Pixel Processing

Juggler

User Guide

7THSENSE

Juggler : User Guide

Trademark Information

The 7thsense logo, and various hardware and software product names are trademarks of 7thSense Design Ltd. Product or company names that may be mentioned in 7thSense publications are tradenames or trademarks of their respective owners, and such trademarks may also be registered in their respective countries. Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Copyright Information

Printed: April 2021

This edition is for software version Compere 1.0.2
Document ref.: M512-3

E: info@7thsense.one
W: 7thsense.one

Contents

Contents

Introduction

Juggler™ and Compere™ go together. Juggler is the hardware that processes the pixels and Compere, like an event manager, is the control software and user interface that tells one or more Jugglers what to do. Together, any range of digital inputs can be transformed and directed to an any range of outputs, because Juggler modules can be daisy-chained on a data bus to expand inputs, outputs and bandwidth as required. Compere does much more besides in its Visualise role, but this guide focuses on Juggler system set up.

Juggler is a highly modular FPGA-based pixel processor, providing the highest hardware performance and efficiency, combined with ultimate connectivity. It is specified according to application, with a very wide choice of input and output options: DisplayPort™ 1.2, HDMI™ 2.0, SDI, and with Video over IP in development. The Juggler system can process a data bandwidth of approximately 14 GB/s. As an example, that equates to a canvas size of 16K × 4K at 60 fps 10-bit 4:4:4.

The configuration of a particular Juggler module will optimise use of the available bandwidth according to requirement, and the Juggler concept is one of tremendous flexibility. Modules can be daisy-chained via a high-speed box-to-box fibre-optic link, to expand I/O type and capacity. Juggler can also display all inputs via an NDI thumbnail matrix on the front panel.

Juggler and Compere

For a first-time user, the concepts of the pixel processor can seem a bit confusing, so here is an overall picture of how the key components work together.

Why Juggler?

Juggler is not just a standalone rack module. We imagine it as stage performer in an event, where as many Jugglers and other performers as you want can join, take part, or leave, all under the direction of a Compere, or stage manager that holds the event together.

Juggler is a modular hardware base for full high-bit depth pixel-based colour processing, warp and blend. It accepts a scalable number of inputs and delivers a scalable number of outputs, in many protocols. 2D, passive and active stereo 3D are fully supported.

With latency reduced to an absolute minimum, Juggler is ideal for live performance. All sources can be genlocked to frame accuracy, with instant system failover across all connected devices via frame-data monitoring (an API is available for third-party sources).

Compere

Compere (pronounced 'KOM pair') is the software system that manages all Juggler connections in a 'stage set' (we call it the Project Group), defines inputs, outputs and manages displays. A version of Compere runs in every Juggler on a Linux OS that drives the front panel touch display and communicates with the FPGA.

Compere provides the user interface (UI) for a Juggler system. Here you create Project Groups of Jugglers and configure all the inputs and outputs of each Juggler involved. All group activity can then be managed via external control from multiple network points.

Compere does much more, but for configuring your Juggler system with its media inputs and display outputs, you need to become familiar with the Juggler side of Compere.

Connect up your hardware before you venture into Compere. Once that is done, connect into the system from a workstation control PC with Compere running, and every part of the system will be recognised and identified, along with all their available ports, in order to create the relationships between each Juggler, their input sources and output devices.

Projects and Groups

Projects are a bit like stage sets, with scenery and equipment ready for a play or show. When you create or see a Project in Compere, you will see a list of what belongs together for the 'stage set': Jugglers, projectors, 3D models and so on, that are assigned as a Project Group.

All the details and settings are held in the Project file, which governs how every member of the Project Group behaves. Whichever instance of the Compere software takes control (the one assigned as server), like a stage manager, coordinates and distributes this set of stage instructions to all instances of Compere, which includes every Juggler in the Group. If any of these members, as clients, wants to make changes at any time, the changes are handed back to the server to redistribute around the Group.

Any client joining the Project Group is handed the common current Project file and performs as directed. In this way any PC running Compere on the network can join the Project Group, become the server or stay as a client and interact with the rest of the Group. This might be to make static changes in the 'stage set' or to move things around dynamically, such as picture-in-picture elements.

If network connection is lost by any Juggler in a Project Group, the local copy of the Project will keep running until it can reconnect and re-sync with the Group, at which point any changes to the Project file will be received from the assigned (or a reassigned) server.

The wider network

There can be more than one Project Group, each with an assigned server to coordinate it. They can all be on the same network, and any member can be moved from one Group to another as required. If you need other Groups, they can be created and named, and available members assigned to them instead of the initial default (unnamed) Group.

Features and capability of the Juggler pixel processor

- Modular FPGA-based pixel processor.

- Configurability: multiple Juggler modules can be daisy-chained via high-speed fibre-optic link, to add additional input and output capability within a 16K × 4K 10-bit 4:4:4 canvas, or equivalent bandwidth.

- Video scaling, video matrixing and picture-in-picture capability, genlocking, framerate conversion.

- Failover redundant source management with 7thSense IntelligentSource™ (API available).

- DisplayPort™ 1.2, 12G SDI and HDMI™ 2.0 (optional HDCP™ compliant)

- DisplayPort output SST to MST conversion.

- Supports 10-bit 4:4:4 – internal processing at high precision.

- Colourspace mapping via 3D LUT.

- External control over IP.

- User interface: 7thSense Compere software.

- Warp, blend, and black level output support via MPCDI (2D MPCDI).

- NDI® 4.5 confidence monitoring.

System Architecture

- Every Juggler has four video processing channels.

- Each Juggler unit has a flexible design for provisioning a number of inputs and outputs of various and mixed types.

- Jugglers can be daisy-chained by utilising a high capacity data bus.

- The number and balance of inputs and outputs in a Juggler system is completely scalable.

- Any input – live capture, streamed or prerecorded – can be processed by any Juggler to any output and placed anywhere in the overall output space.

- Redundancy and failover can be built into a Juggler system

As this example shows, the Juggler system maps the whole-site canvas, uniting, processing and placing all sources with all displays, expanding inputs and outputs as required:
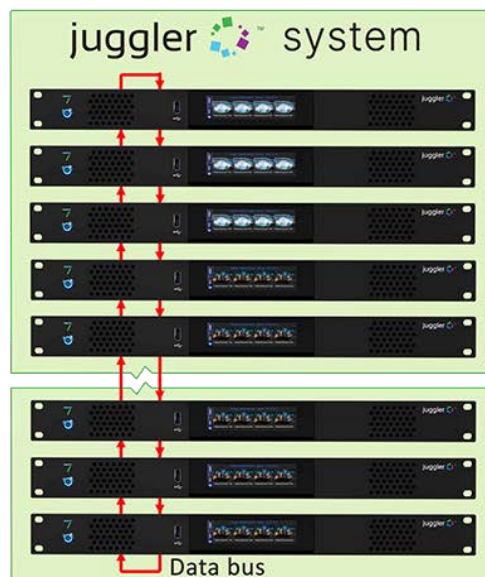
What Juggler Can Do

Here are two examples of Juggler applications that demonstrate its potential roles.

Example 1: IntelligentSource™ failover

Use cases: Ultra-reliable auto failover, per input, or whole system, seamless source switching

- A large 16K × 4K canvas

- IntelligentSource

- Format conversion

- Warp and blend
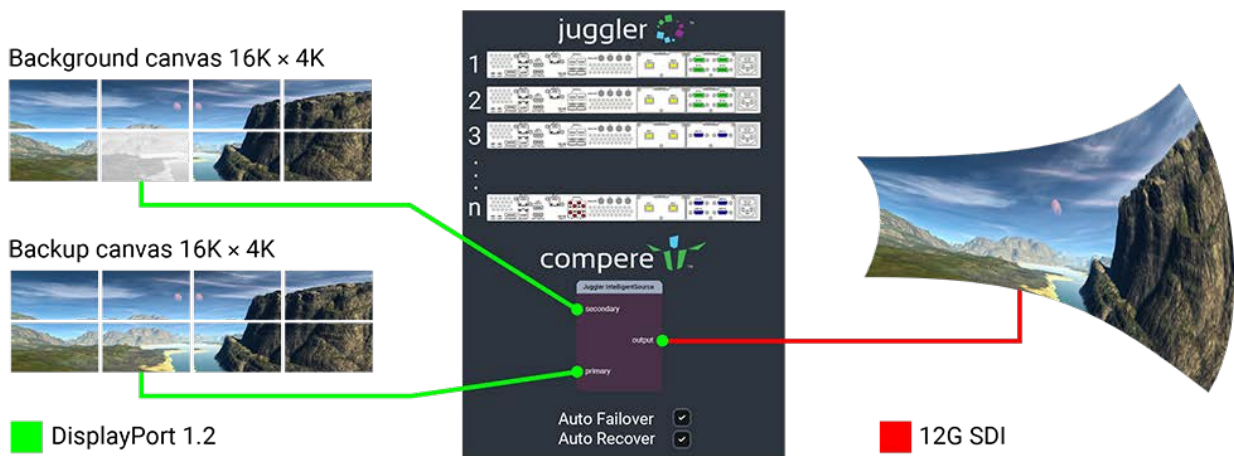
Example 2

- A large 16K × 4K DisplayPort input canvas

- IntelligentSource failover

- Picture in picture (PiP)

- Scaling

- Warp and blend
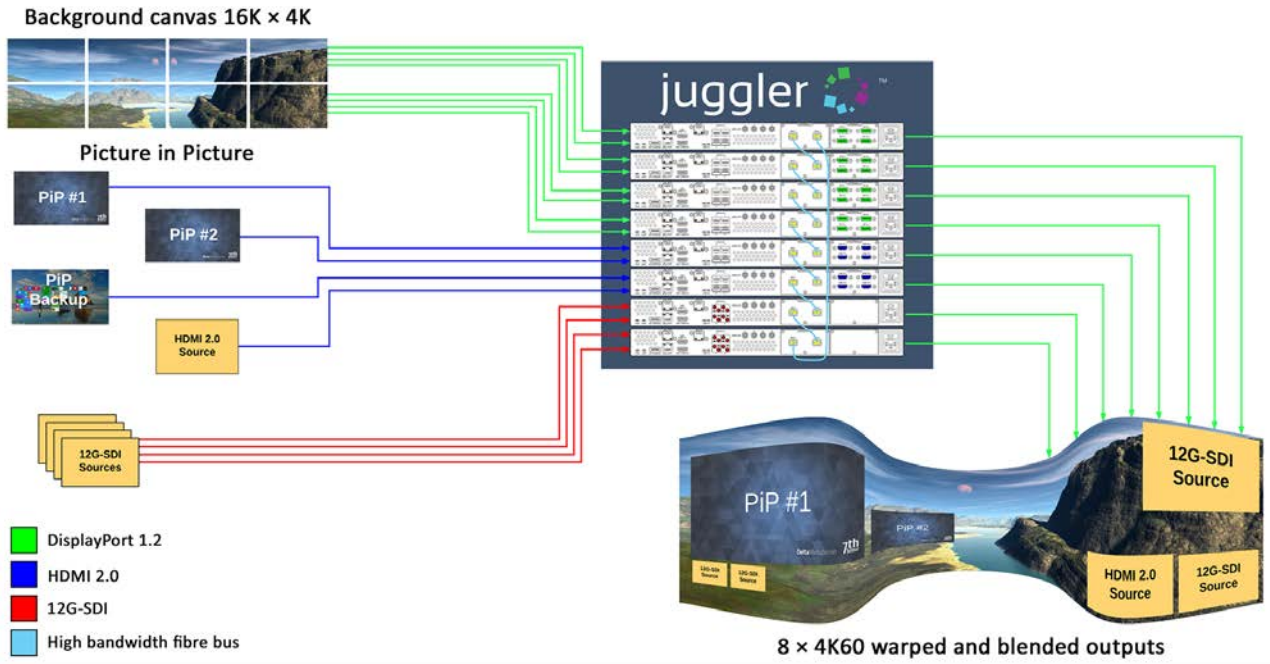
Front Panel: Controls

Power button

Juggler must first be connected to an appropriate power source using the supplied IEC power cord. The front panel power button will show a dim blue ring indicating this connection. Press and release the power button to start the unit. The power button will show a bright blue ring and the display panel will go through a start-up cycle before settling to status information and the Juggler logo.

Reset

Should you need to reset the internal FPGA board only, the power button can be pressed and held for 3 seconds then released when the blue light dims. The blue button light will flash for 2 seconds before returning to normal status.

USB port

For service use only.

Touch-screen display

The touch screen display is a tabbed swipe and scroll screen: see <u>Front Panel Display</u> [13].

Front Panel Display

The Juggler front panel touch-sensitive display is driven by a Linux CPU board running a minimised instance of Compere, which also, as a network client, shares the common Project file for the whole networked system.

The display presents a menu system that conveys elements of the Compere graphical interface, allowing direct interaction with, and monitoring of a single Juggler by selecting, scrolling and swiping.

Video tab

This tab provides a direct NDI display of media being processed by the unit, as a row of video channels 1 to 4.
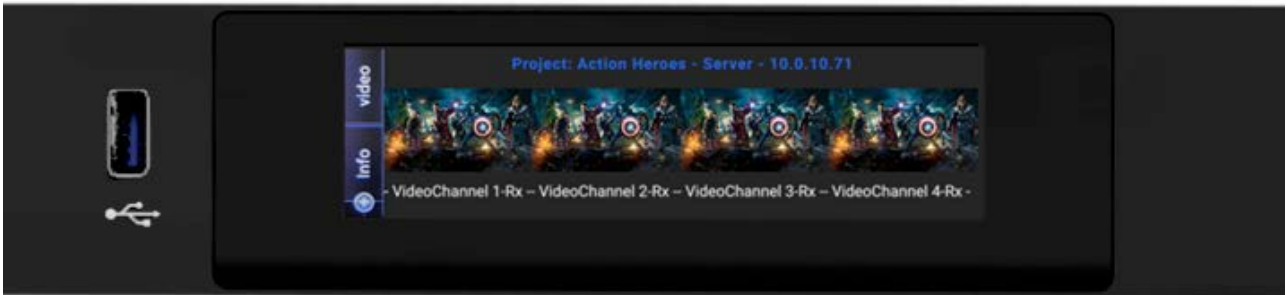
Info tab

This shows the Juggler name, IP addresses, network mode, Project name and connection status, and a second swipe-to page for editing the static IP address and subnet.

Set Juggler Static IP and Subnet

This can be done from the Juggler front panel touch screen.

