Chyron®

VSAR 2.1.0

Setup Guide

Contents

WHAT'S IN THIS DOCUMENT

This document describes how to install, setup and test the PRIME VSAR suite 2.1.0

Help and support

For contact information or our online helpdesk, please visit our support page at https://chyron.com/support/.

Disclaimer: Our products are subject to continual development and improvement. Therefore, while the information in this document was complete and accurate when it was written, additions or modifications to the products may cause changes to the technical and functional specifications. No rights can be derived from this document.

Versions

PRIME VSAR	2.1.0 or higher
Data Engine	0.77.7 or higher
PRIME VSAR Tools	2.0.0
Sentinel Driver	7.6.0
License Updater	2.5.0 or higher
CAMIO	4.11.2 or higher
iSQ	3.7.1 or higher
Endpoint	1.2.2 or higher
CESIUM	5.5.0

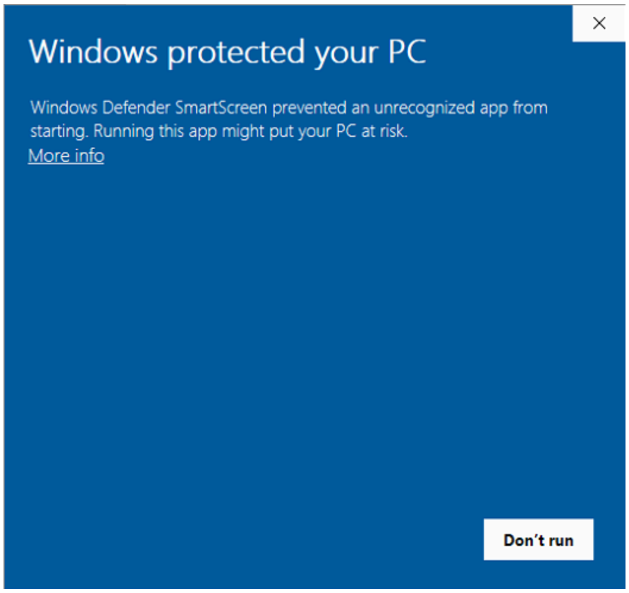
PRIME VSAR INSTALLATION

Installation and configuration steps.

📝Supported platforms are **Windows 10 x64 and Windows 11 x64**.

Installation Note

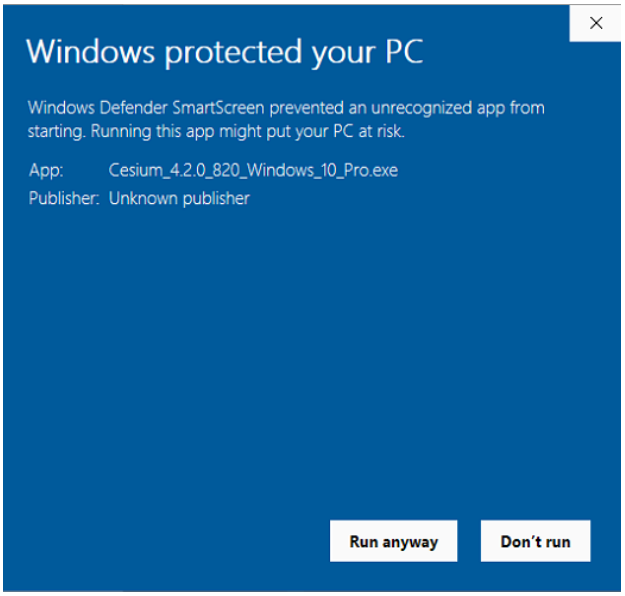
Possible ***Windows Defender*** prompt, when trying to run installers. In some configuration, one may get a warning like the following:



Windows protected your PC

Windows Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

In that case, click on the "More info" link and then "Run anyway"



Windows protected your PC

Windows Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.

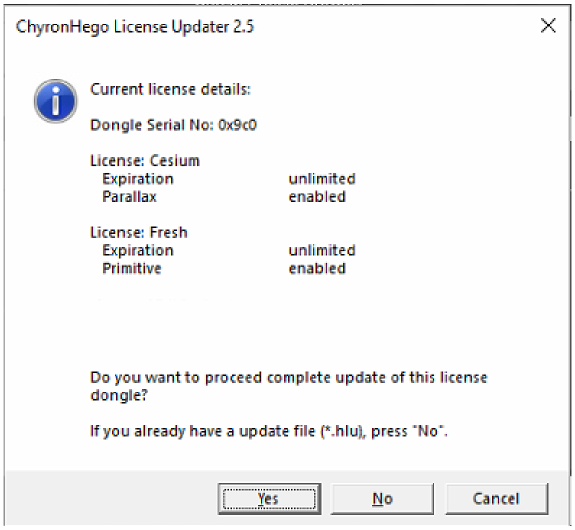
App: Cesium_4.2.0_820_Windows_10_Pro.exe
Publisher: Unknown publisher

License

The following should be done only once, during the first installation.

For any license update, step 3) to 8) may be re-run, after features being added through Download Area (DA).

1) Plug the provided dongle in a USB port.
2) Install ***Sentinel System Driver***.
3) Run ***ChyronHegoLicenseUpdater***.
4) The updater will now show the current licenses.
5) Click **Yes**.
6) Make sure your license has been updated from within the DA.
    a) VSAR(Fresh)
        i) Option: Primitive Library
    b) Cesium
        i) Option: Parallax
7) The updater will now show the licenses to be updated.
8) Click **Yes** to update your license.
9) After this License updater will show you what licenses, options and what validity you have.
10) Please make sure this matches with your ordered configuration - namely Hal Inputs and outputs.



Current license details:

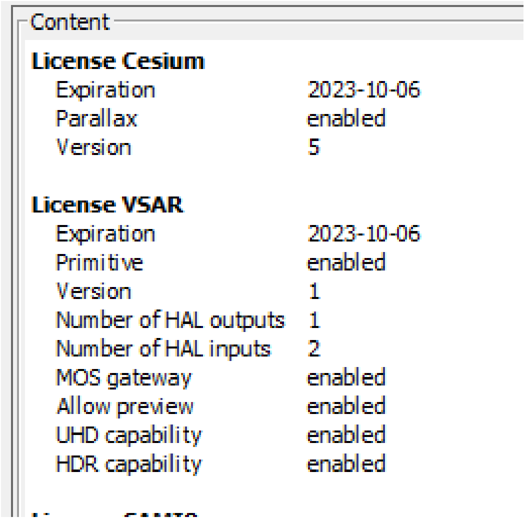
Dongle Serial No: 0x9c0

License: Cesium	
Expiration	unlimited
Parallax	enabled

License: Fresh	
Expiration	unlimited
Primitive	enabled

Do you want to proceed complete update of this license dongle?

If you already have a update file (*.hlu), press "No".



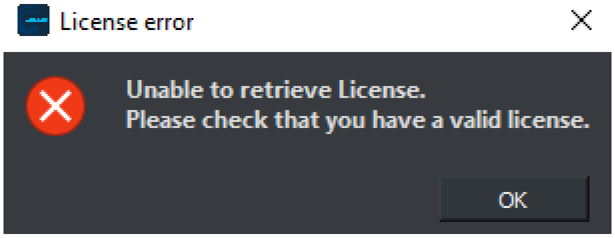
License Cesium	
Expiration	2023-10-06
Parallax	enabled
Version	5

License VSAR	
Expiration	2023-10-06
Primitive	enabled
Version	1
Number of HAL outputs	1
Number of HAL inputs	2
MOS gateway	enabled
Allow preview	enabled
UHD capability	enabled
HDR capability	enabled

Cesium

Cesium is the software that manages translation of any (raw or pre-processed) tracking data coming from any provider into virtual 3D space coordinates.

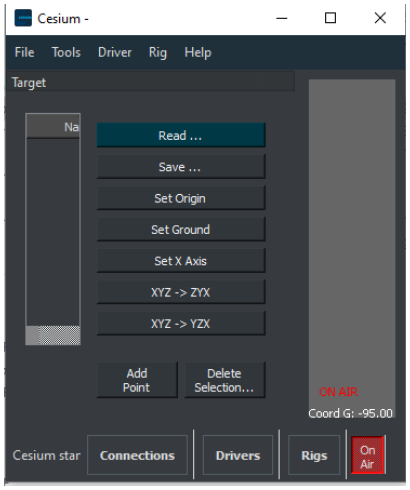
1) Install ***Cesium*** running its corresponding ***Windows*** installer (should look like *Cesium_5.5.NNN_Windows_10_Pro.exe* - *NNN* being release ID).
2) Follow the instructions above, if ***Windows Defender*** prompts for an unrecognized application.
3) After the installation is done, you may want to try and run ***Cesium***.
   The following pop-up signals the software is not able to find the license.



License error

Unable to retrieve License.
Please check that you have a valid license.

   If not already done, plug in the USB dongle with the provided license and run again. ***Cesium*** should open fine.



4) Open any of your tracking file or one provided as sample that can be found at *%ProgramFiles%\Chyron\Cesium\data\cesium\samples*
   For instance *Simulation.cs*.
5) For setting up customized and/or your actual tracking environment, refer to the dedicated *Cesium* manual.
6) To check proper connection in ***PRIME VSAR***, see below, in Configuration Panel.

PRIME VSAR Tools

PRIME VSAR Tools provide a couple of utilities that help ***PRIME VSAR*** users to manage the environment, like connection with the ***CAMIO*** Universe and other remote access software.

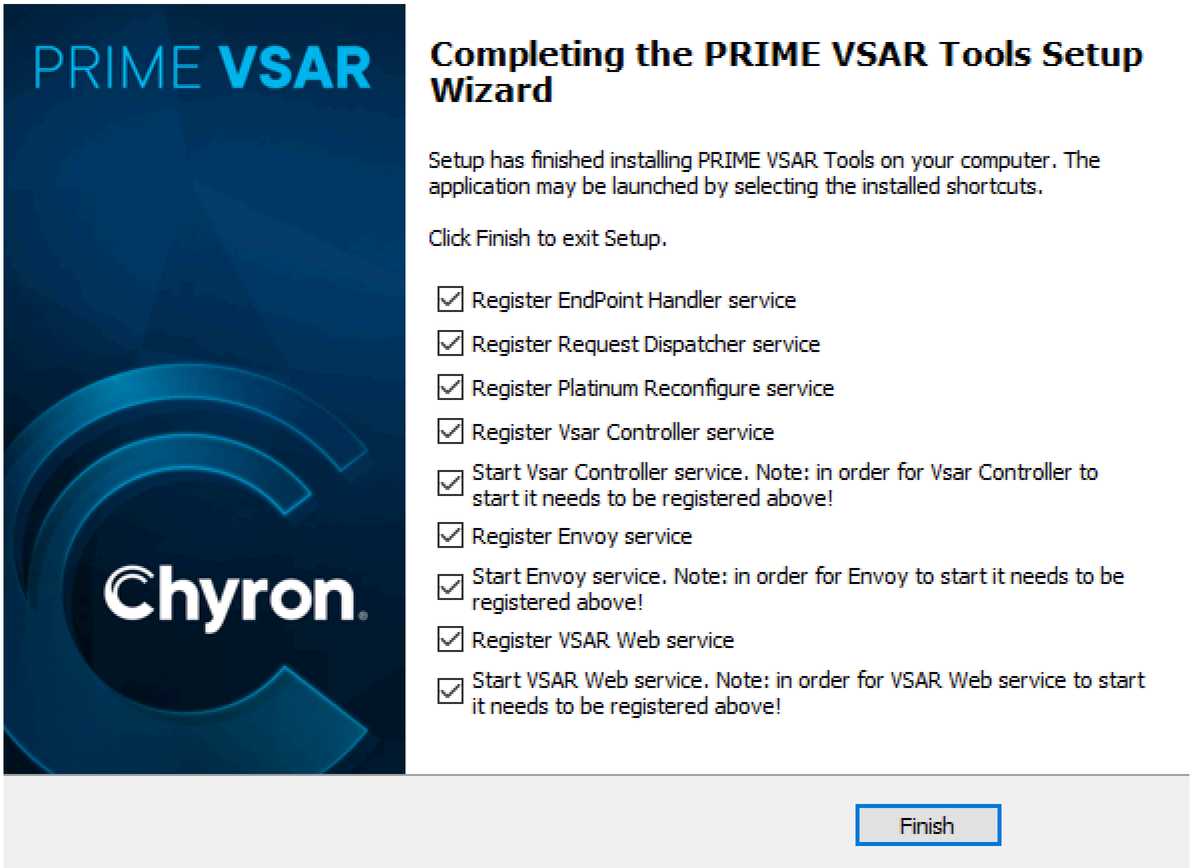
1) Install ***PRIME VSAR Tools***, using related ***Windows*** installer.
2) Follow instructions above, if ***Windows Defender*** prompts for an unrecognized application.

PRIME VSAR Web

Gives the ability to remotely control aspects of **PRIME VSAR** from a web browser. It's part of the **PRIME VSAR** Tools installation.

Make sure the version of **PRIME VSAR** Tools matches the **PRIME VSAR** version or install **PRIME VSAR Tools** that is included in the **PRIME VSAR** installation to ensure full compatibility.

Services used by **VSAR Web** can be registered at the end of **PRIME VSAR Tools** installation:



Completing the PRIME VSAR Tools Setup Wizard

Setup has finished installing PRIME VSAR Tools on your computer. The application may be launched by selecting the installed shortcuts.

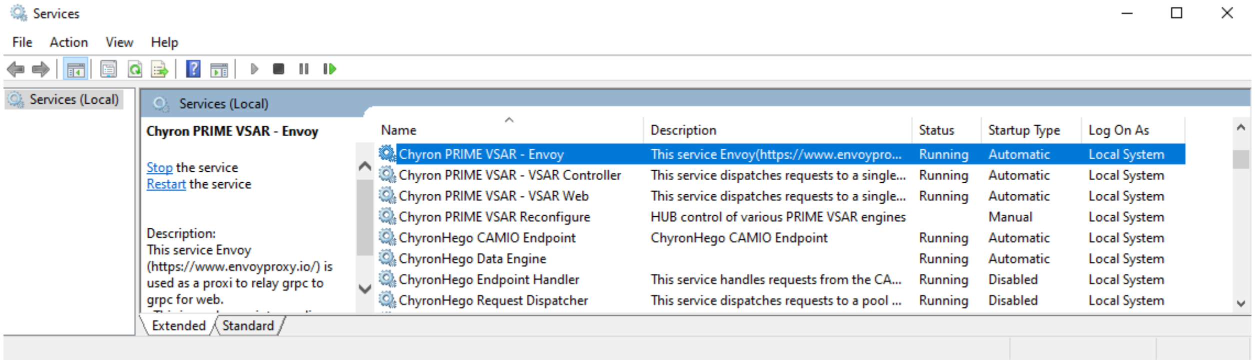
Click Finish to exit Setup.

- ☑ Register EndPoint Handler service
- ☑ Register Request Dispatcher service
- ☑ Register Platinum Reconfigure service
- ☑ Register Vsar Controller service
- ☑ Start Vsar Controller service. Note: in order for Vsar Controller to start it needs to be registered above!
- ☑ Register Envoy service
- ☑ Start Envoy service. Note: in order for Envoy to start it needs to be registered above!
- ☑ Register VSAR Web service
- ☑ Start VSAR Web service. Note: in order for VSAR Web service to start it needs to be registered above!

We highly recommend enabling and starting all the marked services after installation.

Along with the **VSAR Web service** there is also **Envoy service** and **VSAR Controller service** that needs to be registered and running in order for VSAR Web to function properly.

Services can be seen if they are registered and are running in the **Services window** (Win + R - “services.msc”):



Chyron PRIME VSAR - Envoy

Stop the service
Restart the service

Description:
This service Envoy (https://www.envoyproxy.io/) is used as a proxi to relay grpc to grpc for web.

Name	Description	Status	Startup Type	Log On As
Chyron PRIME VSAR - Envoy	This service Envoy(https://www.envoypro...	Running	Automatic	Local System
Chyron PRIME VSAR - VSAR Controller	This service dispatches requests to a single...	Running	Automatic	Local System
Chyron PRIME VSAR - VSAR Web	This service dispatches requests to a single...	Running	Automatic	Local System
Chyron PRIME VSAR Reconfigure	HUB control of various PRIME VSAR engines		Manual	Local System
ChyronHego CAMIO Endpoint	ChyronHego CAMIO Endpoint	Running	Automatic	Local System
ChyronHego Data Engine		Running	Automatic	Local System
ChyronHego Endpoint Handler	This service handles requests from the CA...	Running	Disabled	Local System
ChyronHego Request Dispatcher	This service dispatches requests to a pool ...	Running	Disabled	Local System

VSAR Web configuration

By default **VSAR Web** is hosted on the Local machine and can be accessed on http://localhost:8080/ or on local network with the ip address of the machine followed by the port (:8080). This port can be changed in the website configuration file VsarWebProd.env that is located in the **PRIME VSAR Tools** installation folder bin\config\VsarWebProd.env.

There is a separate configuration file **VSAR Web**, the static website located in **PRIME VSAR Tools** installation folder*%ProgramFiles%/Chyron/PRIME_VSAR-Tools_xxx/bin/build/config.js*. (where xxx is a current VSAR Tools version) this file contains:

- port to the Envoy service
- refresh time
- material settings (range, advanced) for chroma keyer

Envoy service configuration

VSAR Web expects the **Envoy service** to be on the same machine.

Envoy works as proxy for **VSAR Web**, it is a third party tool, more about Envoy proxy on (https://www.envoyproxy.io/)

Configuration for **Envoy service** is located in the **PRIME VSAR Tools** installation folder (mentioned above) **bin\envoy.yaml** . Inside the config section vsar_controller_service there is an ip address and port that should be set to the machine that has **VSAR Controller service** installed.

VSAR Controller service configuration

VSAR Controller allows VSAR Web to run commands on multiple VSAR Renderers, this allows the division of work between multiple machines.

When adding new VSAR to VSAR Controller

1. Make a copy of the Unreal project
2. (this is needed only if project are on the same machine) Change the port in the project files Config\DefaultMithril.ini (create the file if it doesn't exist) and entry
   example:
   [/Script/Mithril.MtGrpcHandler]
   Address=localhost:50051
3. Add new VSAR to VSAR Controller configuration located in PRIME VSAR Tools installation folder bin\config\config.ini
   a. add name of the VSAR to VSAR_NAME_LIST
      example:
      [VSAR_NAME_LIST]
      VSAR_LIST=MY_VSAR,NEW_VSAR
   b. add entry with the suffix "_URL"
      example:
      [NEW_VSAR_URL]

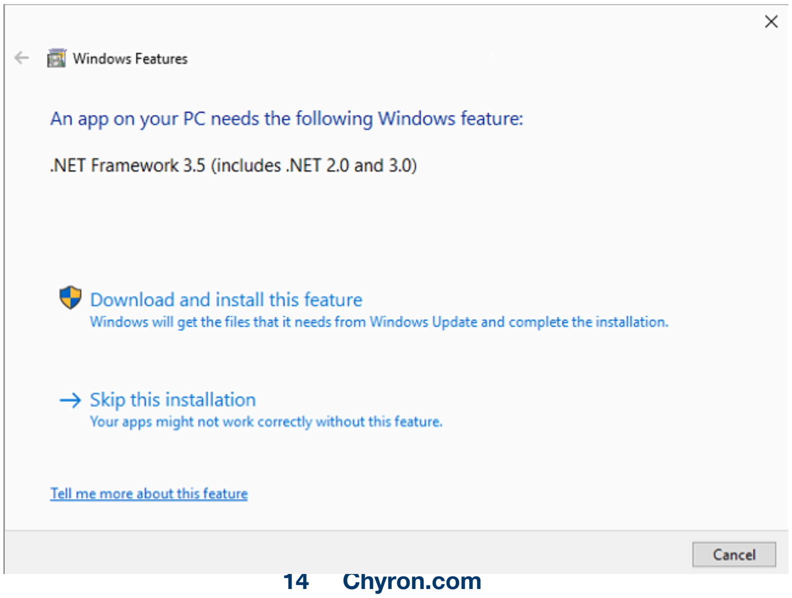
```
VSAR_IP=127.0.0.1
VSAR_PORT=50051
VSAR_POOL=1
VSAR_ENABLED=1
```

PRIME VSAR

Multiple PRIME VSAR versions can be installed on the same machine, there is no need to uninstall a previous version prior to installing a new one. **Warning**: it is recommended to uninstall any previous version prior, if expecting to work with only the last one.

1) If you received the PRIME VSAR installer as a .zip file (eg: PRIME_VSAR_2.0.0-Setup.zip), first start by unzipping it.
2) Install **PRIME VSAR**, by double clicking the .exe file (should look like PRIME_VSAR_2.0.0.xxxx-Setup.exe).

   Note: *PRIME_VSAR-PDB_2.0.0.xxxx-Setup.exe* installer is with Debug functions, ignore if not to be used accordingly.
3) By default, the installer proposes to install in a folder named after the release version (for instance *C:\Program Files\Chyron\PRIME VSAR-2.0.0*). This way different versions of PRIME VSAR may co-exist.
4) Follow instructions above, if Windows Defender prompts for an unrecognized application.
5) If not already installed, you may be prompted for **.NET Framework 3.5** installation.
6) When asked about components you wish to install, install all components shown in the installer except the Prime VSAR Tools **if installed in the previous steps**, or unless you have a specific reason not to install Prime VSAR Tools.



Windows Features

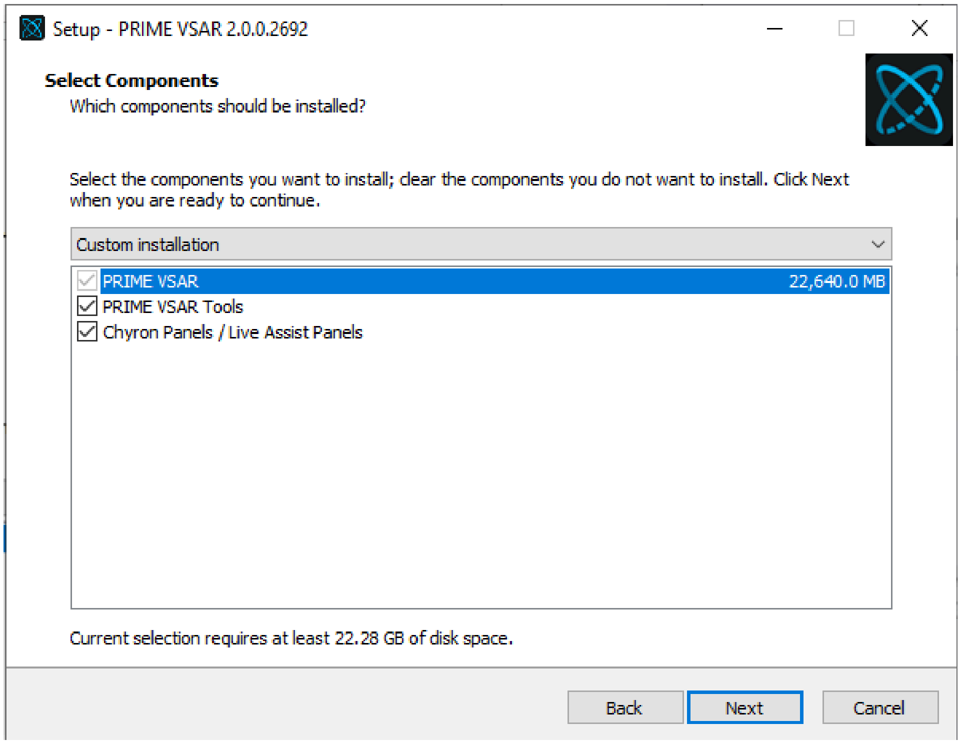
An app on your PC needs the following Windows feature:

.NET Framework 3.5 (includes .NET 2.0 and 3.0)

Download and install this feature
Windows will get the files that it needs from Windows Update and complete the installation.

Skip this installation
Your apps might not work correctly without this feature.

Tell me more about this feature



Setup - PRIME VSAR 2.0.0.2692

Select Components

Which components should be installed?

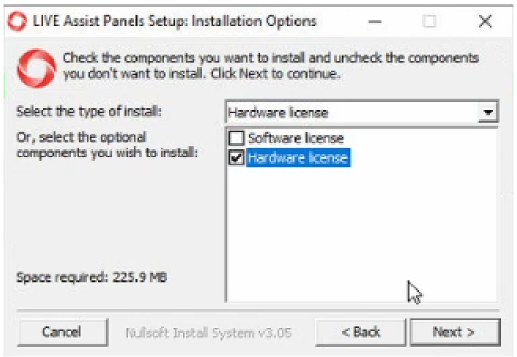
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

Custom installation

- ☑ PRIME VSAR 22,640.0 MB
- ☑ PRIME VSAR Tools
- ☑ Chyron Panels / Live Assist Panels

Current selection requires at least 22.28 GB of disk space.

7) During the installation process of the Chyron Panels (Live Assist Panels), select the **Hardware license** option



LIVE Assist Panels Setup: Installation Options

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

Select the type of install: Hardware license

Or, select the optional components you wish to install:

- ☐ Software license
- ☑ Hardware license

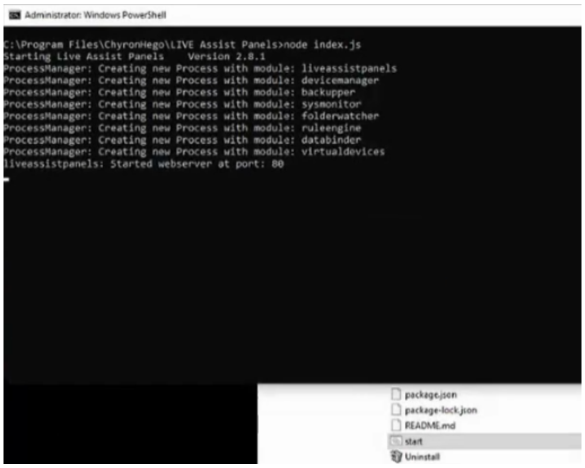
Space required: 225.9 MB

8) After the installation of LAP/Chyron Panels an automatic installation of Mercury panel will be executed.
9) When prompted to restart your computer, do so.

Chyron Panels (Live Assist Panels)

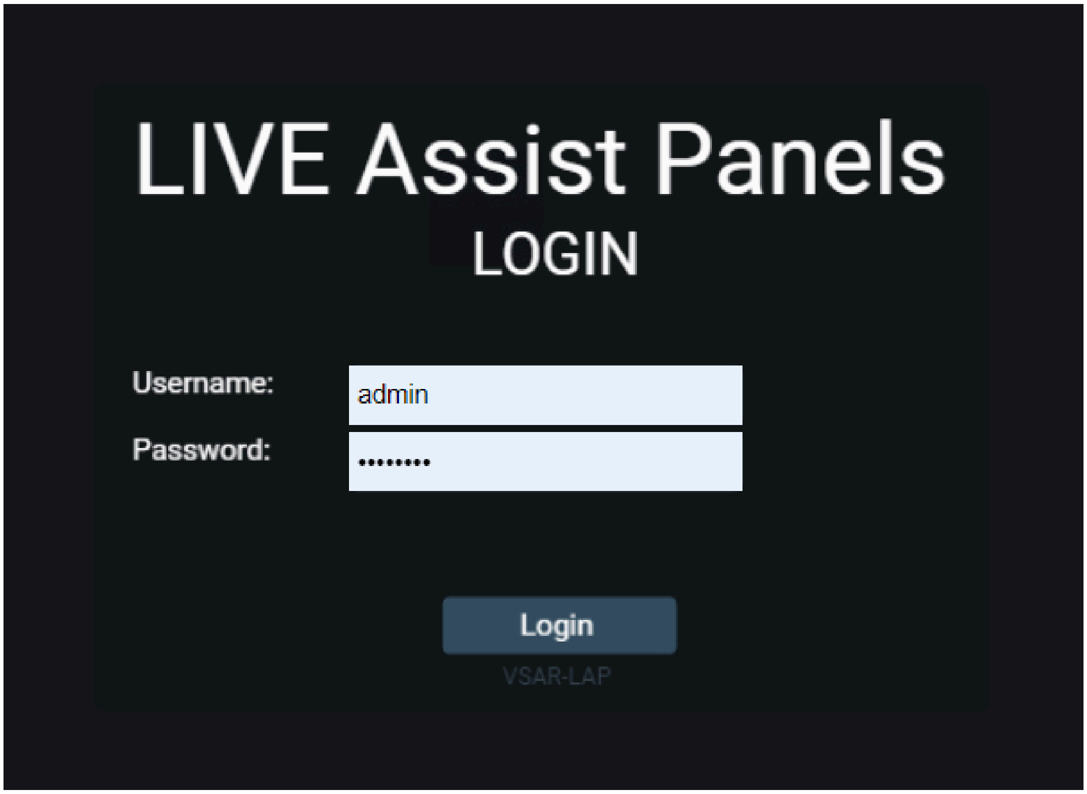
The **Chyron Panels (Live Assist Panels)** web server has to be started manually by clicking on the **LIVE Assist Panels Desktop shortcut**.

- Run it to start the web server. You should get the following output:



```
C:\Program Files\ChyronHego\LIVE Assist Panels>node index.js
Starting Live Assist Panels    Version 2.8.1
ProcessManager: Creating new Process with module: liveassistpanels
ProcessManager: Creating new Process with module: devicemanager
ProcessManager: Creating new Process with module: backupper
ProcessManager: Creating new Process with module: sysmonitor
ProcessManager: Creating new Process with module: folderwatcher
ProcessManager: Creating new Process with module: ruleengine
ProcessManager: Creating new Process with module: databinder
ProcessManager: Creating new Process with module: virtualdevices
liveassistpanels: Started webserver at port: 80
```

- Open Chrome and type in the following URL to open the Panel web page: http://localhost. Login default credentials are **admin / adminLAP**



Username: admin

Password: ••••••••••

VIDEO SETUP

HAL Video Setup

HAL stands for Hardware Abstraction Layer. It is a common interface to video I/Os for *Chyron* products (***PRIME***, ...).

HAL can support different manufacturer boards (called Interface). The inputs and outputs are configured through an Xml file.

Driver installation

Download and install ***Matrox*** drivers Matrox/HAL_driver_10.3.101

To register ***HAL*** for ***Matrox***, open a CommandPrompt, or Bash in **Admin Mode**. Navigate to your Prime VSAR distribution, and then get into directory:

```
> cd…"C:\Program Files\Chyron\PRIME_VSAR-x.x.x.x\Engine\Plugins\Chyronhego\Mithril\ThirdParty \Hal\lib\x64"
```

Then enter the command:

```
> regsvr32 MvIOHAL.dll
```

If for some reason you need to unregister the dll, run:

```
> regsvr32.exe -u MvIOHAL.dll
```

Matrox Board Connections

This document describes the different possible connections of Matrix boards.

for LE5:

HAL Interfaces Configuration

Open the base configuration file *<InstallationDirectory>Engine/Plugins/Chyronhego/Mithril/Config/BaseMithril.ini*, you can then activate and deactivate interfaces by changing the lines:

```
[/Script/HalMedia.Hal]
bInterfaceMatroxEnabled=true
bInterfaceDeltacastEnabled=false
```

Note that interfaces are evaluated in the upper order. This means that in the Xml configuration file, the first board will be a *Matrox* board (if any), and then a *Deltacast* board (if any).

Note that the *Deltacast* interface is not designed for production.

HAL.xml: Video I/O Streams Configuration

Video Streams are configured in the xml file: *%LOCALAPPDATA%/Chyronhego/Fresh/HAL.xml*.

Hal.xml configuration file is loaded when *PRIME VSAR* starts; it allows to configure genlock, inputs and outputs of the different video boards present on the machine. So whenever you edit this file, you need to restart PRIME VSAR.

⚠️When you change video format in Hal.xml from 25/50 fps to 29.97 VSAR needs to be restarted twice, in order for the change to take effect on the Matrox. This is a known issue.

Syntax

```
<halconfig>
        <board>
                <genlock standard="XXX" source="XXX" input="0"/>
                <output standard="XXX" enabled="true" haskey="false" hphase=”0” vphase=”0” fifo="3" 4k2si=”false” tcpcesium="false" cesiumip="127.0.0.1" cesiumport="22222">
                <output/> ...
                <input standard="XXX" enabled="true" haskey="false"/ fifo="3" timeout="50" “useassync”=”false”>
                <input/> ...
        </board>
        <board> ... </board>
</halconfig>
```

Available types

- Available *standards* for <genlock> <output> and <input> tags are:
    - *S259M_NTSC,*
    - *S259M_PAL,*
    - *S274M_1080i_59,*
    - *S274M_1080i_60,*
    - *S274M_1080i_50,*
    - *S274M_1080p_59,*

- ○ *S274M_1080p_60,*
- ○ *S274M_1080p_50,*
- ○ *S274M_1080p_29,*
- ○ *S274M_1080p_30,*
- ○ *S274M_1080p_25,*
- ○ *S296M_720p_50,*
- ○ *S296M_720p_59,*
- ○ *S296M_720p_60,*
- ○ *S4K_2160p_25,*
- ○ *S4K_2160p_29,*
- ○ *S4K_2160p_30,*
- ○ *S4K_2160p_50,*
- ○ *S4K_2160p_59,*
- ○ *S4K_2160p_60,*
- ○ *SFILM_720p_23,*
- ○ *SFILM_720p_24,*
- ○ *SFILM_1080p_23,*
- ○ *SFILM_1080p_24,*
- ○ *SFILM_2160p_23,*
- ○ *SFILM_2160p_24*
- ○ Boolean values accept “true|1” and “false|0”

- Note: formats in red have known issues and may not work properly

<genlock>

Available genlock ***source*** are: *internal|analog|input.*

- In internal mode, the board is not genlocked to an external signal.
- In analog mode, the genlock is received on the dedicated Sync BNC.
- In input mode, the genlock is taken from the ***inputs*** SDI video input. ***Input*** attribute is only relevant when *source* is set to *input*.

<output>

If ***enabled*** is set to “false” the output will be disabled. Using this you can configure eg.: that the 1st connector is disabled and that you want to use 2. connector only.

If ***haskey*** is true, the output will be set in Fill+Key; if set to false, the output will be Fill only. Note that if **haskey** is enabled. As key is generated, it will be generated automatically into another output connector.

⚠️ when **haskey=false** there is color conversion in place that can result in worse chrominance resolution, for the highest resolution we recommend to use **haskey=true**

fifo is the number of frames buffered on the video device; there are two working modes:

- **FIFO Mode**: In fifo mode, frames computed by VSAR are pushed on the output connector; there can't be more than *fifo* buffers on the output queue. Setting a value of minimum 2 is required, 3 is recommended. In this mode, VSAR cannot guarantee that Cesium data will be properly in sync with the real camera.
- **Delay Mode**: In Delay mode, a fixed delay is guaranteed between a given input on the Matrox board, and a given output. For example, 6 frames can be guaranteed between SDI input 0 and output 2. To use this mode. You need to select an input as the 'reference delay' input (see ***useassync*** in <input> below).

hphase and ***vphase*** set the video phase of each output separately (they can be edited in the GUI, but they are currently not reported in the xml file).

If ***4k2si*** is true then 4K signal is transmitted in 2 sample interleave format; otherwise four quadrant split is used. *Note*: this feature may not work on some Matrox cards, use mvConnectorConfig (see below) to configure 2SI.

If ***tcpcesium*** is true then tcp messages will be sent to cesium to genlock the UdpGenlock Cesium drivers. This value is usually set to false in standard operation. ***cesiumip*** and ***cesiumport*** allow to set the Cesium you want to send sync messages to.

If **HDR** is true the camera rendering is done in larger color resolution (16f) and then send using packed 10b signal (YCrCb_422_10b) to output

<input>

Input has the same attributes as output.

timeout is the duration (in ms) after which a video source is considered missing.

useassync is a boolean used to select an input source as the 'Delay mode' reference. If several inputs have the tag ***useassync*** set to true, then only the last one is considered as the sync source.

⚠️ When using ***useassync,*** VSAR main thread is "locked" to the input, meaning that when input is not available VSAR can get stuck.

⚠️When using ***useassync*** input in MediaPlayer it is possible on close of the source that output may drop up-to 2 frames, we recommend in production to keep it either opened or closed.

Notes/Troubleshooting:

- When a new "output" (resp. "input") is parsed the previously parsed output (resp. "input") is copied to the new one. This allows the user not to copy all the attributes on each line.
- This is also true for the "standard" attribute of genlock, which will be copied to the next input and output lines.
- The tcpcesium is reset to false when a new output is built.

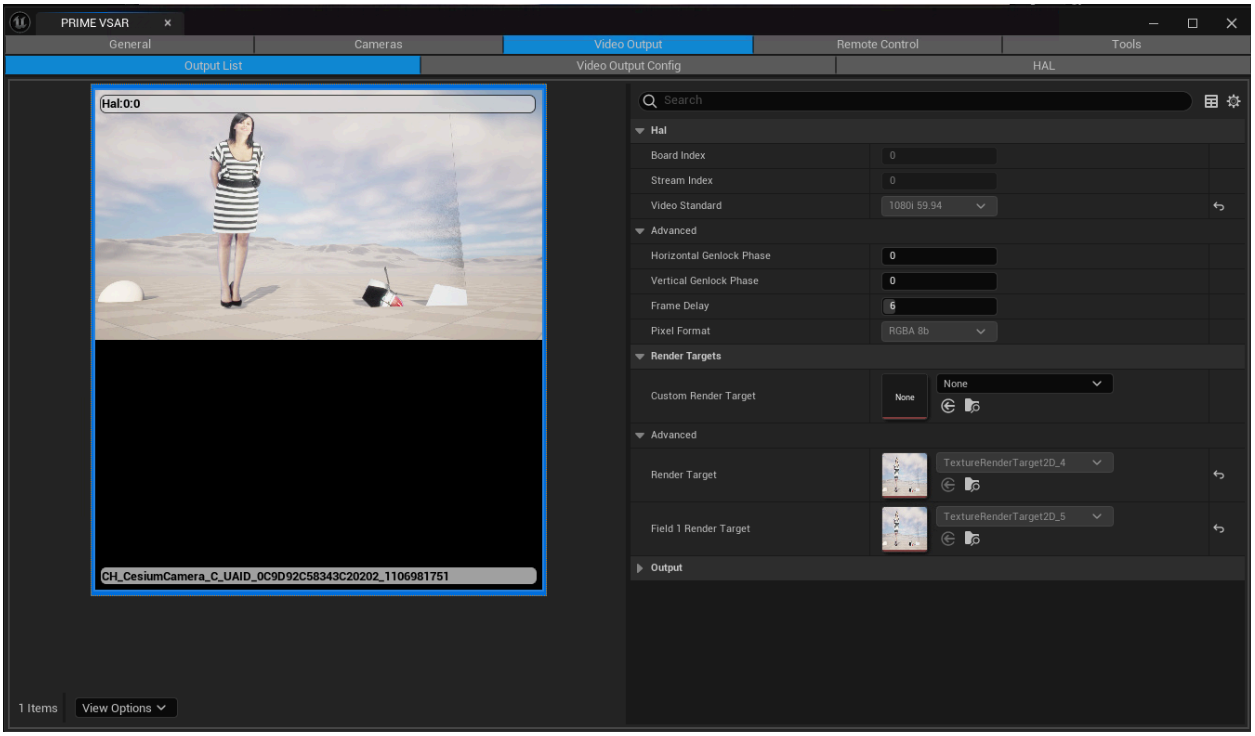
- For Matrox, if you change the framerate (eg: 50->59, or 59->60) then you may have to restart twice PRIME VSAR before it's taken into account.
- For ***Matrox***, it seems that changing haskey needs a reboot. In fact haskey="true" -> haskey="false" seems to work, but the other way seems very challenging. The same applies for 4K outputs.
- If you have more <output> or <input> declared in the xml file than the actual hardware has, they are silently ignored. Thus it's harmless to leave extra <output> or <input> lines.

