parameter within the camera rig connected to the Viz Engine. This **Camera Delay** has to be determined experimentally.

For the camera data which is sent to the Viz Engine, add to this **Camera Delay** the **Processing Delay** from the Delay Value Table below.

8.2.4 Tracking Hub Delay for Reality Connect

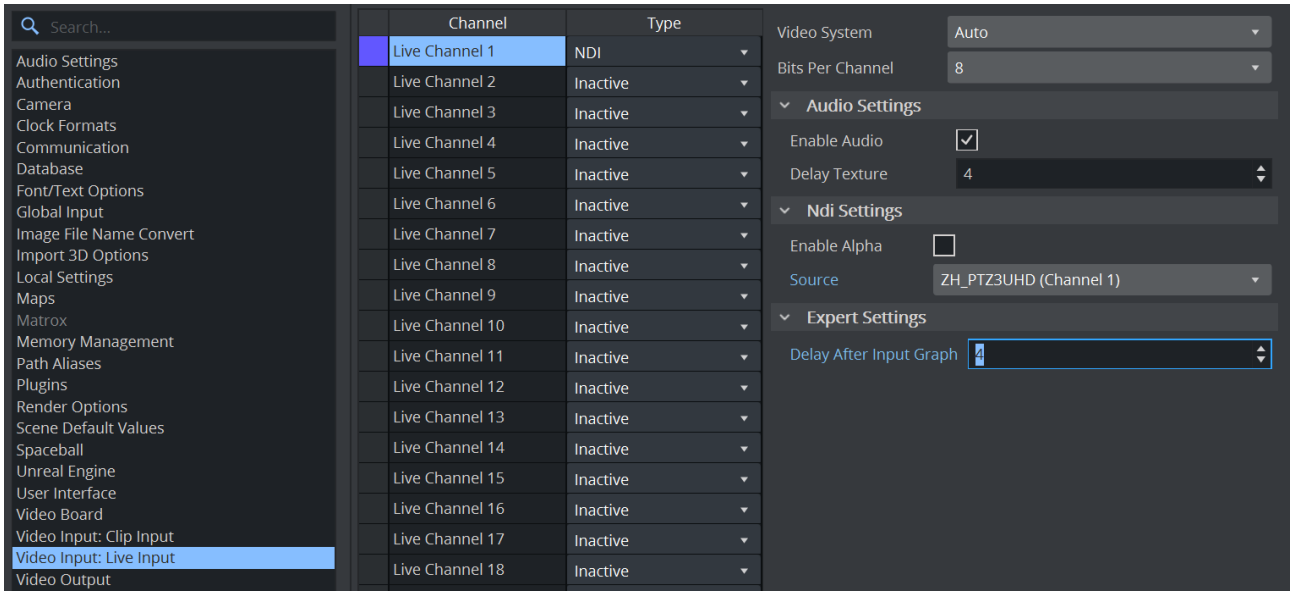
Synchronize the camera data with Reality Connect using the **Overall Delay (Fields)** parameter within the camera rig connected to Reality Connect.

As **Overall Delay (Fields)** set the previously determined **Camera Delay** (*without* the **Processing Delay**).

8.2.5 Viz Engine Delay after Inputgraph

This delay synchronizes the talent tracking data received from Reality Connect with the Viz Engine rendering.

Change the **Delay After Input Graph** for your camera input source to the **Processing Delay** value (in this case LIVE1):



Choose the **Processing Delay** value according to the Delay Value Table at the end of this chapter.

8.3 Delay Value Table

Performance Mode	Smoothing Strength	Processing Delay
Every sync	Off	3
	Light	4

Performance Mode	Smoothing Strength	Processing Delay
	Medium	5
	Strong	7
Every second	Off	4
	Light	5
	Medium	6
	Strong	8
Every fourth	Off	8
	Light	8
	Medium	8
	Strong	10