- Touch to select 'Info' tab if it is not already showing.
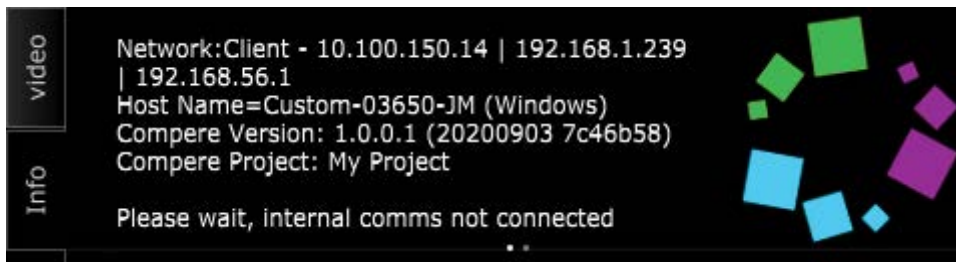
- Swipe from right to left to move to the IP Editor page:
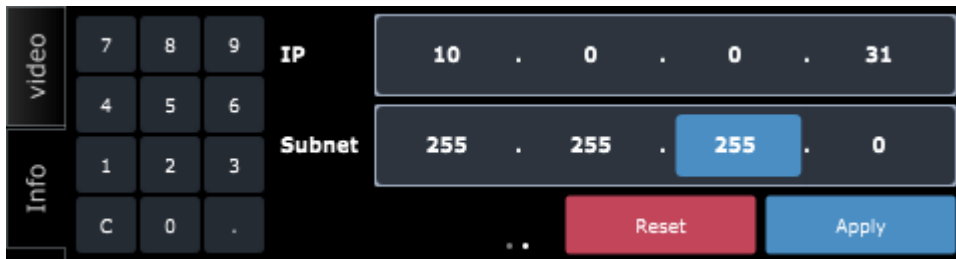
- Touch to select an octet (address element) to change; it will show highlighted blue.
- Use the number pad on the left to enter a value up to 255.

On the number pad:

- o   . (separator) moves to the next octet

- o   C (clear) sets the selected octet value to 0

- Touch 'Reset' to undo all changes, or

- Touch 'Apply' and then Confirm the changes:

Rear Panel Connections

The back panel of each Juggler is factory configurable to order. The required number and type of inputs and outputs for a system of Jugglers is expandable by adding Juggler modules as required.
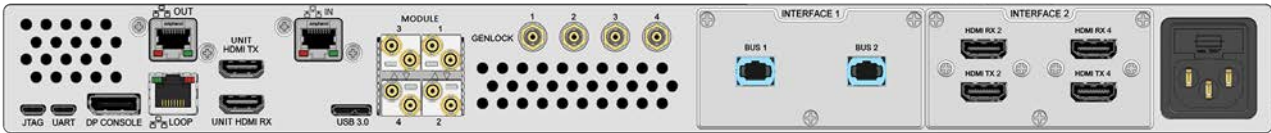
Working from left to right, these connections are available:

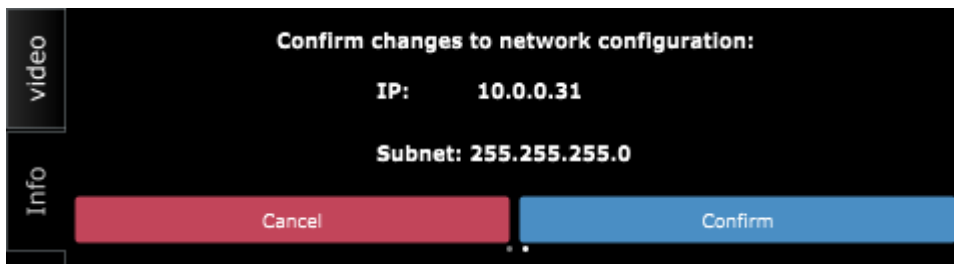|	
JTAG	Service use only.
UART (serial over USB)	Service use only. Connection for Juggler Console to address the FPGA processor.
DP CONSOLE	Not used.
OUT and LOOP	Non-user Ethernet ports relating to front panel control, always connected to each other. (LEDs on OUT are inactive.)
UNIT HDMI Tx and Rx	Used in Juggler modules configured for HDMI, expandable with the I/O Interface panels.
IN (RJ45)	Network port (LEDs on this port are inactive).
USB 3.0	Not used.
MODULE × 4, numbered 1-4 (options)	4 × SFP modules, *either* 12G bidirectional coax (12G-SDI HD-BNC) *or* optical fibre (LC) for SDI or to form a 4-lane ('half') bus between Juggler modules see Connect a Juggler Data Bus [26]
GENLOCK / GPIO – 4 × HD-BNC, numbered 1-4	4 × HD-BNC independent configurable I/O for synchronising; genlock, 3D sync etc. GPIO and Genlock Juggler Systems [20]
I/O INTERFACE 1 and INTERFACE 2 (options)	Choice of 2 × I/O channels of DP or HDMI, or 2 × MPO optical fibre to form an 8-lane ('full') bus between Juggler modules with greater bandwidth than afforded via SFP transceivers see Connect a Juggler Data Bus [26] .

SFP Modules

Note the orientation of SDI modules in particular. These are marked with indented arrows, but note that modules are inserted back to back (lower row inverted):
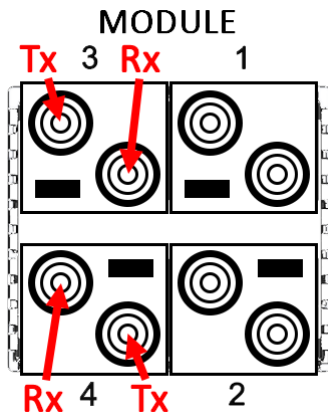
The same orientation is true for fibre-optic modules, but these are keyed. If using the 4-lane 'half'-bus option to daisy-chain Jugglers, see Connect a Juggler Data Bus [26]: 'Half bus'.

> **Important!** Only SFP modules approved by 7thSense should be fitted to the Juggler unit, fitting non-approved SFP modules may cause damage to the Juggler or prevent the unit operating.

Loop cable (every Juggler)

This is a necessary connection, to be kept as short as possible. A 12 cm, black Ethernet cable is provided for this with every Juggler. It must not be disconnected.

Input and Output Options

In a multi-Juggler system with a data bus, any input of any Juggler can be sent to the bus and taken off the bus as an input to any other Juggler. These internal interconnections are done in Compere, the software user interface. For example, this means that an HDMI input can be sent out elsewhere from the system via SDI.

There are three types of internal connection in a Juggler or a Juggler system:

1. Some Juggler inputs may match its outputs directly, for example providing layers from a media server to a whole-canvas background, requiring no processing on the way. These can save bandwidth by using the 'Bypass' feature.

2. Other Juggler inputs may occupy an entire video channel, in which case they go directly from input port to output port.

3. For placing and scaling any Juggler input on the system Canvas, Canvas Windows are matched onto the input ports, which in turn are passed to Canvas Outputs on output ports.

➤ For an explanation of terms and parts of the system Canvas, see Display Configuration: Canvas Setup 78

Connection configurations in Juggler modules

Juggler supports connectivity via HDMI™ 2.0, DisplayPort™ 1.2 and 12G SDI. Some typical configurations of I/O options are shown here.

Media inputs and outputs

HDMI-configured Juggler with a total of 3 × Tx + 3 × Rx HDMI (shown blue) and 2 × Tx + 2 × Rx SDI (shown red):
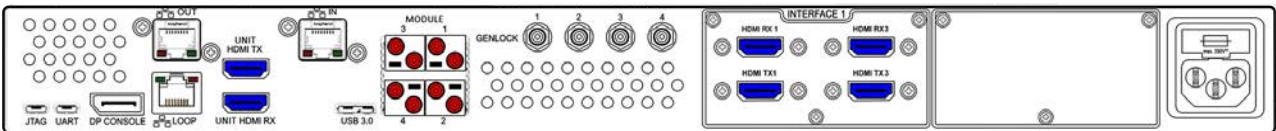
HDMI and **DisplayPort** configured Juggler with 2 × Tx + 2 × Rx HDMI (shown blue) and 2 × Tx + 2 × Rx DisplayPort (shown green):
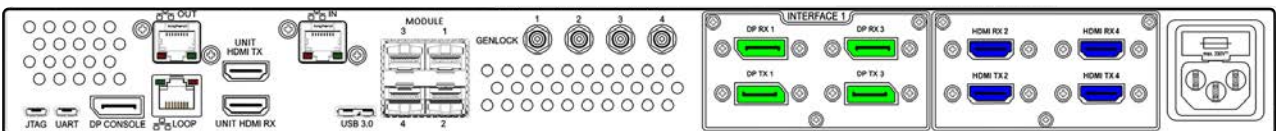
DisplayPort Juggler with 4 × Tx + 4 × Rx DisplayPort (Unit HDMI not enabled):
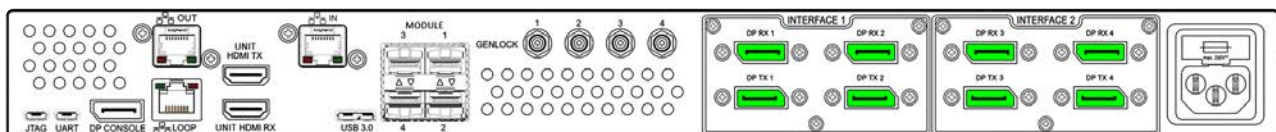
SDI-enabled Jugglers are equipped with up to four SFP modules (shown red):
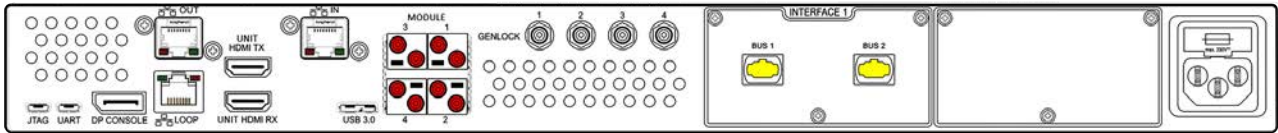
Data bus

Each Juggler has four processing channels. By bussing Jugglers together, additional processing channels can be added to the Juggler system. To **bus** Juggler modules together, use optical fibre cables. Full 8-lane bandwidth MPO I/O ports (shown yellow) may be used, or 4-lane SFP modules (shown magenta). See also <u>Connect a Juggler Data Bus</u> [26]:
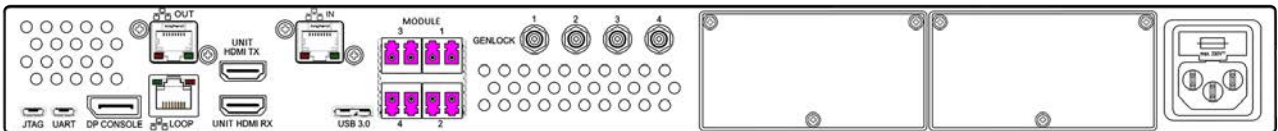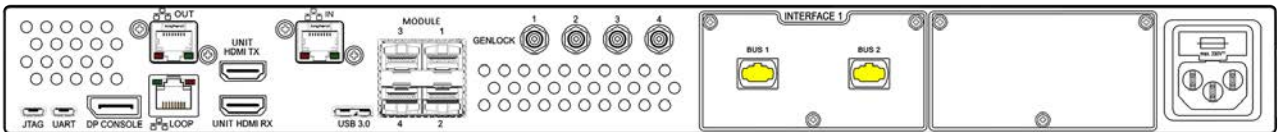
These are configured typically alongside 2 × DP or 2 × HDMI in the second interface.

Bus capacity

The bus configuration determines the capacity in terms of resolution and equivalent bandwidth. To increase bandwidth, lanes 4 and 8 can share their capacity with lanes 1-3 and 5-7. Effectively this becomes a 6-Lane 'High Bandwidth' bus, with lanes 4 and 8 unavailable to separate video channels.

Bits per Pixel determines capacity of your bus for your media requirement.

Bus Mode Bits Per Pixel	Max Resolution at 60fps, or equivalent bandwidth	4-Lane Bus (HD-BNC SFP modules)	8-Lane Bus (MPO Interface)
30	UHD (3840 × 2160)	4 buses (2 + 2)	8 buses (4 + 4)
24	4800 × 2160	4 buses (2 + 2)	8 buses (4 + 4)
30 High-Bandwidth Mode	4800 × 2160	n/a	6 buses (3 + 3)

GPIO HD-BNC Ports

Each Juggler back panel has four HD-BNC connnectors. These can be independently designated as inputs, or outputs that can be assigned to the Juggler's video channels, in the Juggler properties of Compere.

All Juggler ports are configured independently in Compere. Here is the GPIO section in Juggler properties:
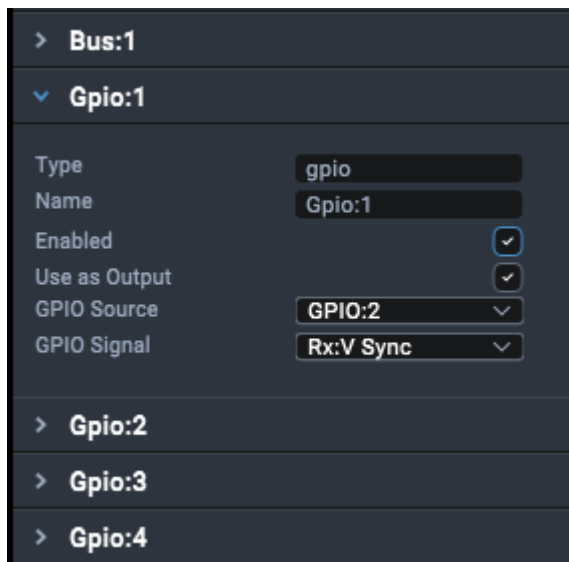
Name

Identifies the physical GPIO.

Enabled

For GPIO ports to be live, they must be enabled. If you need to disable the port, do that here.

Use as Output

If not an output, an enabled port is assigned to receive a signal from a source.

GPIO Source (when used as an output)

To output a signal requires a source. This could be a video channel media source sync, or another feed via another GPIO port.

GPIO Signal (when used as an output)

Select an available signal from the selector:
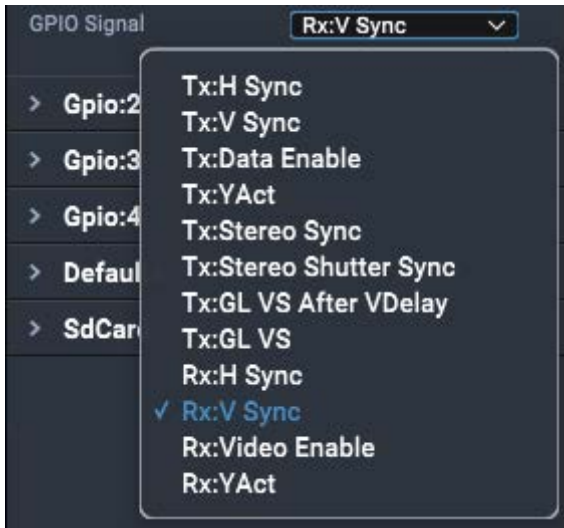
These GPIO ports can also be used to support genlocking. For example, on one Juggler in a system, GPIO:1 can be used to collect the VSync of a media input, pass it out to GPIO:2, collect on GPIO:1 of the next, out from GPIO:2 and so on to daisy-chain down the line.

GPIO Genlock Example

Example: Use an input VSync

In the Juggler system, the first in line is designated as providing the source for genlock.

In Compere

This first Juggler's GPIO:1 is set to be an **output**, using a video channel media input as its **source**, and the **signal** it is sending as the V Sync is is receiving from this source (expressed as Rx:V Sync) (see terms above).