Delay Mode Notes:

In Delay mode, you need to have set at least one input with "useassync"="true" (see Example4 below). This input is called 'delay reference input'.

- If an input has the "useassync" tag set, all the outputs will be considered as configured in "Delay Mode", thus their fifo attribute will silently be set to 0.
- You need to have the genlock in sync with the 'delay reference input'. If you use "source"="analog" then the analog genlock should be in sync with the 'delay reference input'. Alternatively, you can set "source"="input" "input"="XX" where XX is the index of the 'delay reference input', starting at 0 (the first input has index 0, the second index 1, and so on).

To tune the delay between the output and the 'delay reference input', get in VSAR Control Panel > Video Output > Outputs > Frame Delay. Frame delay is set independently for each output. The default is 6.



Hal
- Board Index: 0
- Stream Index: 0
- Video Standard: 1080i 59.94

Advanced
- Horizontal Genlock Phase: 0
- Vertical Genlock Phase: 0
- Frame Delay: 6
- Pixel Format: RGBA 8b

Render Targets
- Custom Render Target: None

Advanced
- Render Target: TextureRenderTarget2D_4
- Field 1 Render Target: TextureRenderTarget2D_5

Output

Examples

Example 1: the following file will configure 2 inputs and 2 outputs in Fill Only:

```
<halconfig>
        <board>
                <genlock standard="S274M_1080i_59"/>
                <output/>
                <output/>
                <input/>
                <input/>
        </board>
</halconfig>
```

Example 2: the following file will configure 2 inputs and 2 outputs in Fill Only, but only the first output will send sync data to cesium.

```
<halconfig>
        <board>
                <genlock standard="S274M_1080i_59"/>
                <output tcpcesium="true"/>
                <output/>
        <input/>
        <input/>
    </board>
</halconfig>
```

Example 3: the following file will configure 2 inputs and 1 output (the second on the board, since the first one is disabled) in Fill Only.

```
<halconfig>
    <board>
        <genlock standard="S274M_1080i_59"/>
        <output enabled="0"/>
        <output/>
        <input/>
        <input/>
    </board>
</halconfig>
```

Example 4: the following file will configure 1 input as the sync source, 2 inputs as live video inputs, 1 output in ‘Delay Mode’, and genlock as Analog. Note that in this configuration, Input0 and AnalogGenlock should be in sync.

```
<halconfig>
    <board>
        <genlock standard="S274M_1080i_59" source="analog">
        <output enabled="1" fifo="0">

        <input useassync="true"/>
        <input useassync="false"/>
        <input/>

    </board>
</halconfig>
```

Matrox I/O reconfiguration

If you need to reconfigure the number of video inputs and outputs: open a command prompt with Admin rights:

```
> cd “C:\Program Files\Matrox DSX-TopologyUtils\drivers”
```

To get the list of commands (optional)

```
> mvConnectorConfig.exe -help
```

To get the Matrox Card Serial Number

```
> mvConnectorConfig.exe list
```

To display the card’s current configuration

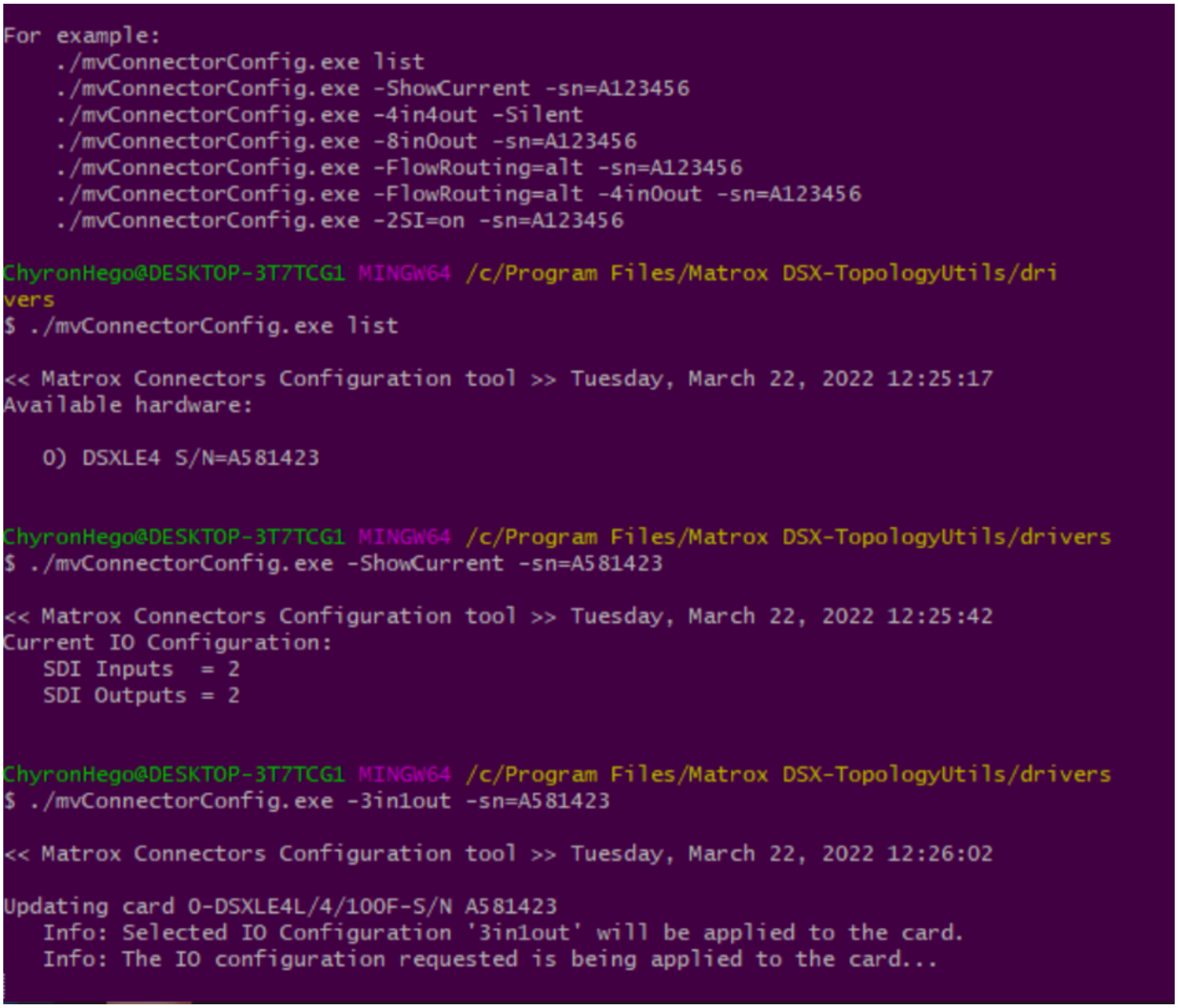
```
> mvConnectorConfig.exe -ShowCurrent -sn=<Matrox Card Serial Number>
```

To re-affect the number of video inputs and outputs (x and y values respectively)

```
> mvConnectorConfig.exe -<x>in<y>out -sn=<Matrox Card Serial Number>
```

The last command takes a while to execute, then you have to reboot the machine.

Example usage for reconfiguration of 4 spigot card from 2 in 2 out into 3 in 1 out:



```
For example:
    ./mvConnectorConfig.exe list
    ./mvConnectorConfig.exe -ShowCurrent -sn=A123456
    ./mvConnectorConfig.exe -4in4out -Silent
    ./mvConnectorConfig.exe -8in0out -sn=A123456
    ./mvConnectorConfig.exe -FlowRouting=alt -sn=A123456
    ./mvConnectorConfig.exe -FlowRouting=alt -4in0out -sn=A123456
    ./mvConnectorConfig.exe -2SI=on -sn=A123456

ChyronHego@DESKTOP-3T7TCG1 MINGW64 /c/Program Files/Matrox DSX-TopologyUtils/dri
vers
$ ./mvConnectorConfig.exe list

<< Matrox Connectors Configuration tool >> Tuesday, March 22, 2022 12:25:17
Available hardware:

   0) DSXLE4 S/N=A581423


ChyronHego@DESKTOP-3T7TCG1 MINGW64 /c/Program Files/Matrox DSX-TopologyUtils/drivers
$ ./mvConnectorConfig.exe -ShowCurrent -sn=A581423

<< Matrox Connectors Configuration tool >> Tuesday, March 22, 2022 12:25:42
Current IO Configuration:
   SDI Inputs  = 2
   SDI Outputs = 2


ChyronHego@DESKTOP-3T7TCG1 MINGW64 /c/Program Files/Matrox DSX-TopologyUtils/drivers
$ ./mvConnectorConfig.exe -3in1out -sn=A581423

<< Matrox Connectors Configuration tool >> Tuesday, March 22, 2022 12:26:02

Updating card 0-DSXLE4L/4/100F-S/N A581423
   Info: Selected IO Configuration '3in1out' will be applied to the card.
   Info: The IO configuration requested is being applied to the card...
```

PRIME VSAR OVERVIEW

PRIME VSAR is a plugin for ***Unreal Engine 5*** (here version 5.3.2), and is developed as such. That being said, it installs its own modified version of Unreal Editor.

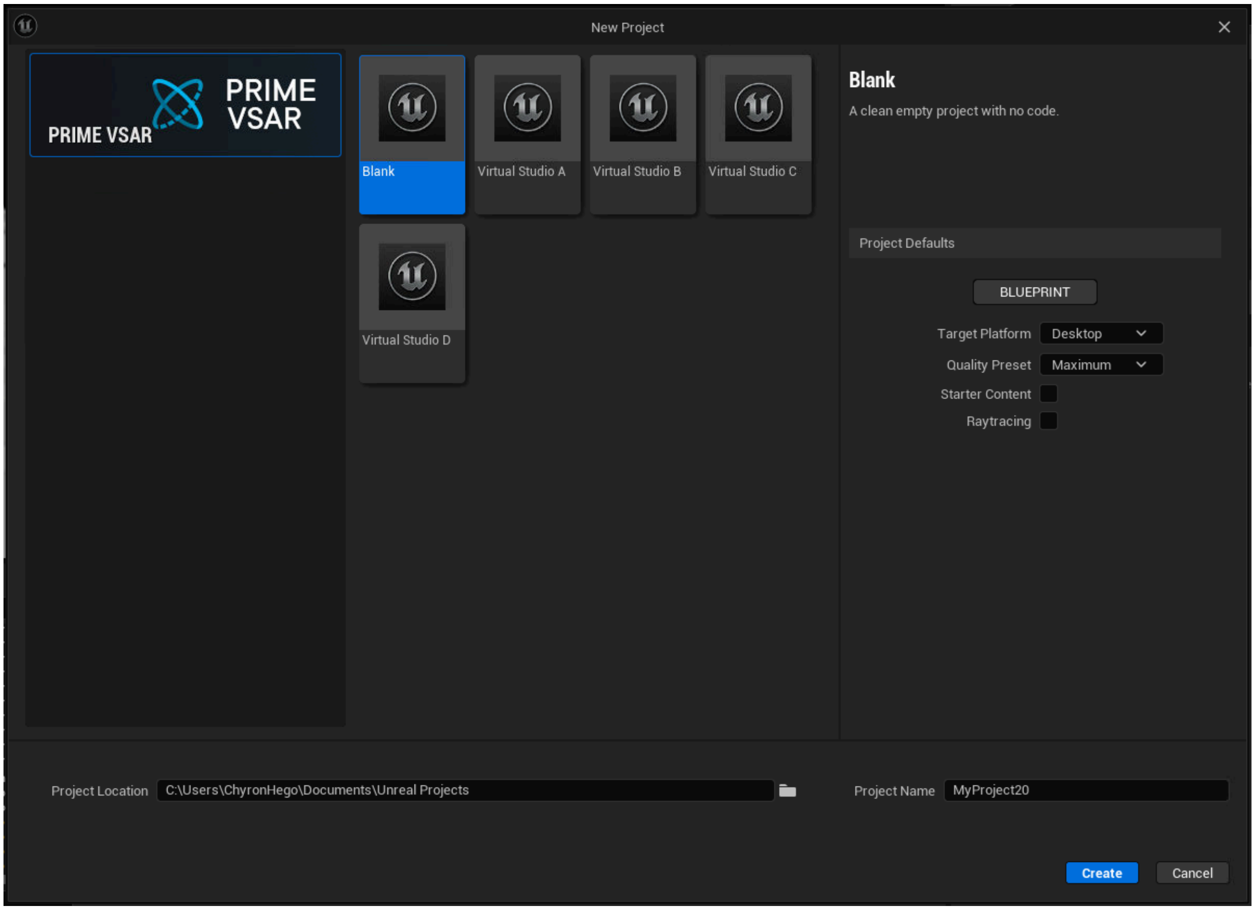
PRIME VSAR brings the following elements to *UE5*:

- Video I/O.
- Interface with tracking (***Cesium***).
- Primitives (a set of procedural geometries, for instance to address some business graphic purposes).
- Interface with ***CAMIO*** Universe.
- A/B Switch (a tool to manage transitions between media, comparable to a small video mixer/switcher).
- Exposes ***UE5*** and ***PRIME VSAR*** core elements to ***LUA*** scripting.
- A subset of related functions/materials/components.

Most of these elements can be managed via the Configuration Panel.

Creating a PRIME VSAR Project

1) Start ***PRIME VSAR*** (**Warning**: this may take time to load, especially during the first start, (due to the shader compilation) see remark below.
2) Create a new or open an existing project.



Blank

A clean empty project with no code.

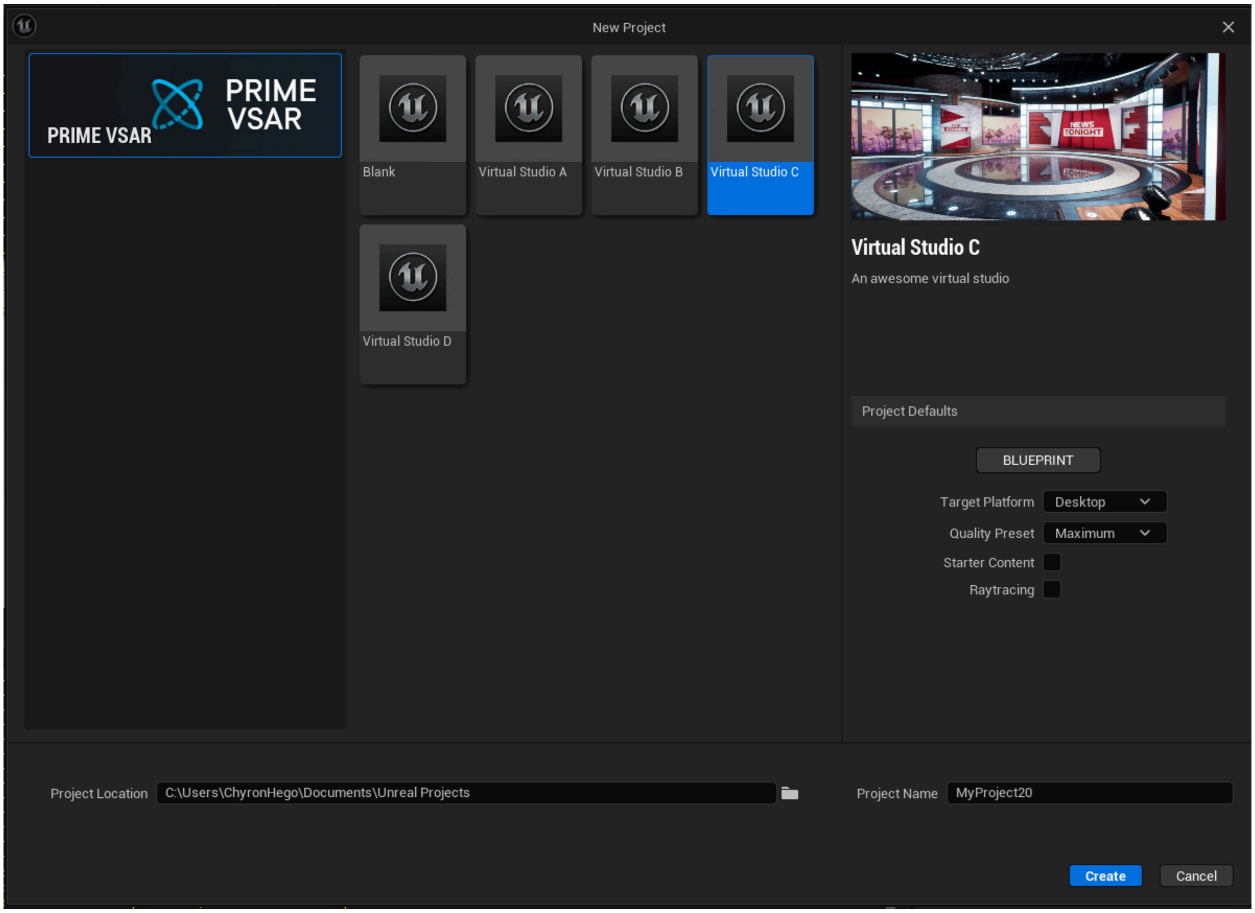
Target Platform: Desktop

Quality Preset: Maximum

Project Location: C:\Users\ChyronHego\Documents\Unreal Projects

Project Name: MyProject20

Select ***PRIME VSAR*** *Example Projects* by clicking on it to have it highlighted and enable the *Next* button.



Virtual Studio C

An awesome virtual studio

Project Defaults

BLUEPRINT

Target Platform: Desktop

Quality Preset: Maximum

Starter Content □

Raytracing □

Project Location: C:\Users\ChyronHego\Documents\Unreal Projects

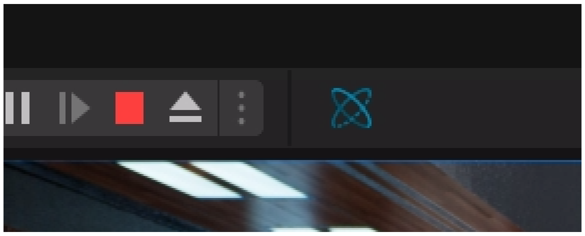
Project Name: MyProject20

PRIME VSAR comes with a couple of **Virtual Studio samples** originating from ***Unreal*** marketplace.

As for any *Unreal* projects, settings may be set up before creating it, but may be later changed. It is recommended to have **Maximum Quality** and **Desktop/Console** by default.

PRIME VSAR Configuration Panel

Open the **PRIME VSAR Config Panel** by clicking on the top **ToolBar** close to **Settings**.

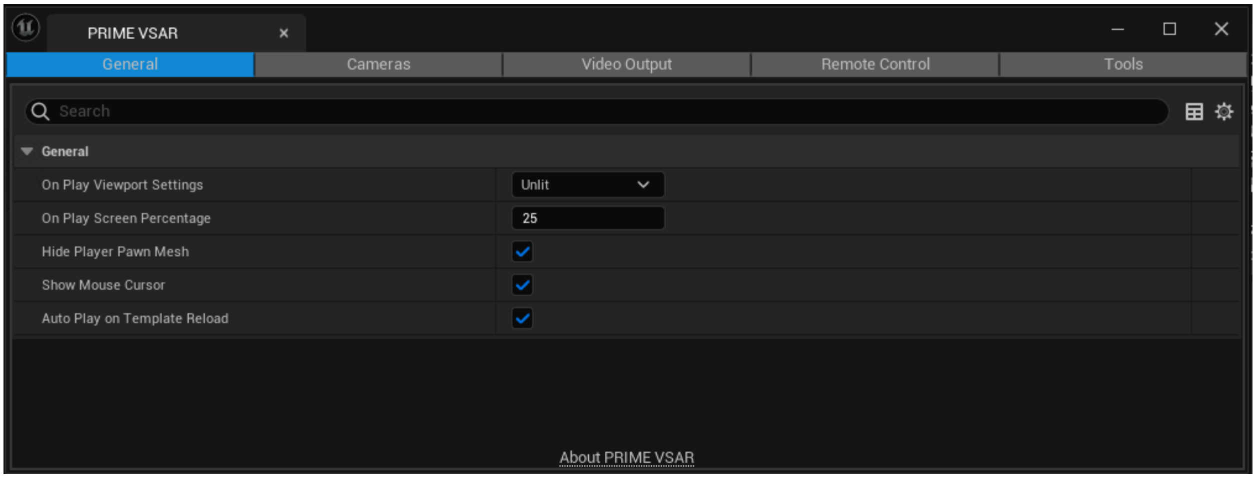


PRIME VSAR related info is located here.

The panel is divided into the following sections/tabs:

- General
- Cameras
- Video Output
- Remote control
- Tools

General



General

On Play Viewport Settings: Unlit

On Play Screen Percentage: 25

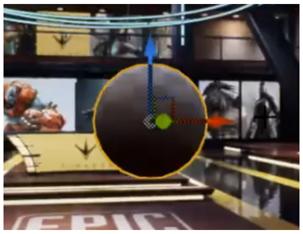
Hide Player Pawn Mesh: ☑

Show Mouse Cursor: ☑

Auto Play on Template Reload: ☑

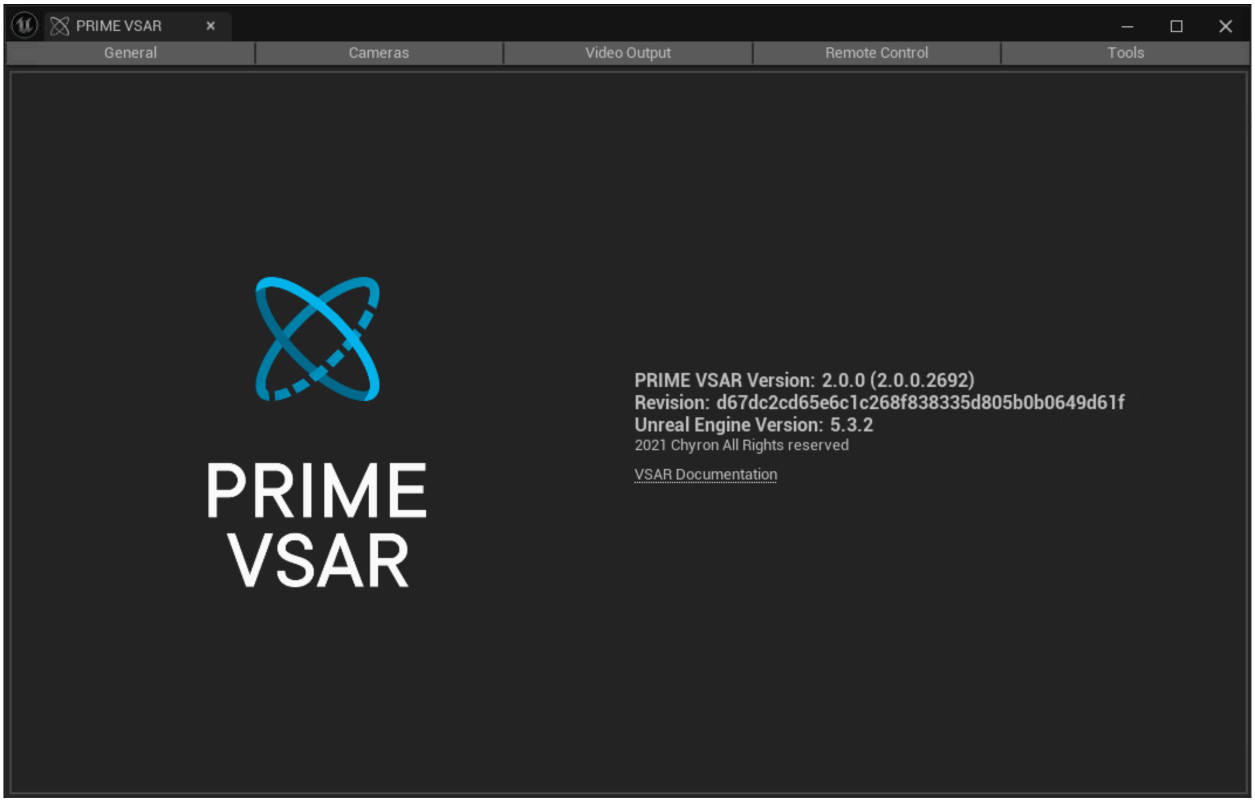
Options listed in this tab can be used while running in Play mode, mainly to decrease performance needs.

- **On Play Viewport Settings**:
  - **Nothing:** this settings leaves the quality of rendering on default presets.
  - **Unlit:** play with "Unlit" lightning mode - this decreases performance demands on the rendering
  - **Disable Viewport:** this completely disables viewport, which saves the most performance for the output rendering, Warning: Screen space reflections do not update and GPU Particles don't work when the viewport is disabled. There is large memory allocation over time that will require VSAR restart.
- **On Play Screen Percentage**:
  Reduces rendering of the viewport resolution by percentage
- **Hide Player Pawn Mesh**:
  (Checkbox checked by default) In Unreal, a default Player Pawn is created at play time as a sphere. Irrelevant for PRIME VSAR usage



- **Show Mouse Cursor:**
  (Checkbox checked by default) Shows (or not) the cursor in the viewport while in play mode.

The *About PRIME VSAR* link at the bottom of the window will display the about dialog, showing current ***PRIME VSAR*** release info. Pressing any tab will switch back to its related content.

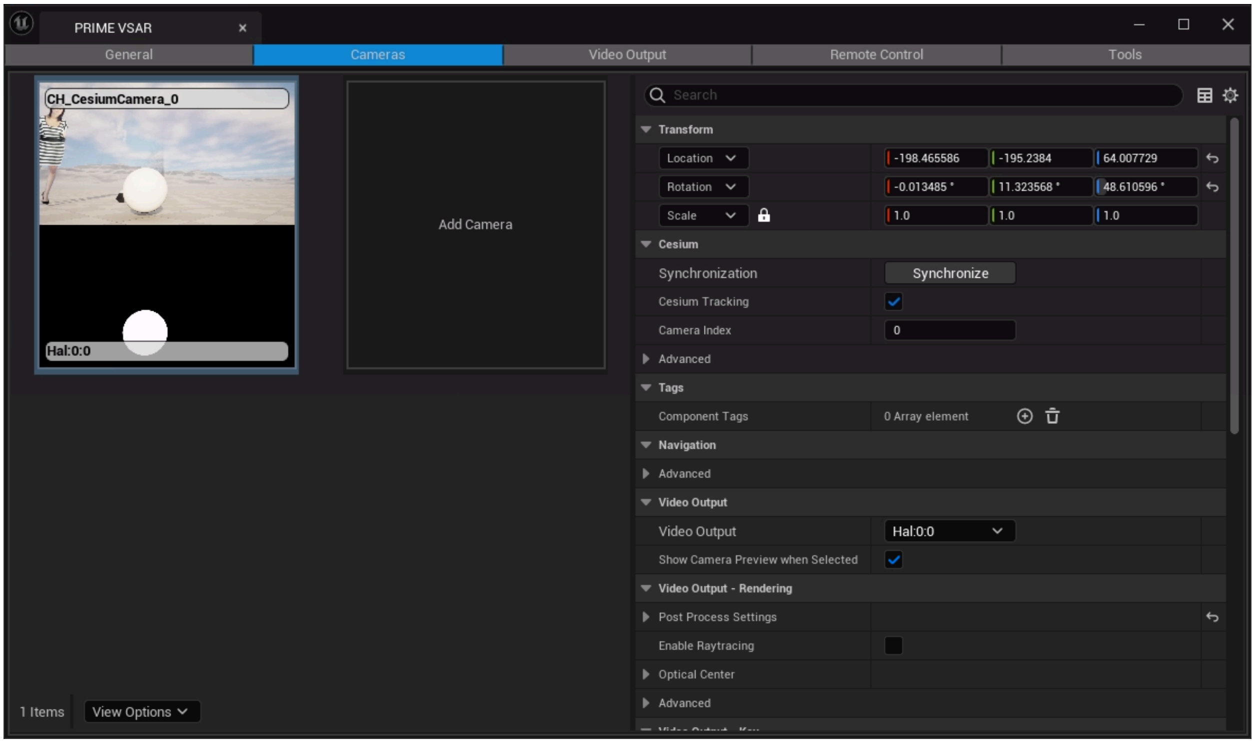


PRIME VSAR Version: 2.0.0 (2.0.0.2692)
Revision: d67dc2cd65e6c1c268f838335d805b0b0649d61f
Unreal Engine Version: 5.3.2
2021 Chyron All Rights reserved

VSAR Documentation

Cameras

For managing ***PRIME VSAR*** cameras.



Transform
Location -198.465586 -195.2384 64.007729
Rotation -0.013485 ° 11.323568 ° 48.610596 °
Scale 1.0 1.0 1.0

Cesium
Synchronization: Synchronize
Cesium Tracking ☑
Camera Index 0
Advanced

Tags
Component Tags 0 Array element

Navigation
Advanced

Video Output
Video Output Hal:0:0
Show Camera Preview when Selected ☑

Video Output - Rendering
Post Process Settings
Enable Raytracing ☐
Optical Center
Advanced

A ***PRIME VSAR*** camera, by default named *CH_CesiumCamera* (at first glance and for almost all applications) have 2 purposes:

- Generate an output onto **Matrox** playout ports through HAL or through **NDI (Not supported in 2.0.0 release)**
- Manage the communication with **Cesium** and thus the tracking data

In the above image, there is 1 available camera.

To create a new one, click the **Add Camera** button.

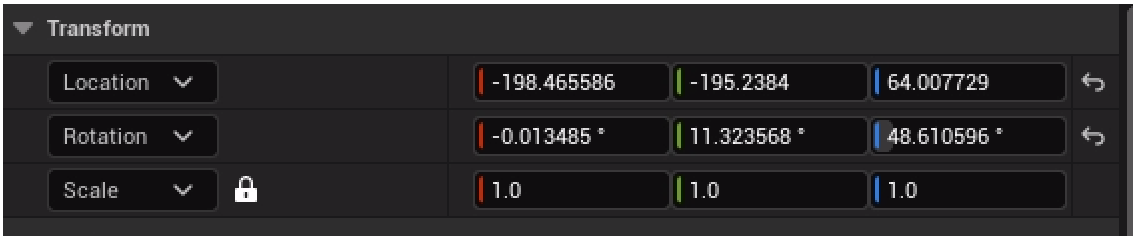
⚠️ Maximum number of Cesium Cameras is 12

To set/get (and reveal) the video parameters of the camera, just select (click on) it.

Note: after upgrading ***PRIME VSAR*** version, in some cases, opening an older ***PRIME VSAR*** .uproject with a previously created ***PRIME VSAR*** camera may show inconsistencies. It may be useful to delete and recreate it.

The right side of the panel shows (a subset of) the Details panel of the selected camera, also available in ***Unreal***'s standard **Details** one.

Transform



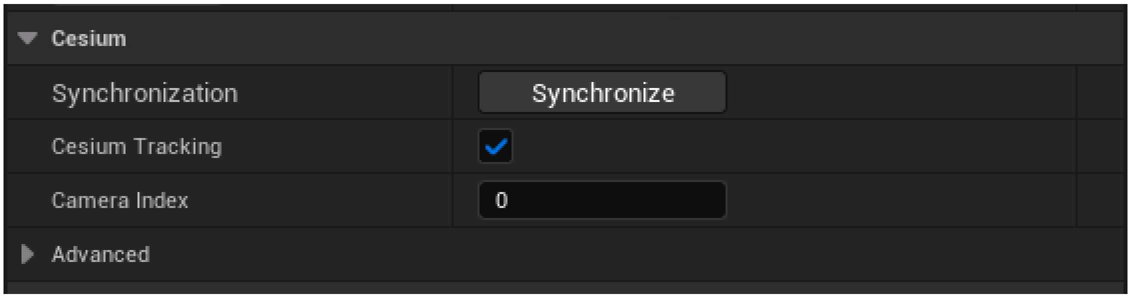
Transform

Location	-198.465586	-195.2384	64.007729
Rotation	-0.013485 °	11.323568 °	48.610596 °
Scale	1.0	1.0	1.0

This shows the coordinates of the ***PRIME VSAR*** camera in space as for any ***UE** Actor*.

If Cesium is *Activated* (see below) these values come straight from ***Cesium*** data, and thus cannot be edited.

Cesium



Cesium

Synchronization	Synchronize
Cesium Tracking	☑
Camera Index	0

Advanced

Synchronization

The Synchronize button allows for forcing immediate re-synchronisation with incoming *Cesium* data. Useful if you notice some introduced delay between real and virtual.

⚠️Unexpected delay may happen after dropping frames, and/or if you do some editing manipulations inside the *Editor*.

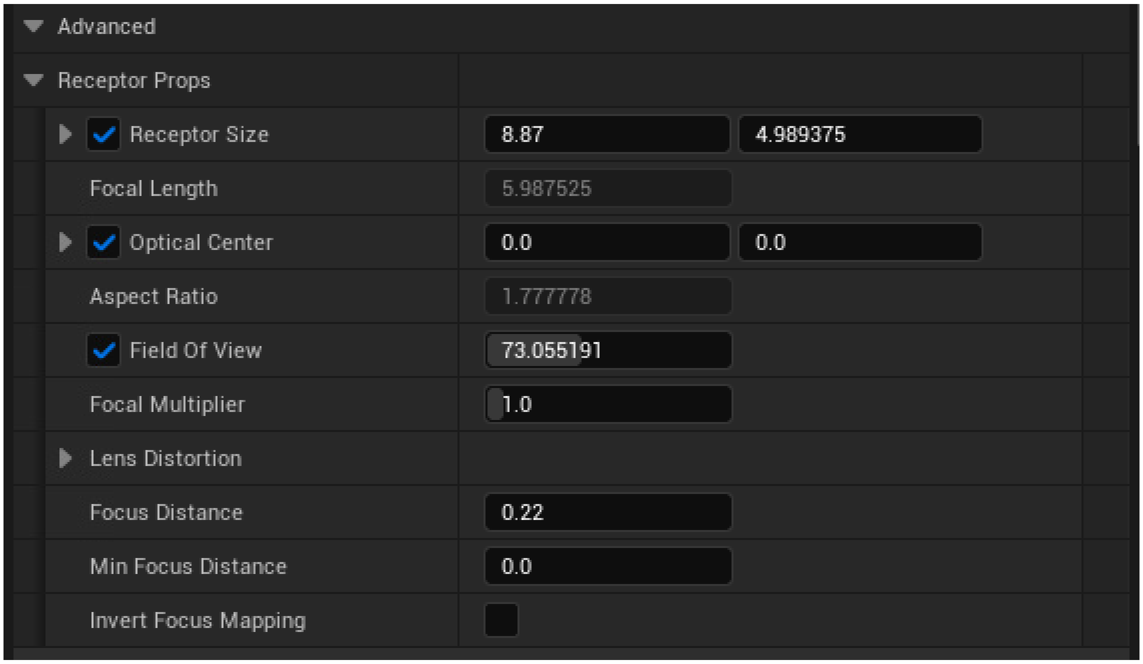
Cesium Activated

If unchecked this camera won't follow ***Cesium*** information. Should be checked for normal operation (camera tracking scenario).

Camera Index

Cesium software can handle multiple cameras. This figure selects which camera we want to use. Index starts at 0. This number relates to the Rig in **Cesium** (for example if we have two rigs in Cesium and we want to use the second one we would input 1here)

Advanced Display (rev arrow ▼)



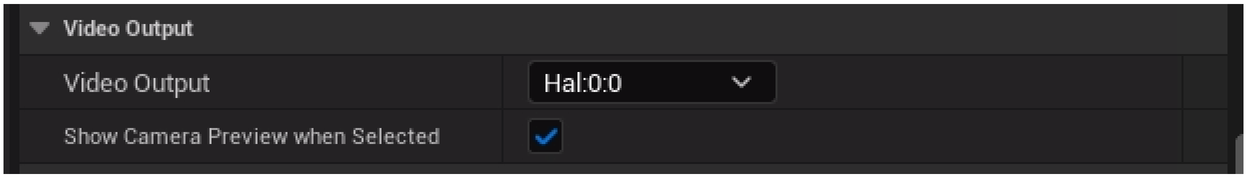
▾ Advanced

▾ Receptor Props

▸ ☑ Receptor Size	8.87	4.989375
Focal Length	5.987525	
▸ ☑ Optical Center	0.0	0.0
Aspect Ratio	1.777778	
☑ Field Of View	73.055191	
Focal Multiplier	1.0	
▸ Lens Distortion		
Focus Distance	0.22	
Min Focus Distance	0.0	
Invert Focus Mapping	□	

These values display the current value of the cesium Camera. This feature can be used to check that you actually receive data from ***Cesium***. If you move the camera these values should change (this can be verified eg. by using simulation driver in Cesium).