The next Juggler in line has its **GPIO:2** set to be in **input**, in other words, simply receiving from the attached cable.

Its **GPIO:1** is set to be an output, using its GPIO:2 as its source (the signal need not be specified).

This is repeated down the line, the last of course not needing to be assigned a sending GPIO.

Connections

Five Jugglers would therefore be cabled as follows:
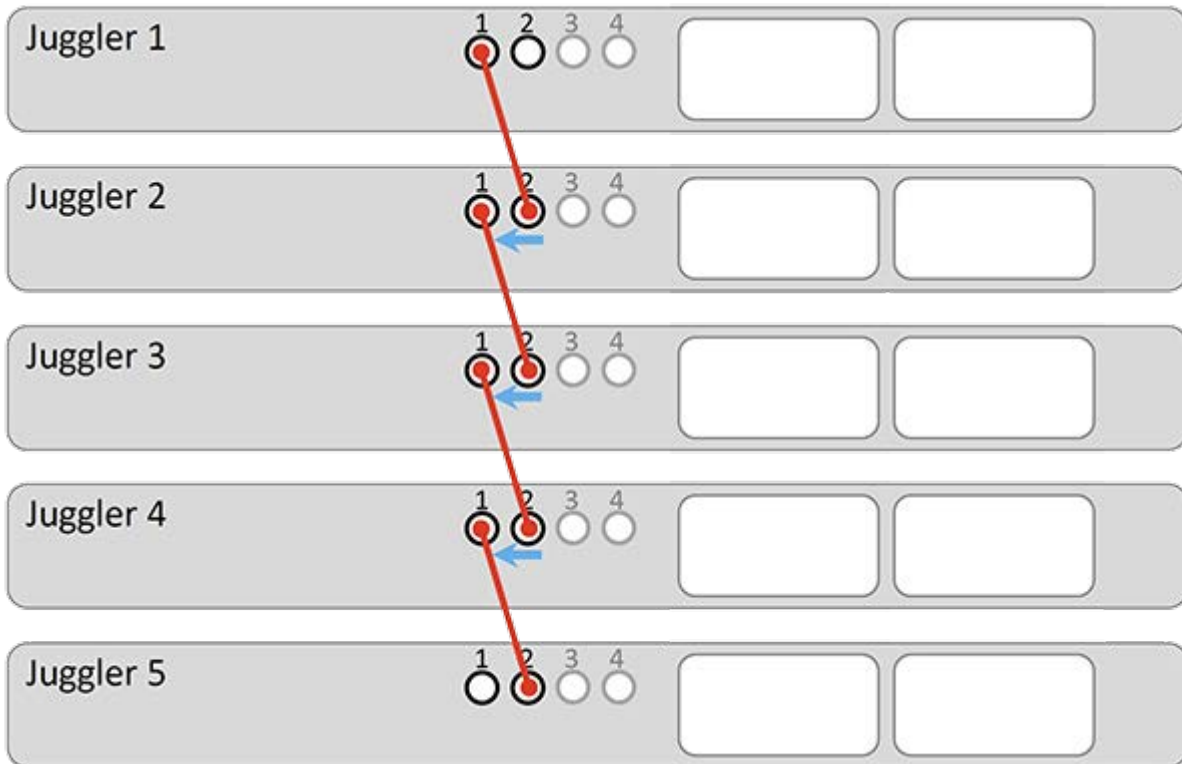
Genlock Mode

Now that every unit has a genlock signal, each that has a signal received on its GPIO:2 can use that for the output **genlock mode** in Compere.

The *first* Juggler has no signal on GPIO:2, so uses the same signal received on video channel 2:

Workflow: Create a Juggler System

Configure the Connections: physical

1. Unbox Jugglers and perform an all-round physical check for any signs of transit damage.

2. Rack the Jugglers so that the units are fully supported at the back and bolted securely at the front.

3. Connect power, press the front panel power button, and confirm that each unit comes to life.

4. The front panel will go through an initialising routine and settle to show an IP address for the unit and the Juggler logo.

5. Reassign IP addresses [14] for each Juggler as required.

6. Switch off all units.

7. Connect any data bus [26] or buses between Jugglers that will combine to form a system.

8. Connect all input devices available at this stage, and required output devices.

9. Switch on or boot all the components of your system. You may find it best to establish your own preferred order, for example projectors first, working back through devices that take most time to boot up.

Configure the Connections: Compere Juggler view

➢ See Terms and Definitions Used in Compere [37]

The precise sequence will depend on the requirements of the overall system of Juggler modules, but this outline gives you the sequence of steps.

Assemble your Compere group

1. With Compere installed on a remote ('control') PC [35], log into the network.

2. Open Compere and make sure it is in Juggler View [38] (top left icon), go to *Configuration > Network Configuration* [68]. You should see all Jugglers and any other devices running Compere.

3. In Network Configuration, ensure that all the devices running Compere that are involved in your Project, belong to the same Project Group [75]. Create your Project Group(s) here as necessary.

4. Assign one member (Juggler) of the Project Group (any) as server, with a binding NIC, and all the rest as clients.

5. All Jugglers in the same Project Group as you as the Control PC, will now be visible in the Project panel on the left of the Compere screen.

6. Drag each of these Jugglers from the Project Panel into the Nodes assembly panel [94] and arrange.

7. Connect the bus nodes between the Jugglers [103] in Compere.

Assemble your display output

Note that if your Juggler configuration is for MPCDI display input, Canvas, Outputs and Windows will be imported with the data and do not require intervention.

8. In the Compere Canvas assembly panel [78], create your overall canvas to size. The Canvas components all need to be created here or, if available as MPCDI, this file can simply be dragged from the Project Panel into the Canvas assembly panel.

9. In the Canvas panel, add and arrange your Canvas Outputs [80]. These will adopt the resolution of the Juggler Tx ports to which you attach them in the Nodes panel.

10. Assign the Canvas for the Project Group to each Juggler.

Configure your Juggler input and output connections

11. For each Rx connection [100] available per Juggler (the number is the channel, 1 to 4 per Juggler), make sure the correct EDID is present for the graphics card feeding each Juggler input.

12. For each Tx connection [100] available per Juggler, the EDID should be autodetected from the output device. If not going directly to an output device, select and apply an EDID from the connection's Properties panel.

13. 'Windows' in the Canvas Outputs are areas in the whole Canvas to which Juggler inputs will be sent. Add and configure [84] your Windows, and name them for easy identification.

14. Make connections between Juggler Inputs to Canvas Outputs and Windows [108] in the Nodes area, either directly through a Juggler, or from the input on one Juggler to the output of another via the data bus, and/or if using MPCDI data, introducing processing with warps, blends and colour convergence.

Connect a Juggler Optical Data Bus

Jugglers in a system should be racked adjacent to each other, since optical connections must be short and loop freely. If you are using multiple Jugglers on a data bus, make these connections first, before inputs and outputs; they can become quite congested with many units.There are two bus configurations, 4 and 8 lane, with each lane having a capacity of up to the equivalent of UHD 60fps 10-bit 4:4:4.

> **Please note** that as a critical component, optical cables used should be supplied or approved by 7thSense. Cables do not need to be short but the minimum bend radius of a free-hanging cable should be not less than 10 times the cable outside diameter.

This page covers the physical cabling. For an explanation of how this works, and the use of the data bus in the Compere software, see:

➢ [Bus Connections in a Juggler System](#) [103]

8-Lane 'Full' Bus

With the MPO (Multi-Fibre Push-on) optical fibre option, twice the bandwidth is available. Each fibre-optic cable carries 8 lanes, 4 in each direction:
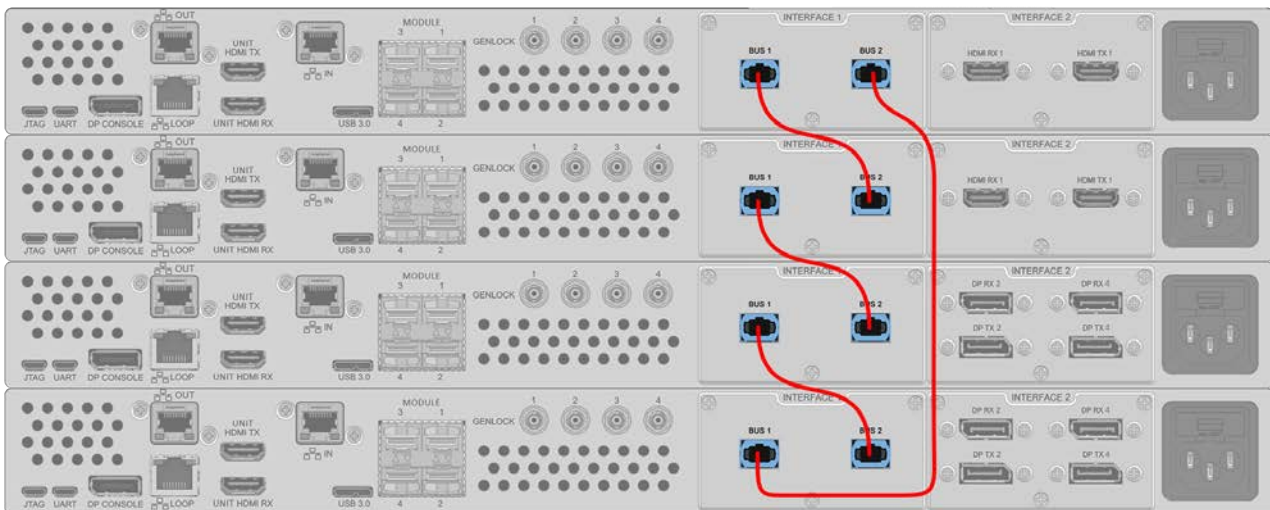
Units are daisy-chained unit to unit, 1 to 2 and loop back from the last unit to the first in the same way.

Cables

Use MTP/MPO 12-fibre 50/125 patch cables, type B, Female to Female.

4-Lane 'Half' Bus

Using the 'half-bus' optical transceivers, Juggler modules can be bused together using on-board SFP modules. Each fibre optic cable carries 1 lane in each direction.
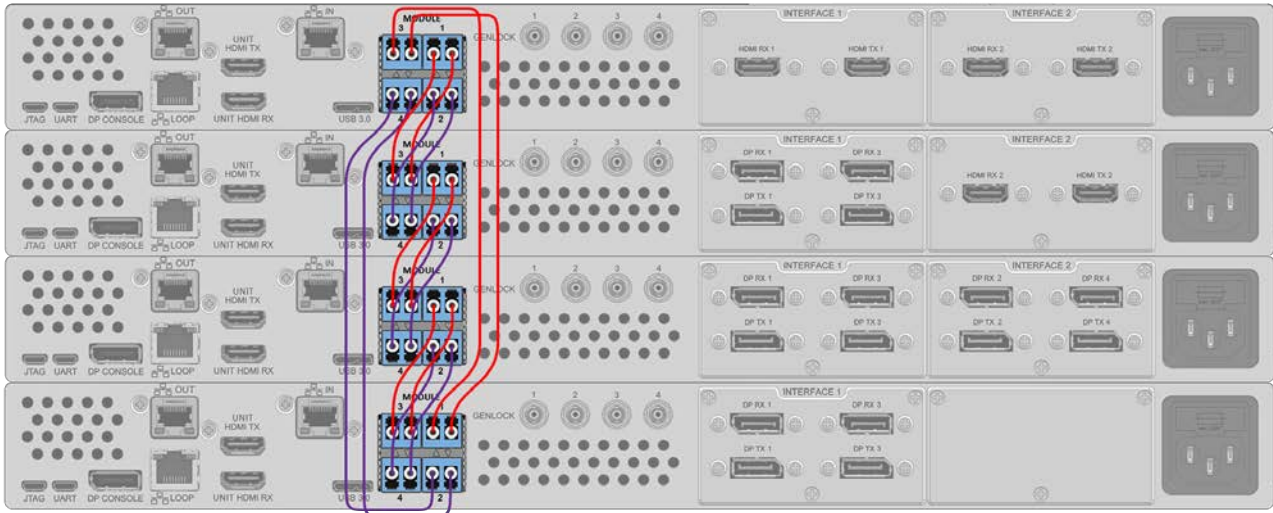
Units are daisy-chained progressively unit to unit, 1 to 3, 2 to 4 and loop back from the last unit to the first in the same way.

We recommend the use of paired crossover patch cables, which will have colour or tag identifiers. It can be helpful to choose an identifier to work with (yellow, a number etc.) and keep these on the outside as you work down one side and up the other. In this example, yellow-sleeved and tagged A:

Bus Configurations

Current modes of operation are:

- bus the complete canvas

- bus individual inputs

- bus thumbnails for monitoring purposes.

We configure your Juggler system based on your requirements.

Care with Optical Connectors

Optical connectors are a Class 1 laser product.

Never look directly into the end of fibre cables until you are positive that there is no light source at the other end.

Cleaning Optical Connectors

The most common problem with fibre optical connectors is contamination – residues or dust. Plastic easily generates static, which attracts fine particles, too small to see with the naked eye,

---

and any contamination will affect performance or cause failure. **Avoid troubleshooting by inspecting first.**

© 2021 7thSense

Inspect before you connect

Hand-held fibre inspection microscopes may seem expensive, but prevention is cheaper than troubleshooting, and it is essential to ensure optical connector end-faces are clean *before you connect*. Problems arise when components are left unconnected and uncapped, so keep caps on until the point of connection. Residues can even be present in an out-of-the-bag cable, or from the moulded cap itself, and can transfer to a bulkhead connector.

Inspect, clean, reinspect

We recommend that the polished ends of fibre leads should be cleaned with a push-click pen cleaner. Dry cleaning is always recommended; alcohol-based agents should *not* be used. Specialist cleaning kits are available that ensure contaminants are not just moved about, or reintroduced by a second use of a tool.

After cleaning, reinspect. You may still have contamination. Do not connect until you know a connection is clean on both sides.

Connecting to a Network and Devices

Juggler systems must be connected on a single network each with a static IP address [14]. All devices running Compere will be visible in any Compere UI, along with their host names, where their network binding NICs can be assigned along with their roles and any descriptive comments.

➤ See Network Configuration [68]

Process

Having established any data bus between Jugglers [26], connect your source inputs (from Juggler physical input ports) and display outputs (to Juggler physical output ports). First identify the Rx and Tx ports – and in particular the numbering on the rear panel – so you understand how these are represented in Compere's representations. All live ports are automatically identified in Compere, and it is there that you will make the internal and process connections.

You can regard outputs and inputs across the system as independent. Unlike many devices, the Juggler in and out (Tx and Rx) are not linked one-to-one. You will decide in the Compere control software precisely how input streams are processed, where they end up via which outputs and into what visual space.

Switch everything on. Jugglers take a short while to self-configure, and then you will see the Juggler logo show, along with the unit IP address. The order in which you switch system components on should not matter. Compere will recognise every member of the shared network, and all their connection types.

Compere on Jugglers

Compere is already installed for the internal operation of each Juggler on Linux. It handles all system communications including the front panel display. To configure your system over the Juggler network, install Compere on a local PC on the same network.

Versions

Juggler projects require compatible versions of the software. If incompatible versions are detected in a Project Group, these will show in the Network Discovery Panel.