Video Output



▾ Video Output

Video Output	Hal:0:0 ⌄
Show Camera Preview when Selected	☑

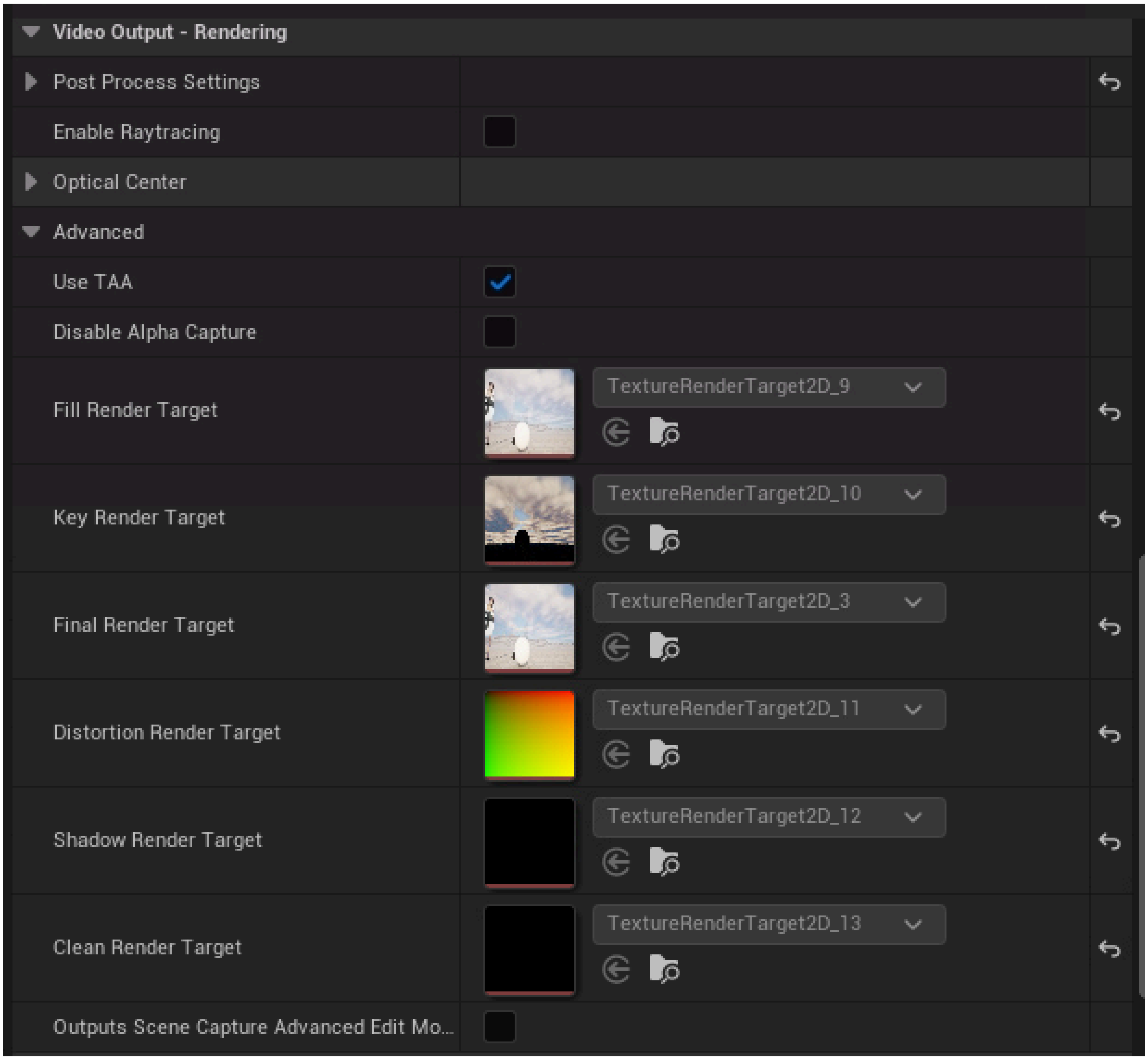
To select which available output will be used for this camera.

The dropdown list is contextual to what outputs are available in the session.

See below regarding Video Output tab.

Defaults to *Hal:0:0* (for a standard workstation with one ***Matrox*** board installed).

Video Output - Rendering



- Video Output - Rendering
  - Post Process Settings
  - Enable Raytracing □
  - Optical Center
  - Advanced
    - Use TAA ☑
    - Disable Alpha Capture □
    - Fill Render Target: TextureRenderTarget2D_9
    - Key Render Target: TextureRenderTarget2D_10
    - Final Render Target: TextureRenderTarget2D_3
    - Distortion Render Target: TextureRenderTarget2D_11
    - Shadow Render Target: TextureRenderTarget2D_12
    - Clean Render Target: TextureRenderTarget2D_13
    - Outputs Scene Capture Advanced Edit Mo... □

Post Process Settings

A list of *Unreal* camera's rendering parameters.

Optical Center

When it comes to calibration/registration/targeting of the real camera, you need to check the optical center and target positions. Optical center display parameters are set here:

- **Optical Center Show**

  Displays or not the optical center. During normal operations it should be unchecked.
- **Optical Center Color**

  Color of the center lines

- **Optical Center Thickness**
  Thickness of the center lines

Note: this displays a cross on the whole screen, both in RGB and Alpha channels, so that it's always in front of everything when using any keyer with external matte feature.

Advanced settings (revealed when clicking on the down arrow ↨).

Use TAA

Checked by default. For using Temporal Anti-Aliasing on the output. This setting smooths out jagged edges of objects in the scene.

⚠️**For this to work TAA needs to be also selected in the project setting as the Anti-Aliasing mode.**

Disable Alpha Capture

To disable management of alpha channel.
Could help increase performances if no alpha is needed. (for most use-cases by very little)

Enable Distortion

To be checked when *Cesium* data is providing lens distortion info.

Then a list of ***Unreal*** *Render Targets* is used for generating the output.
Each of them may be visualized in *Unreal*'s texture editor by double clicking on it.

Fill Render Target

RGB components for the rendered video output.

Key Render Target

Alpha components for the rendered video output.

Final Render Target

Combination of both above.

Distortion Render Target

The texture used for rendering distortions.

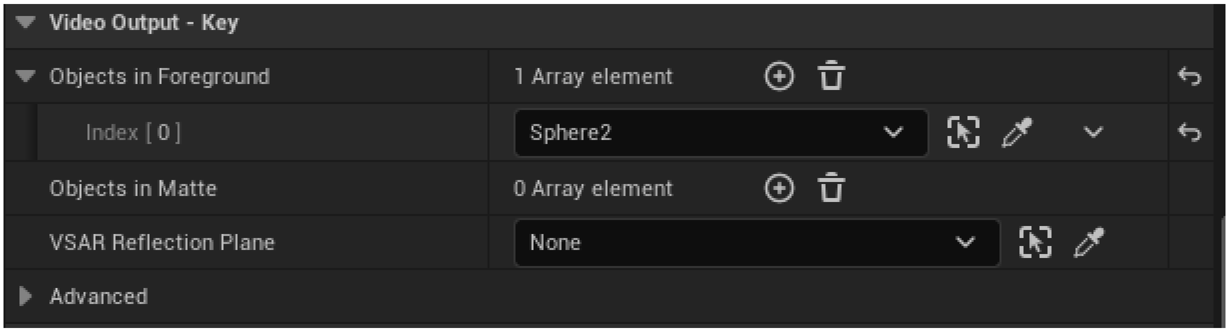
Shadow Render Target

The texture used to render shadows and reflections in Augmented Reality mode.

Outputs Scene Capture Advanced Edit Mode

Enabling this will display the Fill_SceneCaptureComponent and Key_SceneCaptureComponent allowing to edit more advanced parameters.

Video Output - Key



Video Output - Key

Objects in Foreground: 1 Array element

Index [0]: Sphere2

Objects in Matte: 0 Array element

VSAR Reflection Plane: None

Advanced

Objects in Foreground

Set the list of Actors that are going to be displayed in the Fill and Key signals. Usually used to have objects in front of the talent.

Should usually be used for the *Targets* (see below).

Objects in Matte

Set the list of Actors that are going to be displayed in Key signal only. Usually used to hide parts of the real sets (ceilings, walls,...)

Advanced settings (revealed when clicking on the down arrow ↕).

Force Alpha

Forces a full white alpha output.

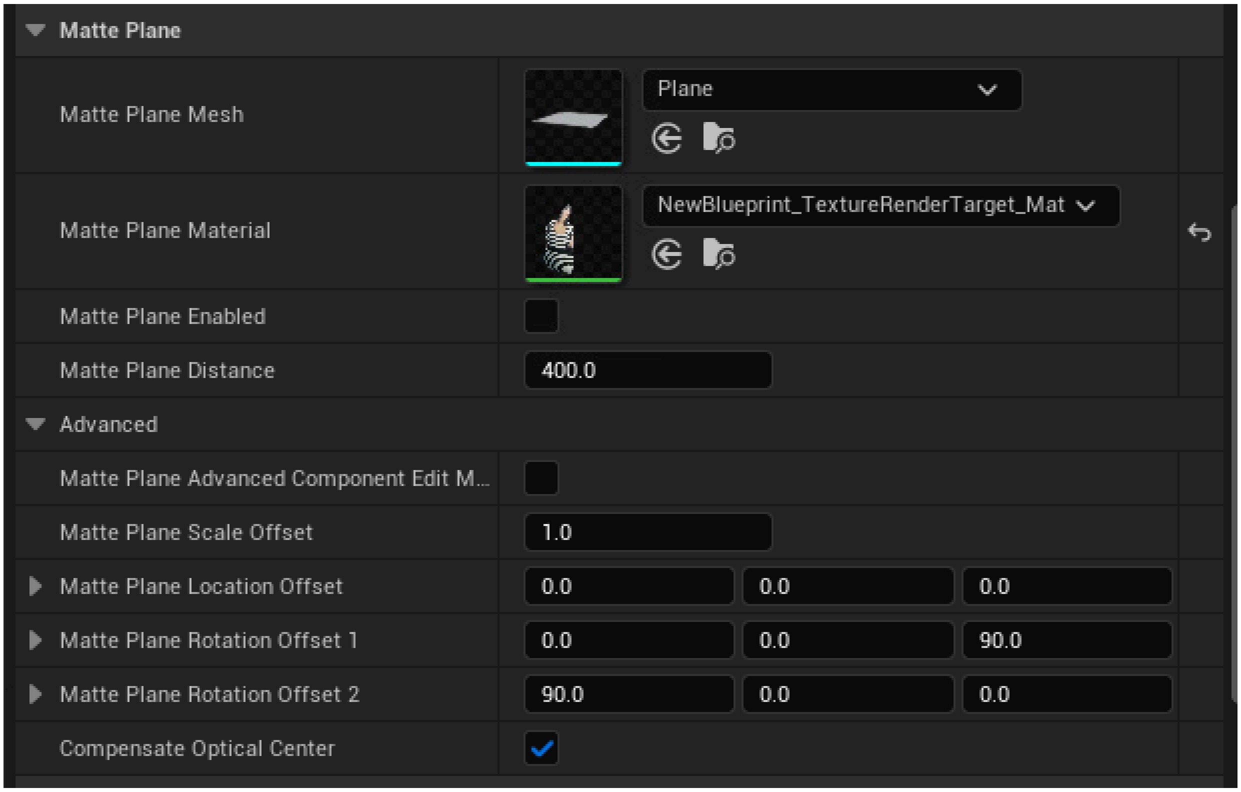
Invert Alpha

Inverts the alpha.

Include Attached Actors

Applies to the actors in the Foreground/Matte lists and also all of their children.

Matte Plane



▼ Matte Plane

Matte Plane Mesh	Plane
Matte Plane Material	NewBlueprint_TextureRenderTarget_Mat
Matte Plane Enabled	□
Matte Plane Distance	400.0

▼ Advanced

Matte Plane Advanced Component Edit M...	□		
Matte Plane Scale Offset	1.0		
▶ Matte Plane Location Offset	0.0	0.0	0.0
▶ Matte Plane Rotation Offset 1	0.0	0.0	90.0
▶ Matte Plane Rotation Offset 2	90.0	0.0	0.0
Compensate Optical Center	☑		

The **Matte Plane** is a surface always facing the camera, and exactly fulfilling the ***PRIME VSAR*** camera field of view. If the camera moves or rotates it will follow it (it is attached to the camera).

Its purpose is to immerse the Video In (talent) into the 3D world, in front or behind other compositions

Matte Plane Mesh

The actual geometry used for it. A plane by default.

Matte Plane Material

The material applied on the object. Should be using the Video Input as media source.

Matte Plane Enabled

To use it (or not).

Matte Plane Distance

Distance of the object to the camera in *UE* units (centimeters).

📝 Size is aligned to the camera projection making it look like nothing changed when adjusting Plane Distance.

Matte Plane Advanced Component Edit Mode

When checked, *UE*'s Details panel shows up all the parameters of the *PRIME VSAR* camera, otherwise, only a subset is displayed.

Matte Plane Scale Offset

Adds a scale bias to the Plane. The default value is 1 and makes the plane exactly fill the camera viewport.

Matte Plane Location Offset

Adds a translation bias to the Plane. The default value is 0. and makes the plane exactly fill the camera viewport.

Matte Plane Rotation Offset 1

Adds a rotation bias to the Plane, prior to the translation. The default value is 0. and makes the plane exactly fill the camera viewport.

Matte Plane Rotation Offset 2

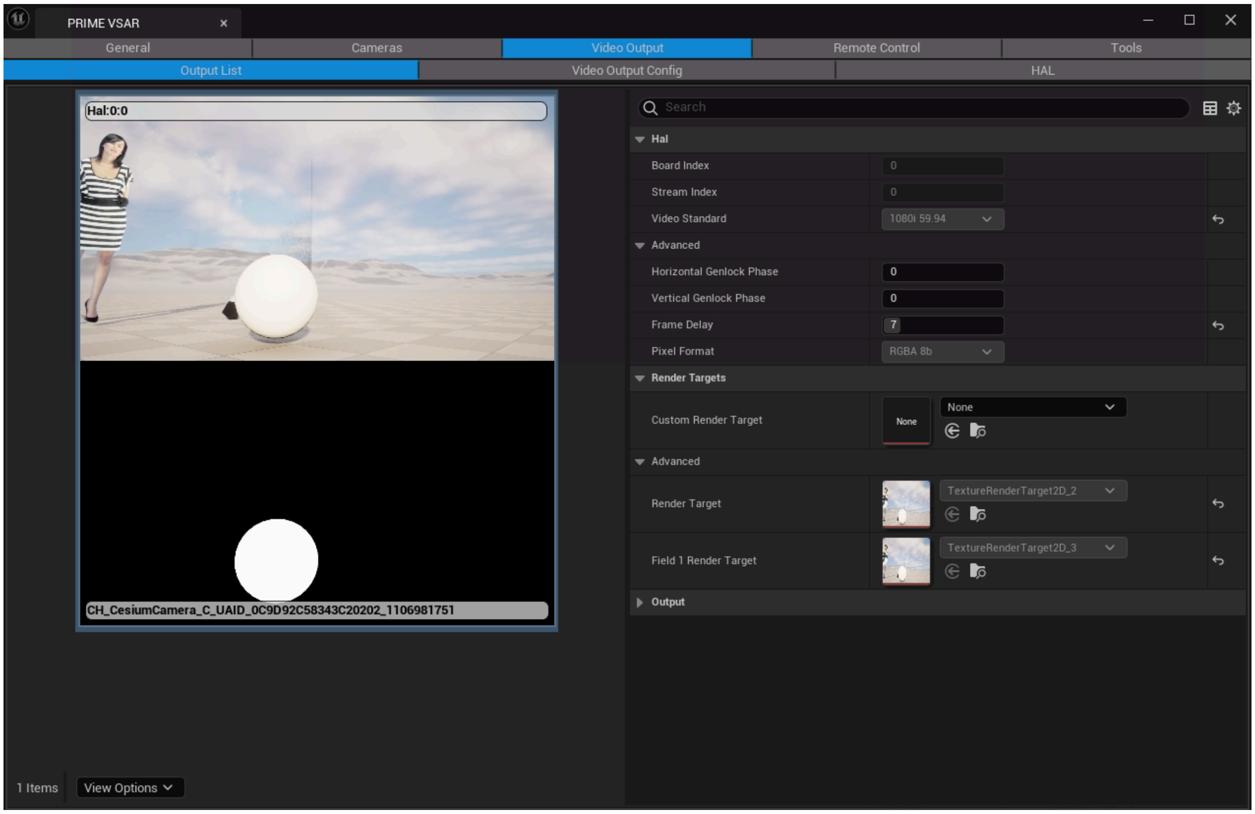
Adds a rotation bias to the Plane, after the translation. The default value is 0. and makes the plane exactly fill the camera viewport.

Compensate Optical Center

Apply or do not apply the Cesium optical center to the plane.

Video Output

For managing all possible Outputs:



Hal
Board Index 0
Stream Index 0
Video Standard 1080i 59.94
Advanced
Horizontal Genlock Phase 0
Vertical Genlock Phase 0
Frame Delay 7
Pixel Format RGBA 8b
Render Targets
Custom Render Target None None
Advanced
Render Target TextureRenderTarget2D_2
Field 1 Render Target TextureRenderTarget2D_3
Output

This Tab displays the following sub-tabs:

- Outputs
- Video Output Configuration
- HAL

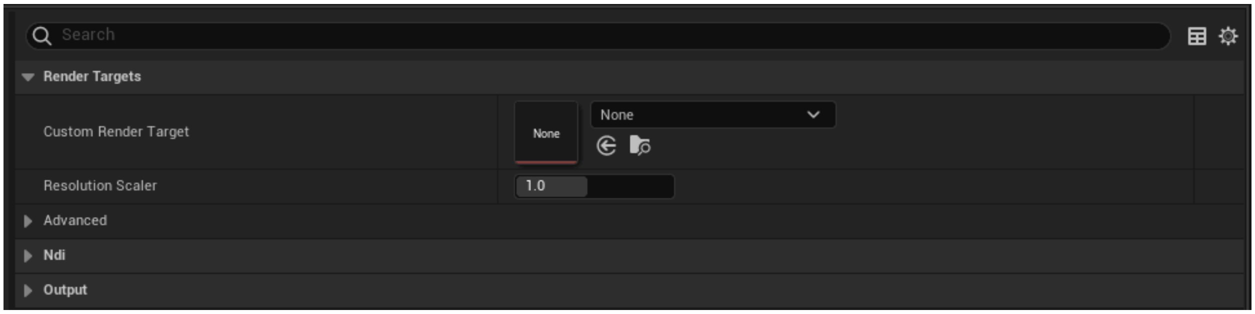
Outputs

This tab details the parameters related to the video outputs available for any *CH_CesiumCamera* (see above).

Each extra output (*NDI*) may be easily deleted using the “X” button on the top right corner.

⚠️(**NDI is Not supported in 2.0.0 release**)

The right side panel displays the different parameters related to the selected output, as Details does.



Search

Render Targets

Custom Render Target: None

Resolution Scaler: 1.0

Advanced

Ndi

Output

Custom Render Target

when inserted render target, this render target will be used as Output instead of the hardware.

Resolution Scaler

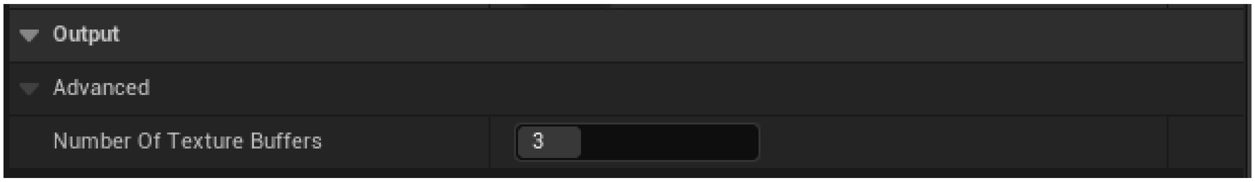
increases or decreases the rendered resolution then it gets coveted to output resolution. For example, when we have 1080p resolution and we set this value to 2 we can use this as a custom super sampling anti aliasing, as it will render in 4k and then be scaled to 1080p.

⚠️for interlaced mode we recommend using only scale larger than 1.5 or TAA but not both at the same time.

Render Targets

Render Targets used for the selected video Output. Also called ‘Render Texture’ in the Game industry, it is used to make an intermediate calculation as a Texture or surface, e.g. to handle the Fill, the Key or any extra composition.

Output



Output

Advanced

Number Of Texture Buffers: 3

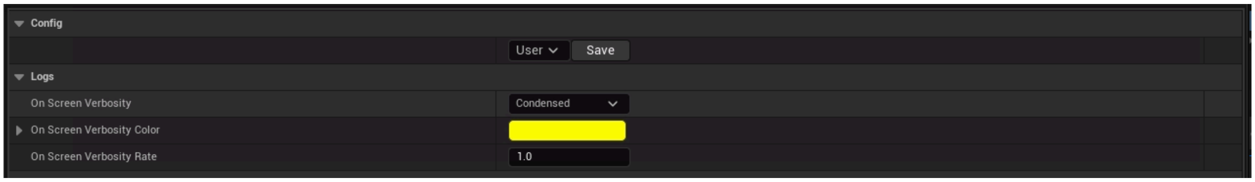
Number of Texture Buffers (from ***Unreal*** ToolTip)

Number of textures used to transfer the texture from the GPU to the system memory.

A smaller number is most likely to block the GPU (wait for the transfer to complete).

A bigger number is most likely to increase latency.

Video Output Configuration



▾ Config

User ▾ Save

▾ Logs

On Screen Verbosity: Condensed ▾

▸ On Screen Verbosity Color: 🟨

On Screen Verbosity Rate: 1.0

Mostly used for eventually generating new NDI outputs (via the “+ NDI Out” button).

⚠️(**NDI is Not supported in 2.0.0 release**)

Config

Advanced: Entry for managing the video configuration file. Configuration files may be User or Project -wise. This file should not be confused with Hal.xml, which is the HAL configuration file.

Default is User-mode.

The user file is %LOCALAPPDATA%/ChyronHego/Fresh/VideoOutputConfig.json\

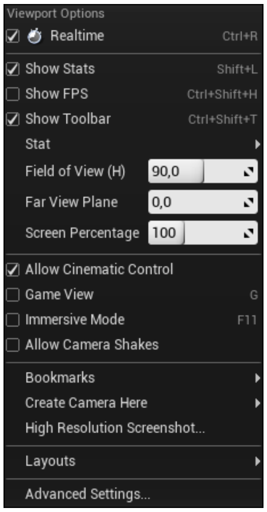
The project file is displayed, as a tooltip, when you hover the mouse on top of the drop down box.

Logs

Entry for managing **On Screen Verbosity**. Changes the level of detail of HUD messages in the editor window.

Helpful for monitoring and analyzing real-time efficiency.

When set to *Condensed* or *Verbose*, the viewer will display subsequent lines informing the state of the calculation, as long as **Viewer**’s parameters have **Show Stats** checked.



Viewport Options

☑ Realtime Ctrl+R

☑ Show Stats Shift+L

□ Show FPS Ctrl+Shift+H

☑ Show Toolbar Ctrl+Shift+T

Stat ▸

Field of View (H) 90,0

Far View Plane 0,0

Screen Percentage 100

☑ Allow Cinematic Control

□ Game View G

□ Immersive Mode F11

□ Allow Camera Shakes

Bookmarks ▸

Create Camera Here ▸

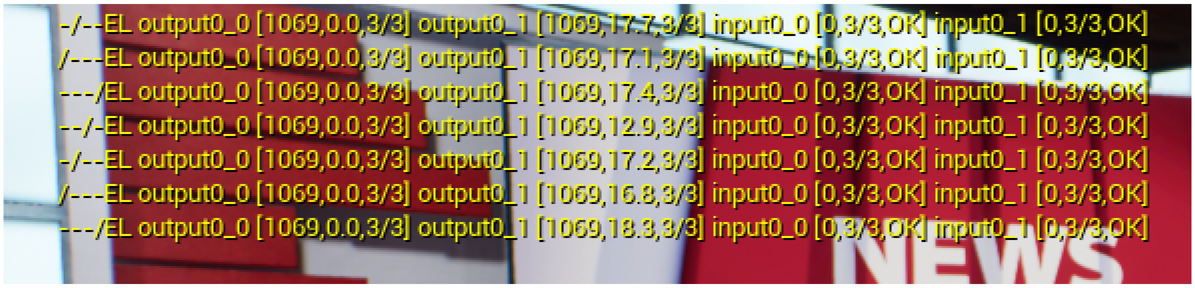
High Resolution Screenshot...

Layouts ▸

Advanced Settings...

The time rate between each line is driven by the **On Screen Verbosity Rate** (in seconds).

For example in Condensed mode, you may read:



```
-/--EL output0_0 [1069,0.0,3/3] output0_1 [1069,17.7,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
/---EL output0_0 [1069,0.0,3/3] output0_1 [1069,17.1,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
---/EL output0_0 [1069,0.0,3/3] output0_1 [1069,17.4,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
--/-EL output0_0 [1069,0.0,3/3] output0_1 [1069,12.9,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
-/--EL output0_0 [1069,0.0,3/3] output0_1 [1069,17.2,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
/---EL output0_0 [1069,0.0,3/3] output0_1 [1069,16.8,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
---/EL output0_0 [1069,0.0,3/3] output0_1 [1069,18.3,3/3] input0_0 [0,3/3,OK] input0_1 [0,3/3,OK]
```

Meaning:

E : externally genlock required

L: genlock is locked.

output0_0: logs for the first output

1069: number of drop frames since PRIME VSAR started

0.0: idle time (milliseconds)

3/3: output fifo fill. In normal operation this should be N/N.

output0_1: logs for the second output

1069: number of drop frames since PRIME VSAR started

18.3: idle time (milliseconds). Means that there are 18.3 milliseconds left when rendering a frame. A frame usually last 40ms, or 33.6 ms. So 18.3 is a good score. Note that only the last output will contain the relevant idle time.

3/3: output fifo fill. In normal operation this should be N/N.

input0_0: logs for the first input

0: number of dropped frames since PRIME VSAR started

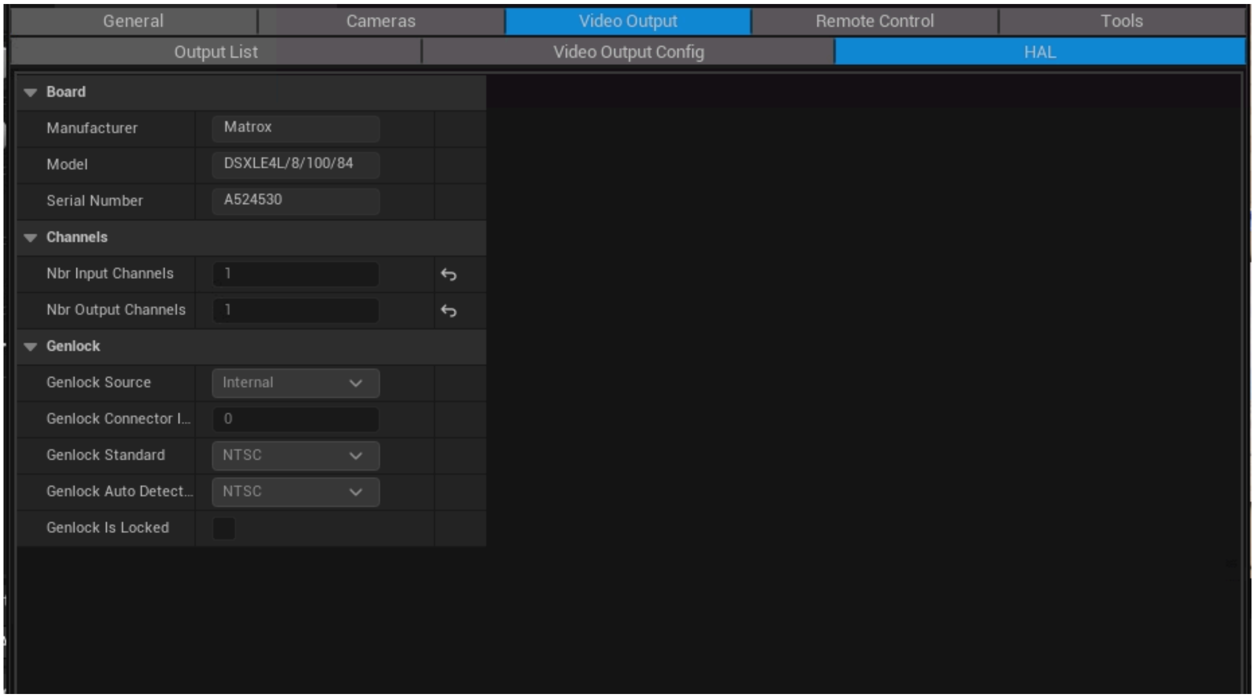
3/3: input fifo fill. In normal operation this should be 0|1/N. Having 3/3 means that the video is not currently used in the scene

OK: video is present on the incoming BNC.

input0_1: logs for the second input. Same fields as upper.

HAL

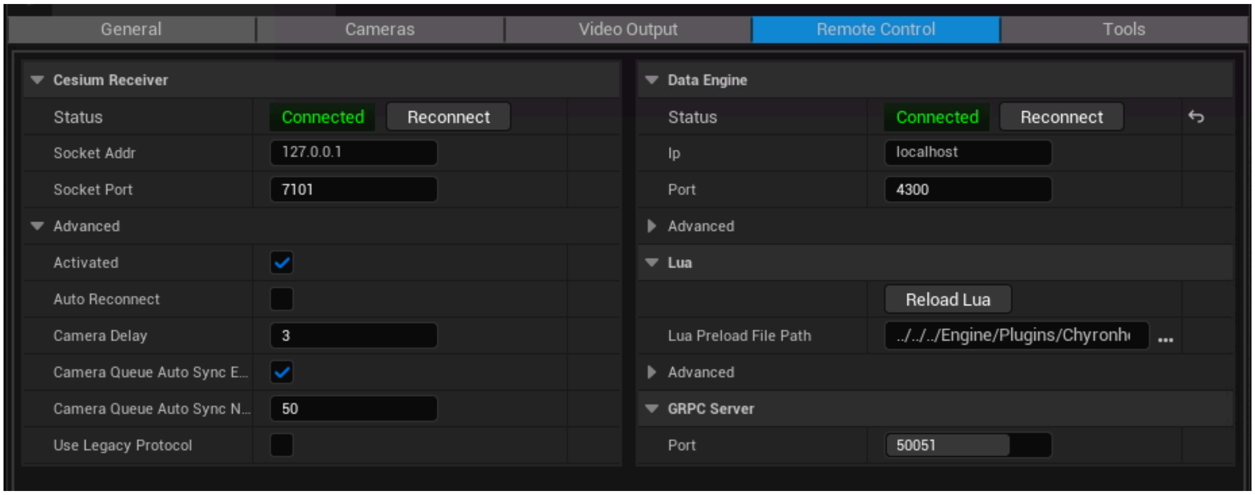
A read only entry displaying how ***HAL*** is configured. This configuration can be changed by editing the HAL.xml file and restarting PRIME VSAR.



Board	
Manufacturer	Matrox
Model	DSXLE4L/8/100/84
Serial Number	A524530
Channels	
Nbr Input Channels	1
Nbr Output Channels	1
Genlock	
Genlock Source	Internal
Genlock Connector I...	0
Genlock Standard	NTSC
Genlock Auto Detect...	NTSC
Genlock Is Locked	□

Remote Control

For managing communication with external utilities, mainly *Cesium* (tracking), *DataEngine* (*CAMIO* and remote *Lua* triggering) and *Lua* scripting.



Cesium Receiver
- Status: Connected [Reconnect]
- Socket Addr: 127.0.0.1
- Socket Port: 7101
- Advanced
  - Activated: ☑
  - Auto Reconnect: ☐
  - Camera Delay: 3
  - Camera Queue Auto Sync E...: ☑
  - Camera Queue Auto Sync N...: 50
  - Use Legacy Protocol: ☐

Data Engine
- Status: Connected [Reconnect]
- Ip: localhost
- Port: 4300
- Advanced

Lua
- [Reload Lua]
- Lua Preload File Path: ../../../Engine/Plugins/Chyronh...
- Advanced

GRPC Server
- Port: 50051

Cesium Receiver

Status

Connected is displayed in Green when the communication occurs with ***Cesium*** (this does not necessarily mean that ***PRIME VSAR*** is receiving proper data, see below Testing). A Red label is displayed if *Cesium* is not connected (running).

The **Reconnect** button helps forcing reconnection if need be (network interruption, …)

Note:If *Auto Reconnect* is checked (see below), then reconnection will happen automatically.

Socket Addr

The name/ip of the machine where *Cesium* is running

Socket Port

The port onto which *Cesium* delivers data.

Defaults to 7100 for Legacy protocol. (in VSAR “Use Legacy Protocol” under VSAR Config -> Remote Control -> Cesium Receiver -> Advanced, we recommend to not have this checked and use the new protocol instead)

Defaults to 7101 for the new protocol.

Extra advanced Parameters (revealed when clicking on the down arrow ↕).

Default values should cover most of the usual cases:

Activated

Should be checked in normal operation. If unchecked, cesium camera data won't be read anymore. This can be useful for outputting video for non-cesium cameras.

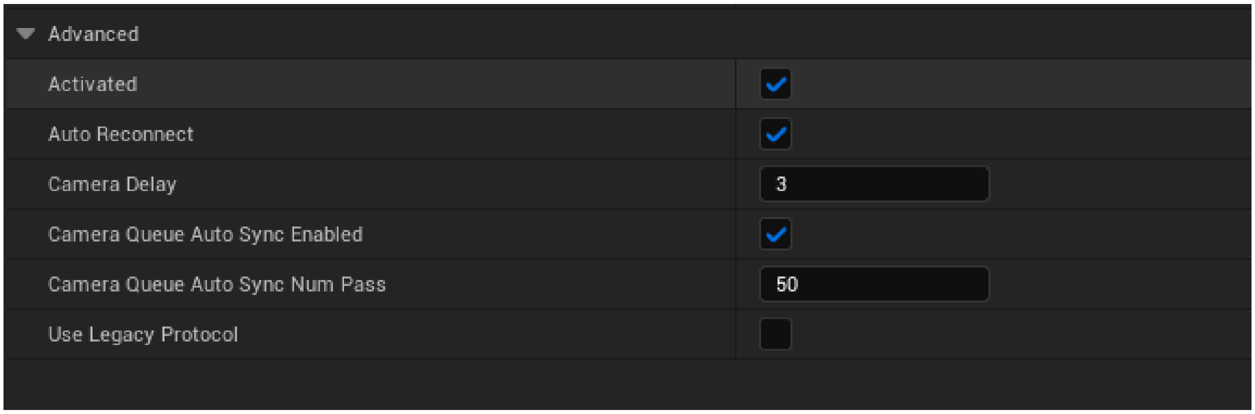
Auto Reconnect

If checked, *Cesium* will automatically be reconnected when a disconnection happens. In normal operation, it should be checked.

Camera Delay

Queue is used to smooth out network data transmission. Incoming data from cesium is appended to the Queue(s) and then used by an individual camera on tick. If the rate of camera data is not genlocked to *PRIME VSAR* video frame rate, then at some moment a desynchronisation of the queue may happen. VSAR automatically tries to sync the Queue to be as close to the aimed Camera Delay while the camera is not moving.

- **CameraDelay** - how many frames of delay should be between Cesium and VSAR. This also determines the Queue size, which is CameraDelay times two plus one, In example of delay 3 the Queue size will be 7.
- **CameraQueueAutoSyncEnabled** - enables auto Synchronization functionality where the frame counter jumps in time to current Queue size - CameraQueueSyncOffset, after camera synchronization has happened camera may jump in space.
- **CameraQueueAutoSyncNumPass** - number of frames that are desynced before Synchronization is triggered (**CameraQueueAutoSyncEnabled** has to be enabled)



▼ Advanced

Activated	☑
Auto Reconnect	☑
Camera Delay	3
Camera Queue Auto Sync Enabled	☑
Camera Queue Auto Sync Num Pass	50
Use Legacy Protocol	☐

Data Engine

Remote triggering is handled via ***CH Data Engine***.

Data Engine remote control is mainly used by **Chyron Panels (*LAP)*** and ***CAMIO*** (among others).

Status

Connected is displayed in green when the connection with *DE* is alive. If it's not the case, click **Reconnect**.

Reconnect

Force the reconnection to the *DataEngine*.

IP

IP address of the machine where the DataEngine is running. 127.0.0.1 stands for localhost.

Port

Port used by the ***DataEngine*** (defaults is 4300)

Extra advanced Parameters (revealed when clicking on the down arrow ↨):

Auto Reconnect

When checked, as soon as the ***DataEngine*** disconnects, it will be automatically reconnected.

Reconnect Delay

Timeout in seconds, before the ***DataEngine*** is declared disconnected.

Lua

Lua Preload File

When PRIME VSAR is started, a Lua initialization is loaded. The default file is: *../../../MtLuaPreload/Main.lua*.

It will load several Lua libraries useful for VSAR. If you need to add your own Lua initialization scripts, most of the time, instead of changing the path of this file, it's better to edit this file and add your own file entries.

Clicking Reload Lua will execute the file content again.

Extra advanced Parameters (revealed when clicking on the down arrow ↨):

Bucket

The *DataEngine* bucket used to communicate with *PRIME VSAR* (Default: *ue4*).

Lua in Key

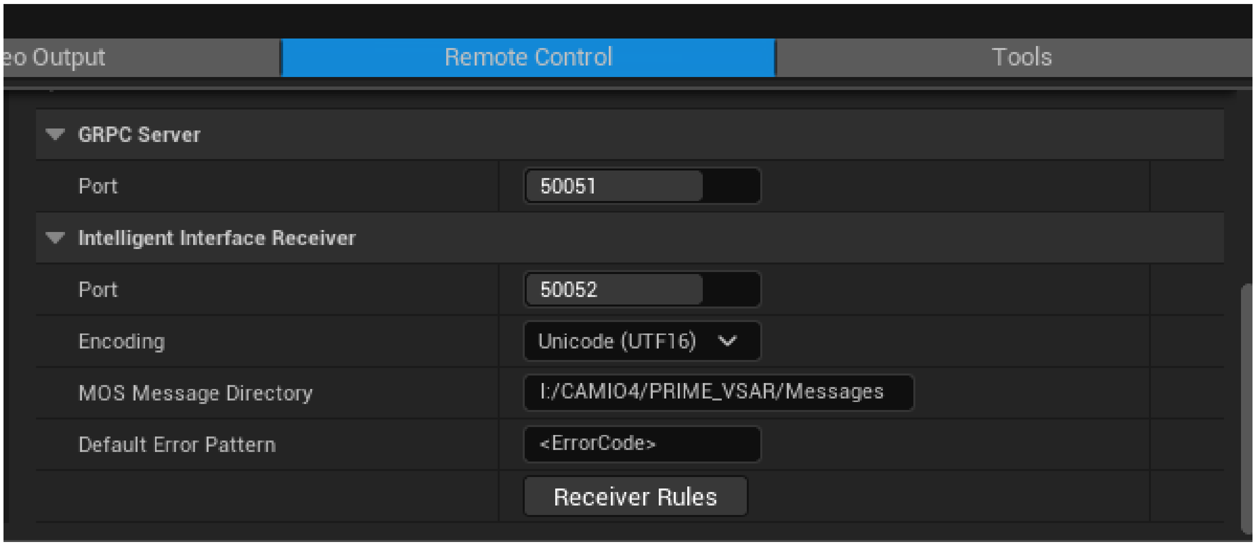
Bucket's key where *PRIME VSAR* should read the Lua commands to execute (Default: *lua_in*)

gRPC Port

sets port on what the gRPC Server is run on, this connection is important for VSAR Controller service.