The current version of Compere shows on a Juggler front panel display, in the Compere title bar, and in *Help > Show About*. To see the version of a remote Juggler on the same network, select it in a Compere Project Panel and find 'Compere version' in the Properties panel.

Upgrading Compere on Jugglers

You may need to install or update a version of Compere on a Juggler or many Jugglers in a Project Group. This can be done remotely from a Control PC to distribute the installer. Select the required Jugglers in the Nodes panel, right-click and select 'Upgrade Compere'. Browse to the Linux installer on the control PC. This will distribute the file to all Juggler Atom controllers. The Selected Jugglers can then be rebooted remotely from the same menu and will be upgraded.

Juggler maintenance

Several operations are available from a control PC running Compere, by selecting a Juggler or Jugglers in the Nodes ⁹⁴ Panel of Compere and using the right-click context menu.

Upgrading Juggler firmware

It may be necessary to upgrade the firmware of a Juggler. This option is also available through Compere, addressing the Juggler in the Nodes panel, from the right-click context menu.

Internal SD card

This is where a number of configuration files reside, and it can be useful to clear it down. From the Juggler context menu select 'Reset Juggler SD Contents'. This will clear all added files including warp and blend, MPCDI, added EDID files.

Factory Reset

This context menu will temporarily remove a Juggler from the Project, reboot and and reset it to clean state.

Compere MinGUI

Running Compere with the full graphical user interface and controls is CPU intensive. The Juggler Atom CPU runs without this, as client or as server, but also drive the front panel display for information and with a limited set of touch controls.

There can be a case for using a PC with a lower than recommended specification as a project group server, but not as a control PC with the full UI. The PC becomes part of the project group and is addressable by external control commands, just as any Juggler.

By running Compere with these additional arguments, it will run from the PC system tray. (Create a shortcut and add these arguments after the .exe command.)

C:\7thSense\compere\compere.exe networkserver mingui systemtray
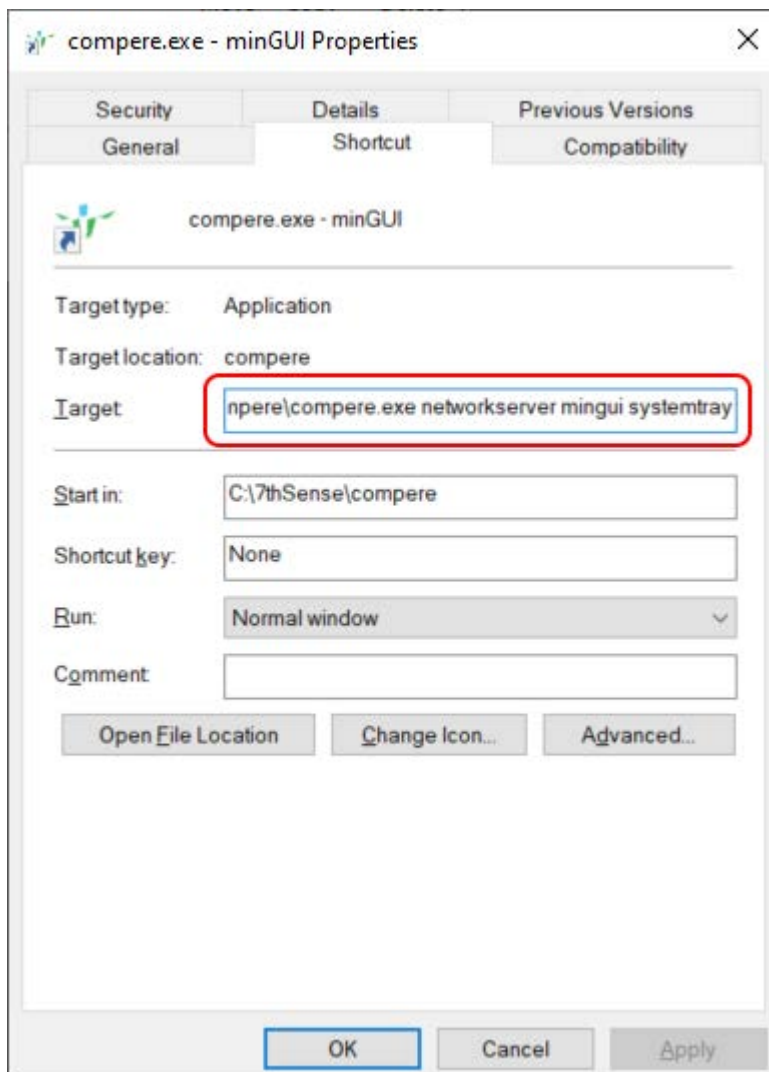
Add a shortcut key or run at startup as required. Compere in its reduced form will be run from the system tray, right-click the icon to quit. There is no further user interaction.

Installation of Compere is otherwise the same as for the full user interface for a control PC:

➢   [Installing Compere on a Control PC](#) ³⁵

Introducing Compere UI for Jugglers

Once Jugglers have been physically connected to each other, to the network and to related devices (see Connect a Juggler Data Bus ²⁶ and Connecting a System) ¹⁰⁸ they need connecting internally, from inputs to outputs via whatever processes are required. This is all achieved using Compere's UI on a control PC, to create a complete 'stage-management' script that contains the connections, configuration properties, positions and arrangements of all the equipment and players that comprise what is called a 'Project'.

➢ See overview: Juggler and Compere ⁷

Project Group, Server, Clients

Active instances of Compere that have a shared involvement in a Project, must belong to a Project Group. This will include all the Jugglers, plus any control PC that plays a part in managing the Project. One, and only one, instance of Compere in a Project Group has the role of server to the rest as clients. This role includes owning the Project script, receiving any changes to this from clients, and redistributing any changes to the rest of the Project Group.

It does not matter which instance of Compere is the server, and the role can be handed to any Group member. If a server leaves the Group for any reason, the Project will keep running, but intercommunication stops and changes will not be redistributed until there is a Group server again. A group Juggler is always the preferred choice – a control PC does not need ever to play the server role. The function of the control PC is to be able to use the graphical interface to view and edit the Project Group and all its members and their roles, and/or to be used for external control.

➢ Special case: PC as Project Group server ³²

The function of the Jugglers in a Project is to assemble one or more shared visual canvases between them, i.e. a combined total pixel space. These pixels may be displayed anywhere in the real world, and you will place individual inputs into the system onto any number of locations in this overall space. Each Juggler in a system has the advantage of handling a range of input and output connection standards, including DisplayPort™ 1.2, HDMI® 2.0 and 12G SDI, and when put on a common fibre-optic data bus they can share a single very large canvas.

> **Names**: Almost every element of a Juggler system has a name in its properties. Plan a logical scheme of naming so that everything is easily referenced. Names are used as parameters in External Control ¹¹⁵ .

Network

To start, the network that the Juggler system and associated devices live on needs to be set up in Compere, grouping devices for a Project (see Network Configuration ⁶⁸).

Nodes

The next task is to describe the connections of the Jugglers so that source devices address the appropriate inputs, and outputs are assigned and configured to the right display devices. You have wired the outside world; the inside world of the system is 'wired' in Compere. This is all done in Compere's Juggler view, in the Nodes assembly panel (94). Here you will add representations of all the Jugglers in your system, connect up any data bus that the system shares (see Connect a Juggler Data Bus), and connect media streams from anywhere in the system, insert media processes, through to display devices and areas anywhere in the system.

Canvas

You will now configure where the received (input) pixels will be displayed within the total output canvas 'map' available to the Juggler system. This is done in Compere's Juggler view, in the Canvas assembly panel (78). In the overall canvas you will define canvas outputs delivered by the output devices. Inputs are assigned to canvas windows anywhere within these areas. It won't matter if the window is spread over multiple outputs; all pixels in the total canvas are available to the Juggler system.

Juggler systems can accommodate multiple canvases.

Calibrated canvases can be imported as MPCDI files.

Media throughput: NDI® Monitoring

When everything looks right, you want real media to play onto your canvas. You might want to use simple colour or numbered media per channel to start with, to check all your flows. Compere also provides NDI Monitoring (an option in the properties of each Juggler). This enables you to visually trace media input, through outputs and into canvas windows in the Compere assembly panels. These should correspond with the real world displays.

External Control

Everything in a Juggler system, constructed using Compere, has its own unique label (UUID). Any of its components or properties, including position, size, opacity, visibility, projector beams etc. can therefore be addressed remotely using External Control (115) commands and a show controller.

Installing Compere on a Control PC

Compere installs to C:\7thSense\compere.
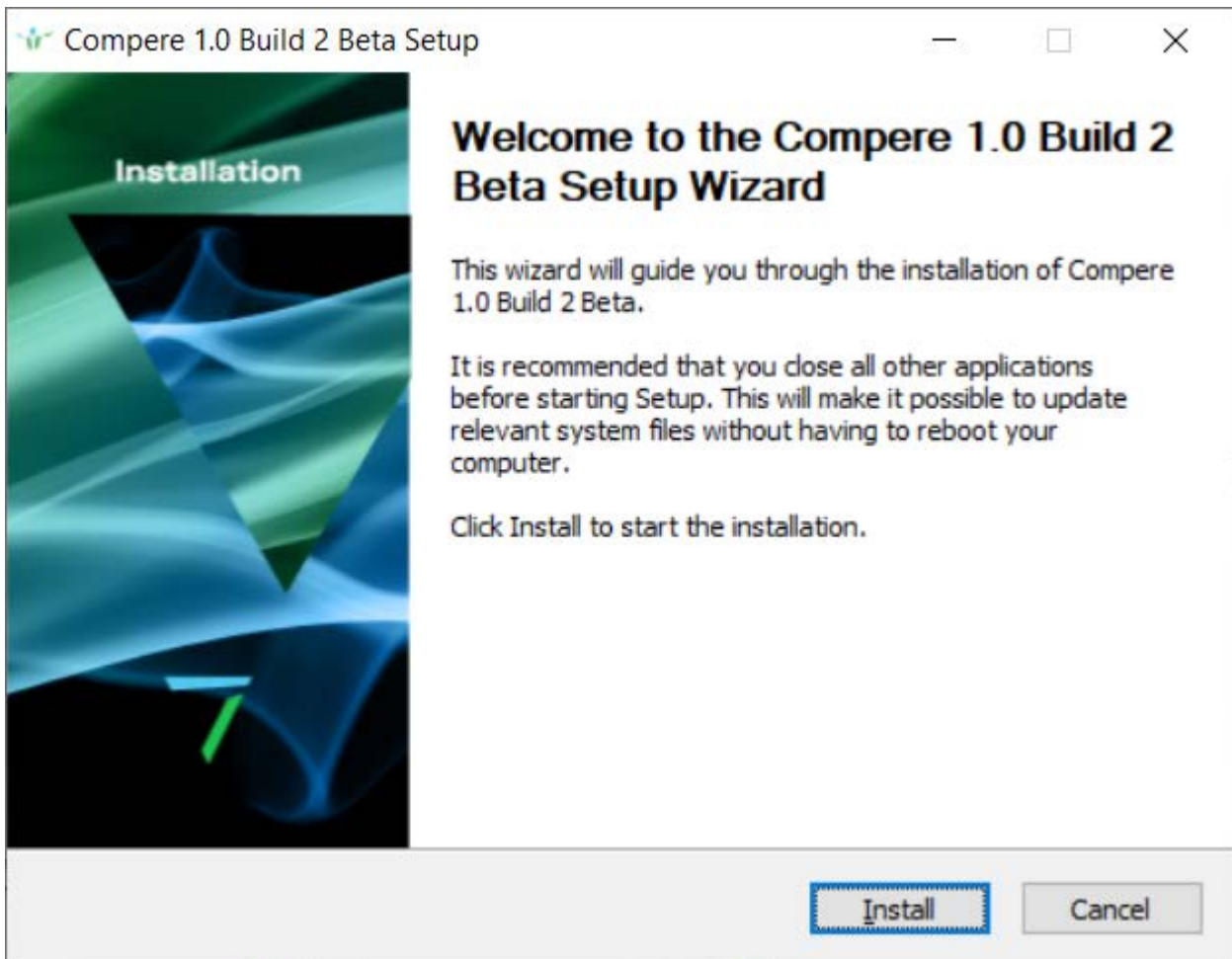
Minimum requirements

A control PC to run Juggler need not normally be of particularly high spec., but memory and number of cores govern performance. For system configuration:

- Minimum quad-core, Intel® Core™ i5 processor

- 16 GB RAM (recommended)

- a dedicated GPU

- OpenGL 3.3 core support

- Vulcan 1.1

- Windows 10

As redundancy, a lightweight networked PC can hold the Project dataset and communicate with a Juggler system.

Terms and Definitions Used in Compere

We use a lot of terms that are specific to the Juggler-Compere world, so it's worth being familiar with them.

> **Tooltips** are implemented throughout Compere, with longer descriptions for labels in panels and toolbars.

Term	Definition
Assembly panel	There are three panels for assembling systems on a grid: Canvas for the display, Nodes to connect Jugglers, and Visualise to create 3D projection spaces.
Bus	The Juggler fibre-optic data bus connects Jugglers together to direct any input through any process to any other Juggler output in the system. The bus can carry individual media streams or the whole system canvas, and its overall bandwidth is dependent on the unit configuration.
Bypass	A media stream input is directed in the same Juggler to a matched-type and resolution output, requiring no processing.
Canvas panel	The assembly panel containing the total available video space ('the Canvas'). This panel can contain multiple display canvases on which Canvas Outputs can be positioned.
Canvas Output	A section of a composited canvas that is taken to an output.
Canvas Window	A source of video that can be positioned on a Canvas.
Clone	xml descriptor for a project entity, e.g. canvas+outputs; projectors; Juggler configuration; 3D model arrangements. Contains all properties and can be saved and reused by loading into in any project. Unlike Presets, clones are saved without UUIDs.
Connector	A video interface point for a device displayed in Node view. May be of any type (e.g. DisplayPort, HDMI, SDI, or bus), input or output.
External Control	A system of commands to address and message any element in a Project.
Grid	A background scaling and positioning guide used in the Assembly panels: Canvas, Nodes and Visualise.
IntelligentSource™	7thSense automatic failover system.
Juggler	A Juggler unit of any configuration of inputs and outputs.
Layout	The selection of dockable panels within which a Juggler project is addressed. Users can save the Layout they create and prefer. Views are default task-oriented Layouts.
Mini Map	A small preview representing the current viewing position and enabling navigation, amongst all nodes available in the Nodes panel.
Network Client	Any network device running an instance of Compere that is not a Network Server.

Term	Definition
Network Discovery	The tab in the Network Configuration panel that lists and profiles all available Compere hosts on the network.
Network Server	Network device assigned as communication 'project controller' across all devices.
Nodes panel	The assembly panel in which connections between devices and processes are made. Example nodes include Juggler, Juggler-Matrix, NatNet receiver, NDI receiver, Projector, Image, Mesh model, etc., which are represented in diagrammatic form in Compere.
Panel	A movable, dockable panel providing: • **assembly** (e.g. Nodes; Canvas; Visualise) • a **contents** list (e.g. Project; Resources) • **configuration** options (e.g. the Properties panel populated by selection from a list; the Network Discovery panel) • information (e.g. logs; system config table; task tracking)
Pin	An internal connection on a Node to link it to another similar pin.
Preset	Presets capture arrangements and settings for components in a Project. When recalled to the same Project, nothing is added to the Project, but dimensions and locations etc. of existing project components will be set again to those of the saved Preset. *Example of use*: multiple arrangements of windows in a canvas or canvases.
Project	A single dataset of information for a system of Jugglers, connected devices and display configurations, shared and used by all Compere nodes. (Think stage, lighting and cast of performers with various roles.) The project dataset includes connections and configurations of all associated devices and assets. Every element in a Project has its own universally unique identifier (UUID). All project data can be stored and recalled, in a .prj format.
Project Group	A network Project Group comprises Jugglers and devices on the same network assigned to work together in a specific Project.
UUID	Every element in a Compere Project is assigned a Universally Unique IDentifier so that it can be addressed by any other part of the system, for example by external control commands.
Video Channel	Transport route through a Juggler. There are up to 4 video channels per Juggler.
View	The main toolbar View buttons present default Layouts that are task oriented. 'Juggler' is a useful Layout for setting up hardware input, output and data bus connections through to the output canvas. 'Visualise' is a useful Layout for setting up a theatre scene in 3D space.
Visualise panel	The assembly panel in which a 3D view of projectors and surfaces in an illuminated theatre space, can be created and manipulated.

The Compere Juggler View

Switch on each part of your system of inputs, outputs and Jugglers in turn.

When you join a network with the Compere UI on a PC, all Jugglers available on that network will be represented in the Project panel on the left side. They will be in a group under a title that shows how many are present.
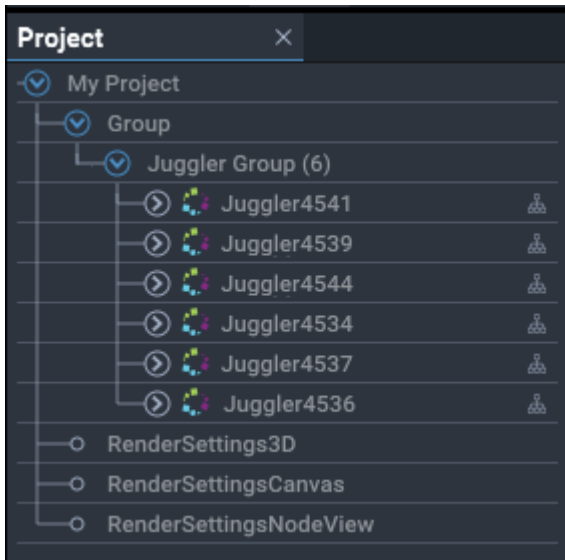

From here, you will drag Juggler names into Compere to further configure and connect them.

When a Juggler is physically connected to display outputs and is switched on, a test pattern can be displayed. Until media inputs, internal pathways and outputs are defined, these will be the reassurance that your systems are connected. How these fill the screen is subject to application of EDIDs on the Juggler outputs.

Open Compere on a Control PC and you will be in a new empty Project. We shall learn more about Compere Projects [58] later; for now, get to know how Compere is put together.

If not in Juggler View, select the Juggler icon. Note the default Nodes and Canvas assembly panels, and the sidebar panels. You may want to maximise the screen with the default overall Layout:
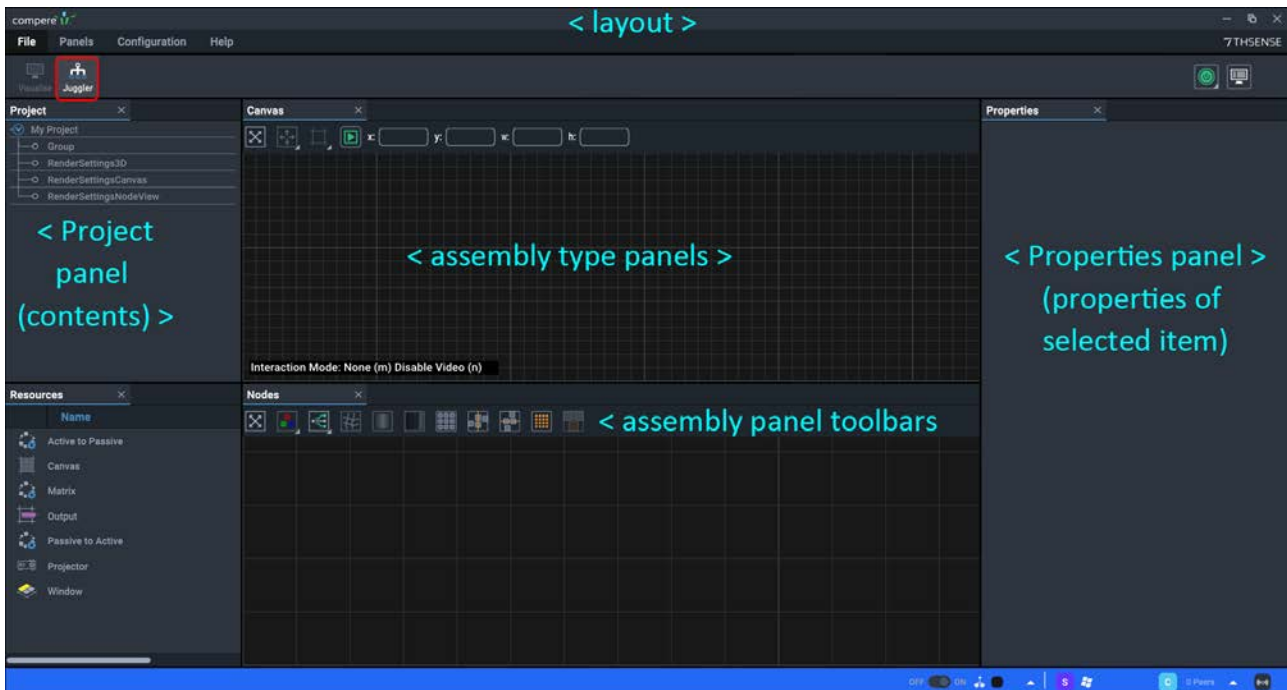
The initial default 'Juggler' Layout comprises:

- the main (top) menu

- the main toolbar area

- the 'Project' panel (default: upper left), listing all configurable elements

- the 'Properties' panel (default: right) for any selected element in a Project

- the 'Resources' panel (default: lower left), showing a range of asset types that can be added to a Project

- a grid 'assembly panel' with a tab marked 'Canvas'

- a grid 'assembly panel' with a tab marked 'Nodes'

- a coloured status bar.

If any of the panels is not present (you can close any element) they can be reopened from the top menu item *Panels*, or by restoring the default view, size and screen location from *Panels > Reset UI*. If you rearrange the parts as you prefer for different tasks, you can save and retrieve custom Layouts.

Note that an inactive Compere licence will display a watermark across the assembly panels.

The alternative default Layout view is 'Visualise', which introduces a 3D scene assembly panel for modelling a 3D space, with interactive projectors, displays and surfaces to configure. This is described separately.

Project

The Project panel is a branching structure of all elements connected in a system, including Jugglers, input sources and output devices, 3D models, canvases and display surfaces. These can be dragged into assembly panels as appropriate.

When you start Compere for the first time, you will find a new empty Project. You can use this, change the name, and save it. Projects are held as extensive xml files, by default stored in C:\Users\[username]\AppData\Local\Compere\Projects, with the extension .prj. This is configurable in Compere > Preferences.

The Project is where you will assemble everything required to 'set the stage'. Your Jugglers will appear in here when present in the same project group, as will your canvases, outputs, projectors, 3D models and any other component as you add them in.

An instance of Compere will hold its own Project only when offline. As a Group client it will inherit the Group project from the Group server. As Group server it will distribute and assimilate changes made within the Group. When offline a project can be saved locally.

Properties

The Properties panel contains all the features for any element in the Project. Properties are grouped by type, under which details can be collapsed and expanded. Use the Search box to filter the panel display by type.

This panel is populated when a Project element is selected, and comprises check boxes, text entry and value fields. Values can be entered, incremented with up and down arrows, or by using sliders.

Some properties lists are very long, so properties may be grouped by type, or filtered using the search box. When multiple items are selected in the Project, only shared common properties and values will be shown.

Resources

The Resources panel lists internal Compere objects that can be inserted into the project, including Canvas, Window, Juggler node components, and projectors.
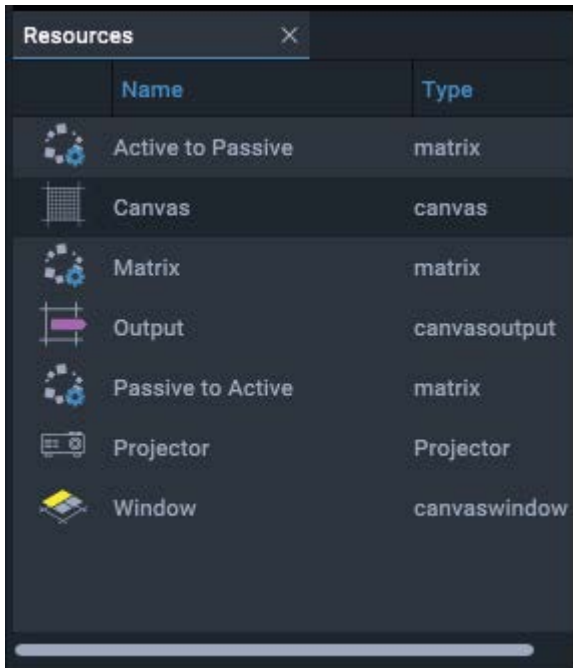
Add, Select, Edit, Remove, Delete objects

- Objects can be dragged from the Resources panel onto an assembly panel. This simultaneously adds them as branches of the Project panel.

- Objects shown in the Project panel can be dragged into an assembly panel.

- Objects are selected within an assembly panel to position, size or edit them, by double-clicking them directly, or selecting them by name from the Project panel.

- When any object in an assembly panel is selected it will show an orange border, and its name in orange in the Project panel. Its properties are then shown in the Properties panel.

- To delete any asset, select it in the Project panel and press delete, or select it in the assembly panel and press Ctrl+Delete.

- To remove any asset from an assembly panel, select 'Remove' from the right-click context menu.

- Links between nodes and/or pins are deleted by selecting the receiving connector, and pressing delete.

Canvas

The Canvas assembly panel represents the total pixels being processed by an entire Juggler Group. In this overall space, Canvas Outputs (channels) are defined. Media inputs (Canvas Windows) are then placed anywhere in the overall canvas. The Canvas assembly panel appears in the default Layout for the Juggler View.

Nodes

This assembly panel creates all the throughput connections and processes within the Juggler system. Representations of each Juggler are shown here, with each video interface connector (bus and AV inputs and outputs), along with information on external connections and internal routing. Nodes is the assembly panel where you configure everything that comes into and goes out from each Juggler, and how these are connected and processed internally. NDI feeds are added and managed here. The Nodes assembly panel appears in the default Layout for the Juggler View.

Using Compere Layouts

The Compere user interface is completely flexible. Panels can be resized, docked and nested however you want, or detached and floating, to work on multiple screens. They are docked in rows, columns or tabs (layered). Any item can be a docking space for others.

Any element of the Layout can be resized. When the whole space is resized, assembly panels (those with a grid) resize proportionately, whilst other panels do not. When more than one panel is in a row or column, they resize together.

Elements are positioned by dragging their title tabs.

- Click and drag a panel boundary to resize it.

- Click and drag a panel title to rearrange it. Dragging will bring up a cross pattern of white squares. As you drag across the screen, this pattern will relate to the vertical divisions of the default docked Compere Layout. Panels can be arranged horizontally or vertically in these zones, or layered (centre square). As the panel title is dragged over the pattern, the squares light up, and where the panel will be is highlighted. Drag free of the squares (no highlight) to float a panel. Here the Resources panel is being moved to dock below on the-left:
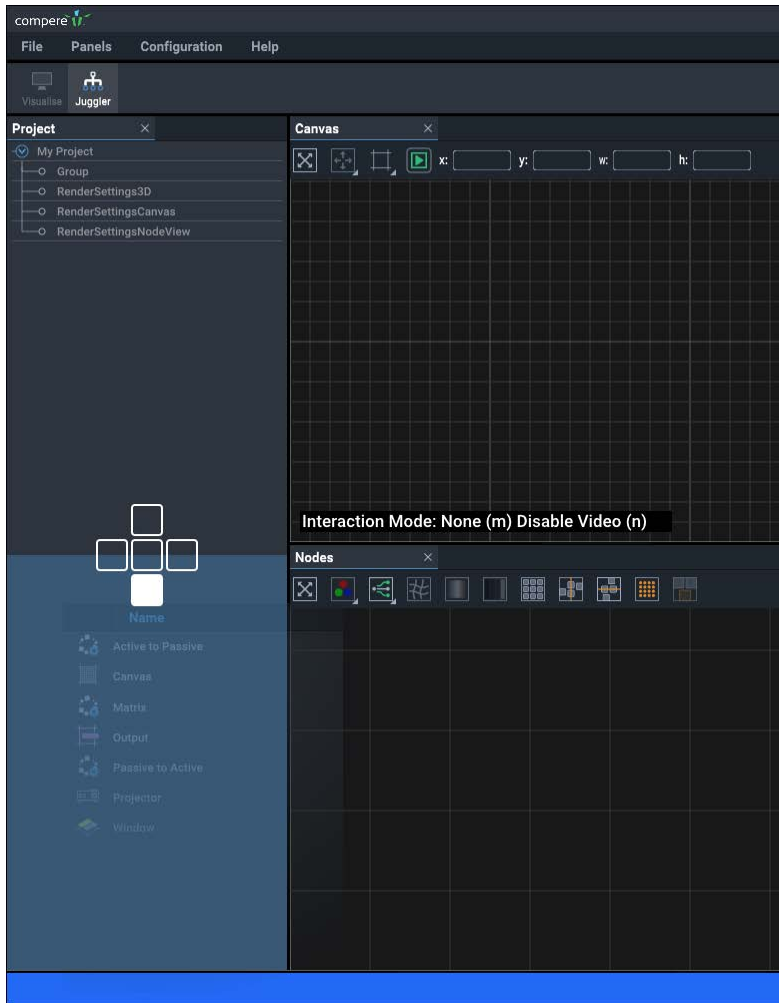
- Any panel can be closed, and redisplayed from the *Panels* menu item. If any panel is duplicated, these are just multiple instances, but can display different areas of the panel simultaneously. For example you may want to see different parts of a complex Project next to each other, or fill a separate display screen with a single assembly panel.

- Menu *Panels > Reset UI* will restore the default fully-docked arrangement of panels to the default Layout relevant to either Juggler or to Visualise View.