Intelligent Interface

Intelligent Interface is remote control protocol. Works similar to Prime Intelligent Interface. VSAR receives messages on UDP on the specified **Port**. Received message is decoded by selected **Encoding** format and then matched against command patterns.



GRPC Server

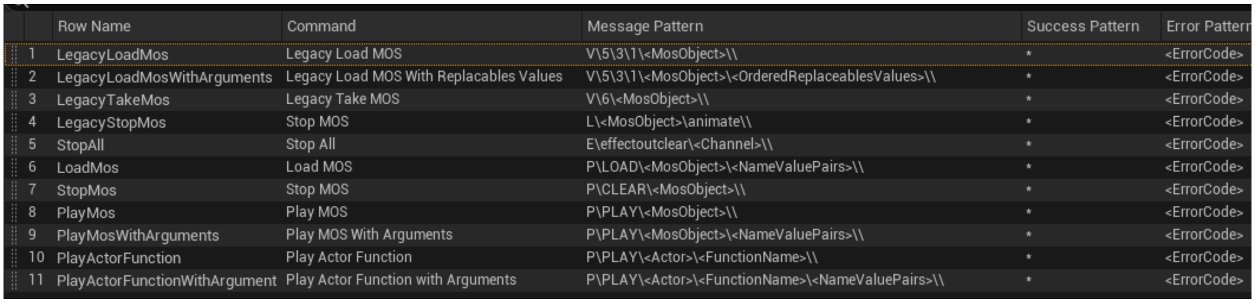
Port	50051

Intelligent Interface Receiver

Port	50052
Encoding	Unicode (UTF16)
MOS Message Directory	I:/CAMIO4/PRIME_VSAR/Messages
Default Error Pattern	<ErrorCode>

MOS Message Directory is path where should be MOS files located and in this directory will VSAR try to find MOS files in case, that message does not contain a full path to MOS but just its ID or basename. This property can be changed using blueprint node **SetMOSMessageDirectory.** When changed via blueprint, property editor in VSAR config will not be updated, but value will be changed. To see this change, close VSAR Config panel and open again. **Default Error Pattern** is used in case no rule will match incoming messages or selected commands in all matched rules are missing some arguments. All of those settings are project specific. During execution of some commands, the opened project can be switched.

By clicking on the **Receiver Rules** button, DataTable asset editor containing rules table is opened. Here, there are predefined rules with their patterns, and user can alter patterns, responses or behaviors.



	Row Name	Command	Message Pattern	Success Pattern	Error Patter
1	LegacyLoadMos	Legacy Load MOS	V\5\3\1\<MosObject>\\	*	<ErrorCode>
2	LegacyLoadMosWithArguments	Legacy Load MOS With Replacables Values	V\5\3\1\<MosObject>\<OrderedReplaceablesValues>\\	*	<ErrorCode>
3	LegacyTakeMos	Legacy Take MOS	V\6\<MosObject>\\	*	<ErrorCode>
4	LegacyStopMos	Stop MOS	L\<MosObject>\animate\\	*	<ErrorCode>
5	StopAll	Stop All	E\effectoutclear\<Channel>\\	*	<ErrorCode>
6	LoadMos	Load MOS	P\LOAD\<MosObject>\<NameValuePairs>\\	*	<ErrorCode>
7	StopMos	Stop MOS	P\CLEAR\<MosObject>\\	*	<ErrorCode>
8	PlayMos	Play MOS	P\PLAY\<MosObject>\\	*	<ErrorCode>
9	PlayMosWithArguments	Play MOS With Arguments	P\PLAY\<MosObject>\<NameValuePairs>\\	*	<ErrorCode>
10	PlayActorFunction	Play Actor Function	P\PLAY\<Actor>\<FunctionName>\\	*	<ErrorCode>
11	PlayActorFunctionWithArgument	Play Actor Function with Arguments	P\PLAY\<Actor>\<FunctionName>\<NameValuePairs>\\	*	<ErrorCode>

Patterns:

Patterns like the Prime Intelligent Interface consist of literals and arguments. Backslash is reserved character for argument delimiting. Arguments are enclosed in <>. Argument names are limited to alphanumeric characters and underscore. Using _ instead of <ArgumentName> will discard argument value. (*P\PLAY_\<FunctionName>*\\ will discard the first argument and use the second argument as FunctionName) Everything else (outside arguments and discard character _) is considered a literal and will be matched case insensitive.

This means that for Pattern: "**TEST::\abracadabra\<ArgOne>_**" following messages would match:

TEST::\abracadabra\1\I

teSt::\AbracadabrA\with space\5

test::\Abracadabra\with:comma\5

test::\Abracadabra\with_underscore\5

For pattern: "**SHOW\<key>:<val>**"

show\aaa:bbb\ (key will be "aaa" and val will be "bbb")

Special arguments:

<OrderedReplaceablesValues> = backslash delimited arguments

<NameValuePairs> = \name1\value1\name2\value2\ ….

<Values> = \name1:value1\name2:value2\ ….

Special Arguments for reply pattern:

<Command> = whole received message

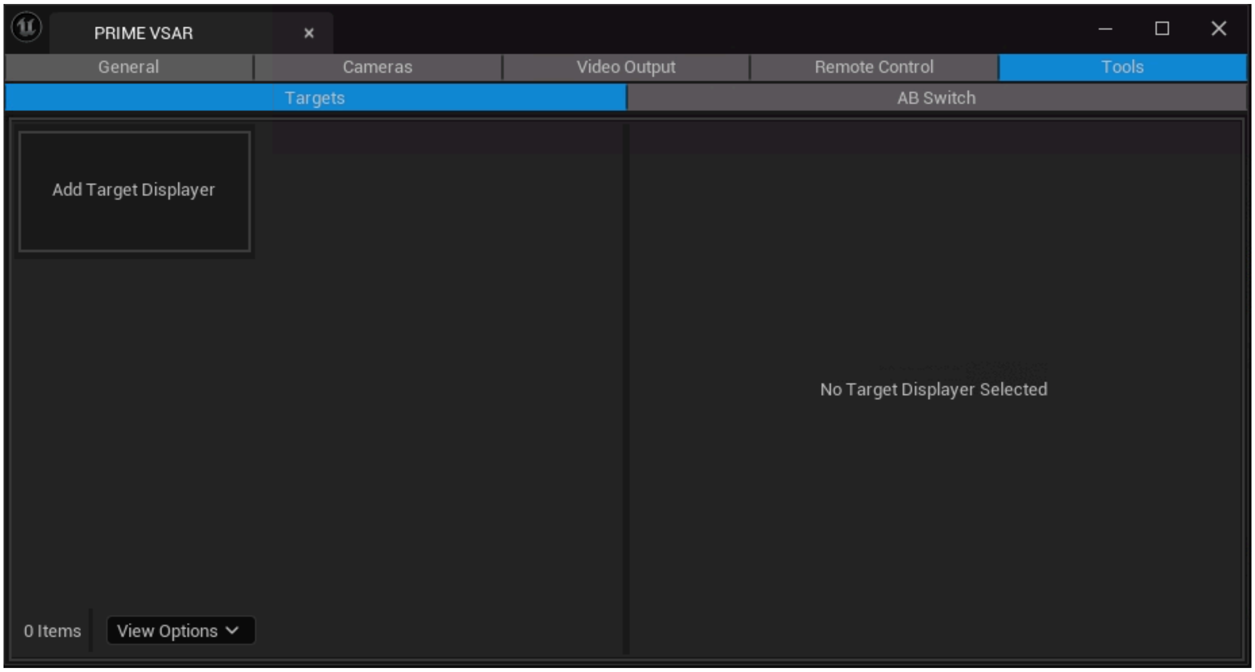
<ErrorCode> = 4 digit hexadecimal error code (just in Error Reply)

<ErrorMsg> = Verbose error message (just in Error Reply)

Command	Needed Arguments	Behaviour
Legacy Load MOS	MosObject	Loads MOS object, replace replaceables, call Cue
Legacy Load MOS With Replaceables Values	MosObject, OrderedReplaceablesValues	Loads MOS object, replace replaceables (values from OrderedReplaceablesValues override MOS values, order is determined by MOS file), call Cue
Legacy Take MOS	MosObject	Loads MOS object, call Take (no replaceable is affected)
Stop MOS	MosObject	Loads MOS, call Stop
Stop All		Iterate trough all CamioTemplates and call Stop on each
Load MOS	MosObject, NameValuePairs	Loads MOS object, replace replaceables (values from NameValuePairs override MOS values, values are replaced by name of pair - order is not important), call Cue
Play MOS	MosObject	Loads MOS object, replace replaceables, call Take
Play MOS With Arguments	MosObject, NameValuePairs	Loads MOS object, replace replaceables (values from NameValuePairs override MOS values, values are replaced by name of pair - order is not important), call Take
PlayActorFunction	Actor, FunctionName	Find actors by name, if not able, find actors by label (regular expression matching), call function / invoke event
Play Actor Function with Arguments	Actor, FunctionName, NameValuePairs	Find actors by name, if not able, find actors by label (regular expression matching), call function / invoke event

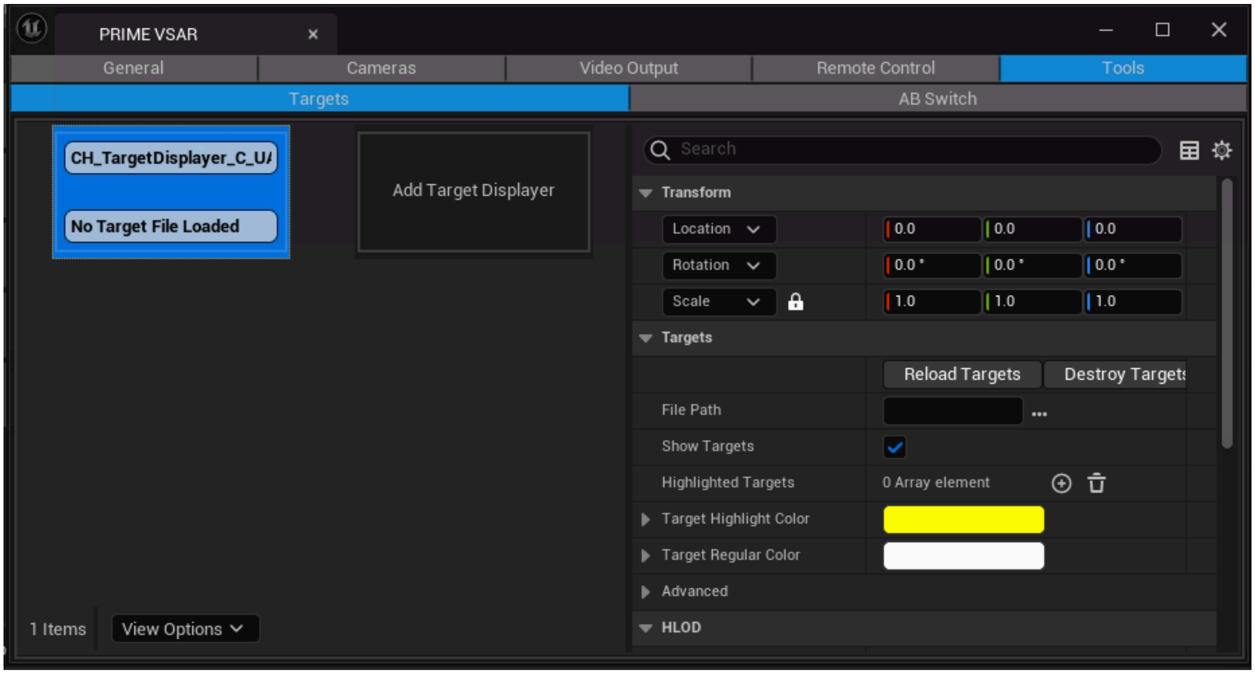
Tools

A tab for managing specific dedicated tools like the *Targets* and *ABSwitch* features.



No Target Displayer Selected

Targets



Location 0.0 0.0 0.0

Rotation 0.0 ° 0.0 ° 0.0 °

Scale 1.0 1.0 1.0

File Path

Show Targets ☑

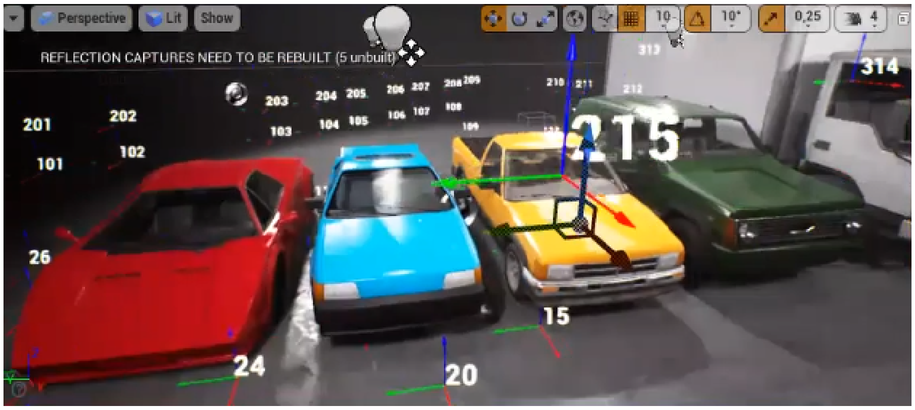
Highlighted Targets 0 Array element

Target Highlight Color 🟨

Target Regular Color ⬜

Used in Tracking mode, Targets are geometrical references (“dummies”) that should reflect physical (XYZ) real positions in your studio. They’re parented to a specific actor usually named *CH_TargetDisplayer*.

Their main purpose is to have in the 3D (virtual) environment references to the real world, either for calibrating/calibration checking or virtual object positioning.



If there are no Target Displayer objects created yet, click Add Target Displayer to create a new one.

Then, load the .tgt file by either entering the file path or clicking the "..." button, and press Reload Targets.

As for Cameras and Video Output tabs, the right side of the panel shows the Details of the selected element.

Transform: allows to globally translate, rotate, scale the targets. Note that in normal operation (ie to have a perfect match between *Cesium* and *PRIME VSAR*), the Transform should be set to Identity; that is: *"Location=0,0,0", "Rotation=0,0,0", "Scale="1,1,1"*.

Reload Targets: deletes and reload the targets based on the specified *File Path*.

Destroy Targets deletes all the targets contained in this target displayer object.

File Path: specifies the path to the target file (.tgt) to display. Usually, this file is the same as the one used in *Cesium*.

Show Targets: if unchecked the targets won't be displayed in the scene. In production, it should be unchecked.

Adding targets to Highlighted Targets will enhance display of the selected targets. You can control their color with Target Highlight Color, and control the color of other targets with Target Regular Color.

It is a good practice to have the targets displayed in foreground (see Video Output - Key above).

AB Switch

AB Switch is a tool to feature texture transitions mapped on a 3D object (for instance a virtual screen in your scenes where we expect transitions to happen).

It should behave close to standard A/B (or Program/Preview) transition effects in a mixer.

See ABSwitch chapter in Prime VSAR UserGuide for usage.

TESTING

Test Prime VSAR / Cesium connection

1) Get in Config Panel, ***Cesium*** and check that ***Cesium*** is Connected in green.
2) Check that Activated is checked.
3) Check that the corresponding ***Cesium*** Camera has:
    a) ***Cesium* Activated** checked.
    b) Correct ***Cesium*** Camera Index (usually 0).
4) Do a zoom movement and check that Field of View is moving accordingly.
5) Select the camera in the “World Outliner”, get in **Details** panel and check that camera position and orientation are changing while you are moving the real camera.

Deeper testing can include:

1) IP/port configuration of ***Cesium*** and ***PRIME VSAR***.
2) Check that ***Cesium*** camera in ***Cesium*** is *Online* and is correctly reporting data (see Cesium Documentation).

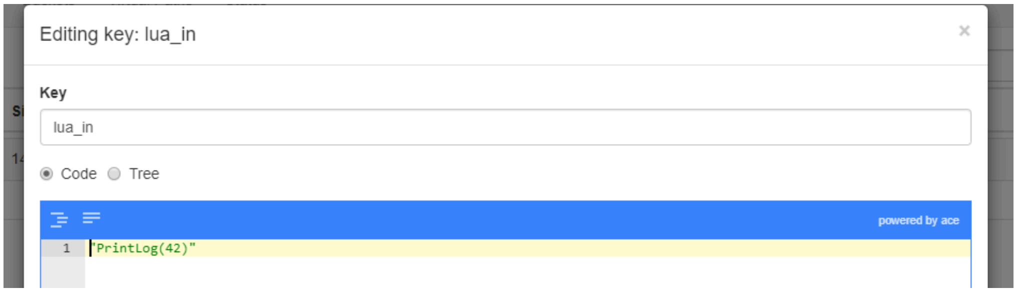
Cesium: Jitter and Glitches while you move the real camera

Check the following in that order:

1) Is ***PRIME VSAR*** dropping ? See above for **On Screen Verbosity**. Check **Output Log** for any error messages regarding dropped or late frames.
2) Is ***PRIME VSAR*** video output properly set-up? Without moving the camera, run an animation, it should be flawless. Alternatively you can create an Actor with a Rotating Component (this will save you time wrt to creating an animation). Solutions could be: bad video cables, bad overall video path configuration, field inversion, …
3) Is ***PRIME VSAR*** properly genlocked to the cesium camera data? ***PRIME VSAR*** video output and ***Cesium*** camera data device should be genlocked to the same genlock source. If it’s not the case, and if you are in autoReconnect mode you should have regular “Camera:resync” messages in the Console output of ***Unreal***.

Test DataEngine/PRIME VSAR connection

1) Open a web browser and type *http://<ip>:4300*. For IP use the IP of the PRIME VSAR system.
2) Open bucket **ue4**. If it does not exist, create it. ("ue4" is used for backwards compatibility)
3) Open key **lua_in**. If it does not exist, create it.
4) Type *PrintLog(42)*



Editing key: lua_in

Key: lua_in

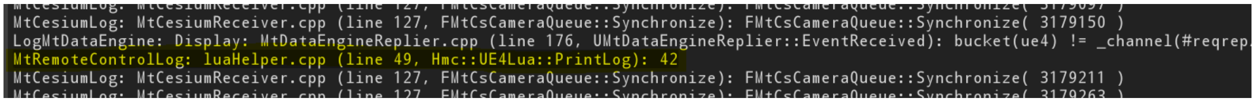
● Code ○ Tree

```
"PrintLog(42)"
```

5) Click Save (the request is send on Save)

If everything goes well, ***PRIME VSAR*** should print 42 in its **Output Log** (*Window>Developer Tools>Output Log*) something like:

MtRemoteControlLog: luaHelper.cpp (line 49, Hmc::UE4Lua::PrintLog): 42

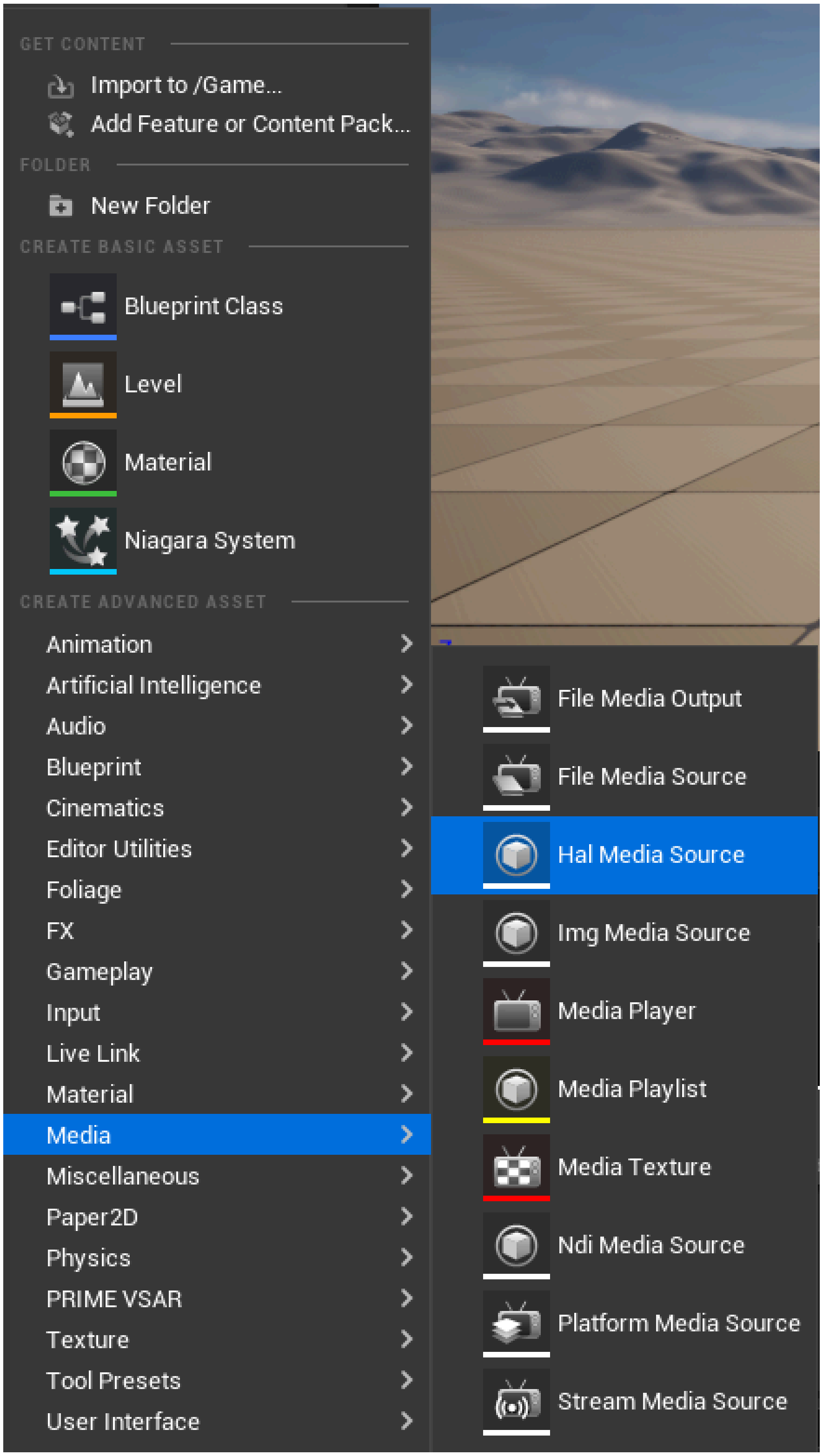


```
MtCesiumLog: MtCesiumReceiver.cpp (line 127, FMtCsCameraQueue::Synchronize): FMtCsCameraQueue::Synchronize( 3179150 )
LogMtDataEngine: Display: MtDataEngineReplier.cpp (line 176, UMtDataEngineReplier::EventReceived): bucket(ue4) != _channel(#reqrep
MtRemoteControlLog: luaHelper.cpp (line 49, Hmc::UE4Lua::PrintLog): 42
MtCesiumLog: MtCesiumReceiver.cpp (line 127, FMtCsCameraQueue::Synchronize): FMtCsCameraQueue::Synchronize( 3179211 )
MtCesiumLog: MtCesiumReceiver.cpp (line 127, FMtCsCameraQueue::Synchronize): FMtCsCameraQueue::Synchronize( 3179263 )
```

Test Video Input

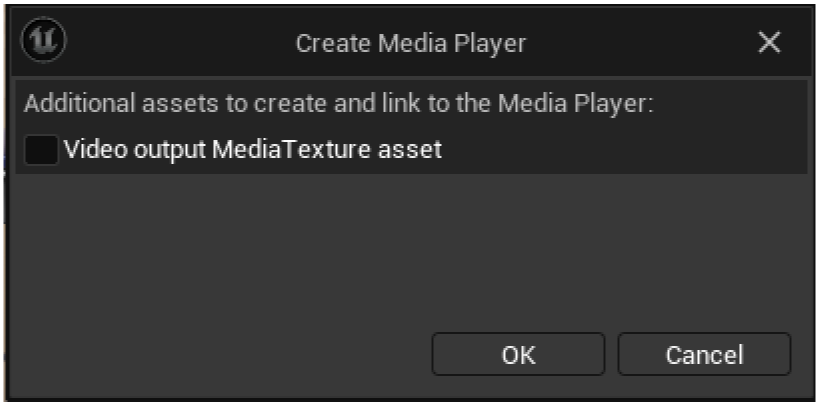
PRIME VSAR relies on *HAL* to manage video in a general manner.

1) From ***Unreal*** Content browser, add a *Add/Import>Media>HAL Media Source*.



2) From ***Unreal*** Content browser, add a *Add/Import>Media>Media Player*.
You may consider checking the checkbox that labels **Video output Media Texture**

asset, since that would be needed to be used on a material as a texture.



Create Media Player

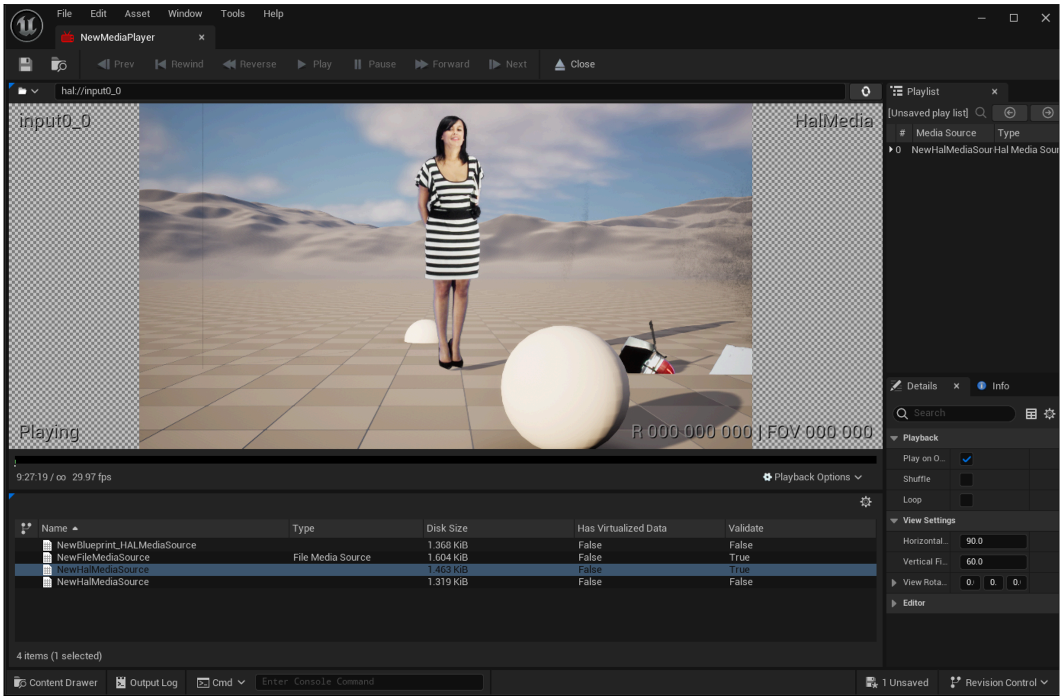
Additional assets to create and link to the Media Player:

□ Video output MediaTexture asset

Use the predefined **HAL Media Source** in the Media player. The URL address bar should display something like *hal://input0_0*.

You should be able to monitor the video from there.

3) If expecting to use it on an object through a textured material you should have checked ***Video output Media Texture asset*** at the previous step, or, alternatively, from ***Unreal*** Content browser, add a *new>Media>Media Texture* and use the previous *Media Player* as Media.



hal://input0_0

9:27:19 / ∞ 29.97 fps

[Unsaved play list]

#	Media Source	Type
0	NewHalMediaSour	Hal Media Sour

Playback

- Play on O... ☑
- Shuffle □
- Loop □

View Settings

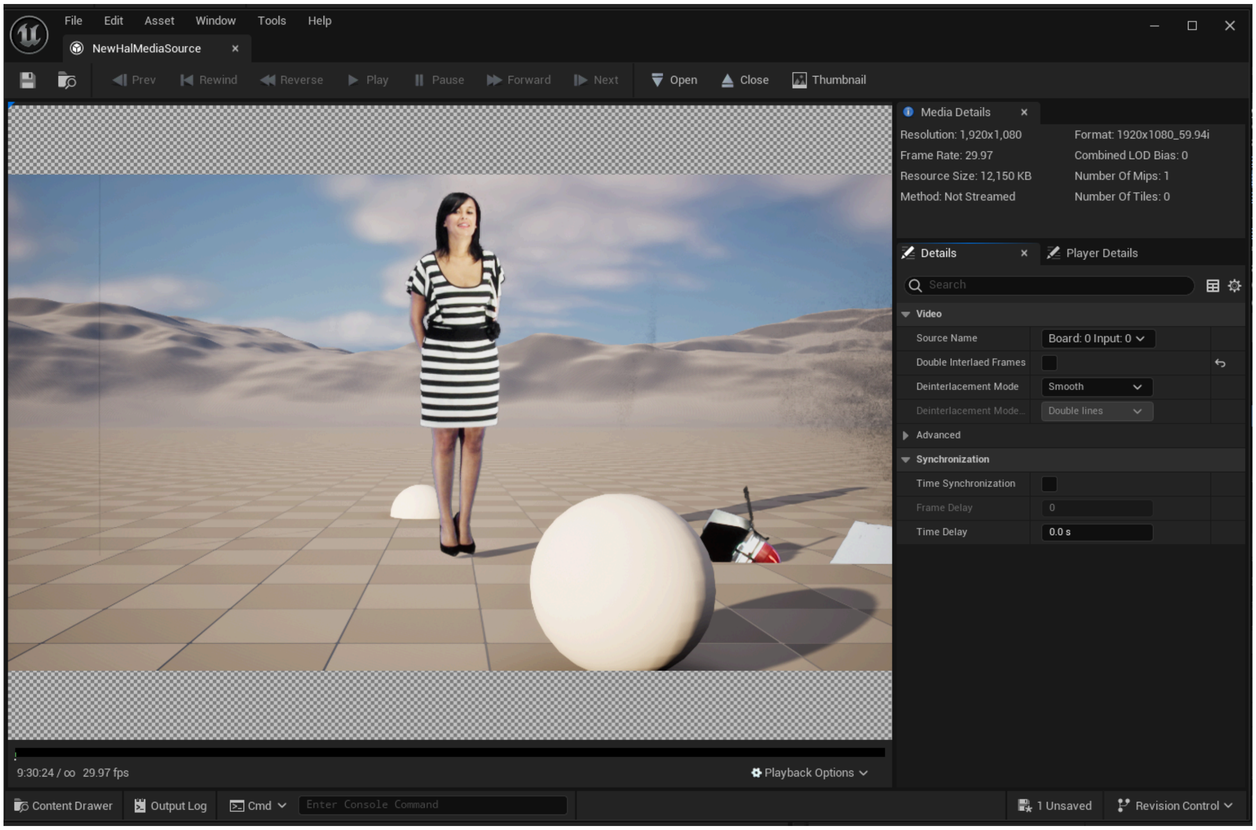
- Horizontal... 90.0
- Vertical Fi... 60.0
- View Rota... 0. 0. 0.

Editor

Name	Type	Disk Size	Has Virtualized Data	Validate
NewBlueprint_HALMediaSource		1.368 KiB	False	False
NewFileMediaSource	File Media Source	1.604 KiB	False	True
NewHalMediaSource		1.463 KiB	False	True
NewHalMediaSource		1.319 KiB	False	False

4 items (1 selected)

HAL Media Source



Resolution: 1,920x1,080
Frame Rate: 29.97
Resource Size: 12,150 KB
Method: Not Streamed
Format: 1920x1080_59.94i
Combined LOD Bias: 0
Number Of Mips: 1
Number Of Tiles: 0

Video	
Source Name	Board: 0 Input: 0
Double Interlaed Frames	□
Deinterlacement Mode	Smooth
Deinterlacement Mode...	Double lines
Advanced	
Synchronization	
Time Synchronization	□
Frame Delay	0
Time Delay	0.0 s

Source Name

The input ID as defined in the HAL.xml configuration file, more likely *hal://input0_0*.

Double interlaced Frames

When enabled, the interlaced format input aims to run at full fps. example: 50i running at 50 fps. and when disabled 50i is running at 25 fps

Deinterlacement Mode

Selects the preferred deinterlacing algorithm, applicable only when when Double interlaced Frames is disabled:

- **Keep**: keep both field as is - no deinterlacement
- **Discard**: keep even or odd field and discard second depending on the odd priory
- **Blend**: blends between fields using average
- **Smooth**: automatically detect if deinterlacing needs to be done. Usually this setting should be preferred.

Deinterlacement Mode Doubled

Selects the preferred deinterlacing algorithm, applicable only when when Double interlaced Frames is enabled:

- **Keep**: keep both field as is - no deinterlacement
- **Half size**: reduces vertical resolution by half example: 1920x1080 to 1920x540
- **Double Lines**: doubles the lines for each field, this can reduce perceived vertical resolution by half and can cause Bobbing. - sometimes called “Bob”
- **Blend**: blends between fields using average
- **Lerp**: doubles lines by blending between them using average. can be more performance intensive

Log Drop Frame: logs dropped frames that the Media player has dropped

Max Num Video Frame Buffer: maximum number of frames in buffer before Media player shows them, interlaced format is using two buffer frames each other frame when Double interlaced Frames is enabled

Odd Priory

when using Deinterlacement Mode like Discard this value decides which frames are not discarded. example: if disabled odd fields will be discarded and when enabled even fields will be discarded.

Synchronize with Engine’s Timecode

Irrelevant here.

Time Delay

delay of the playback. the value is in seconds. if we use input with 50 fps - 0.02s here will mean a single frame of delay - this value can’t be larger than Max Num Video Frame Buffer in frames or it will result in black screen as all frames will be dropped before the requested delay is reached. applicable only when Synchronize with Engine’s Timecode is disabled

⚠️ Timestamp is currently not supported

KNOWN ISSUES

Loading stuck a 45%

- This happens during first start of Unreal Engine, or when you open a new project for the first time. No worry it means that the engine is compiling the Engine shaders (check in your task manager, all cpu should have full load).

Procedure to report issues

1) In case of a crash, open your project folder.
2) Go to **Saved/Crashes/**
3) Make a zip with the folder(s) inside and send it to support@chyronhego.com
4) Please provide steps to reproduce
5) Please provide specific VSAR Version, Cesium Version, VSAR Tools Version

In the case of Visual bug:

1) Capture screenshots
2) Send them to support@chyronhego.com
3) Please provide steps to reproduce
4) Please provide specific VSAR Version, Cesium Version, VSAR Tools Version

In the case of I/O Bug/Issue:

1) Copy Hal.xml from *%LOCALAPPDATA%/Chyronhego/Fresh/HAL.xml*
2) Send it to support@chyronhego.com
3) Please provide steps to reproduce
4) Please provide specific VSAR Version, Cesium Version, VSAR Tools Version

In All cases it is also helpful to send over a copy of the project to rule out any project specific issues, if possible.

Black Video Output

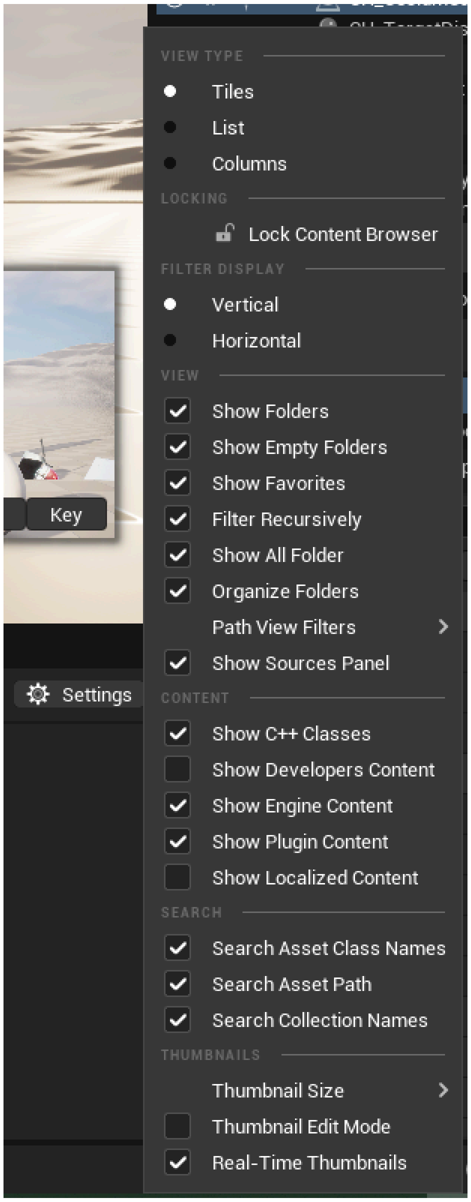
- Check that ***Cesium*** is running, and that a valid lens is loaded in ***Cesium***.

AB Switch isn’t working / Texture is black

On the first installation of ***PRIME VSAR*** there is an issue concerning the load of the shader directory and the compilation: the material is compiled before the shader directory is loaded. The fix needs to be done one time in a new installation

How to Fix

1) In your “Content Browser” you need to show the “Plugin Content” and “Engine Content”
2) To do this, click on the eye “View Options”, and select “Show Engine Content” and “Show Plugin Content”.



VIEW TYPE
● Tiles
○ List
○ Columns

LOCKING
Lock Content Browser

FILTER DISPLAY
● Vertical
○ Horizontal

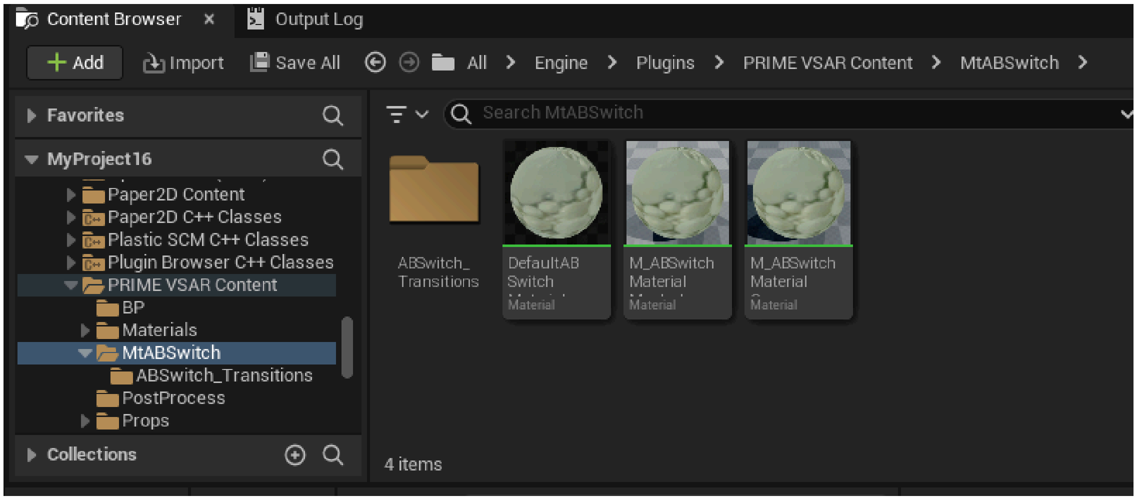
VIEW
☑ Show Folders
☑ Show Empty Folders
☑ Show Favorites
☑ Filter Recursively
☑ Show All Folder
☑ Organize Folders
Path View Filters >
☑ Show Sources Panel

CONTENT
☑ Show C++ Classes
□ Show Developers Content
☑ Show Engine Content
☑ Show Plugin Content
□ Show Localized Content

SEARCH
☑ Search Asset Class Names
☑ Search Asset Path
☑ Search Collection Names

THUMBNAILS
Thumbnail Size >
□ Thumbnail Edit Mode
☑ Real-Time Thumbnails

3) Then open the tree on the left and scroll down to “Mithril Content” and select it.
4) Open the “MtAbSwitch” directory and open the “MtABSwitchMaterial” by double clicking on it.