- To save your own Layout to use again, use the top right toolbar:
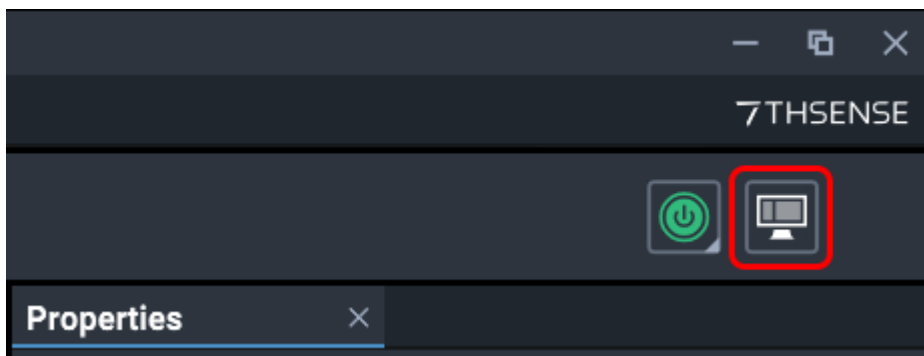
- Assembly panel grids can be dragged around to reposition the view, and zoomed with the mouse wheel. The Visualise assembly panel grid operates in a 3D space and can be rotated about 3 axes.

General mouse and keyboard operations in assembly panels

Mouse / key	Action
Left mouse and drag	Move the grid, an object within the grid, or an object handle.
Right mouse click	Raise a context menu. There may be additional items relevant to the area or object clicked (shows a green border around the selected object).
Left mouse click	Select an item in a list (list panel type, menu, drop-down selector). In the Canvas panel, selects an object according to interaction mode.
Left mouse double-click	Select an object or part (shows an orange bounding box, populates the Properties panel).
Mouse wheel	Zoom
V	Bring all objects in an assembly panel into view.

More specific combinations are available in the individual assembly panel types.

General

- To add items from a Resource panel, drag them into assembly panels or onto items already there.

- To remove items from assembly panels, right-click and select 'Remove'. Note that 'Remove' is not 'Delete', and that removing leaves configuration of properties intact. Deletion from the Project panel also deletes properties.

- To edit the properties of any resource, first select it in the Project panel to list its properties in the Properties panel.

- To delete a link in the Juggler Nodes assembly panel, select an end point of the node and press Delete.

- Assembly panels have their own toolbars, and right-clicking between and on items will raise a context-specific menu.

Main Menu

File Menu

Projects

New

Start a new Project; opens a dialog to name the Project. This is the Project name not the file name, which is given when saving a Project file.

Open

Open a saved Project file. This will open the Project for all connected servers in the network group. A client instance of Compere will send the file to the server which will distribute it to all other clients. (file extension: .prj)

Save As / Save

Enter a filename for your Project, and/or if you are the server, override the default location. For a client, the saved file will go to the Project Group server for redistribution.

Save As also allows you to assign this Project as default, so that Compere will open with this Project each time:
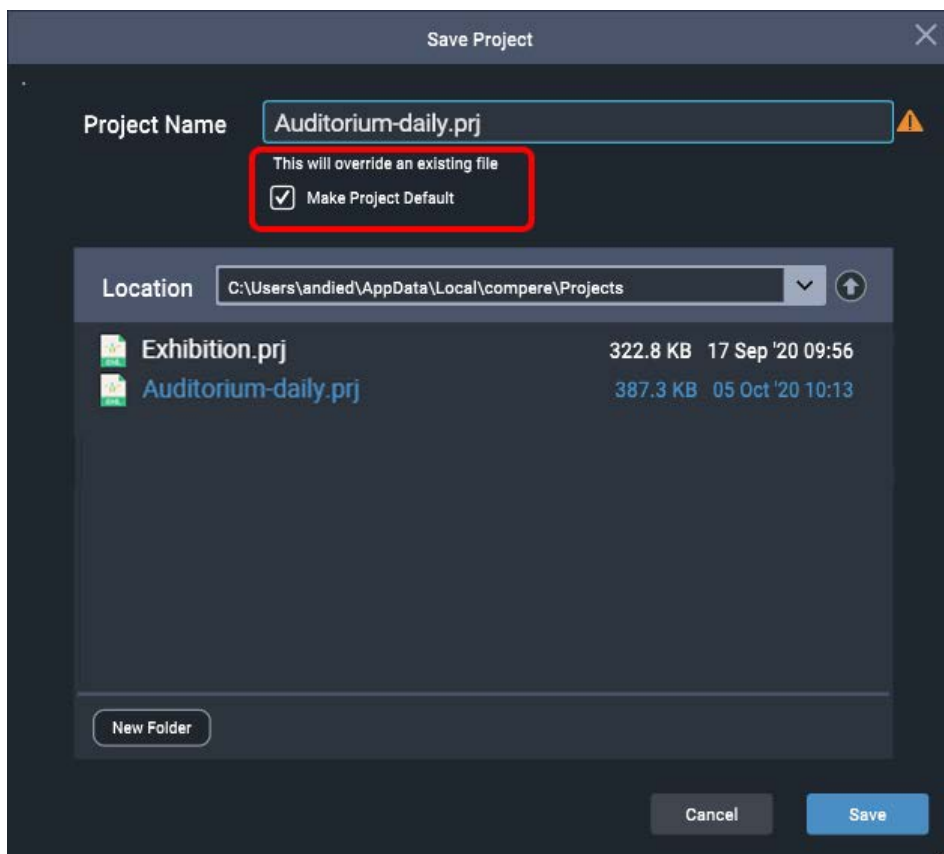
Saving to an existing file name raises a warning.

If you save a different Project as default, this will replace your previous selection. This is only really relevant for a Project Group server, or an offline instance of Compere UI (for example) assembling parts of a project remotely for integration later (see Save/Load Clone below).

Export Project

For a client, or a client not attached to a server, it can be useful to save a copy of a project to a different location, for example to send for support diagnosis.

Set Default Project

Allocate any existing project as default, to open with Compere.

Clear Default Project

If, having set a default project (as above under Save) you want Compere to open without a project running, this will remove the setting you made.

Presets

Recall / Save As / Export

Record multiple arrangements for the same project, typically arrangements of windows on and off a canvas for different display purposes. Properties for all or selected UUIDs are saved and recalled. Recall will replace property values for all matching UUIDs. The Export option saves the Preset file locally, not in the Juggler network. (file extension: .pre)

Save (any) selected elements only, or save the contents of the Canvas panel.

Clones

Load / Save As / Export

Any entity in a Project assembly will typically be configured for the situation. All its properties can be saved for reuse. So if you are configuring a Juggler, or a projector, for example, select it in the Project panel and save it as a Clone. Similarly a canvas and its outputs can be dimensioned and arranged as a standard configuration. You can then load the same Clone file into another project where it will be added to the Project panel. The item can then be dragged, preconfigured, into your Project assembly panel (Nodes, Canvas, Visualise). The Export option saves the Clone file locally, not in the Juggler network. (file extension: .clo)

Save (any) selected elements only, or save the contents of the Canvas panel.

Quit

Close Compere (Project files are not auto-saved).

Panels Menu

Selected items can be used multiple times and are added undocked. Closing or undocking any these elements in a Layout does not lose any content. They can be used independently, for example on separate displays.

➤ See [Compere Layouts](#) [43]

Nodes

Add a Notes assembly panel to the Layout

➤ See [Node View: Connecting Juggler Systems](#) [94]

Canvas

Add a Canvas assembly panel to the Layout.

➤ See [Display Configuration: Canvas Setup](#) [78]

Project

Add a Project Panel to the Layout

Properties

Add a Properties Panel to the Layout

Logs

Open live log messages. See [Logs and Diagnostics](#) [132].

Resources

Add a Resources Panel to the Layout

Task Tracker

Open a process progress [Task Tracker Panel](#) [52].

Visualise

Add a 3D Visualise assembly panel to the Layout. This is a second side to Compere not covered by this Guide.

Reset UI

Restore the default Compere Layout (panel contents, arrangement and size) for the current View (i.e. Juggler or Visualise).
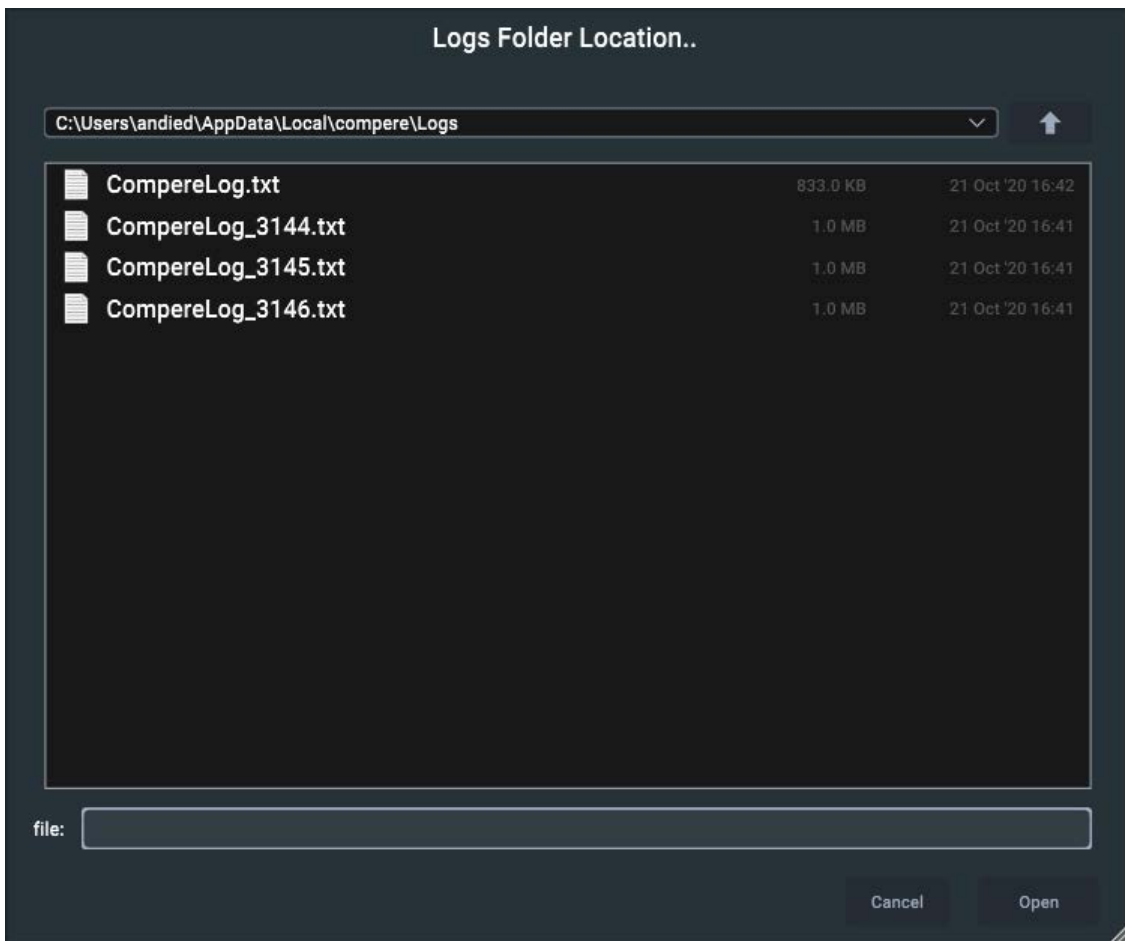
Configuration Menu

System Config Table

Displays a table of all Jugglers with their IP addresses, versions of software and firmware, port characteristics, and capabilities.

Gather Project Group Data

Lists logs per Juggler in the Project Group. Click 'Open' to create a zip file in this location of all logs and the local Compere Preference.pref file.

Network Configuration

Opens the Network Discovery and Project Group panel, for configuring a network of Compere devices.

➢ See Network Configuration [68]

Preferences

Opens the Preferences panel, containing ports and default file locations for the current instance of Compere.

➢ See Preferences Panel [55]

Help Menu

External Controls

List External Control [115] commands that can be sent to Compere to control it in various ways.

Show About
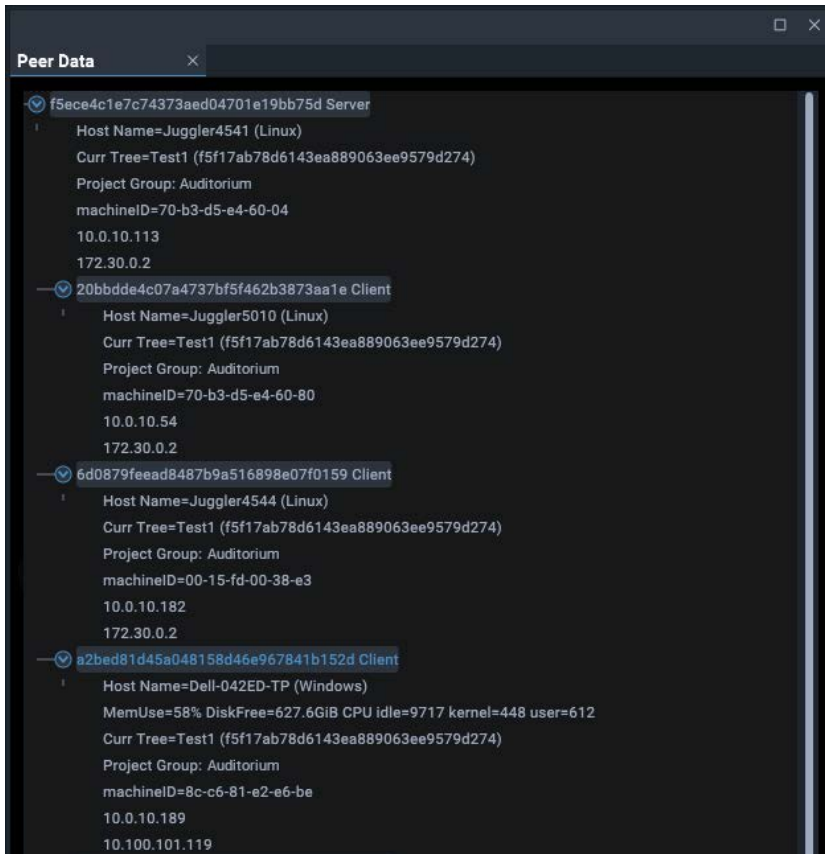
Show the version of Compere on this device.

Debug Menu

By ticking 'Enable Debug UI Elements' in the *Configuration > Preferences* Panel, this additional menu is added. Its contents are for diagnosis and maintenance only:

Peer Data

Server-client data for all instances of Compere in the Project Group, including disk and memory usage, machine Ids and IP addresses.

Note the server (Juggler), clients, and the Windows PC.

Juggler Front Panels
   Replicates a front panel display of the current Compere host.

ThreadComms Log
   Extended system logging information that can be cleared or saved.

Distribute File
   Used to select certain systems files to be delivered to all Jugglers in the Project Group.
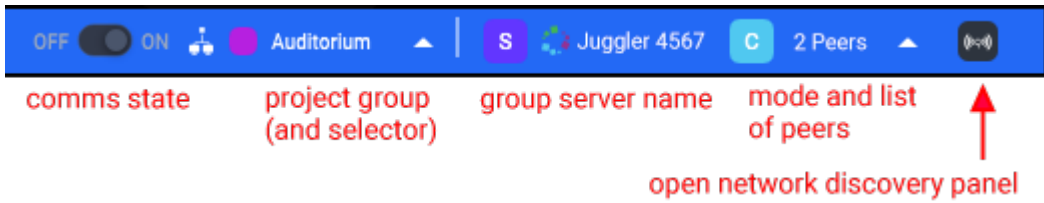
Status Bar

The status bar carries information about the current instance of Compere on the network. It has three colour states:

`grey` : offline. SInce it is disconnected, any information may be residual from the last online state.