PRIME VSAR does not execute Chyron Panels (LAP) commands

1) Check that ***PRIME VSAR*** is in PLAY
2) Check ***PRIME VSAR/DataEngine*** connection troubleshooting (see above)

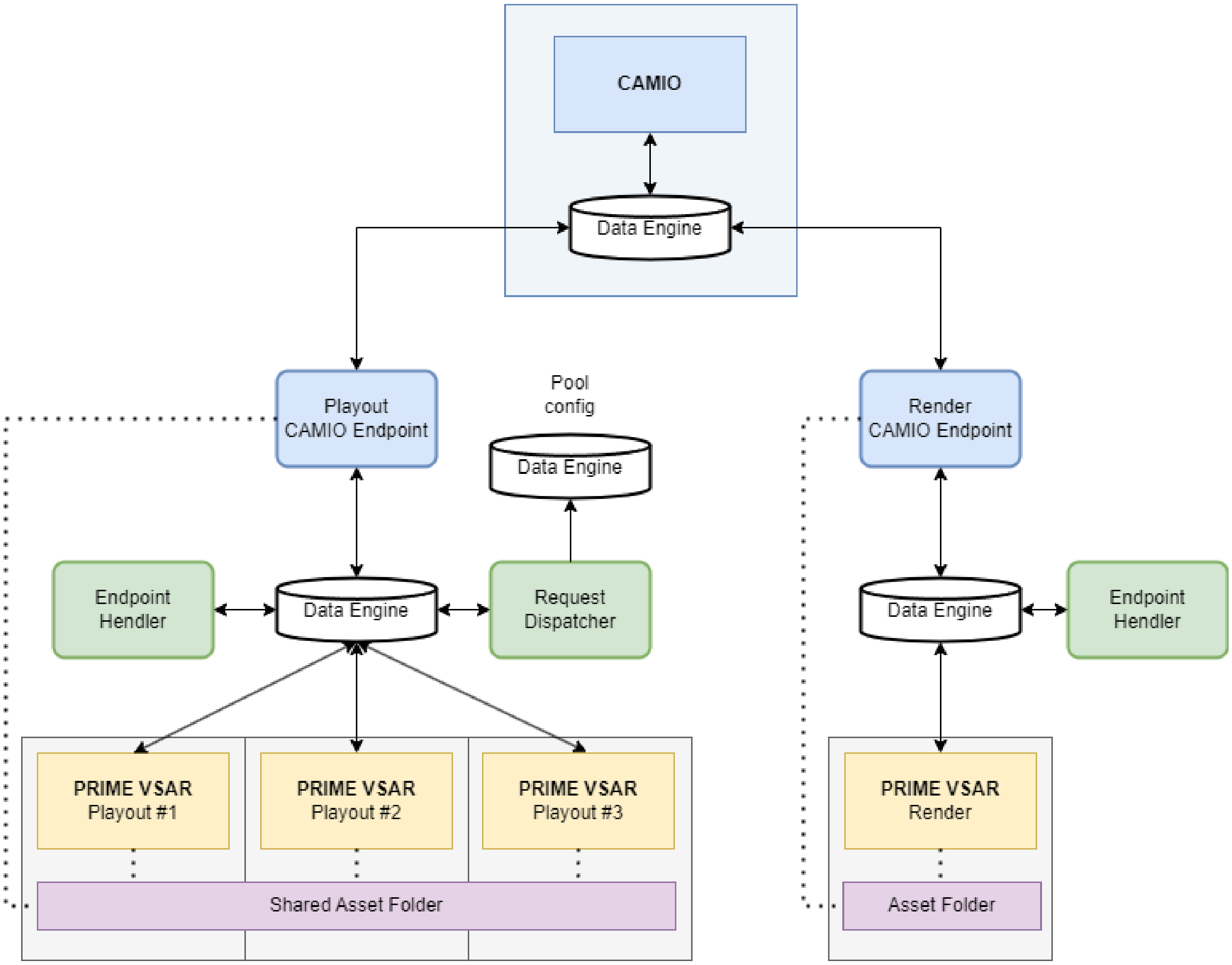
PRIME VSAR - CAMIO INTEGRATION

Integration Overview

The integration of PRIME VSAR into the CAMIO universe relies on the native Data Engine’s Request/Reply messaging system and implements the Bluenet Device API.

4 types of services are involved when CAMIO/iSQ commands are sent to PRIME VSAR:

- Data Engine
- Endpoint
- Endpoint Handler
- Request Dispatcher (optional in case of a single playout device)



Example of installation with a pool of 3 playout devices and 1 Render

Camio Components

Data Engine

The Data Engine is the communication layer of all CAMIO commands with PRIME VSAR.

⚠️ You should install at least one Data Engine for the playout device (or pool of playouts devices) and one for the preview device (cf. Data Flow diagram). The Data Engine on the CAMIO server should be already installed.

⚠️ The Data Engine is bundled with Chyron Panels(LAP), so it should be already installed.

Once installed, a Windows service is started automatically and running in the background.
Download link: https://da.chyronhego.com/da/download-area.php?proID=36

Endpoint

The Endpoint is responsible for managing requests and template asset files from CAMIO. Therefore this service must have access to a shared folder in order to save the files, see the ”Cache directory (Assets folder)” section for more details.

The *Endpoint* could be installed on a dedicated machine or on a playout/preview machine.
⚠️ There should be a single *Endpoint* running for all the playouts and a single one for the preview (cf. Data Flow diagram).
Once installed, a Windows service is started automatically and running in the background.
Download link: https://da.chyronhego.com/da/download-area.php?proID=55

Endpoint Handler

This service handles requests from the Endpoint and manages the states of on-air/cued items. The requests are then forwarded to the final device or to the *Request Dispatcher* in case of multiple devices (i.e. a pool).
The Endpoint Handler executable is included in the PRIME VSAR Tools installer.

⚠️ If PRIME VSAR Tools has been previously installed, the services may need to be stopped manually (Win+R -> services.msc) in before proceeding to the upgrade.

Once PRIME VSAR Tools is installed, you can choose to register the service when asked

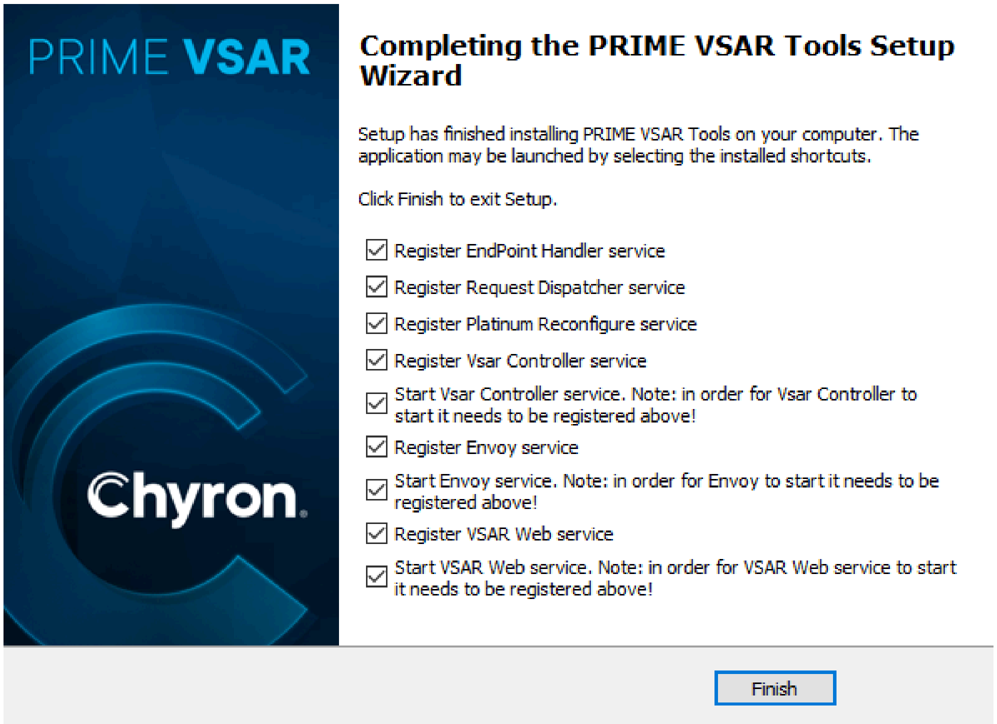
⚠️ The service reads by default its configuration on the local Data Engine, make sure the bucket and key exist before starting the service (see the Configuration section) or it will fail to start.

Request Dispatcher

This service dispatches a request to a pool of devices, for example in case of multiple playouts devices. In the case of a single device, this service is not required.
The Request Dispatcher executable is included in the PRIME VSAR Tools installer.

⚠️ If PRIME VSAR Tools has been previously installed, the services may need to be stopped before proceeding to the upgrade.

Once PRIME VSAR Tools is installed, you can choose to register the service when asked.



Completing the PRIME VSAR Tools Setup Wizard

Setup has finished installing PRIME VSAR Tools on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

☑ Register EndPoint Handler service

☑ Register Request Dispatcher service

☑ Register Platinum Reconfigure service

☑ Register Vsar Controller service

☑ Start Vsar Controller service. Note: in order for Vsar Controller to start it needs to be registered above!

☑ Register Envoy service

☑ Start Envoy service. Note: in order for Envoy to start it needs to be registered above!

☑ Register VSAR Web service

☑ Start VSAR Web service. Note: in order for VSAR Web service to start it needs to be registered above!

⚠️ The service reads by default its configuration on the local Data Engine, make sure the bucket and key exist before starting the service (see the Configuration section) or it will fail to start.

Camio Configuration

Data Engine

⚠️ Make sure the port 4300 is open on all machine were it is installed. You should be able to access http://hostname:4300 (example http://DESKTOP-MF15G8M:4300) from the other machines.

For more information, please refer to the Data Engine Documentation

Endpoint

Config key or config.json

The *Endpoint* could be configured either by a `config.json` file located by default in `C:\Program Files\ChyronHego\ChyronHego CAMIO Endpoint Beta`, or by a `config` key in the bucket `bn.endpoint`
⚠️ The `config.json` file takes precedence over the `config` key, so delete or rename `config.json` if you want to use the `config` key method.
⚠️ The `config.json` file is recreated when reinstalling the service

Example

```
"isqHostName":"ip"
"camioExternal":"camio4"
```

`"isqHostName"`: *If set, Endpoint will use this to find items in running order belonging to this host name instead of its own.*
⚠️ The hostname must match the hostname visible in the rundown's JSON.

`"camioExternal"`: It must be set to “camio4” (lower case) for backward compatibility with CAMIO 4.11.2
⚠️ If not set, the Endpoint will fail to upload previews on CAMIO.

For more information, please refer to the Endpoint Documentation

Or in its folder installation:
e.g. `C:\Program Files\ChyronHego\ChyronHego CAMIO Endpoint Beta\doc\config.md`

Cache directory (Assets folder)

In order to share the assets between multiple playout devices, the Endpoint cache directory must be shared.

These steps can be done manually or can be handled by the tool platinum-reconfigure.

1. Create a folder and set it up as a Windows shared folder
2. Create a symbolic link on each playout device pointing to the shared folder
   This step provides a solution to map UNC paths to “Drive Letter” paths, e.g. with `cmd`:
   `mklink /d c:\sharedfolder \\COMPUTER\sharedfolder`
3. Set the Endpoint `config` key or `config.json` accordingly e.g.:
   `"tempPath": "c:/sharedfolder"`

Endpoint Handler

The *Endpoint Handler* could be configured either by arguments on the command line or by a `config` key in the bucket `hybrid.endpointhandler`

Command line arguments

`-d, --dst <ARG1>`
Destination device where requests should go (e.g. MyPoolId|localhost|192.168.1.42). Mandatory

`-t, --type <ARG1>`
Destination device's type (playout|render). Mandatory

`-f, --shared_folder <ARG1>`
Shared folder of the CAMIO device. Fallback if I disk can not be found

`-h, --host <ARG1>`
IPv4 address|host of this service's DataEngine. DEF: localhost

`-s, --src_host <ARG1>`
IPv4 address|host of the Endpoint's DataEngine. DEF: localhost

`-c, --channel <ARG1>`
Device's number of channels. DEF: 1

`-o, --timeout <ARG1>`
Requests timeout in ms. DEF: 5000

`-l, --log <ARG1>`
Log level. DEF: info (error|warn|info|verbose|debug|silly)

`-v, --version`
Print version number

`--svc_args <ARG1>`
IPv4 address|host of the DataEngine with the service args (svc mode)

Config key

In order to use the `config` key, the argument `--svc_args` must be specified.

⚠️ The `config` key takes precedence over the command line arguments when the argument `--svc_args` is specified and cannot be mixed.
The key is a JSON object with attributes similar to the command line arguments.

```
{
  "dst": "localhost",
  "type": "playout",
}
```

📝Logs are written to `c:/log/ChyronHego/endpoint-handler`

Request Dispatcher

Pool of playout devices

In AR/VR, a virtual set or AR graphics could be viewed from different camera angles.

Thus the concept of a pool of playout devices has been introduced to define multiple devices working together while still being seen as a single device from the CAMIO side.

Request Dispatcher needs to run only on “`master`” VSAR.

Make sure the bucket `hybrid.pools` exists otherwise create it.
See the following documentation for the JSON definition of a pool in the bucket `hybrid.pools`:
hybrid.pools

Example

Key `pool.Pool1` in the bucket `hybrid.pools` :

```
{
  "name": "Pool1",
  "master": "Pt2",
  "slaves": [
    "Pt1",
    "Pt3"
  ],
  "share": {
    "folder": "camio-endpoint",
    "type": "smb"
  }
}
```

`Pt1, Pt2, Pt3` = are Hostnames but it can be also ip address

The *Request Dispatcher* could be configured either by arguments on the command line or by a `config` key in the bucket `hybrid.requestdispatcher`

Command line arguments

```
  -p, --pool <ARG1>
Pool Id where requests will be dispatched (e.g. 'MyPoolId'). Mandatory

  -c, --cfg_host <ARG1>
IPv4 address|host of the DataEngine with the Pool config. DEF: localhost

  -s, --src_host <ARG1>
IPv4 address|host of the Endpoint Handler's DataEngine. DEF: localhost

  -h, --host <ARG1>
IPv4 address|host of this service's DataEngine. DEF: localhost

  -r, --resolve <ARG1>
Request resolution policy. DEF: all (all|any|race)
```

`all:` A request is resolved if all devices reply successfully
`any:` A request is resolved if at least one device reply successfully
`race:` The request resolution is determined by the first (fastest) device to reply successfully or not.

```
  -x, --exclusive
Exclude the master from the pool DEF: not exclusive

  -o, --timeout <ARG1>
Requests timeout in ms. DEF: 2000

  -l, --log <ARG1>
Log level. DEF: info (error|warn|info|verbose|debug|silly)

  -v, --version
Print version number

  --svc_args <ARG1>
IPv4 address|host of the DataEngine with the service args (svc mode)
```

Config key

In order to use the `config` key, the argument `--svc_args` must be specified.

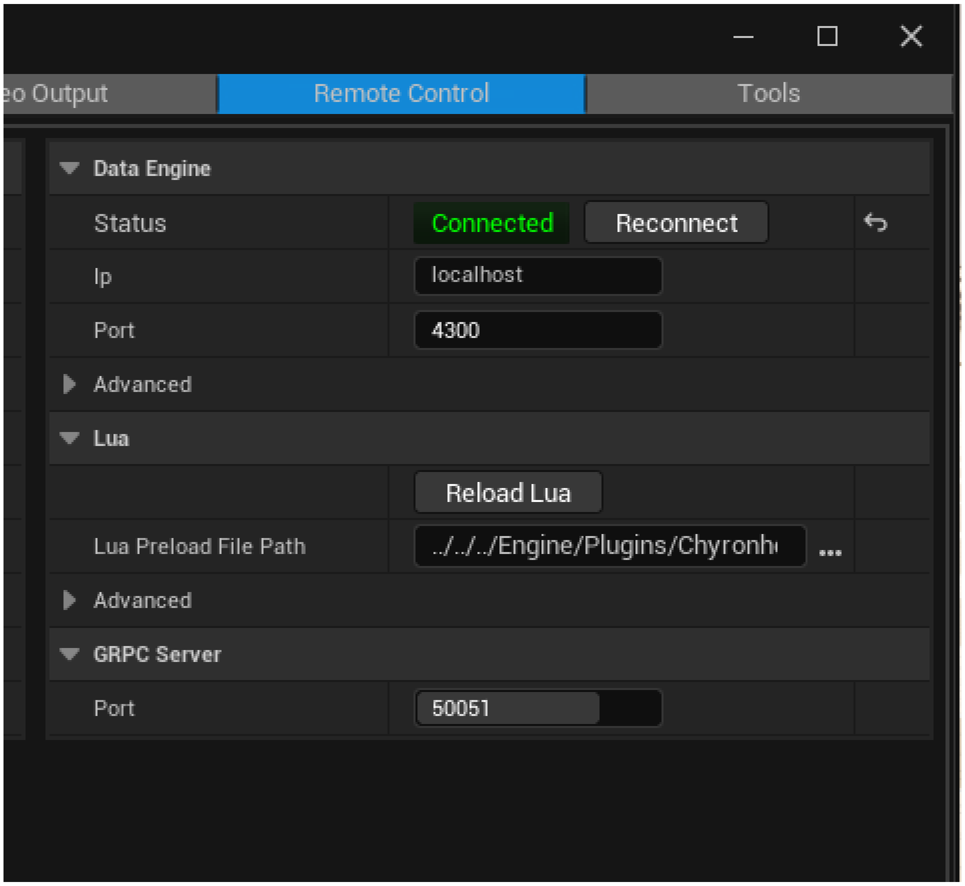
⚠️ The `config` key takes precedence over the command line arguments when the argument `--svc_args` is specified and cannot be mixed.
The key is a JSON object with attributes similar to the command line arguments.

```
  {
    "pool": "Pool1"
  }
```

📝Logs are written to `c:/log/ChyronHego/request-dispatcher`

PRIME VSAR Configuration

In the section “Remote Control” of PRIME VSAR’s Config panel, make sure it is connected to the Data Engine used by the Endpoint Handler, i.e. defined by the argument --host (or by the Request Dispatcher in case of a pool of playout devices) :



Data Engine

Status: Connected

Ip: localhost

Port: 4300

Lua

Lua Preload File Path: ../../../Engine/Plugins/Chyronh

GRPC Server

Port: 50051

PRIME VSAR’s Config Panel

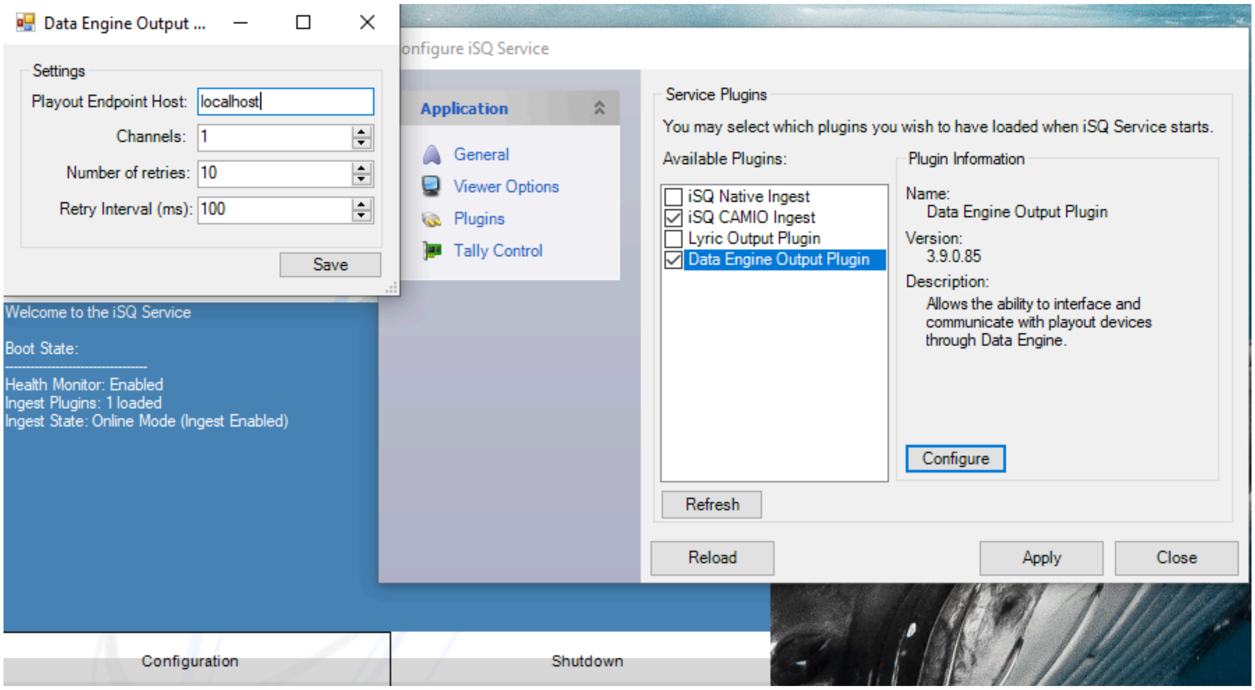
Platinum Reconfigure

platinum-reconfigure.js listens to the dataengine to change the system configuration accordingly.

Prerequisites

- VSAR Tools (2.0.0+) installed on each playout. It includes the service platinum-reconfigure
- ISQ service installed on each playout. Only one service is needed at a time running on the master device but we prepare the other devices in advance in case of a pool reconfiguration

The Playout Endpoint Host is set to localhost (i.e. iSQ Service and Endpoint will be running on the master device)



Settings

Playout Endpoint Host: localhost
Channels: 1
Number of retries: 10
Retry Interval (ms): 100

Service Plugins

You may select which plugins you wish to have loaded when iSQ Service starts.

Available Plugins:

- □ iSQ Native Ingest
- ☑ iSQ CAMIO Ingest
- □ Lyric Output Plugin
- ☑ Data Engine Output Plugin

Plugin Information

Name:
Data Engine Output Plugin
Version:
3.9.0.85
Description:
Allows the ability to interface and communicate with playout devices through Data Engine.

Welcome to the iSQ Service

Boot State:

Health Monitor: Enabled
Ingest Plugins: 1 loaded
Ingest State: Online Mode (Ingest Enabled)

Reconfigure

```
USAGE: node `platinum-reconfigure.js` [OPTION1] [OPTION2]...
```

The following options are supported:

```
 --de <ARG1\>            IP address of the dataengine on which the script will listen.
```

```
ex: node platinum-reconfigure.js --de=localhost
```

Pool

This tool can receive events to reconfigure the machine in a master/slaves environment pool. A pool is a group of machines with 1 master and several slaves. In this use case, all machines of the pool are expected to run the service `platinum-reconfigure`, working with the same dataengine.

This tool works with hostnames. So hostnames and IP correspondences are expected to be filled in the system.

On Windows:

C:\Windows\System32\drivers\etc\hosts

On the dataengine hybrid.pools bucket, each pool is described in a Key: **pool.poolname** This key needs to be created manually the first time regarding the needs.
Create in bucket hybrid.pools, the key pool.myPoolName (replace myPoolName by the pool’s name, e.g. pool.Pool1) with the following content (example with FRESH3 as master and FRESH4 as slave):

Ex: pool.Pool1 Master on Windows (smb)

```
{
 "name": "Pool1",
 "master": "Pt2",
 "slaves": [
   "Pt1",
   "Pt3"
 ],
 "share": {
   "type": "smb",
   "name": "camio-endpoint",
   "drive": "S"
 }
}
```

In hybrid.pools bucket, each machine listen a Key that tell to which pool it belongs:
device.hostname
Pt1, Pt2, Pt3 = are Hostnames but it can be also ip address

Ex: device.Pt2

```
    {
       "pool": "pool.Pool1"
     }
```

To re-arrange a pool, modify the pool description. Then overwrite the key of the new master. It will receive an event and configure itself as master. Then, when ready, it will write the keys of its slaves so they will reconfigure themselves as slaves.

For reconfiguration, external script are called:

- releaseMe.[sh-bat]
- masterMe.[sh-bat]
- slaveMe.[sh-bat]
- patchconfig.js

During the reconfiguration:

The script will read on the local dataengine to get the share folder path in bn.endpoint bucket, config key, tempPath field. That's where the shared folder is. All machines are expected to have this field on the same value. Default is C:\Users\ChyronHego\AppData\Local\Temp

For the master VSAR device, it will modify **hybrid.endpointhandler** and **hybrid.requestdispatcher** buckets, **config** keys. endpointhandler and requestdispatcher services have to be installed and those keys have to exist.

For all VSAR devices, it will modify **HOME\AppData\Local\Chyronhego\Fresh\Mithril.ini** to update `[/Script/Mithril.MtDataEngineHandler]` section, `Ip` field to the master VSAR hostname.

PRIME VSAR - CAMIO Integration Troubleshooting

Notes

iSQ web interface (experimental) is located at http://localhost:8087/isq

Bluenet (Endpoint) log web interface (experimental) is located at http://localhost:8087/log

Data Engine web interface is located at http://localhost:4300

📝Data Engine logs are written to `c:/log/ChyronHego/dataengine`
📝Endpoint logs are written to `c:/log/ChyronHego/camio-endpoint`
📝Endpoint handler logs are written to `c:/log/ChyronHego/endpoint-handler`
📝Request dispatcher logs are written to `c:/log/ChyronHego/request-dispatcher`

Check list

- **All the machines can ping each other**
- **Endpoint config file is either `config.json` or a `config` key in the bucket `bn.endpoint`**
- Port 4300 is open on every machine hosting a Data Engine
- All services are running: Data Engine, Endpoint, Endpoint Handler, Request Dispatcher (optional, only needed for pool of playouts)
- All MOS/CAMIO server related services are running (e.g. External Render Service)
- All PRIME VSAR are connected to their respective Data Engine

Symptoms

A service is failing to read its config key on the Data Engine

(Failed to read args on the Data Engine, exiting...)

- Verify the Data Engine is running
- Verify the bucket and config key are existing and named accordingly without heading or trailing spaces
- Verify the config key's content is a valid JSON object

The rundown is selectable in iSQ web interface but is empty,
iSQ Viewer displays the rundown but commands have no effect
(`"item not found"` or `"0/0 assets"` in the Endpoint logs)

- Verify Endpoint config file is either `config.json` or a `config` key in the bucket `bn.endpoint`
- Verify the Endpoint configuration key `"isqHostName"`: If set, Endpoint will use this to find items in running order belonging to this host name instead of its own.
- Restart the Endpoint service, and republish the rundown (e.g. in ENPS toggle 'MOS Control Active')

It should display in the log `"Looking for items routed to XXX"`, `XXX` being the `isqHostname` `"X/X assets"` with `X` not equal to 0

⚠️ **The hostname must match the hostname visible in the rundown's JSON.**

The Request dispatcher is not able to see a slave machine (“Device 'xxx' is unreachable” in the logs)

- Check the Request dispatcher is pinging the slave machine
- Check the port 4300 is open on the slave machine (on the slave machine you should be able to access in a web browser the page located at http://localhost:4300)
  You might need to disable/reconfigure the firewall on Windows

The context is not visible in LUCI 4 / 5

- Check the Context Permission in the CAMIO Admin tool (cf. CAMIO documentation)

The playout/preview fails

- Check there is no ID clash between the PRIME VSAR instances. For example 2 previews having the same ID and running at the same time will cause issues.
- Check PRIME VSAR is running in PLAY mode

PRIME VSAR - nDisplay integration

nDisplay integration allows the use of HDMI / DisplayPort output for VSAR with the use of Cesium Camera.

nDisplay Setup

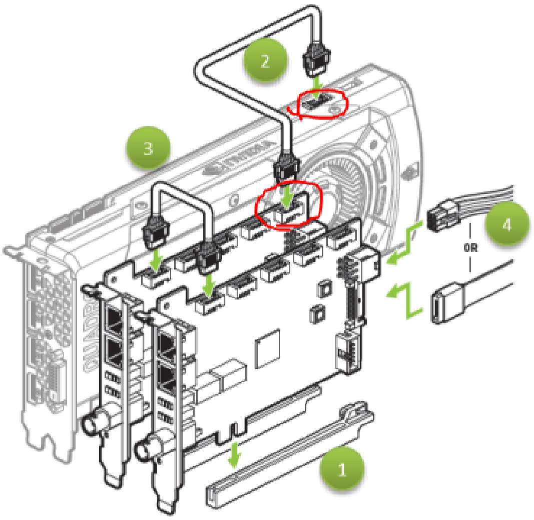
nDisplay Requirements:

- Nvidia Quadro Sync II
- Cat-5 or Cat-6 RJ45 cables. (for multiple machines)
- BNC cable for genlock
- DisplayPort to HDMI convertor (only for HDMI)

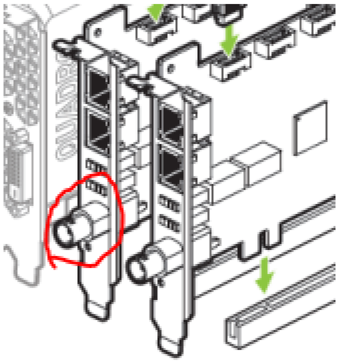
Sync setup

Nvidia Quadro Sync II needs to be installed in the PCIe slot on the motherboard and connected with appropriate cable to the Nvidia GPU. for more information see more at Quadro-Sync-II user guide

The cable should come with the Nvidia Quadro Sync II, Nvidia GPU has the ports for the Quadro Sync II connection at the top; they usually have plastic cover that needs to be removed to uncover the ports and connect the according the Quadro-Sync-II user guide



After the cable is properly connected to the GPU. We need to connect the genlock signal using BNC cable to the spigot.



after the connection is correctly connected make sure to install the the Quadro sync II Driver (the link is for windows 11)

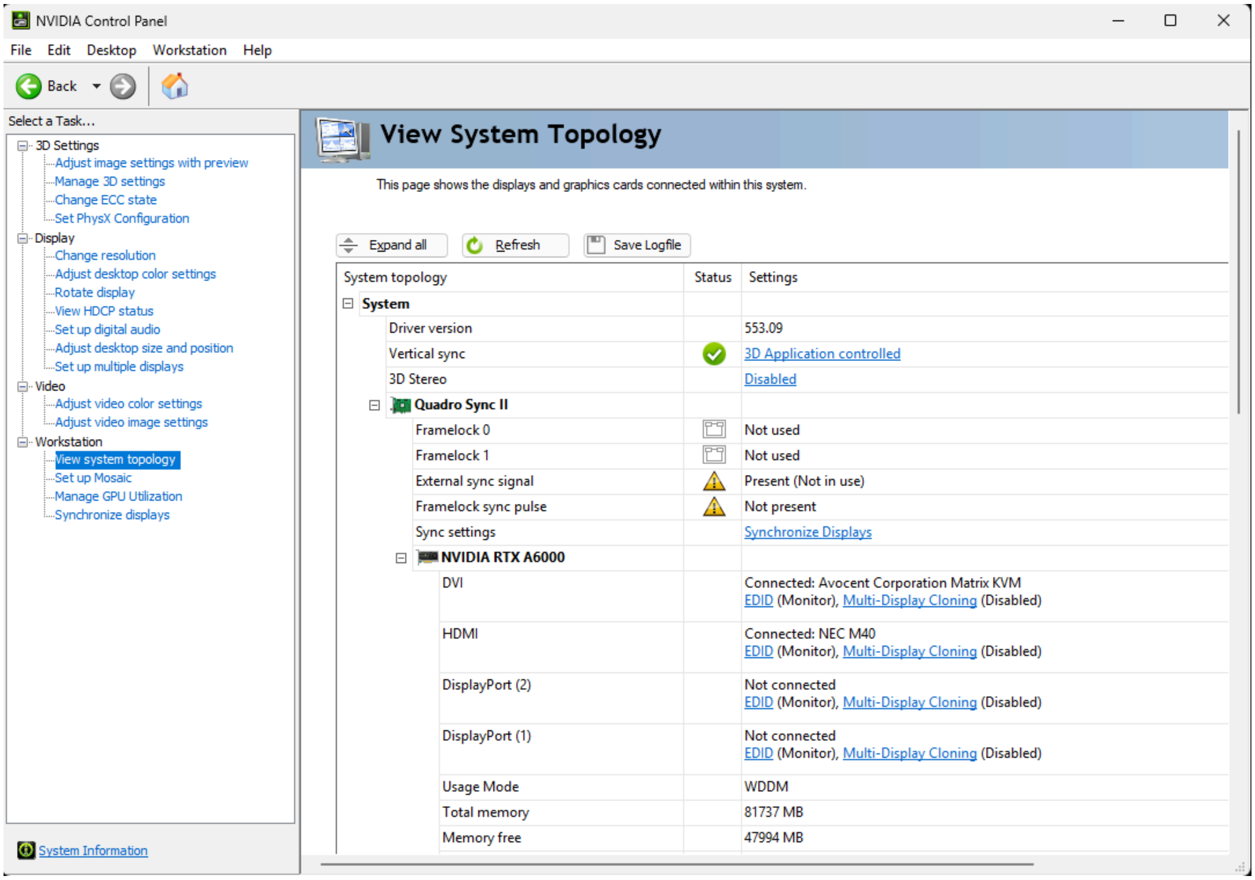
For multiple machines:

Frame Lock: connected to the Cat-5 or Cat-6 RJ45 cables according to the Quadro-Sync-II user guide, make sure to use direct connections without any TCP/IP hardware in between. Coming from the master machine going into the client machines.

Or

External Sync only: connect each Nvidia Quadro Sync II to genlock source and set them up all with external sync source.

when properly set-up on **master** it should say “External Sync signal: Present (Not in use)”

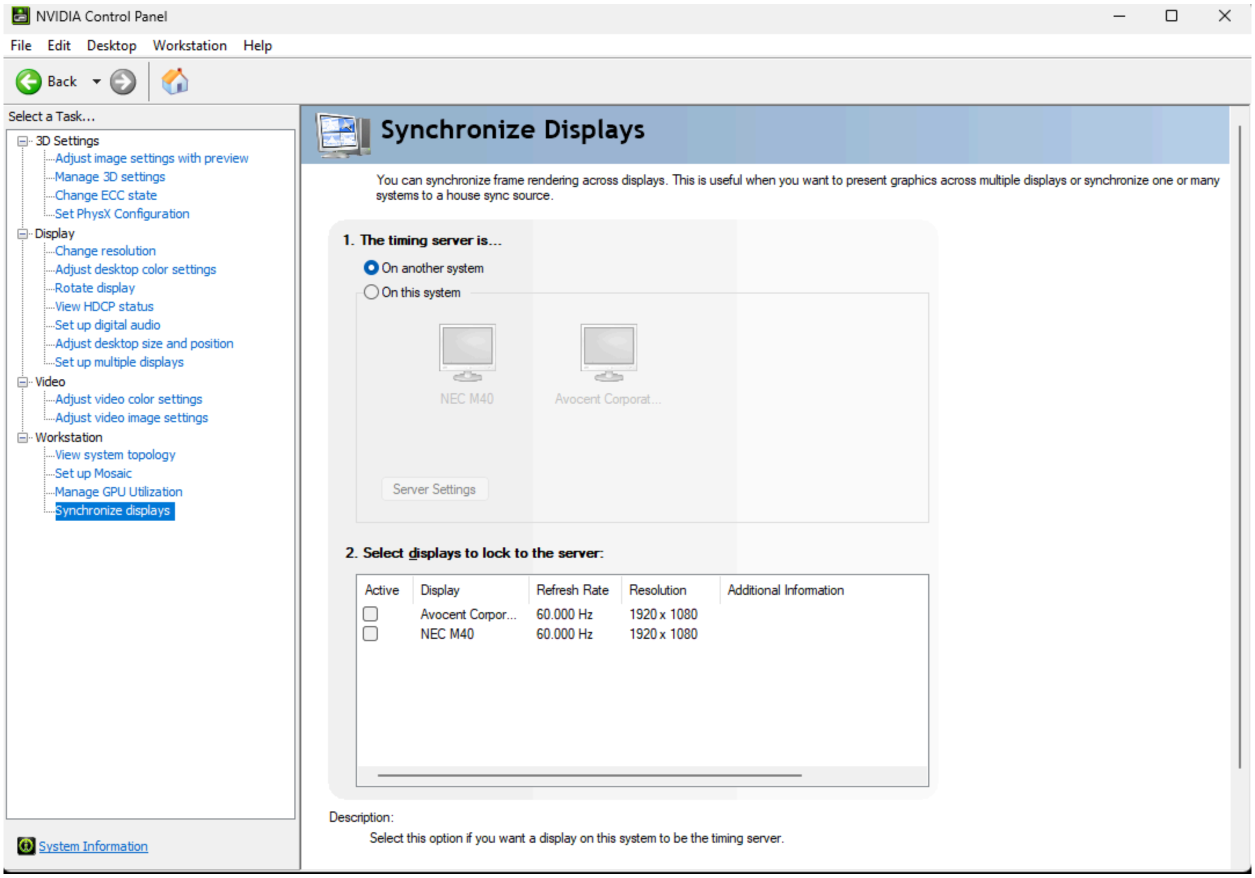


View System Topology

This page shows the displays and graphics cards connected within this system.

System topology	Status	Settings
⊟ **System**		
Driver version		553.09
Vertical sync	✓	3D Application controlled
3D Stereo		Disabled
⊟ **Quadro Sync II**		
Framelock 0	□	Not used
Framelock 1	□	Not used
External sync signal	⚠	Present (Not in use)
Framelock sync pulse	⚠	Not present
Sync settings		Synchronize Displays
⊟ **NVIDIA RTX A6000**		
DVI		Connected: Avocent Corporation Matrix KVM EDID (Monitor), Multi-Display Cloning (Disabled)
HDMI		Connected: NEC M40 EDID (Monitor), Multi-Display Cloning (Disabled)
DisplayPort (2)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)
DisplayPort (1)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)
Usage Mode		WDDM
Total memory		81737 MB
Memory free		47994 MB

go to **Synchronize Displays**



Synchronize Displays

You can synchronize frame rendering across displays. This is useful when you want to present graphics across multiple displays or synchronize one or many systems to a house sync source.

1. The timing server is...

● On another system
○ On this system

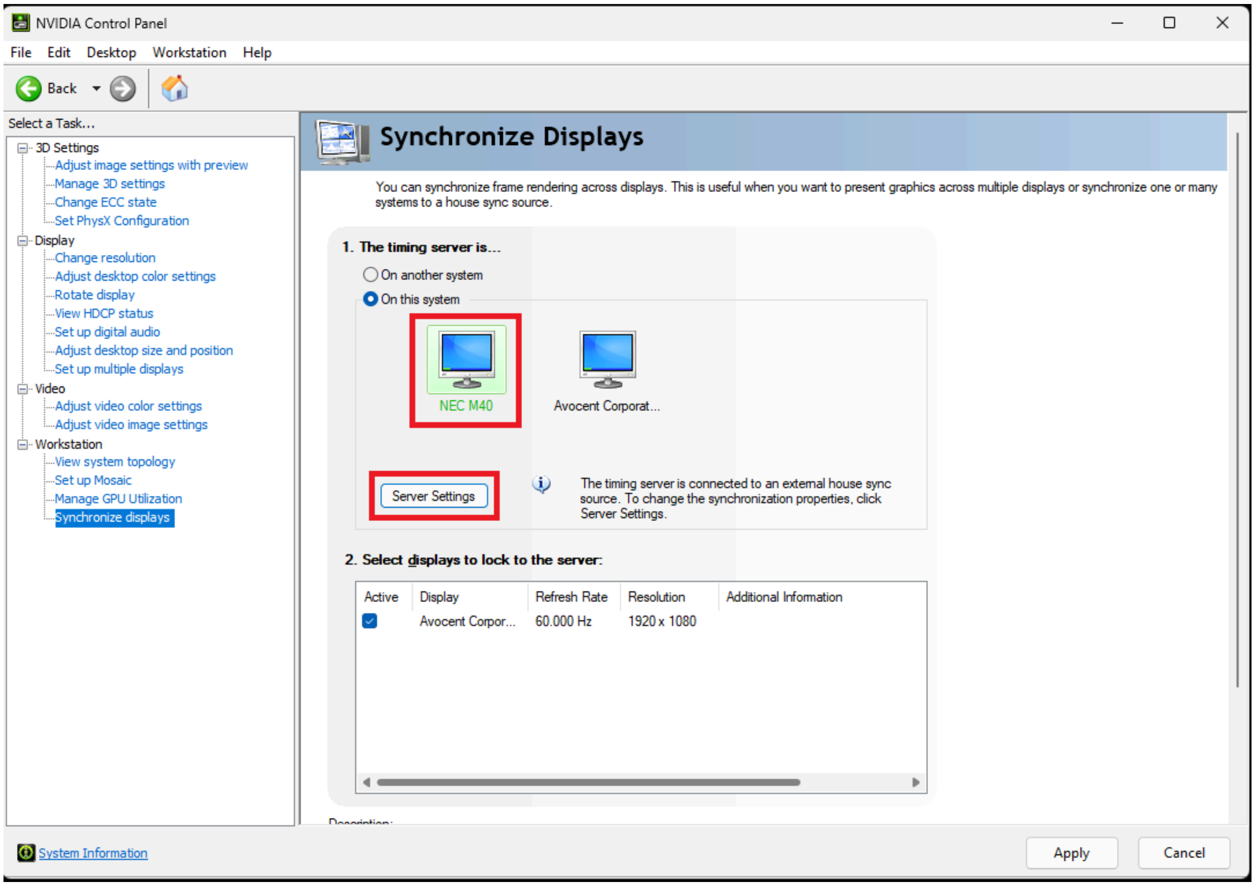
2. Select displays to lock to the server:

Active	Display	Refresh Rate	Resolution	Additional Information
□	Avocent Corpor...	60.000 Hz	1920 x 1080	
□	NEC M40	60.000 Hz	1920 x 1080	

Description:

Select this option if you want a display on this system to be the timing server.

Select **On this system**



Synchronize Displays

You can synchronize frame rendering across displays. This is useful when you want to present graphics across multiple displays or synchronize one or many systems to a house sync source.

1. The timing server is...

○ On another system
● On this system

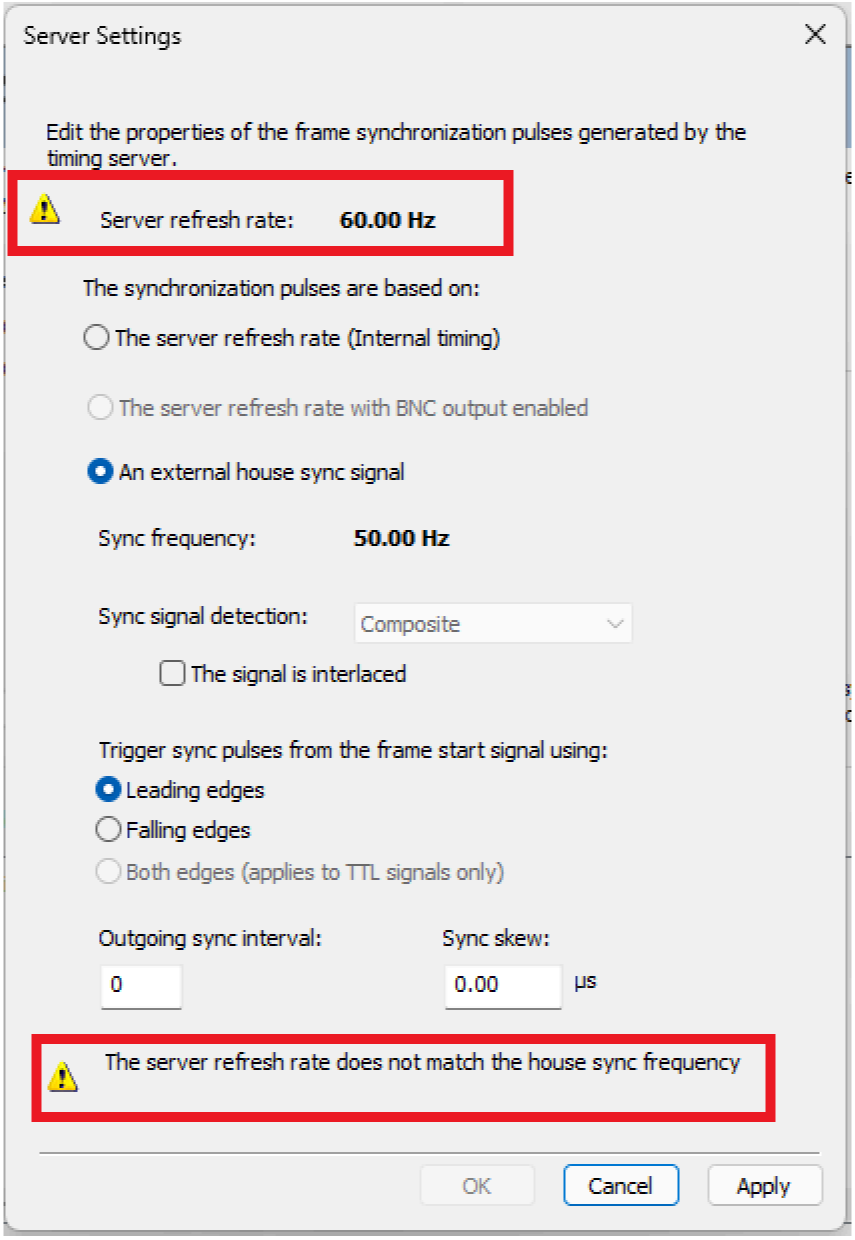
The timing server is connected to an external house sync source. To change the synchronization properties, click Server Settings.

2. Select displays to lock to the server:

Active	Display	Refresh Rate	Resolution	Additional Information
☑	Avocent Corpor...	60.000 Hz	1920 x 1080	

Select the **Main Display** and click the **Server Settings**

if you see that the external sync frequency does not match



Server Settings

Edit the properties of the frame synchronization pulses generated by the timing server.

⚠ Server refresh rate: **60.00 Hz**

The synchronization pulses are based on:

○ The server refresh rate (Internal timing)

○ The server refresh rate with BNC output enabled

● An external house sync signal

Sync frequency: **50.00 Hz**

Sync signal detection: Composite

□ The signal is interlaced

Trigger sync pulses from the frame start signal using:

● Leading edges
○ Falling edges
○ Both edges (applies to TTL signals only)

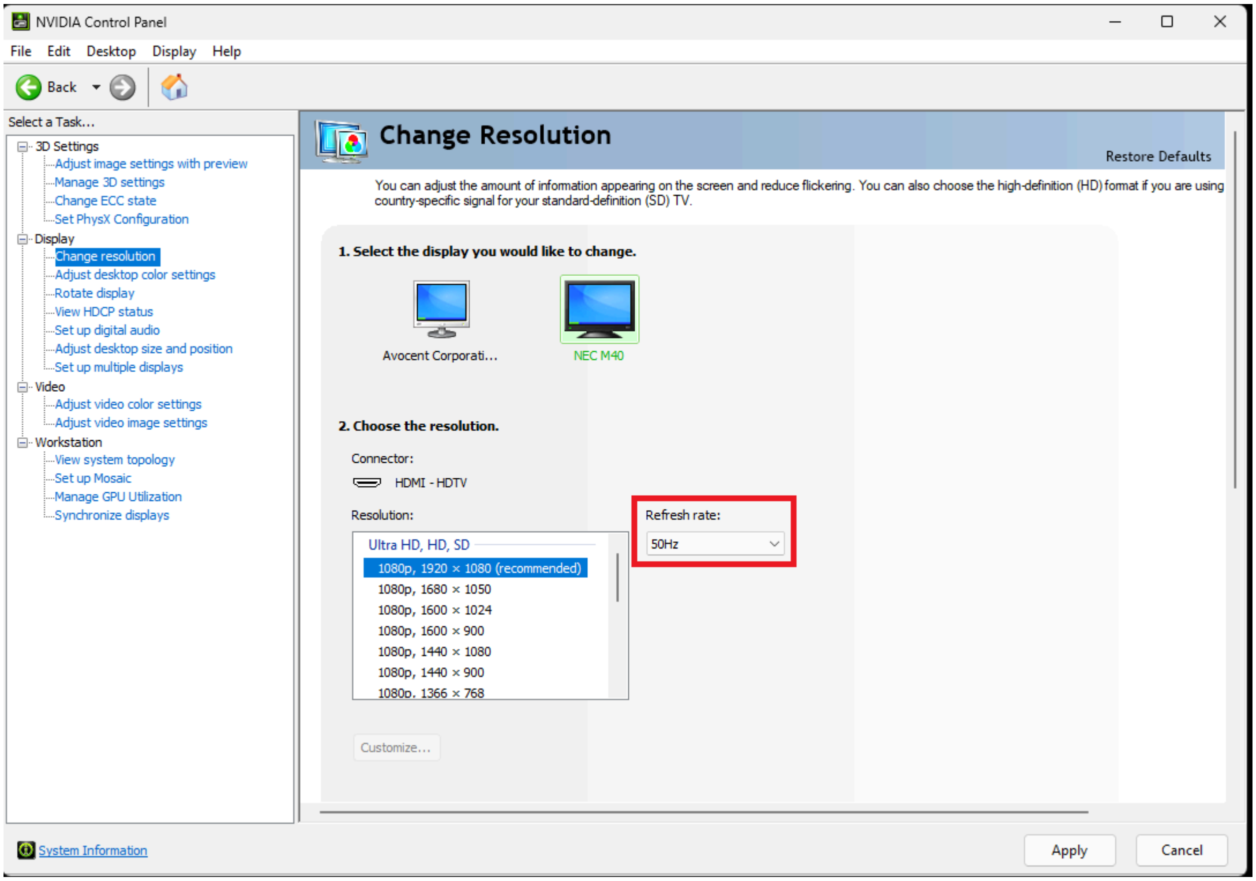
Outgoing sync interval: 0

Sync skew: 0.00 μs

⚠ The server refresh rate does not match the house sync frequency

you can fix that by going into **change resolution -> refresh rate**

⚠️ when you experience blinking of the windows try restarting the display



NVIDIA Control Panel

Select a Task...

- 3D Settings
  - Adjust image settings with preview
  - Manage 3D settings
  - Change ECC state
  - Set PhysX Configuration
- Display
  - Change resolution
  - Adjust desktop color settings
  - Rotate display
  - View HDCP status
  - Set up digital audio
  - Adjust desktop size and position
  - Set up multiple displays
- Video
  - Adjust video color settings
  - Adjust video image settings
- Workstation
  - View system topology
  - Set up Mosaic
  - Manage GPU Utilization
  - Synchronize displays

Change Resolution

Restore Defaults

You can adjust the amount of information appearing on the screen and reduce flickering. You can also choose the high-definition (HD) format if you are using country-specific signal for your standard-definition (SD) TV.

1. Select the display you would like to change.

Avocent Corporati... NEC M40

2. Choose the resolution.

Connector:

HDMI - HDTV

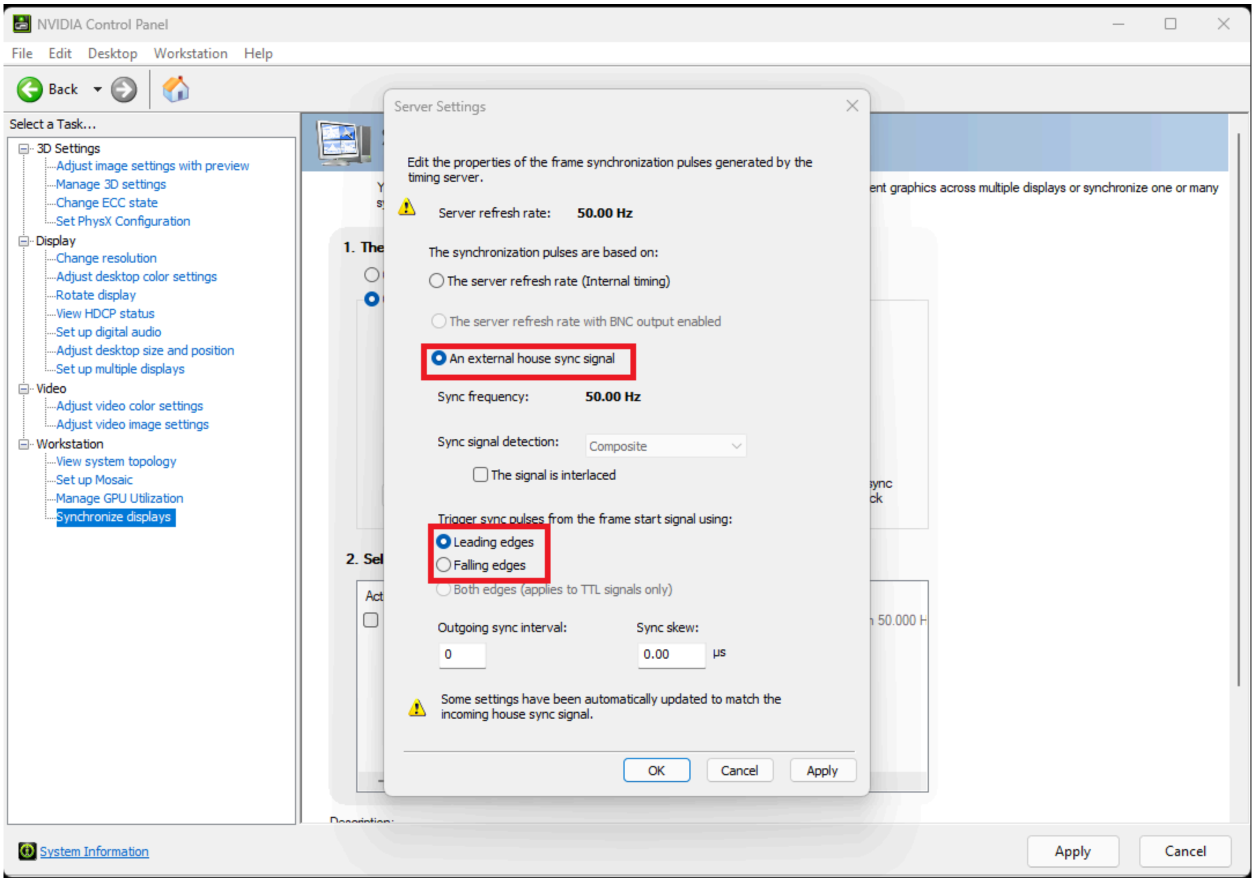
Resolution:

Ultra HD, HD, SD
1080p, 1920 × 1080 (recommended)
1080p, 1680 × 1050
1080p, 1600 × 1024
1080p, 1600 × 900
1080p, 1440 × 1080
1080p, 1440 × 900
1080p, 1366 × 768

Refresh rate:

50Hz

make sure click **Apply**



Server Settings

Edit the properties of the frame synchronization pulses generated by the timing server.

⚠ Server refresh rate: **50.00 Hz**

The synchronization pulses are based on:

○ The server refresh rate (Internal timing)

○ The server refresh rate with BNC output enabled

● An external house sync signal

Sync frequency: **50.00 Hz**

Sync signal detection: Composite

□ The signal is interlaced

Trigger sync pulses from the frame start signal using:

● Leading edges

○ Falling edges

○ Both edges (applies to TTL signals only)

Outgoing sync interval: 0

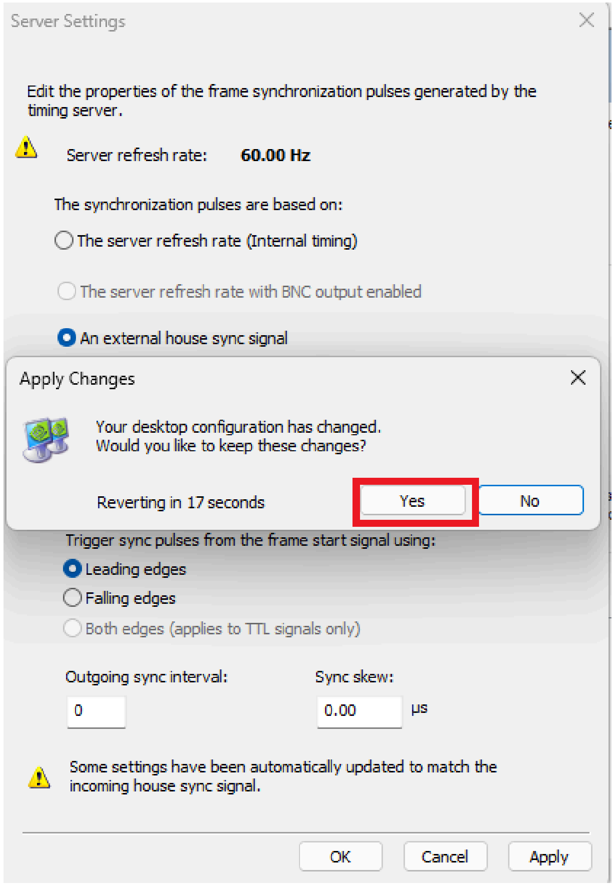
Sync skew: 0.00 µs

⚠ Some settings have been automatically updated to match the incoming house sync signal.

select the **An external house sync signal** and select the Leading/Falling edges depending on the genlock generator.

click **Apply**

make sure to confirm the setting



Server Settings

Edit the properties of the frame synchronization pulses generated by the timing server.

Server refresh rate: **60.00 Hz**

The synchronization pulses are based on:

○ The server refresh rate (Internal timing)

○ The server refresh rate with BNC output enabled

● An external house sync signal

Apply Changes

Your desktop configuration has changed.
Would you like to keep these changes?

Reverting in 17 seconds

Trigger sync pulses from the frame start signal using:

● Leading edges

○ Falling edges

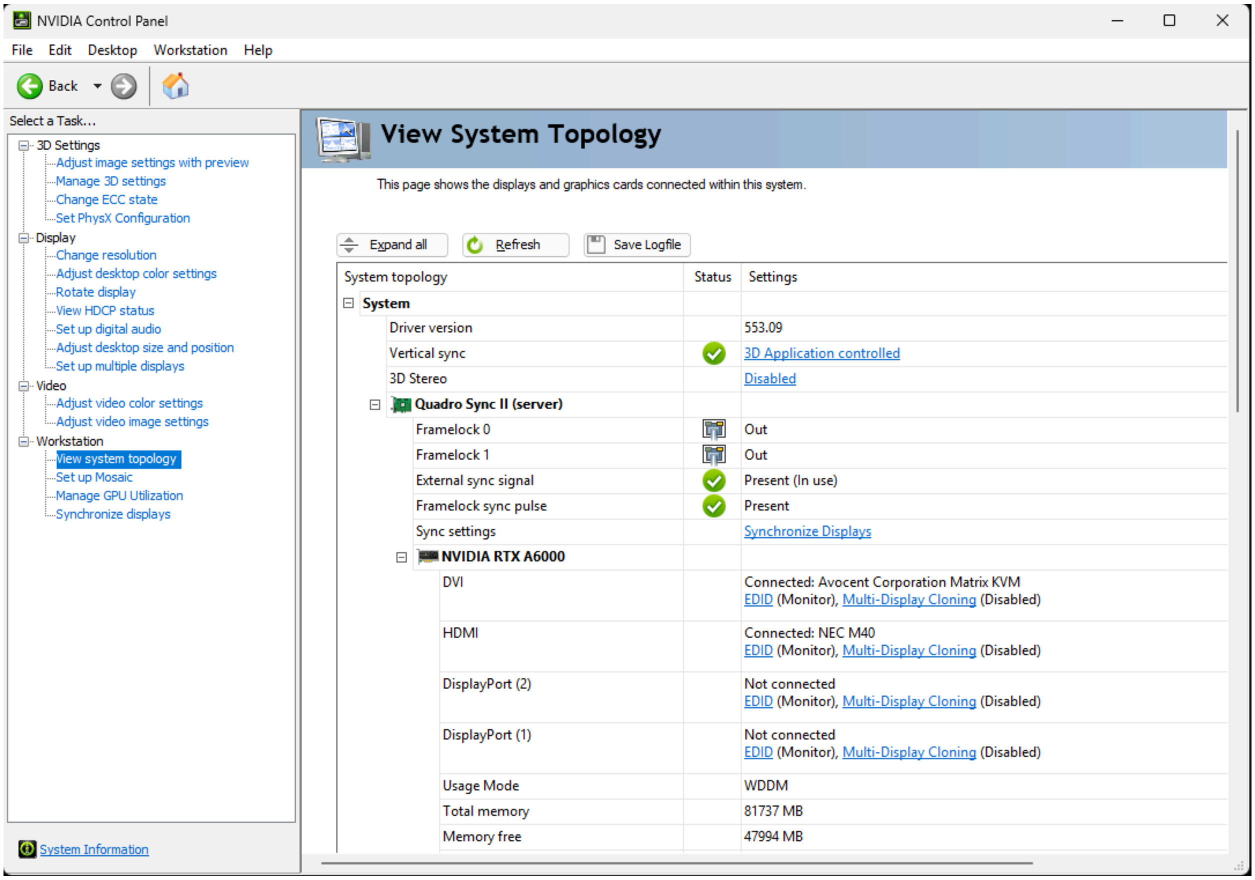
○ Both edges (applies to TTL signals only)

Outgoing sync interval: 0

Sync skew: 0.00 µs

Some settings have been automatically updated to match the incoming house sync signal.

when the genlock is actively in use it should say “External Sync signal: Present (in use)” (may not necessarily mean that all configuration is correct just that it is used)



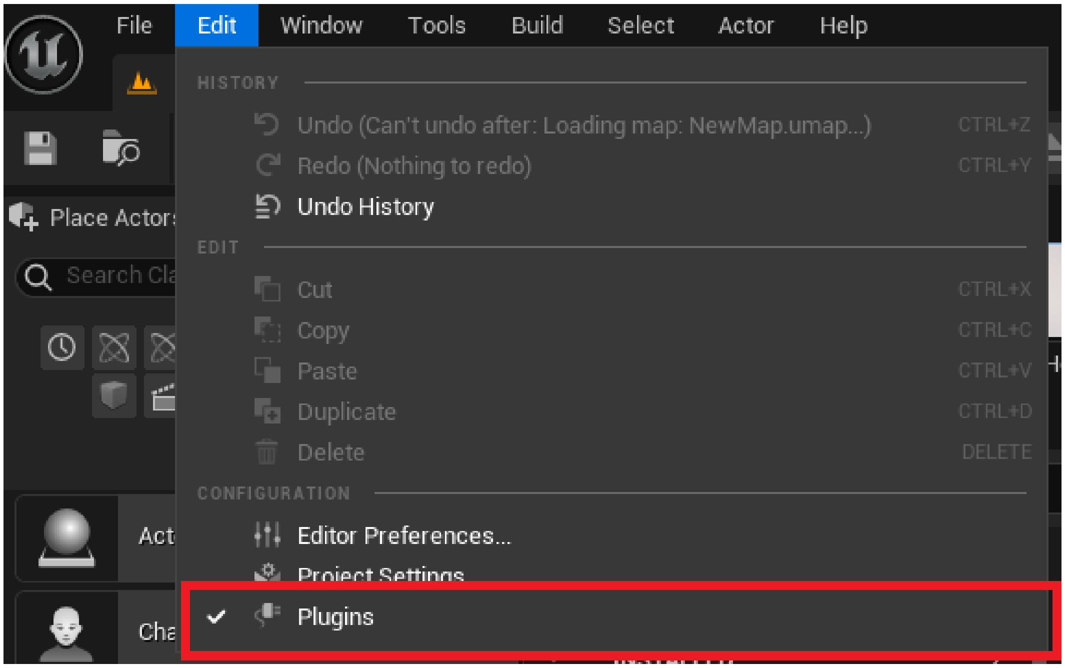
View System Topology

This page shows the displays and graphics cards connected within this system.

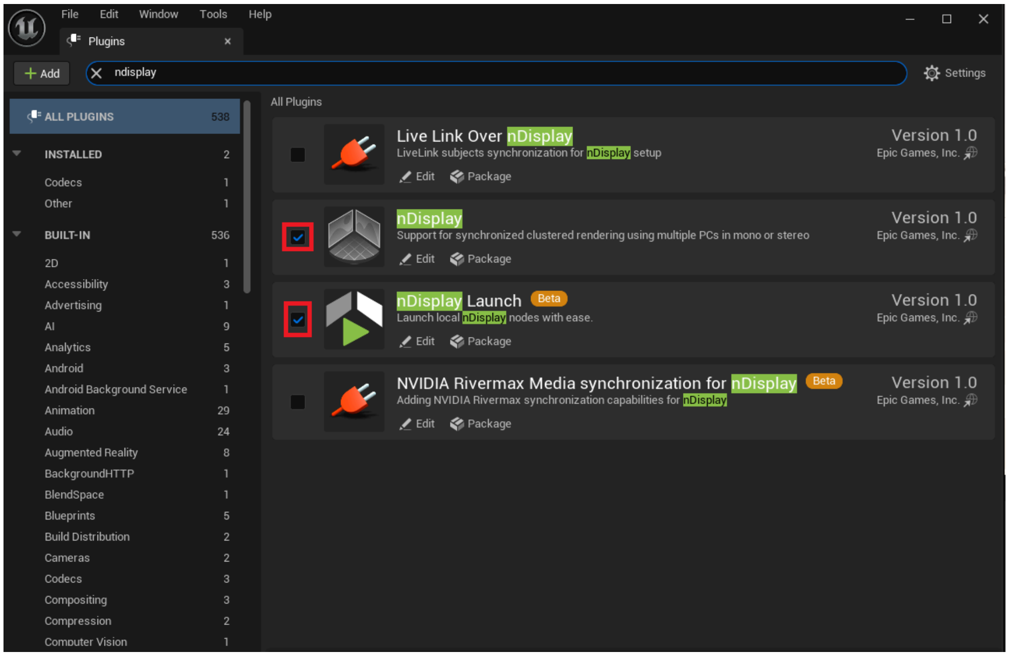
System topology	Status	Settings
System		
Driver version		553.09
Vertical sync	✓	3D Application controlled
3D Stereo		Disabled
Quadro Sync II (server)		
Framelock 0		Out
Framelock 1		Out
External sync signal	✓	Present (In use)
Framelock sync pulse	✓	Present
Sync settings		Synchronize Displays
NVIDIA RTX A6000		
DVI		Connected: Avocent Corporation Matrix KVM EDID (Monitor), Multi-Display Cloning (Disabled)
HDMI		Connected: NEC M40 EDID (Monitor), Multi-Display Cloning (Disabled)
DisplayPort (2)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)
DisplayPort (1)		Not connected EDID (Monitor), Multi-Display Cloning (Disabled)
Usage Mode		WDDM
Total memory		81737 MB
Memory free		47994 MB

Enable Plugins

Open VSAR in **Edit -> Plugins**



search for **nDisplay** and enable **nDisplay** and **nDisplay Launch**



All Plugins

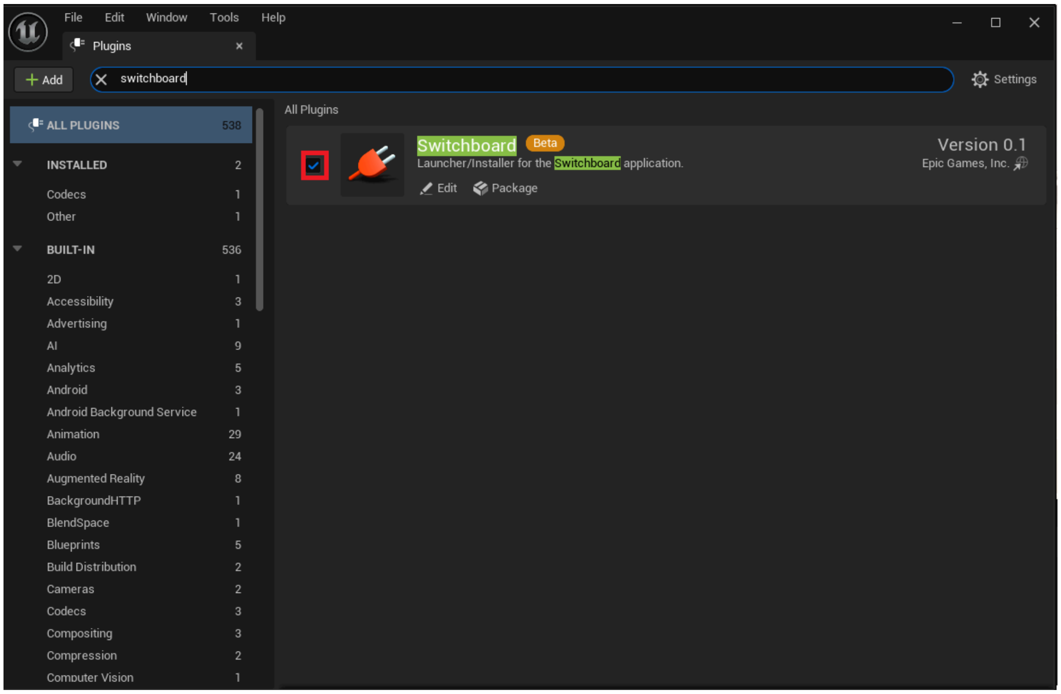
□ Live Link Over nDisplay — Version 1.0
LiveLink subjects synchronization for nDisplay setup
Epic Games, Inc.

☑ nDisplay — Version 1.0
Support for synchronized clustered rendering using multiple PCs in mono or stereo
Epic Games, Inc.

☑ nDisplay Launch (Beta) — Version 1.0
Launch local nDisplay nodes with ease.
Epic Games, Inc.

□ NVIDIA Rivermax Media synchronization for nDisplay (Beta) — Version 1.0
Adding NVIDIA Rivermax synchronization capabilities for nDisplay
Epic Games, Inc.

search for **switchboard** and enable it



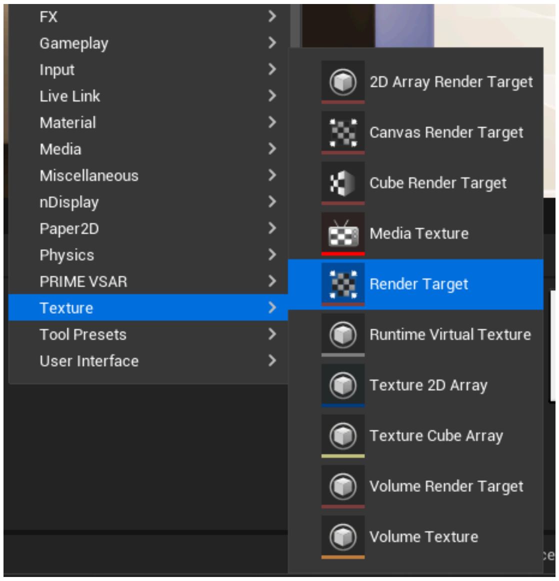
switchboard

☑ Switchboard Beta
Launcher/Installer for the Switchboard application.
Version 0.1
Epic Games, Inc.

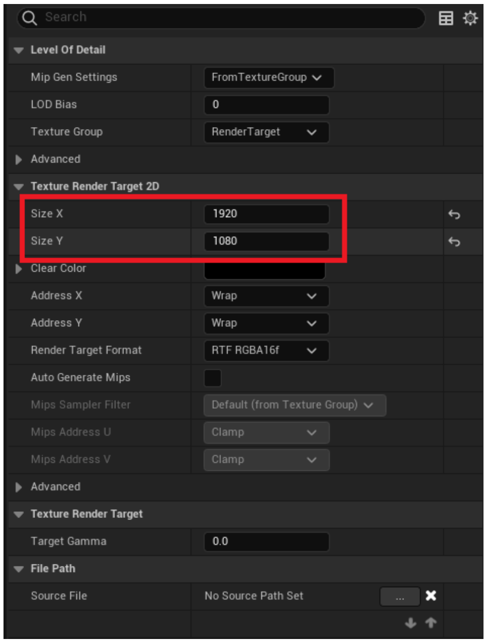
you may need to restart VSAR

Render target

Cesium Camera is using the render target as intermediary between Cesium Camera and nDisplay. Create Render target by Right click in Content browser and select Render target.



double click on the Render target asset and set the Size X, Size Y to use **custom resolution**

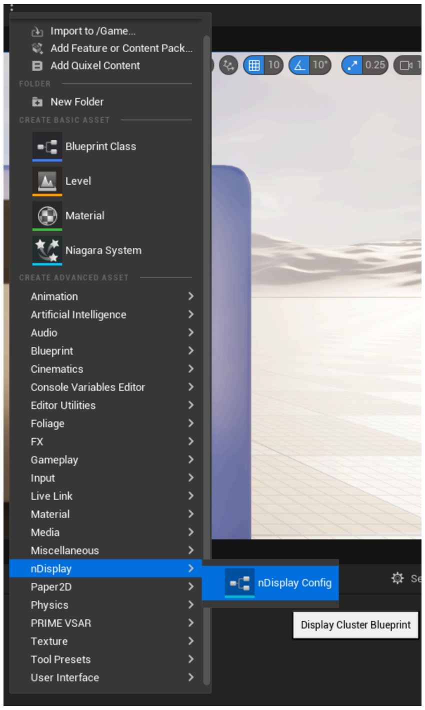


Level Of Detail	
Mip Gen Settings	FromTextureGroup
LOD Bias	0
Texture Group	RenderTarget
Advanced	
Texture Render Target 2D	
Size X	1920
Size Y	1080
Clear Color	
Address X	Wrap
Address Y	Wrap
Render Target Format	RTF RGBA16f
Auto Generate Mips	□
Mips Sampler Filter	Default (from Texture Group)
Mips Address U	Clamp
Mips Address V	Clamp
Advanced	
Texture Render Target	
Target Gamma	0.0
File Path	
Source File	No Source Path Set

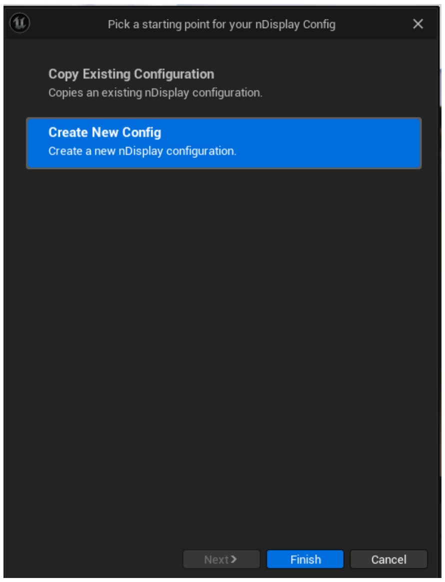
Create one for each Cesium Camera that should have HDMI / DisplayPort as output using nDisplay.

nDisplay config

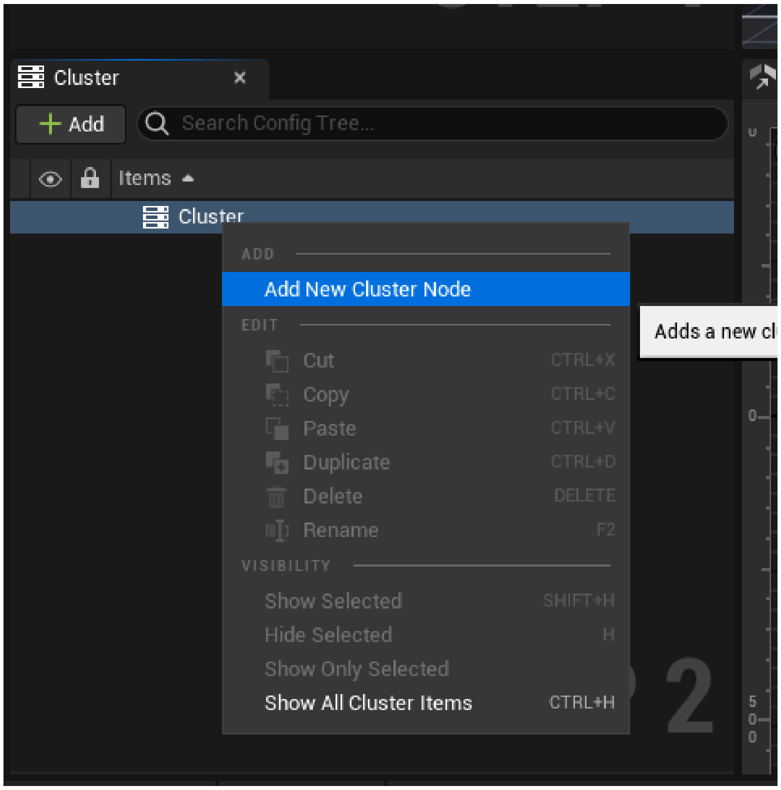
Create **nDisplay config** asset in Content browser, Right click in Content browser and select nDisplay config under **nDisplay -> nDisplay config**

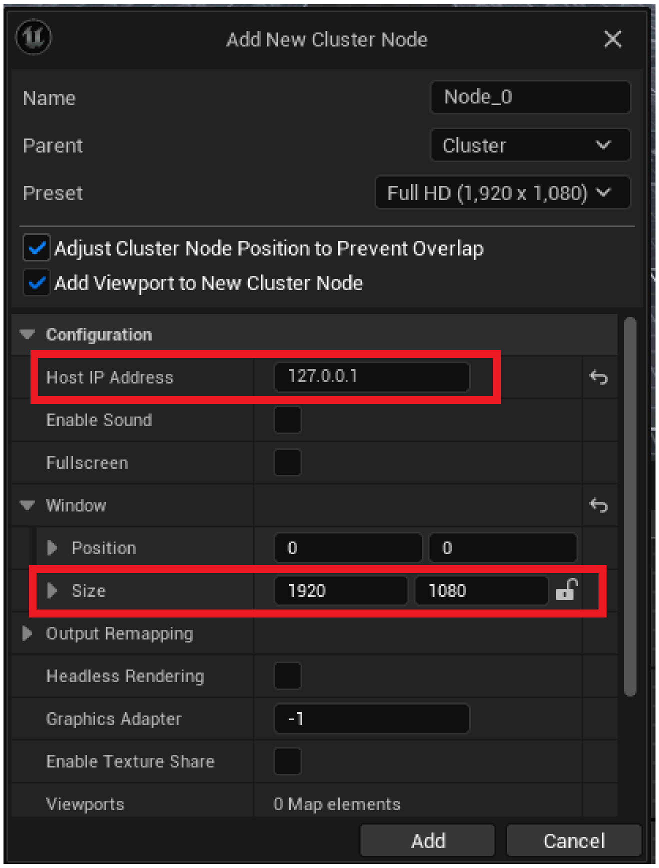


create new config


Pick a starting point for your nDisplay Config
Copy Existing Configuration
Copies an existing nDisplay configuration.
Create New Config
Create a new nDisplay configuration.
Next >
Finish
Cancel

Add New Cluster Node


Cluster
Add
Search Config Tree...
Items
Cluster
ADD
Add New Cluster Node
Adds a new cl
EDIT
Cut CTRL+X
Copy CTRL+C
Paste CTRL+V
Duplicate CTRL+D
Delete DELETE
Rename F2
VISIBILITY
Show Selected SHIFT+H
Hide Selected H
Show Only Selected
Show All Cluster Items CTRL+H



Add New Cluster Node

Name: Node_0

Parent: Cluster

Preset: Full HD (1,920 x 1,080)

☑ Adjust Cluster Node Position to Prevent Overlap

☑ Add Viewport to New Cluster Node

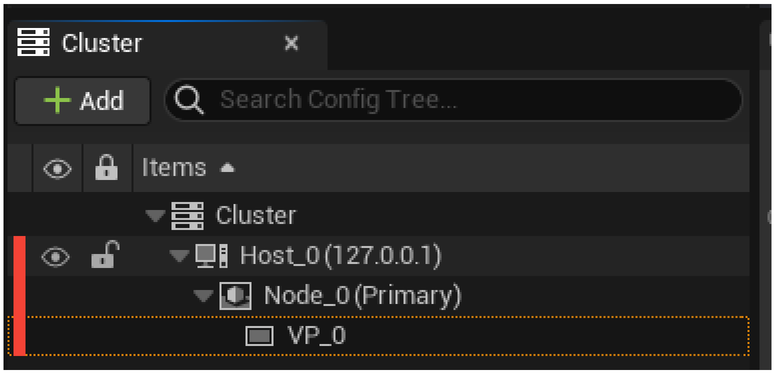
Configuration

Host IP Address	127.0.0.1
Enable Sound	□
Fullscreen	□
Window	
Position	0 0
Size	1920 1080
Output Remapping	
Headless Rendering	□
Graphics Adapter	-1
Enable Texture Share	□
Viewports	0 Map elements

Set the Host IP Address if Cluster Node is not a local host (127.0.0.1). You can set your **custom resolution** in the **Window -> Size**, this is the size of all displays connected to the system.