`red` : online as a client but with no project group server present

`blue` : online and in a project group with a server.

comms state | project group (and selector) | group server name | mode and list of peers

open network discovery panel

In this example, Compere running this UI is in the project group 'Auditorium', with Juggler 4567 acting as group server to 2 client peers. The list available in 'Peers' shows their names and IP addresses.

The symbol next to the server shows its operating system icon: Juggler (Linux), Windows or MacOS.

Task Tracker Panel

Whenever a system task takes place (the example here is importing an MPCDI file, but it could be a system file distribution, restore or update etc.), a small progress meter will appear on the right sight of the main menu bar.
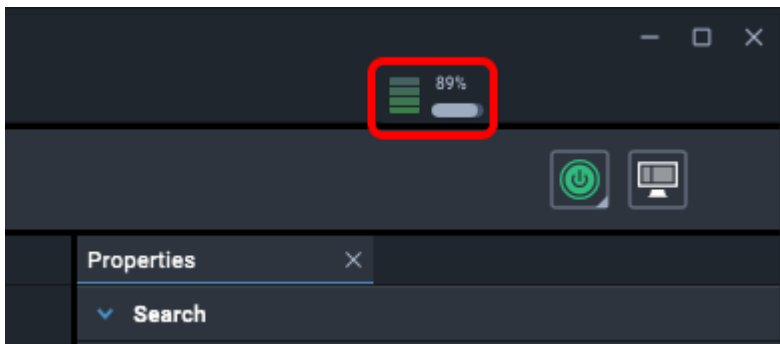
Its colour (red through yellow to green) will reflect current load, and by clicking on it, the Task Tracker panel will open, showing in progress, failed, and completed tasks. This panel is also available from the top menu *Panels > Task Tracker*.

There is a progress bar for each task and each subtask. Blue progress will change to green on completion. For longer tasks, click the arrow (here next to 'MPCDI') to expand the list for progress of subtasks.

When all tasks are complete, the list will clear and the task progress icon will disappear.

Failed tasks can be cleared by clicking the red button next to 'Overall Progress'.
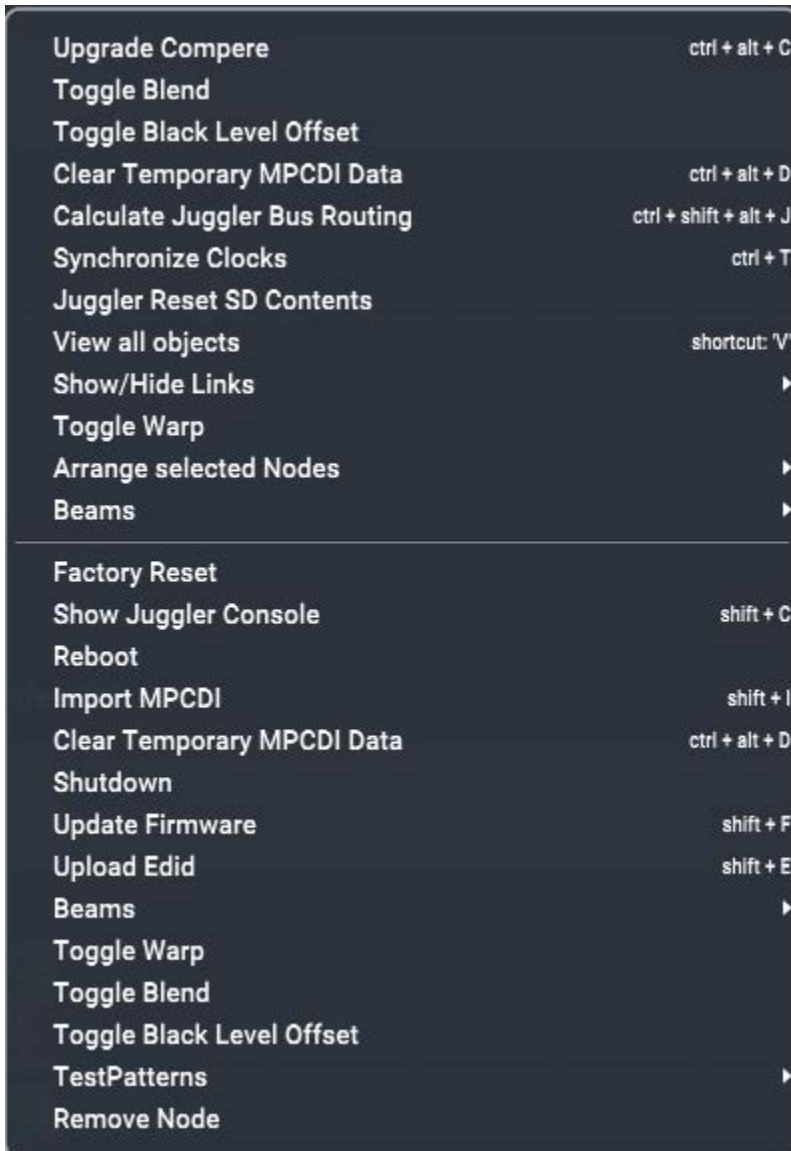
Contextual Menus

Right-clicking on an assembly panel background, or on a resource in an assembly panel, will open a contextual menu, many with shortcut hints. The object selected will change its orange border to green.

Example: a Canvas Window in the Canvas assembly panel, showing a submenu:
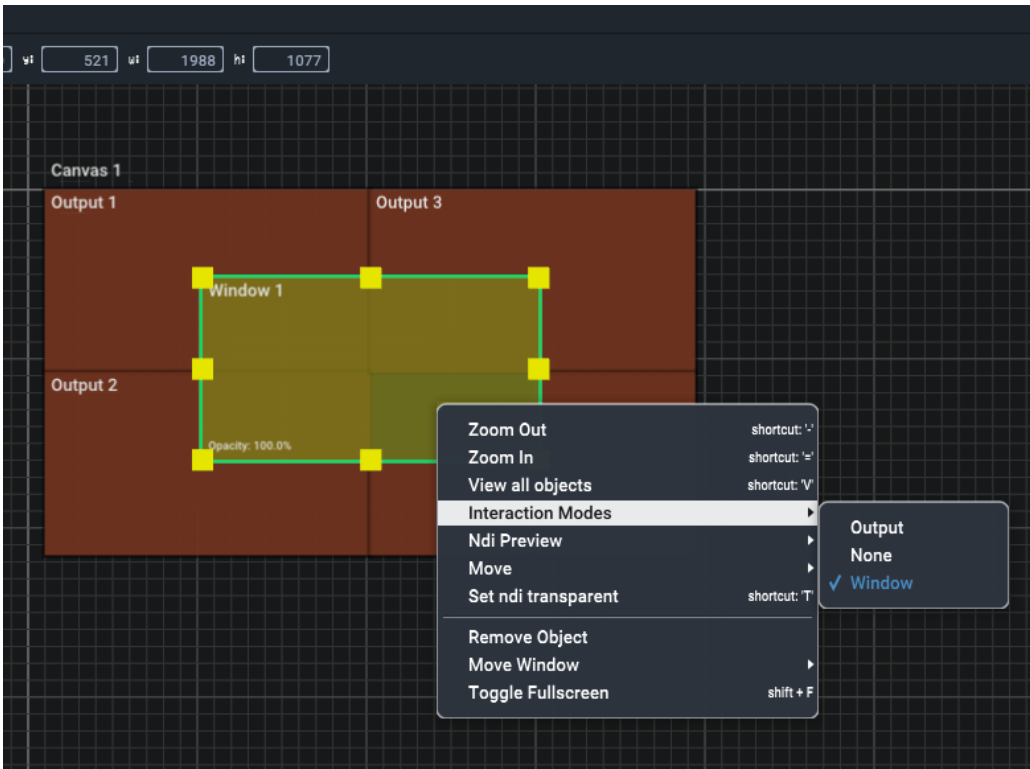
Contextual menus typically have two sections when called for an object rather than the containing panel. The upper section relates to the panel in general, the lower part to the selected object. In this example, the interaction mode is 'Window', so the two-part right-click menu contains generalised options such as Zoom, above object-specific options such as 'Move Window', which in this case has a further option menu. These last shortcuts are alternatives to using this menu, and are replicated as toolbar options.

Shortcut keys are themselves contextual to each assembly panel, though some, such as 'V' to view all objects, will be common. Here is another example – Juggler in the Nodes panel:

Upgrade Compere	ctrl + alt + C
Toggle Blend	
Toggle Black Level Offset	
Clear Temporary MPCDI Data	ctrl + alt + D
Calculate Juggler Bus Routing	ctrl + shift + alt + J
Synchronize Clocks	ctrl + T
Juggler Reset SD Contents	
View all objects	shortcut: 'V'
Show/Hide Links	▶
Toggle Warp	
Arrange selected Nodes	▶
Beams	▶
Factory Reset	
Show Juggler Console	shift + C
Reboot	
Import MPCDI	shift + I
Clear Temporary MPCDI Data	ctrl + alt + D
Shutdown	
Update Firmware	shift + F
Upload Edid	shift + E
Beams	▶
Toggle Warp	
Toggle Blend	
Toggle Black Level Offset	
TestPatterns	▶
Remove Node	

Some context menu items repeat the toolbar button actions. When the menu is divided, the upper section applies to all objects and the lower to the selected object(s).

Preferences Panel

Menu: *Configuration > Preferences*
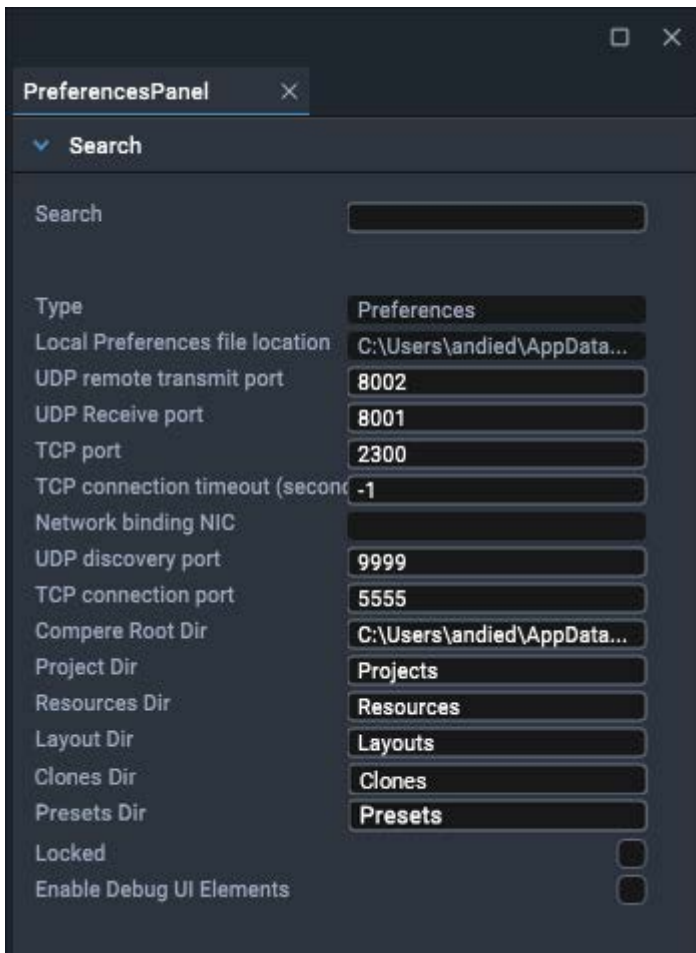
Search
Filter properties

Type
  (not editable)

Local Preferences file location
  File location where preferences are stored on local machine.

UDP remote transmit port
  UDP port number to which external control will transmit.

UDP receive port
  UDP port number external control socket will listen on.

TCP port
  TCP port number external control socket will listen on

TCP connection timeout
  Timeout on an existing TCP connection in seconds in which it'll be closed if no data is received.
  A negative value means there will be no expiry time.

Network binding NIC

> Shows the IP address of the current instance of Compere.

UDP discovery port

> UDP port used when discovering other Compere applications on the network

TCP connection port

> TCP port on which connections will be made between instances of Compere for Project updates.

Enable verbose logging

> Enable to activate extra system logging.

Locked

> Check to avoid accidental changes in the panel.

Enable Debug UI Elements

> Implements an additional [Debug Menu](50) for diagnostics.

File Locations

Compere Root Dir

> Where Compere data is stored, the root for subfolders. Default: C:\Users\[UserName]\AppData\Local\Compere

Project Dir

> Where you choose to store Project (.prj) fles on the Compere server device. Default: [Root Dir]\Projects

Resources Dir

> Where you choose to store Resources for Projects. Default: [Root Dir]\Resources. Add subfolders as required.

Layout Dir

> Where you choose to store Layout (.lay) files. Default: [Root Dir]\Layouts

Clones

> Where you choose to store Clone (.clo) files. Default: [Root Dir]\Clones

Presets

> Where you choose to store Preset (.pre) files. Default: [Root Dir]\Presets

Compere Projects

There is always a Project

Once Compere runs, on a control PC or as a Juggler, a Project is always present. This is true whether offline, or in the role of client or server.

- If this is an unconnected instance, Compere will, on opening, start a new empty project, or run a default project if one has been assigned.

- If the instance is a client or a server, is online, but is not a member of a [Project Group](75) with a server, it will open as if unconnected.

- If the instance is a client, is online, and joins a Project Group with an assigned server, it will receive and run the Project of that server.

- If the instance is a server, is online, and is a member of a Project Group, whatever Project is running (new, empty or default) will persist and be handed to all online Project Group clients present or joining.

As the Project is assembled, all changes made from any instance of Compere are referred back to the Project Group server to redistribute to all clients. Changes to a Project when made offline as a server are distributed on going online. If a client from a Group goes offline, edits the Project and then rejoins, its changes will be lost.

Saving and sharing, projects and parts

You can create and save as many Projects as you want in Compere. Parts of a Project (e.g. a configured Projector, Window, Canvas) can also be saved as Clones or Presets for reuse in Projects, so clear naming is important. Project (.xml) files can be saved locally, but can only be distributed among a Project Group by a server.

Edits to a Project by any member of a Project Group, whether in a client or server role, are collected and distributed to all others via the server.

Similarly, any device running Compere that joins the Project Group as a client will be given the Project file, displacing any other, so that it follows the same script as the 'stage manager' (server).

Server determines the project

Distributed files reside on all instances of Compere on the network (this includes every Juggler and control PC), so that every one has its own directions, and any one can be assigned the role of server. The server coordinates the distribution of the current Project to all clients, receiving edits from any client and distributing all revisions.

Compere can be used offline – as a client but switched to offline – or as a client without a Project Group server present. Project edits while offline will not be saved or distributed, but can be exported locally.