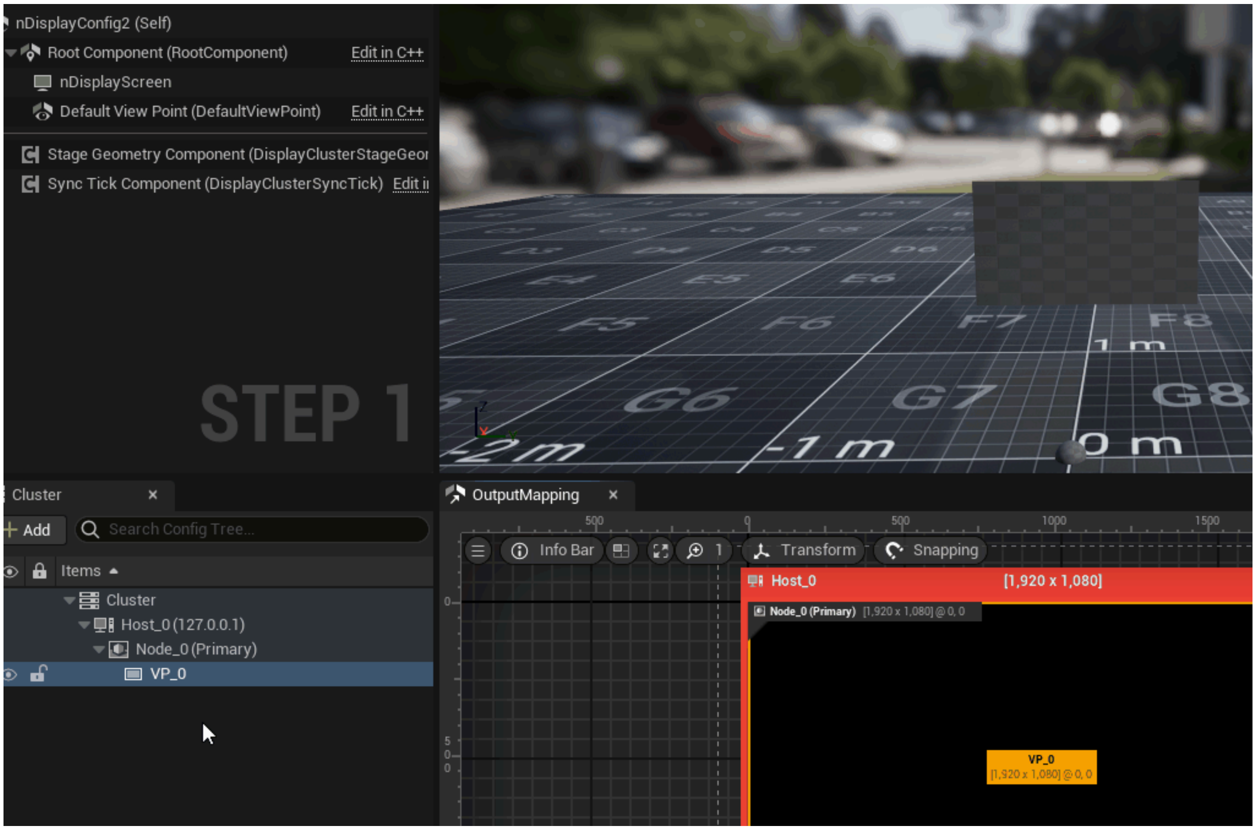
Note: Max size is 8192 x 8192 but it can be increased by DC.OverrideMaxTextureDimension

this should created VP_0

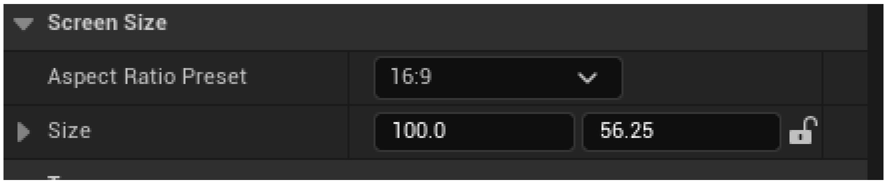


- Cluster
  - Host_0 (127.0.0.1)
    - Node_0 (Primary)
      - VP_0

Drag and drop VP_0 to the nDisplayScreen


nDisplayConfig2 (Self)
Root Component (RootComponent) Edit in C++
nDisplayScreen
Default View Point (DefaultViewPoint) Edit in C++
Stage Geometry Component (DisplayClusterStageGeor
Sync Tick Component (DisplayClusterSyncTick) Edit i
STEP 1
Cluster
Add
Search Config Tree...
Items
Cluster
Host_0 (127.0.0.1)
Node_0 (Primary)
VP_0
OutputMapping
Info Bar
Transform
Snapping
Host_0 [1,920 x 1,080]
Node_0 (Primary) [1,920 x 1,080] @ 0, 0
VP_0 [1,920 x 1,080] @ 0, 0

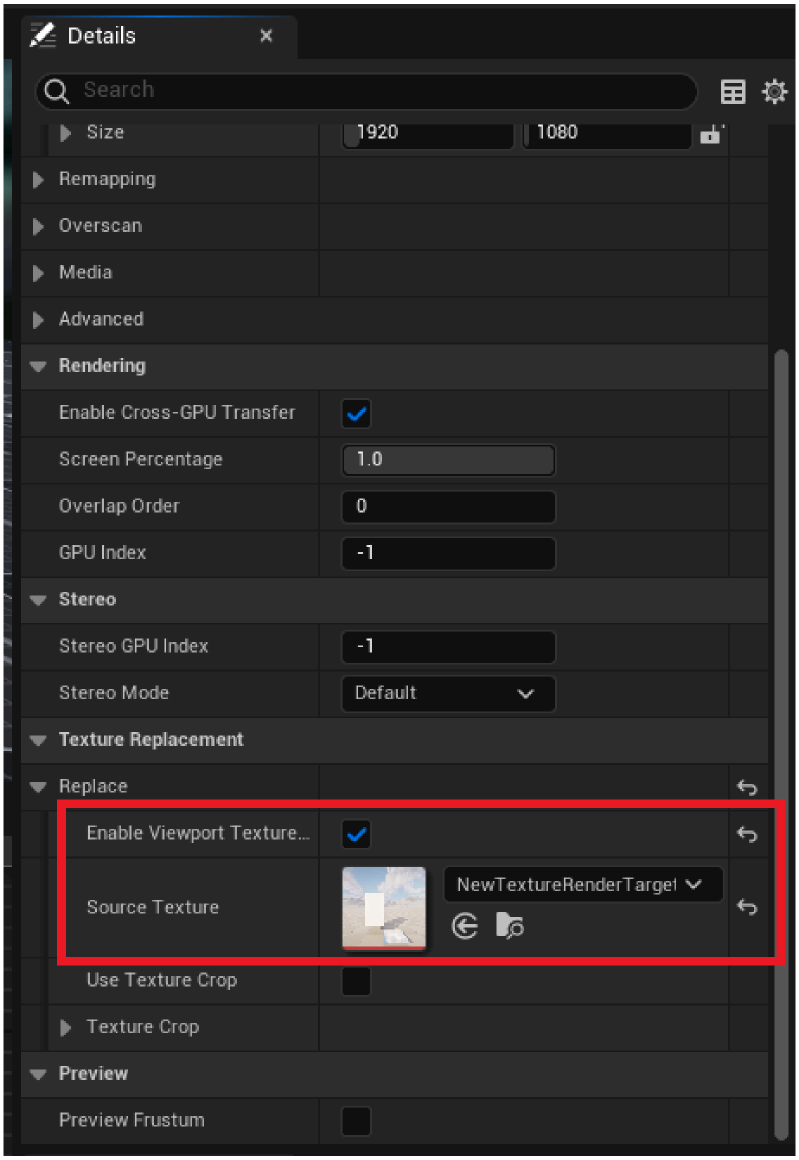
When using **custom resolution** make sure to adjust the **nDisplayScreen** size in the details panel under **Screen Size -> Size**



Screen Size

Aspect Ratio Preset	16:9
Size	100.0 | 56.25

Select the VP_0 in details panel set **Replace -> Source Texture** to the render target created before and make sure the **Enable Viewport Texture Replacement**

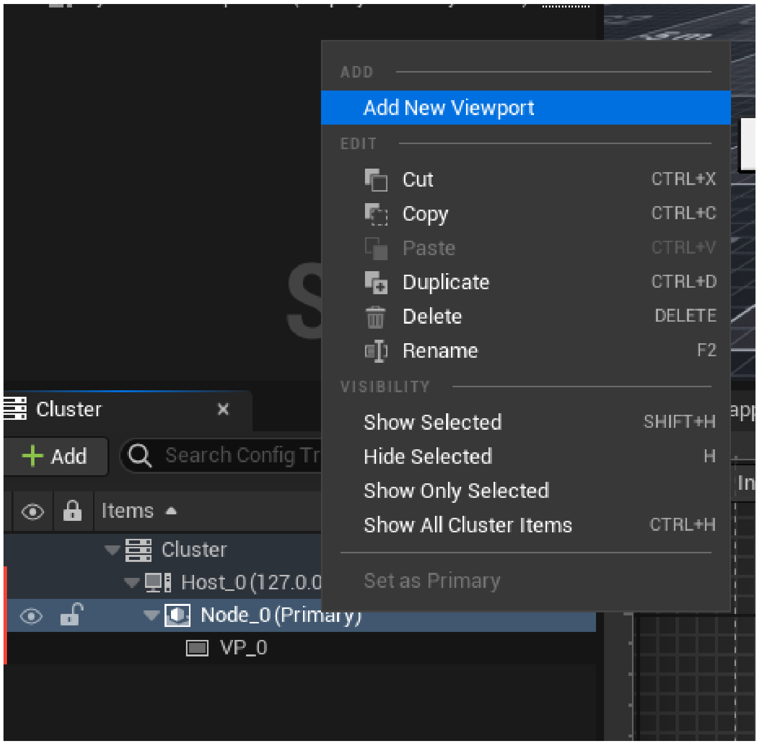


Details

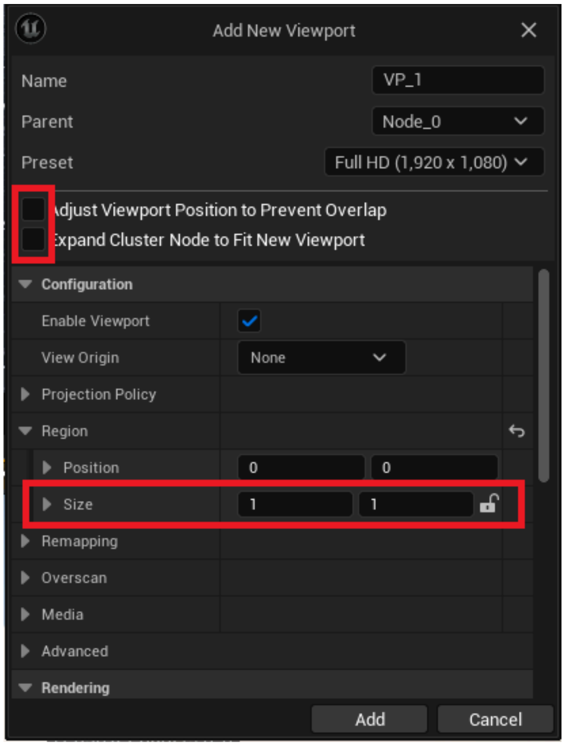
Search

Size	1920 | 1080
Remapping	
Overscan	
Media	
Advanced	
Rendering	
Enable Cross-GPU Transfer	☑
Screen Percentage	1.0
Overlap Order	0
GPU Index	-1
Stereo	
Stereo GPU Index	-1
Stereo Mode	Default
Texture Replacement	
Replace	
Enable Viewport Texture...	☑
Source Texture	NewTextureRenderTarget
Use Texture Crop	□
Texture Crop	
Preview	
Preview Frustum	□

Add second viewport (per node) that the sole purpose of it is to have a valid Viewport due to deferred rendering requiring a valid Viewport (this requirement may be removed in future releases)



with the size of 1x1, we wand the viewport to be as small as possible, and uncheck the adjust Viewport position to prevent overlapping and expand cluster node to fit new viewport.



Add New Viewport

Name: VP_1
Parent: Node_0
Preset: Full HD (1,920 x 1,080)

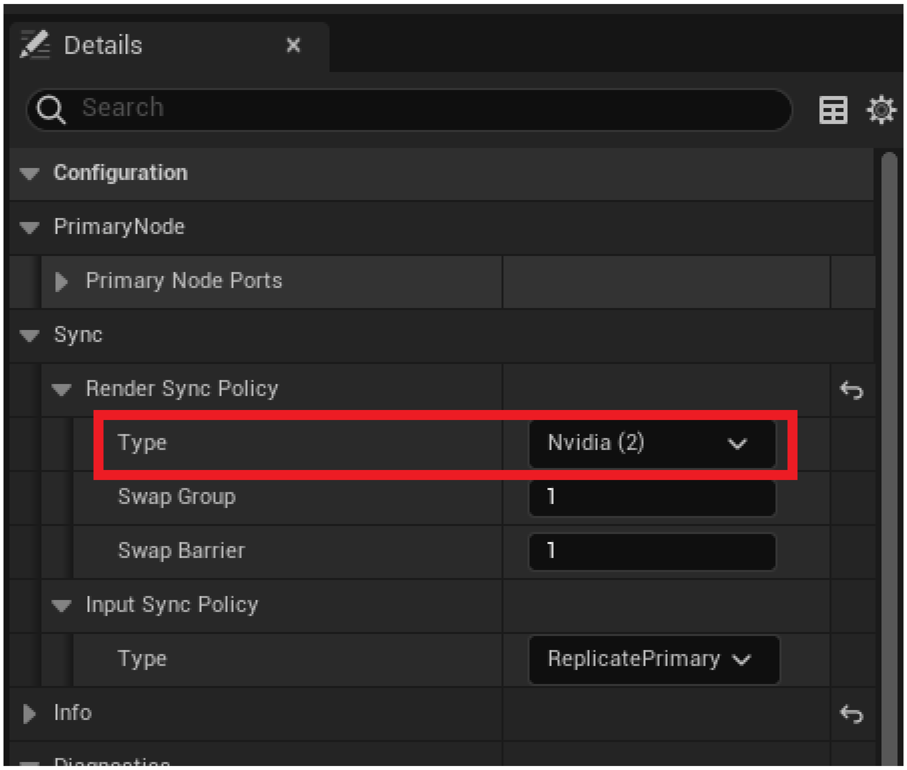
□ Adjust Viewport Position to Prevent Overlap
□ Expand Cluster Node to Fit New Viewport

Configuration
- Enable Viewport: ☑
- View Origin: None
- Projection Policy
- Region
  - Position: 0 0
  - Size: 1 1
- Remapping
- Overscan
- Media
- Advanced

Rendering

Make sure to also drag and drop the this Viewport on the **nDisplayScreen**

Select **Cluster** in details panel under **Sync -> Render Sync Policy -> Type** select **Nvidia (2)**



Render Sync Policy
Type Nvidia (2)
Swap Group 1
Swap Barrier 1
Input Sync Policy
Type ReplicatePrimary

📝 for the Nvidia sync to work correctly it requires the display resolution to much the display resolution, also it does not work properly when application is out of focused or remote desktop user is connected (example: TeamViewer user is connected).

Save and Compile

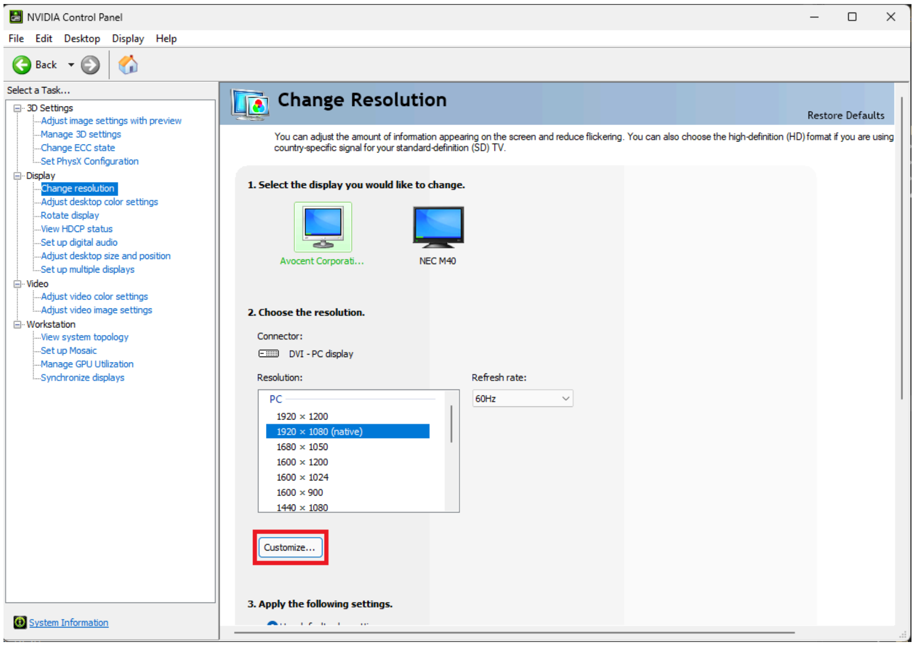
drag and drop the **nDisplay config asset** to scene (level)

Multiple display setup

On system with multiple displays you want to set main Node for **Main Display** as the nDisplay window(s) appears on the Main Display, for secondary display you want to add additional Node set it up same as the first one and then use the Nvidia mosaic to map it to the proper display.

Interlaced display

If your specific use case requires interlaced output you can enabled that for the display in the **Change resolution -> Customize**



Change Resolution

You can adjust the amount of information appearing on the screen and reduce flickering. You can also choose the high-definition (HD) format if you are using country-specific signal for your standard-definition (SD) TV.

1. Select the display you would like to change.

2. Choose the resolution.

Connector: DVI - PC display

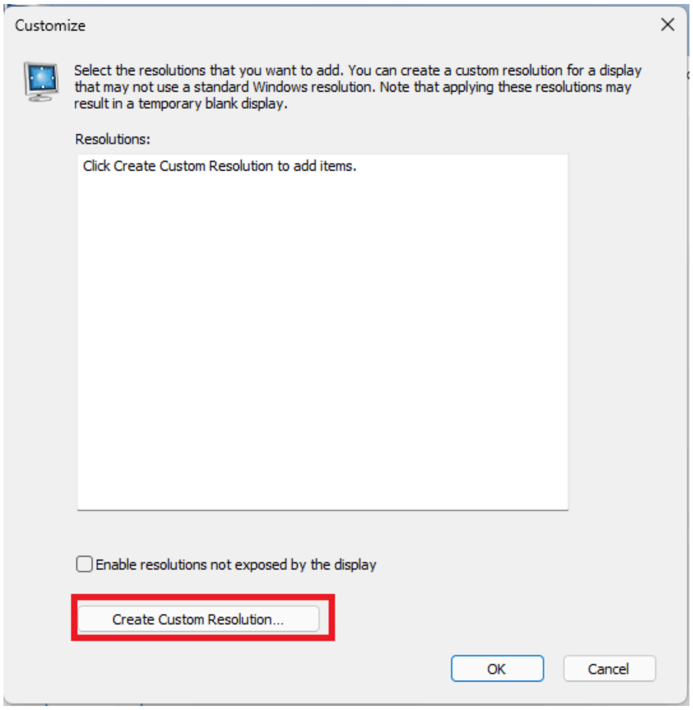
Resolution: PC

- 1920 × 1200
- 1920 × 1080 (native)
- 1680 × 1050
- 1600 × 1200
- 1600 × 1024
- 1600 × 900
- 1440 × 1080

Refresh rate: 60Hz

3. Apply the following settings.

Click the **Create Custom Resolution…**

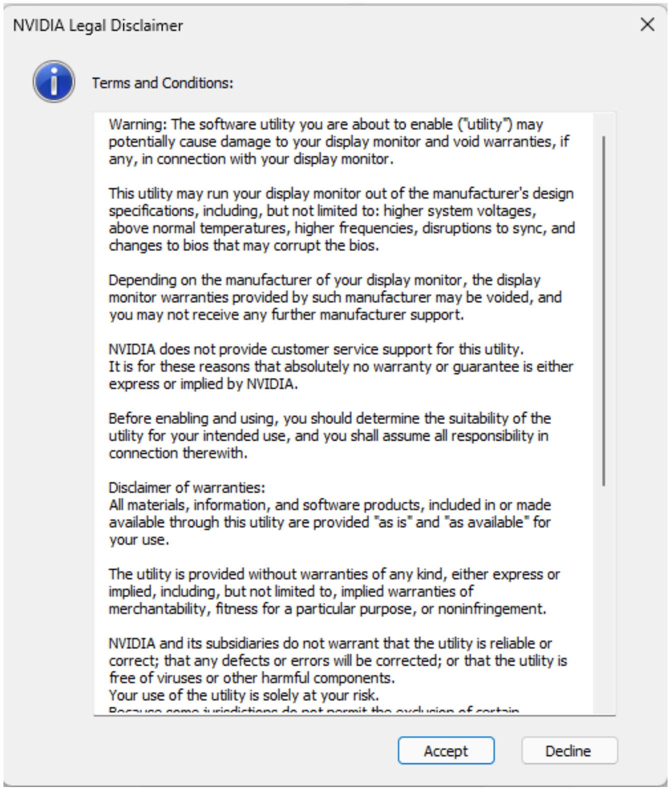

Customize
Resolutions:
Enable resolutions not exposed by the display
Create Custom Resolution...
OK
Cancel

Select the resolutions that you want to add. You can create a custom resolution for a display that may not use a standard Windows resolution. Note that applying these resolutions may result in a temporary blank display.

Click Create Custom Resolution to add items.

□ Enable resolutions not exposed by the display

Accept the Legal Disclaimer (make sure you know what you doing)



Warning: The software utility you are about to enable ("utility") may potentially cause damage to your display monitor and void warranties, if any, in connection with your display monitor.

This utility may run your display monitor out of the manufacturer's design specifications, including, but not limited to: higher system voltages, above normal temperatures, higher frequencies, disruptions to sync, and changes to bios that may corrupt the bios.

Depending on the manufacturer of your display monitor, the display monitor warranties provided by such manufacturer may be voided, and you may not receive any further manufacturer support.

NVIDIA does not provide customer service support for this utility.
It is for these reasons that absolutely no warranty or guarantee is either express or implied by NVIDIA.

Before enabling and using, you should determine the suitability of the utility for your intended use, and you shall assume all responsibility in connection therewith.

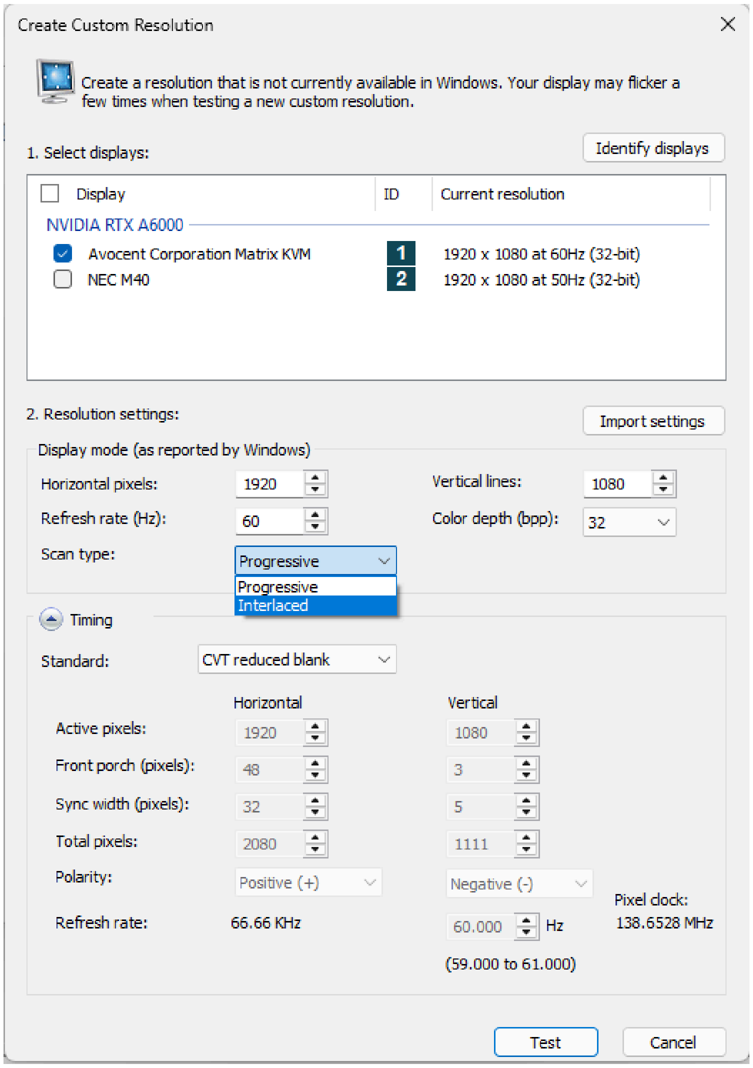
Disclaimer of warranties:
All materials, information, and software products, included in or made available through this utility are provided "as is" and "as available" for your use.

The utility is provided without warranties of any kind, either express or implied, including, but not limited to, implied warranties of merchantability, fitness for a particular purpose, or noninfringement.

NVIDIA and its subsidiaries do not warrant that the utility is reliable or correct; that any defects or errors will be corrected; or that the utility is free of viruses or other harmful components.
Your use of the utility is solely at your risk.
Because some jurisdictions do not permit the exclusion of certain

Select the correct display

In the **Scan type** you can select **Interlaced**



Create Custom Resolution

Create a resolution that is not currently available in Windows. Your display may flicker a few times when testing a new custom resolution.

1. Select displays:

□ Display	ID	Current resolution
NVIDIA RTX A6000		
☑ Avocent Corporation Matrix KVM	1	1920 x 1080 at 60Hz (32-bit)
□ NEC M40	2	1920 x 1080 at 50Hz (32-bit)

2. Resolution settings:

Display mode (as reported by Windows)

Horizontal pixels: 1920

Vertical lines: 1080

Refresh rate (Hz): 60

Color depth (bpp): 32

Scan type: Progressive

Timing

Standard: CVT reduced blank

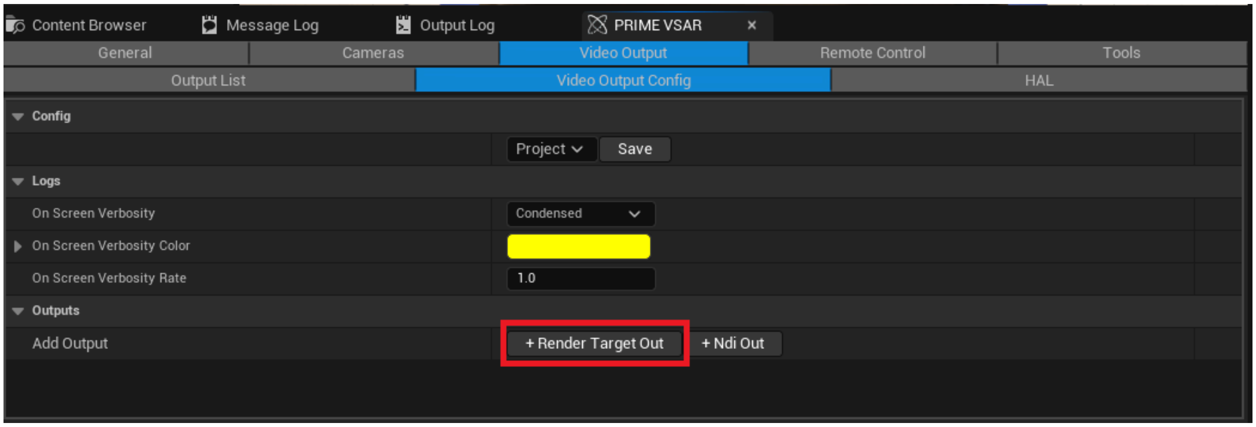
	Horizontal	Vertical
Active pixels:	1920	1080
Front porch (pixels):	48	3
Sync width (pixels):	32	5
Total pixels:	2080	1111
Polarity:	Positive (+)	Negative (-)
Refresh rate:	66.66 KHz	60.000 Hz

(59.000 to 61.000)

Pixel clock:
138.6528 MHz

Cesium Camera Output

in **PRIME VSAR config** add **Render Target Output**



Config

Project Save

Logs

On Screen Verbosity: Condensed

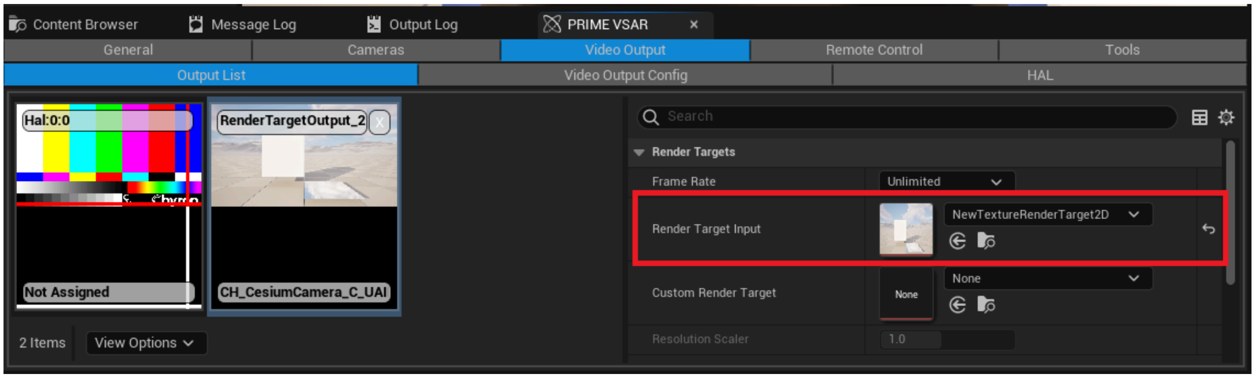
On Screen Verbosity Color

On Screen Verbosity Rate: 1.0

Outputs

Add Output: + Render Target Out + Ndi Out

Set the Render Target input to the before created Render target



Render Targets

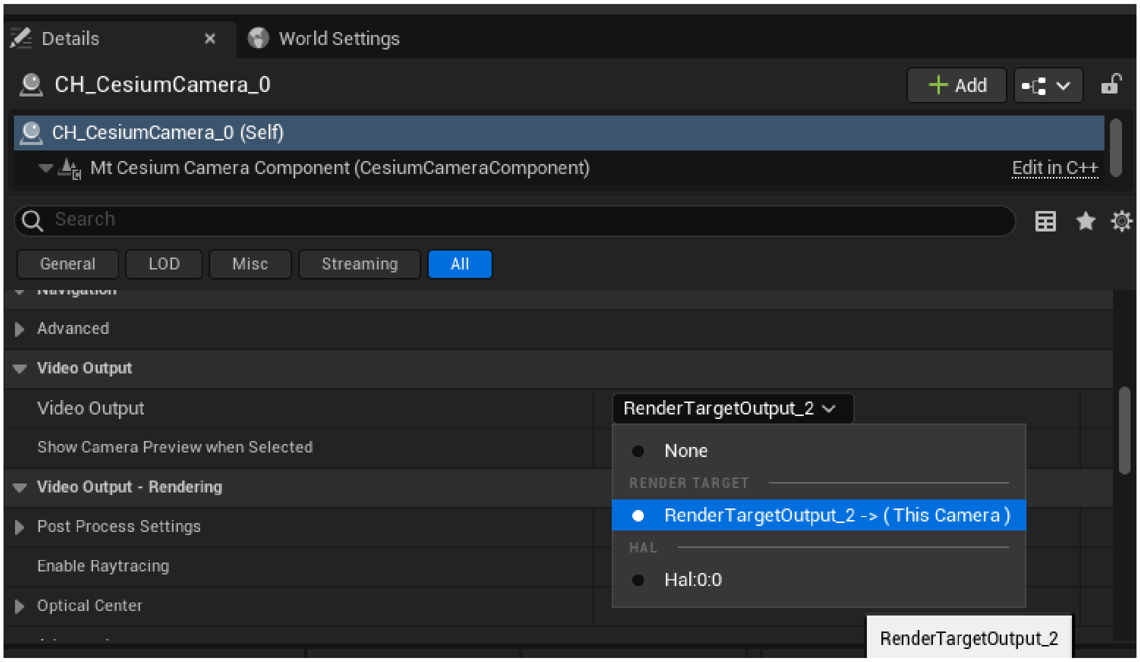
Frame Rate: Unlimited

Render Target Input: NewTextureRenderTarget2D

Custom Render Target: None

Resolution Scaler: 1.0

In the corresponding Cesium Camera in details panel select in the **Video Output -> RenderTargetOutput**



Details × | World Settings

CH_CesiumCamera_0

CH_CesiumCamera_0 (Self)

Mt Cesium Camera Component (CesiumCameraComponent) — Edit in C++

Navigation

Advanced

Video Output

Video Output: RenderTargetOutput_2

Show Camera Preview when Selected

Video Output - Rendering

Post Process Settings

Enable Raytracing

Optical Center

None

RENDER TARGET

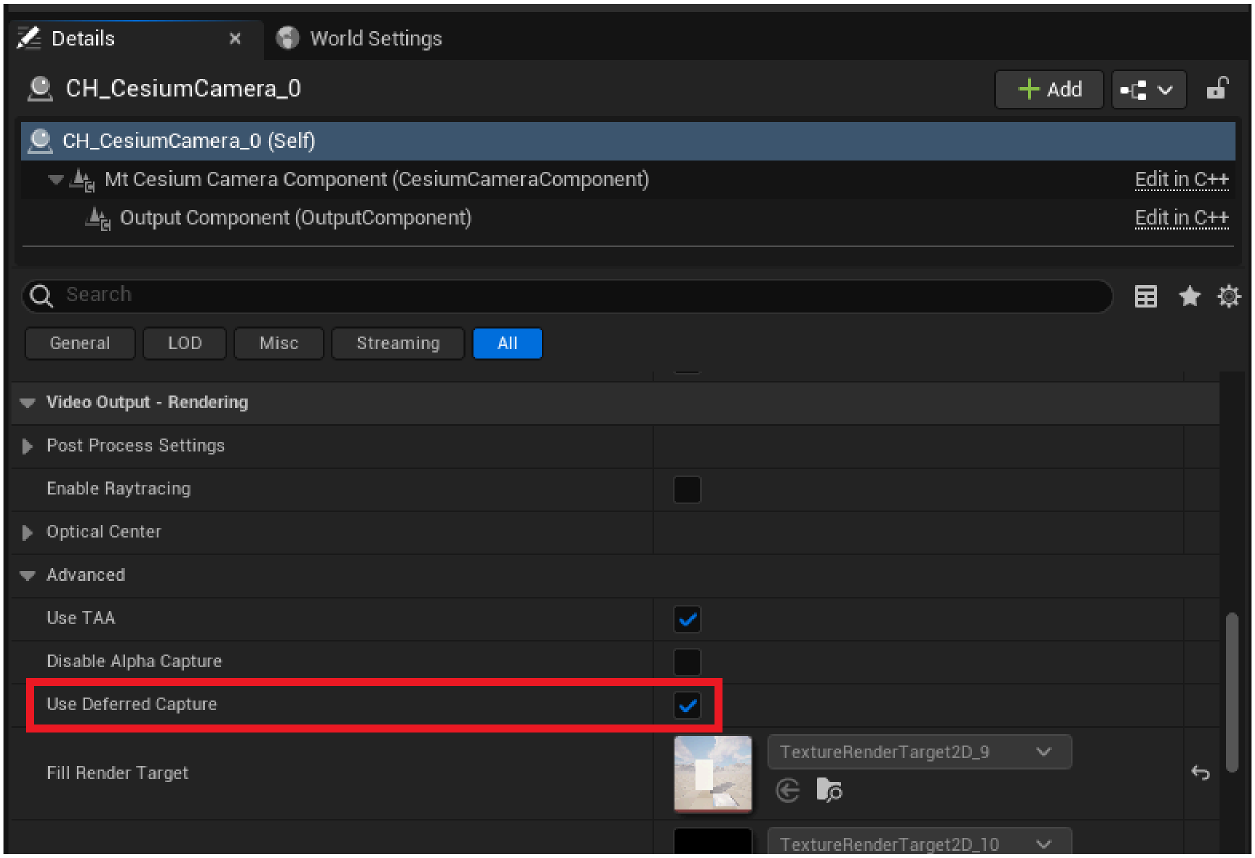
● RenderTargetOutput_2 -> (This Camera)

HAL

Hal:0:0

RenderTargetOutput_2

Make sure that the deferred capture is enabled under **Video Output - Rendering -> Advanced -> Use Deferred Capture**



Details × | World Settings

CH_CesiumCamera_0

CH_CesiumCamera_0 (Self)

Mt Cesium Camera Component (CesiumCameraComponent) — Edit in C++

Output Component (OutputComponent) — Edit in C++

Video Output - Rendering

Post Process Settings

Enable Raytracing □

Optical Center

Advanced

Use TAA ☑

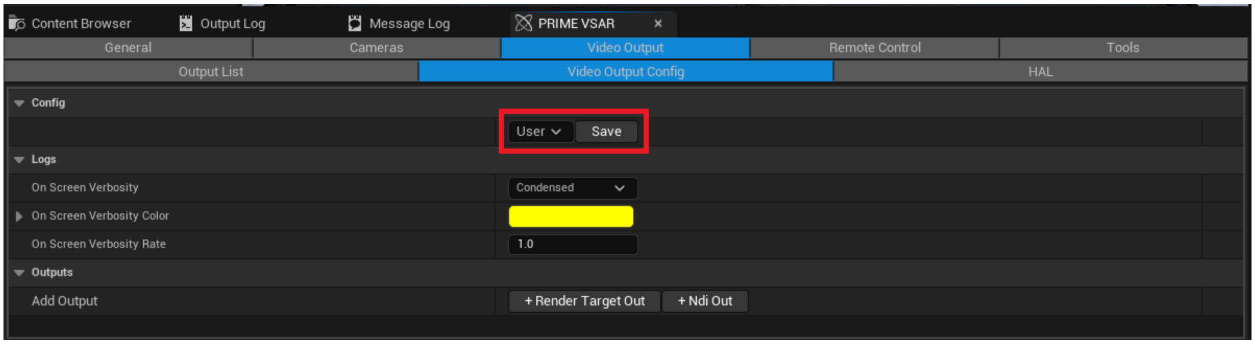
Disable Alpha Capture □

Use Deferred Capture ☑

Fill Render Target: TextureRenderTarget2D_9

TextureRenderTarget2D_10

Make sure to save the output configuration

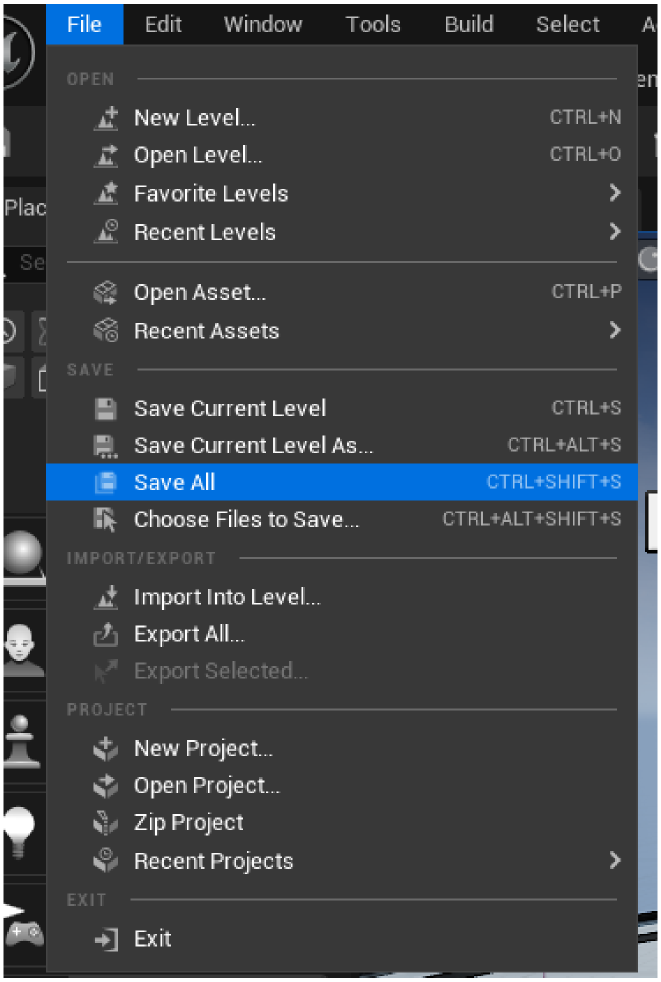


On Screen Verbosity: Condensed

On Screen Verbosity Color

On Screen Verbosity Rate: 1.0

Make sure to **Save all**

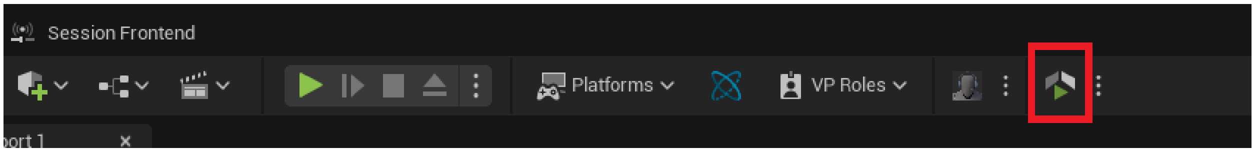


Launching nDisplay

While launching nDisplay it launches a standalone application that may take some time to boot up and will appear black while it is loading.

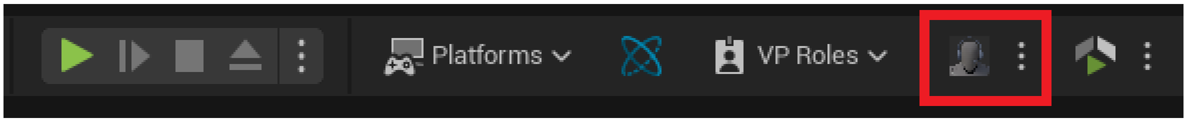
There are two main ways to launch nDisplay configuration:

using Launch nDisplay



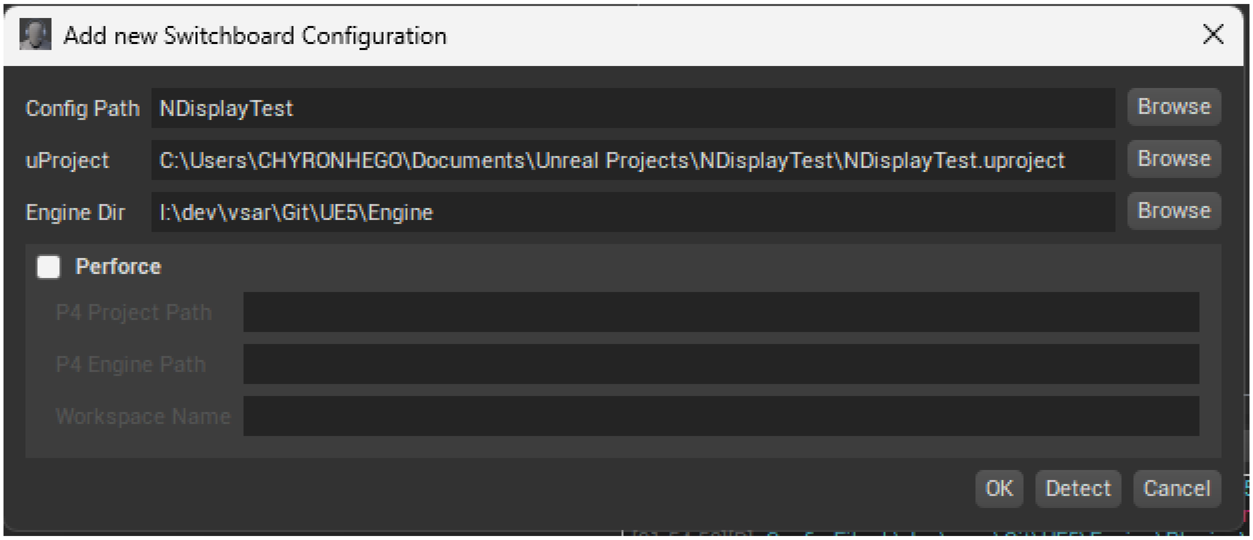
It requires nDisplay config to be in the scene. When using **Launch nDisplay** the **Editor** can consume SDI inputs/outputs due to the Editor and standalone application having the same settings.

using Switchboard



First launch of the switchboard requires installation.

Select .uproject file for project if not populated automatically



Add new Switchboard Configuration

Config Path NDisplayTest

uProject C:\Users\CHYRONHEGO\Documents\Unreal Projects\NDisplayTest\NDisplayTest.uproject

Engine Dir I:\dev\vsar\Git\UE5\Engine

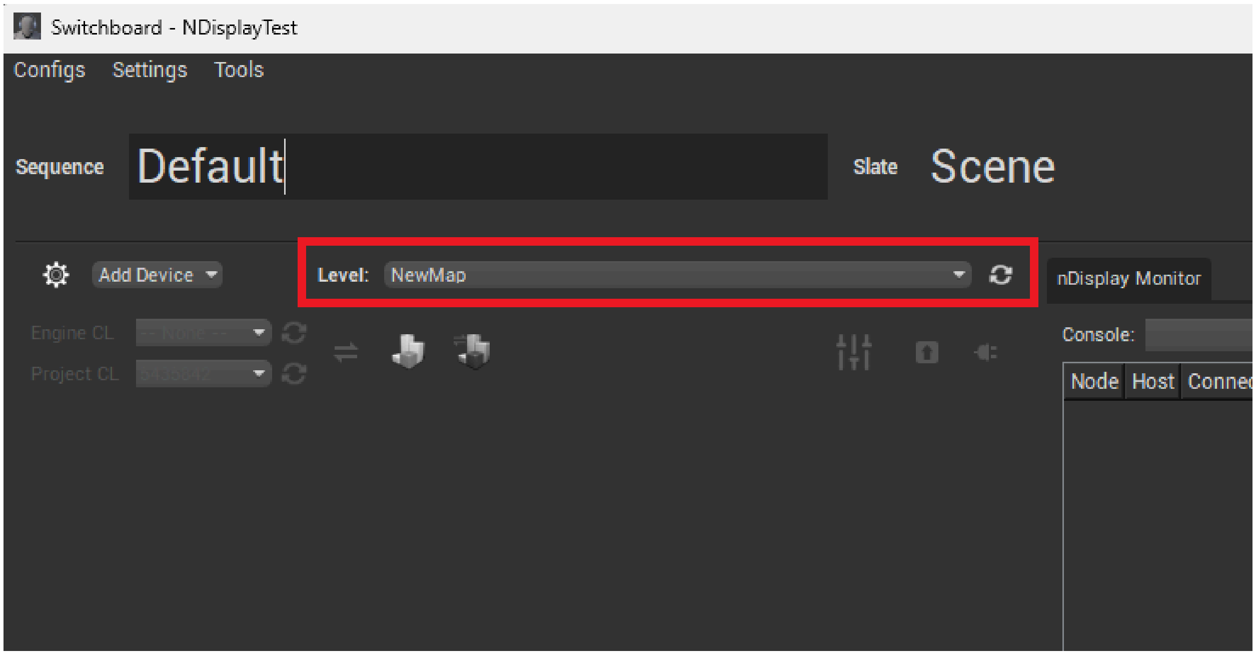
□ Perforce

P4 Project Path

P4 Engine Path

Workspace Name

select correct level

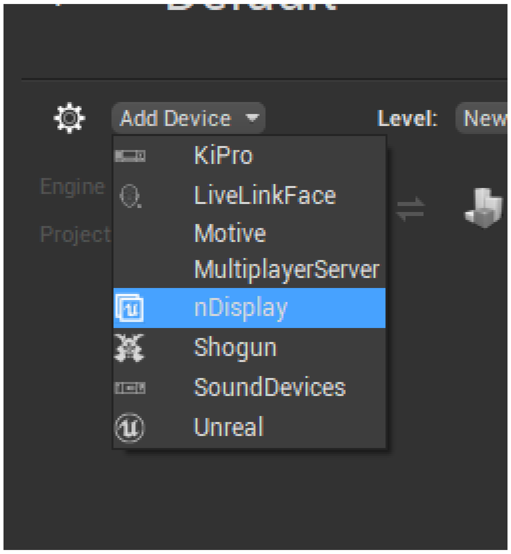


Sequence Default

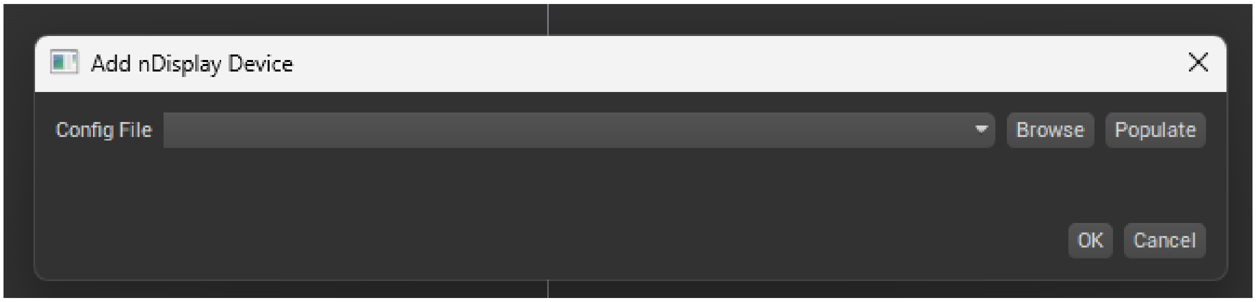
Slate Scene

Level: NewMap

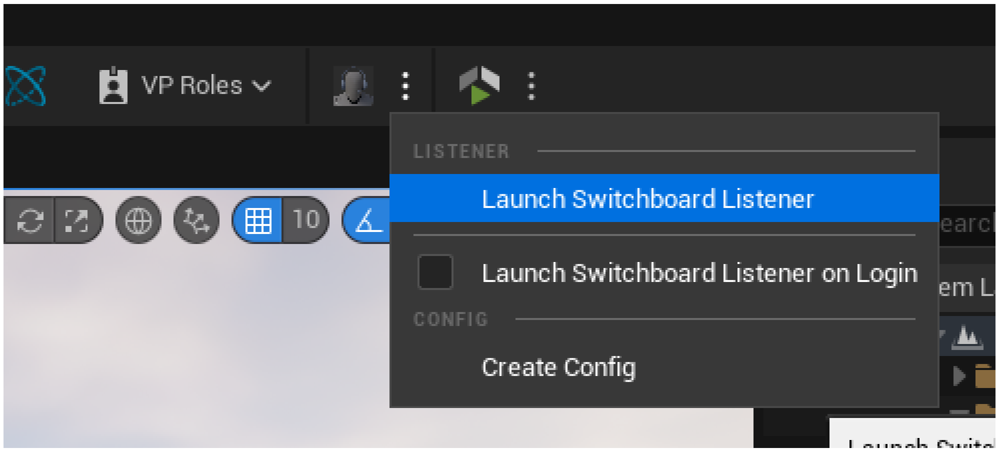
Add Device **nDisplay** one for each



Select **nDisplay Config** file

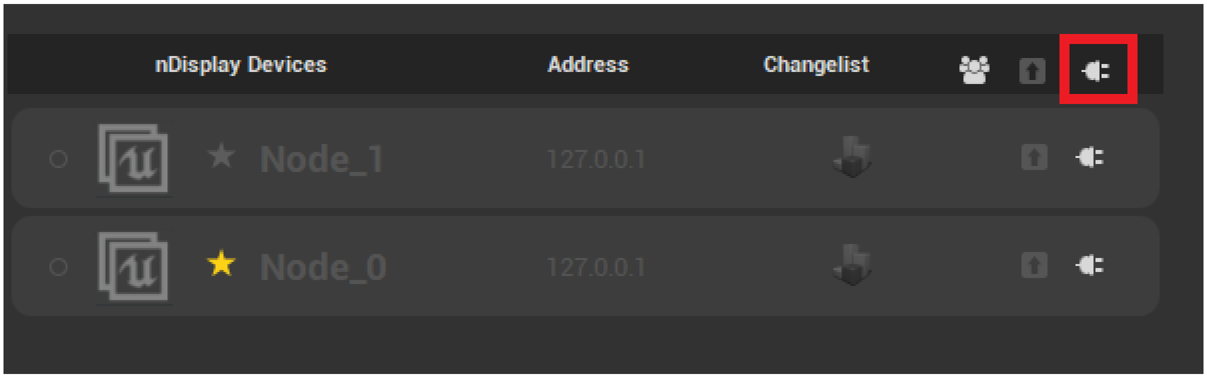


Launch the switchboard listener on all **VSAR** machines you want the nDisplay config to run on


VP Roles
10
LISTENER
Launch Switchboard Listener
Launch Switchboard Listener on Login
CONFIG
Create Config

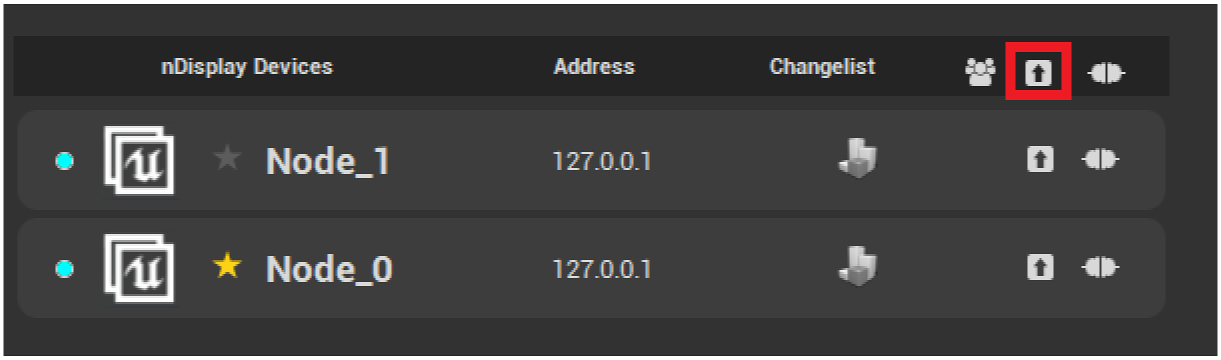
Close **VSAR** on all VSAR machines you want the nDisplay config to run on

Click the connect button to try to connect to all the nodes



nDisplay Devices	Address	Changelist
○ ★ Node_1	127.0.0.1	
○ ★ Node_0	127.0.0.1	

you can start all nodes using the start button (up arrow)



nDisplay Devices	Address	Changelist
★ Node_1	127.0.0.1	
★ Node_0	127.0.0.1	

you can rebuild the node to update it to the project changes, unfortunately this requires development environment

Setting up development environment

Development environment for Unreal Engine 5.3.2 requires:

- Visual Studio 2022 v17.4 or newer
- Windows SDK 10.0.18362 or newer
- LLVM clang 14.0.1
- .NET 4.6.2 Targeting Pack
- .NET 6.0

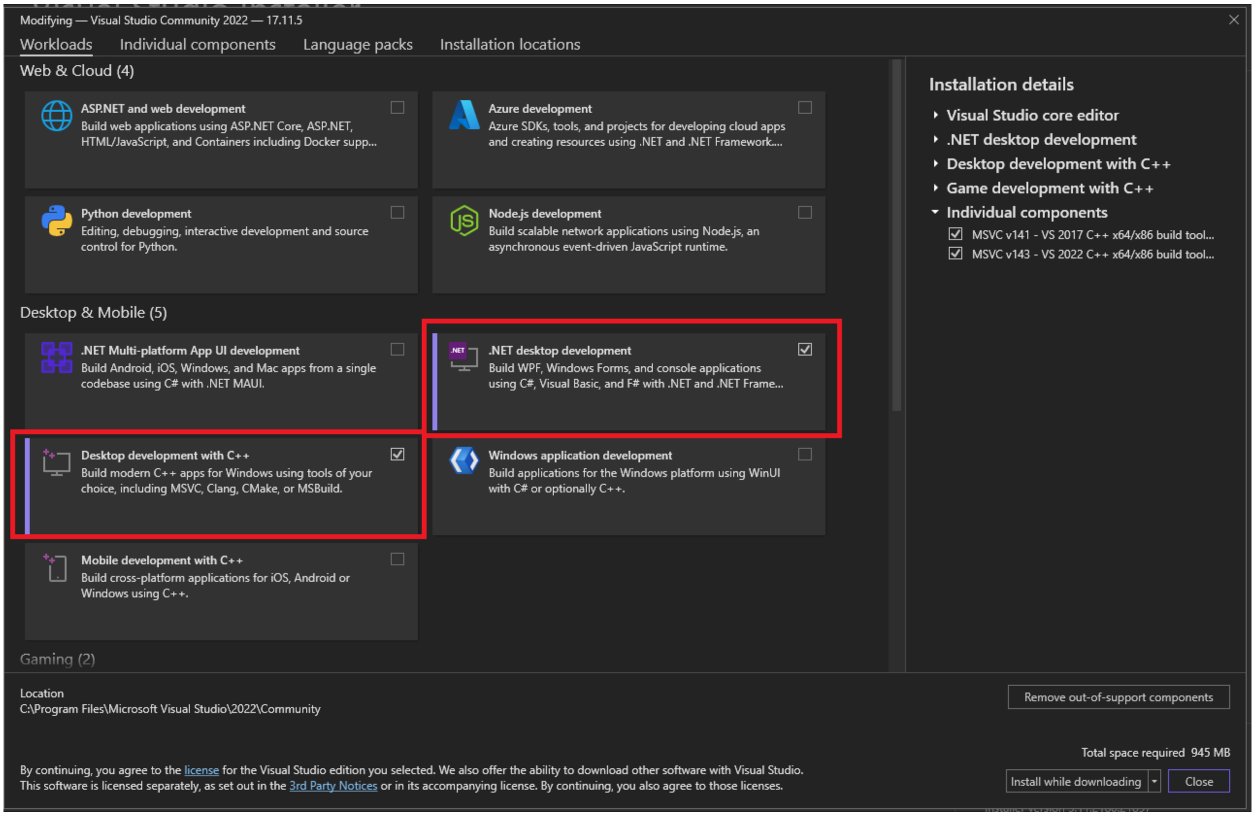
is may change depending on the base Unreal Engine Version

Visual Studio

download Visual studio community 2022

and install it

while installing select **.NET desktop development, Desktop development with c++** and **Game development with c++**



Modifying — Visual Studio Community 2022 — 17.11.5

Web & Cloud (4)

- ASP.NET and web development ☐
  Build web applications using ASP.NET Core, ASP.NET, HTML/JavaScript, and Containers including Docker supp...
- Azure development ☐
  Azure SDKs, tools, and projects for developing cloud apps and creating resources using .NET and .NET Framework....
- Python development ☐
  Editing, debugging, interactive development and source control for Python.
- Node.js development ☐
  Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.

Desktop & Mobile (5)

- .NET Multi-platform App UI development ☐
  Build Android, iOS, Windows, and Mac apps from a single codebase using C# with .NET MAUI.
- .NET desktop development ☑
  Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...
- Desktop development with C++ ☑
  Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.
- Windows application development ☐
  Build applications for the Windows platform using WinUI with C# or optionally C++.
- Mobile development with C++ ☐
  Build cross-platform applications for iOS, Android or Windows using C++.

Gaming (2)

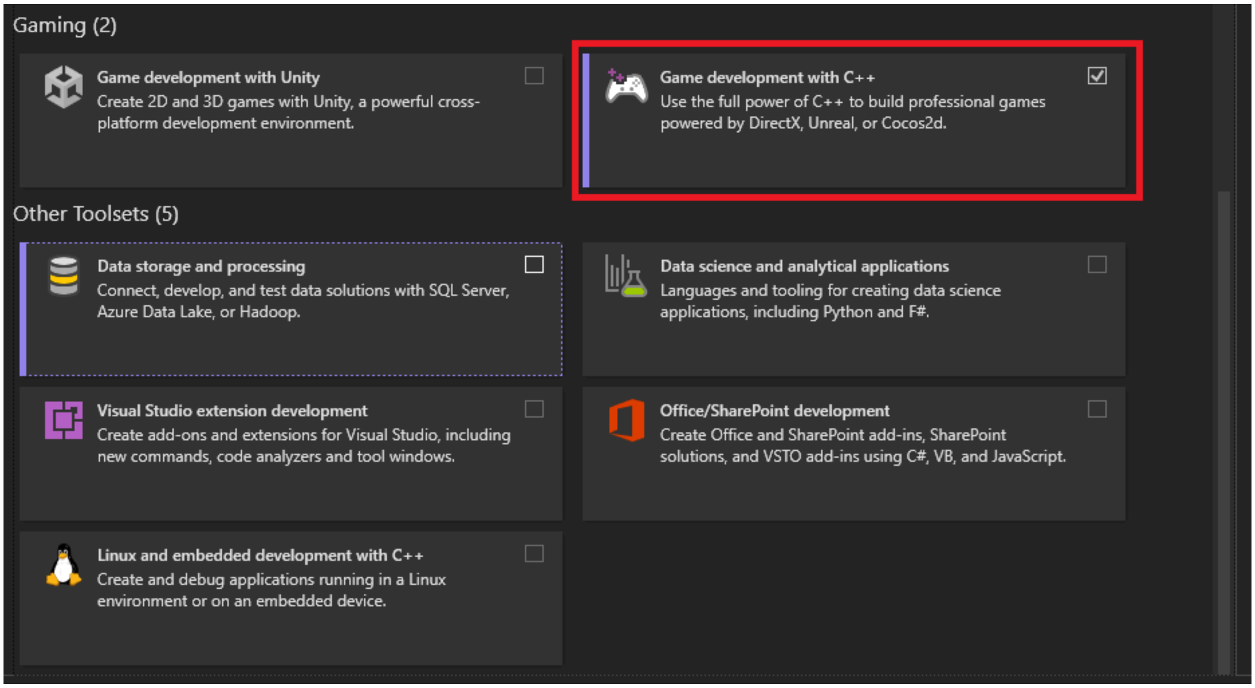
Installation details

- Visual Studio core editor
- .NET desktop development
- Desktop development with C++
- Game development with C++
- Individual components
  - ☑ MSVC v141 - VS 2017 C++ x64/x86 build tool...
  - ☑ MSVC v143 - VS 2022 C++ x64/x86 build tool...

Location
C:\Program Files\Microsoft Visual Studio\2022\Community

Total space required 945 MB

By continuing, you agree to the license for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the 3rd Party Notices or in its accompanying license. By continuing, you also agree to those licenses.



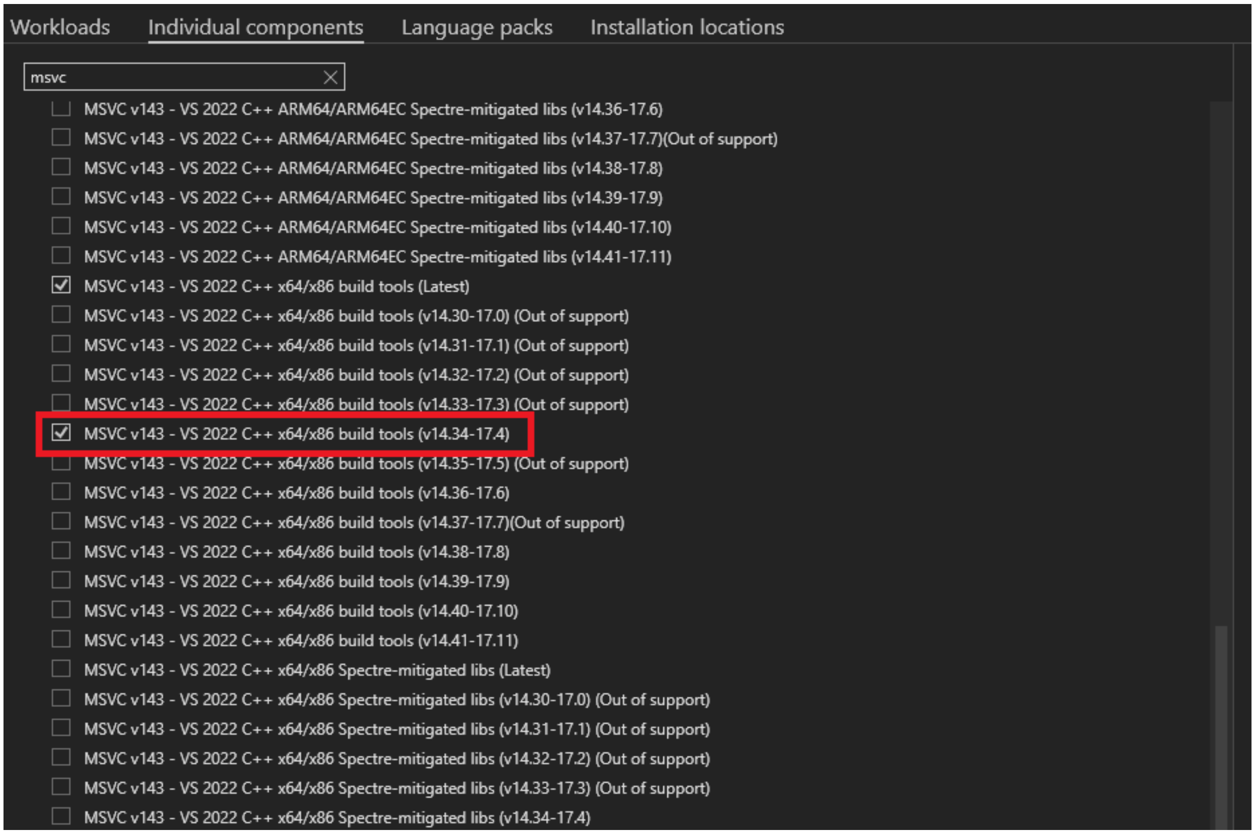
Gaming (2)

- Game development with Unity ☐
  Create 2D and 3D games with Unity, a powerful cross-platform development environment.
- Game development with C++ ☑
  Use the full power of C++ to build professional games powered by DirectX, Unreal, or Cocos2d.

Other Toolsets (5)

- Data storage and processing ☐
  Connect, develop, and test data solutions with SQL Server, Azure Data Lake, or Hadoop.
- Data science and analytical applications ☐
  Languages and tooling for creating data science applications, including Python and F#.
- Visual Studio extension development ☐
  Create add-ons and extensions for Visual Studio, including new commands, code analyzers and tool windows.
- Office/SharePoint development ☐
  Create Office and SharePoint add-ins, SharePoint solutions, and VSTO add-ins using C#, VB, and JavaScript.
- Linux and embedded development with C++ ☐
  Create and debug applications running in a Linux environment or on an embedded device.

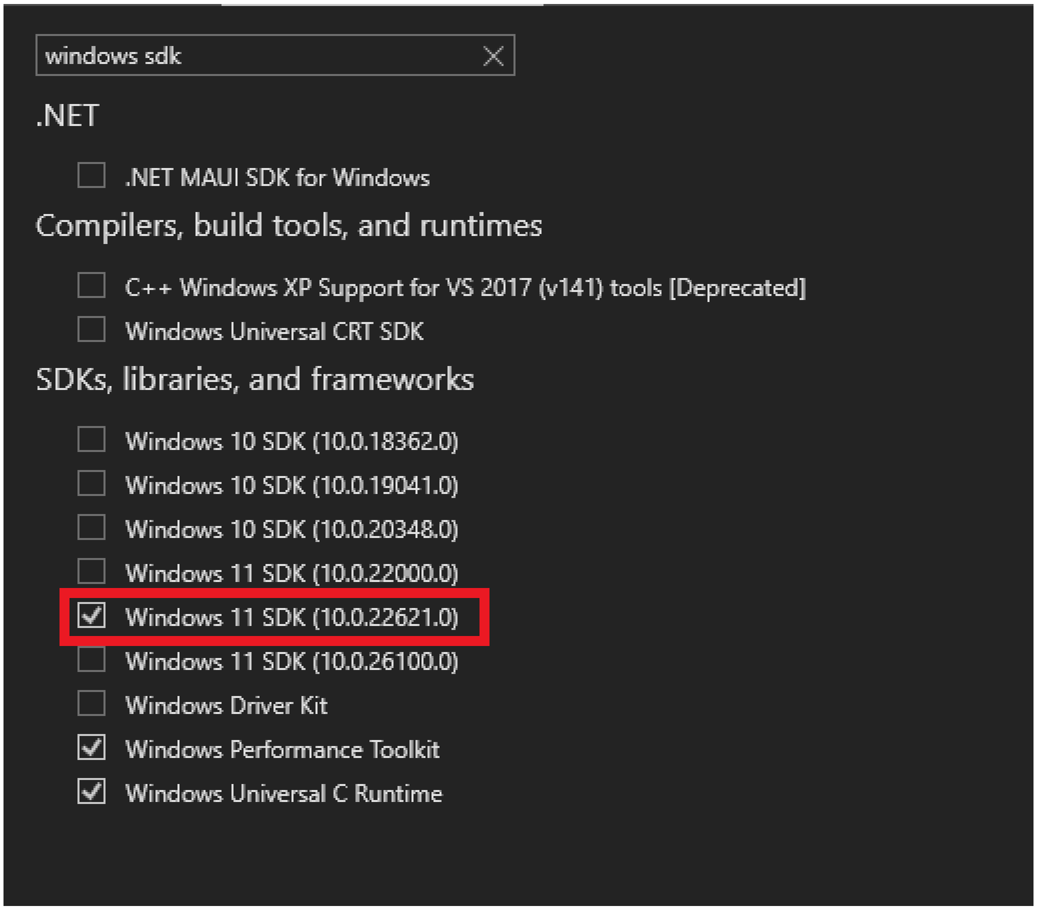
then select individual components search for MSVC and select **MSVC v143 - VS 2022 c++ x64/x86 build tools (v14.34-17.4)**



msvc

- □ MSVC v143 - VS 2022 C++ ARM64/ARM64EC Spectre-mitigated libs (v14.36-17.6)
- □ MSVC v143 - VS 2022 C++ ARM64/ARM64EC Spectre-mitigated libs (v14.37-17.7)(Out of support)
- □ MSVC v143 - VS 2022 C++ ARM64/ARM64EC Spectre-mitigated libs (v14.38-17.8)
- □ MSVC v143 - VS 2022 C++ ARM64/ARM64EC Spectre-mitigated libs (v14.39-17.9)
- □ MSVC v143 - VS 2022 C++ ARM64/ARM64EC Spectre-mitigated libs (v14.40-17.10)
- □ MSVC v143 - VS 2022 C++ ARM64/ARM64EC Spectre-mitigated libs (v14.41-17.11)
- ☑ MSVC v143 - VS 2022 C++ x64/x86 build tools (Latest)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.30-17.0) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.31-17.1) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.32-17.2) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.33-17.3) (Out of support)
- ☑ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.34-17.4)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.35-17.5) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.36-17.6)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.37-17.7)(Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.38-17.8)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.39-17.9)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.40-17.10)
- □ MSVC v143 - VS 2022 C++ x64/x86 build tools (v14.41-17.11)
- □ MSVC v143 - VS 2022 C++ x64/x86 Spectre-mitigated libs (Latest)
- □ MSVC v143 - VS 2022 C++ x64/x86 Spectre-mitigated libs (v14.30-17.0) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 Spectre-mitigated libs (v14.31-17.1) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 Spectre-mitigated libs (v14.32-17.2) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 Spectre-mitigated libs (v14.33-17.3) (Out of support)
- □ MSVC v143 - VS 2022 C++ x64/x86 Spectre-mitigated libs (v14.34-17.4)

then search for windows sdk and select **Windows 11 SDK (10.0.22621.0)** and make sure that no other sdk is check



windows sdk

.NET

□ .NET MAUI SDK for Windows

Compilers, build tools, and runtimes

□ C++ Windows XP Support for VS 2017 (v141) tools [Deprecated]
□ Windows Universal CRT SDK

SDKs, libraries, and frameworks

□ Windows 10 SDK (10.0.18362.0)
□ Windows 10 SDK (10.0.19041.0)
□ Windows 10 SDK (10.0.20348.0)
□ Windows 11 SDK (10.0.22000.0)
☑ Windows 11 SDK (10.0.22621.0)
□ Windows 11 SDK (10.0.26100.0)
□ Windows Driver Kit
☑ Windows Performance Toolkit
☑ Windows Universal C Runtime

and press **install**

restart the machine after the installation