The Compere Project panel

The Project panel on the left lists everything that takes part in a Project. Many items will have elements with their own properties, for example a Juggler can be expanded in the Project 'tree' into its branches of inputs and output ports, and a Canvas can be expanded into its Canvas Outputs and Canvas Windows. When any branch of the Project is selected, it is highlighted orange and its properties are shown in the Properties panel.
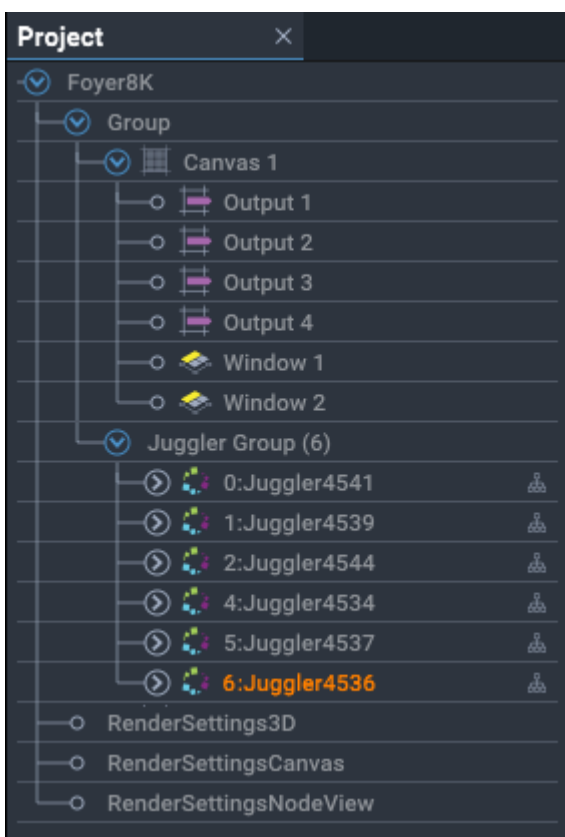
Multiple objects can be selected in the usual way using Shift or Ctrl, and the properties panel will only show shared properties of common values. However, other operations can be performed on multiple objects at once.

Project Properties: Name

The only property to address is the Project Name. This can be edited; it is not the same as the project file name. If you *File > Save As*, the project filename will change but the Project name does not.

When you start a new Project with *File > New*, you need to add a Project name:
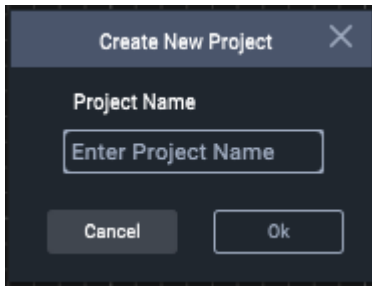
This will then show as the Project name in the Project panel.

The name can be subsequently edited in the Properties panel.

Render Settings

Render Settings are the display properties of the Compere layout panels themselves. This might be to see, or to set, how or from where the panel contents are viewed, or the zoom level. You can also choose whether to see the text labels of objects or not, to declutter a complex view.

Render Settings : Nodes panel

Mini Map is one feature that helps you locate the current viewing position within a large system of Jugglers. Also available from the toolbar button. Check / uncheck this box in the Nodes render settings (Show Mini Map):
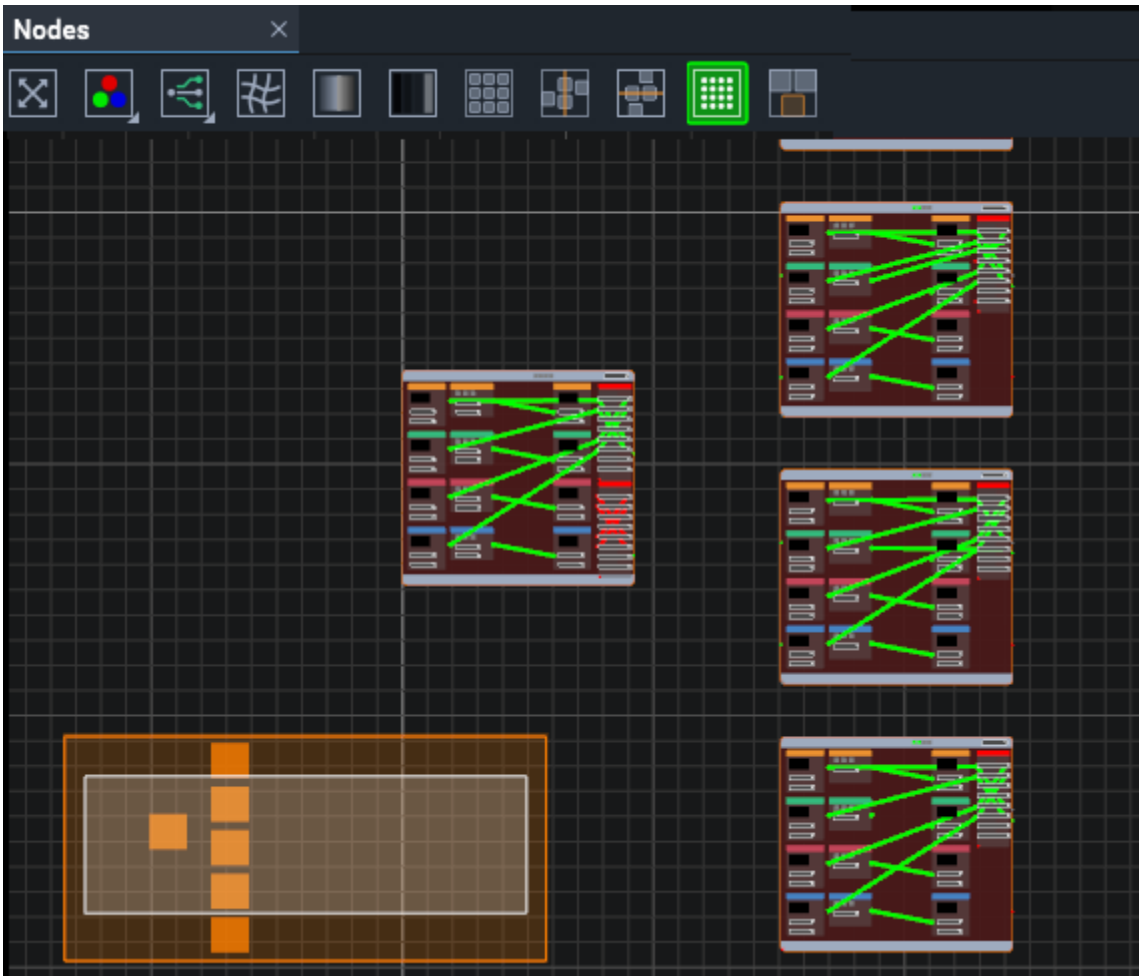
The orange box contains all objects. The lighter area is the current view scope and can be dragged around to view the required Juggler(s) without having to zoom out or in to find your bearings.

How Juggler is Seen in Compere

Each Juggler in a Project has its build configuration of available ports recognised by Compere, and this is exposed in the Properties and Project panels. When a Juggler is added to the Nodes [94] panel, the build options of the Juggler are represented as connector pins: square for external and round for internal. These are used to make connections within and between Jugglers, media sources and output devices. This is the full zoom level of detail for a Juggler node. As the Nodes panel is zoomed out, levels of detail are progressively reduced and ultimately replaced by simple colour blocks.
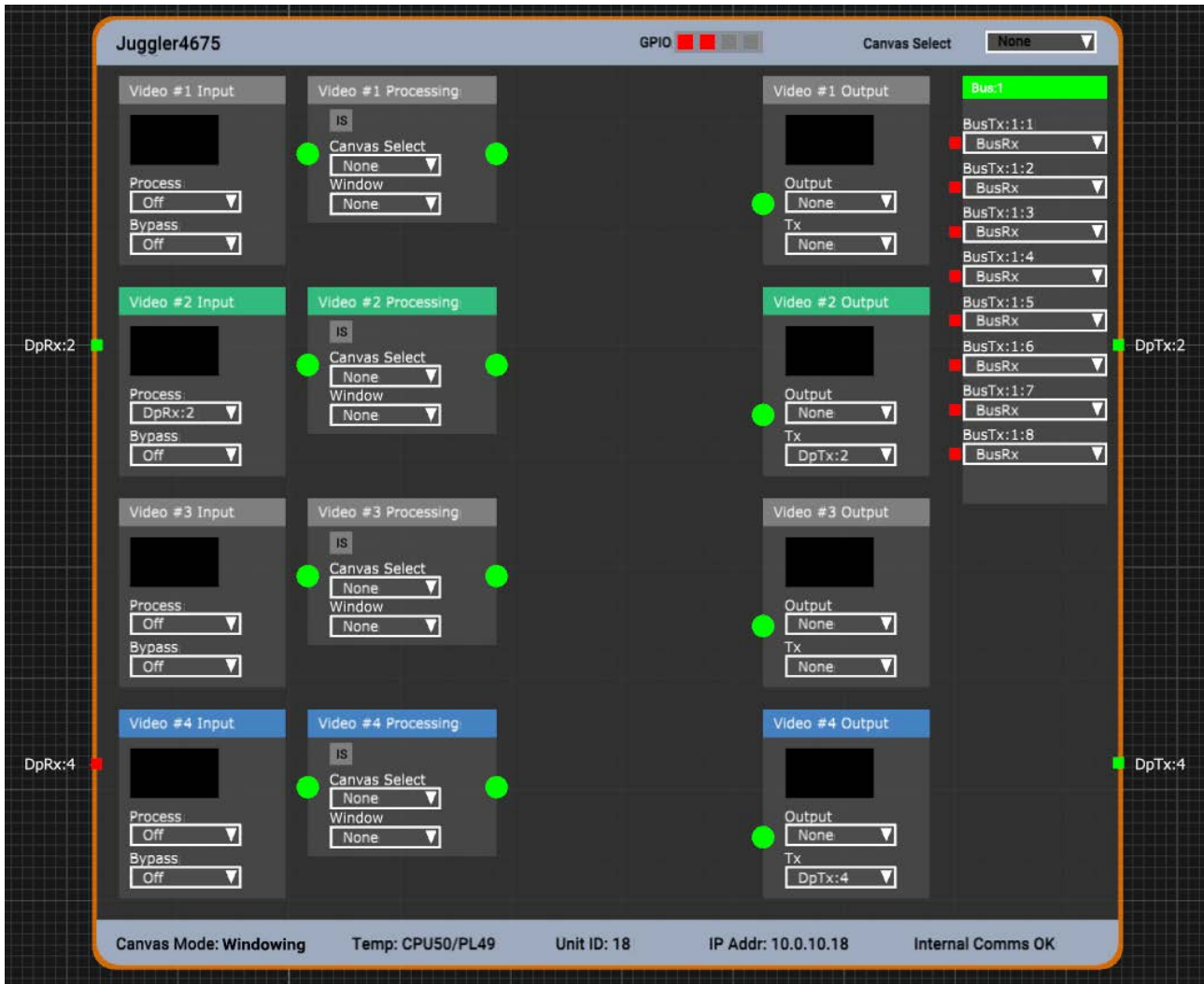
Jugglers in the network are visible in the Project panel as a group. An icon indicates that a Juggler is currently included in the Nodes panel. Each Juggler entry in the Project panel can be expanded into its parts, ports and channels. Select any element to examine and edit its properties in the Properties panel, or select the Juggler for all properties, broken down into the same sections:

Project

- My Project
  - Group
    - Juggler Group (6)
      - 0:Juggler4541
      - 1:Juggler4539
      - 2:Juggler4544
      - 4:Juggler4534
      - 5:Juggler4537
      - 6:Juggler4536
      - media
  - RenderSettings3D
  - RenderSettingsCanvas
  - RenderSettingsNodeView

Properties

Search

Search	

2:Juggler4544

Type	juggler
Name	2:Juggler4544
Enable NDI Confidence Monitors	✓
Enable Auto Identify	✓
Unit Id	182
Canvas	None
Warp Mode	Warp-1x4
Warp Zone Pixels	3840
Warp Zone Lines	2368
Uptime	0d:21h:06m
Auto Bus monitor	
System Pixel Format	
System Bits Per Pixel	

ATOM

External IP Address	10.0.10.182
Juggler Connected	
Enable UDP console	
Juggler Lowlevel SW Version	(2021.01.06-1) Wb-47 B...
Juggler FW Version	200.47
Display Port Version	1.2:2:2
HDMI Version	2.0:0:0
SDI Version	12G:0:2
BUS Version	16G:8:8
Capabilities	Warp/Blend
Compere version	1.0 Build 2 (03af24c)
Pic version	
Front Panel Default	Video

See further:

➢ [Juggler Properties](#) [64]

➢ [Nodes: Connecting Juggler Systems](#) [94]

Juggler Properties

Name

By default, the Juggler serial number. Rename, perhaps to reflect real-world tagging or purpose in your project.

Enable NDI Confidence Monitors

Check this to introduce confidence monitoring on Juggler I/O connectors and in the Canvas assembly panel. In the latter, all video can separately be switched off.

Enable Auto Identify

When enabled, the outputs from this Juggler will throw a sequence of RGB colours in turn onto an output to identify it, when an output is assigned to a video channel.

Unit Id

Last octet digits of the Juggler module's IP address, to provide an identifier within the Project Group. Unit Id is appended to connector identifiers and their type, e.g. when one Juggler AV connection is receiving from an AV connection on another.

Canvas

Select from list of Canvas names created (there may be only 1). Applies to all video channels of the Juggler (same setting as in the Juggler title bar in the Nodes panel).

Warp Mode

There are two current versions of Juggler firmware: warp and blend or windowing. For warp and blend units there are 3 warp modes: 1 × 4, 2 × 2, 4 × 1 (this X-Y orientation setting has to match supplied warp files). Warp Mode describes how the four input channels into the particular Juggler unit should be interpreted. Defining the arrangement of the source channels allows an output to span across multiple inputs.

WarpZonePixels / WarpZoneLines

(Applies to warp and blend firmware version only.) Provides a mechanism to override the size of the input Warp Zones (i.e. the video feed into each Video Channel). If this is left at zero then Juggler will use the incoming video resolution of Video Channel:1 to determine the correct size – however, this can cause issues if the video is not present when the warp is being calculated. **Note:** All Warp Zones must be the same size.

Uptime

Total time continuous running of the Juggler FPGA since last restart.

System Pixel Format

Select as appropriate.

System Bits Per Pixel

Select as appropriate.

ATOM Properties

For reference only, software and firmware configuration versions installed on the internal ATOM CPU running the Juggler.

Inputs and Outputs

Properties for each type of input and output are described separately. The types available depend on the particular build configuration of each Juggler.

System Monitor

Includes internal temperature monitoring of the Juggler module.

Defaults

Default properties relate to Juggler properties held not in the Project, but in the Juggler module's firmware on its internal SD card.

Note: If edited, you must click save data and reboot to effect any changes.

SD Card

The SD card inside a Juggler holds essential configuration files, including EDID, MPCDI, warp, blend and colour correction files. This section will list all files added to the SD card.

To clear all added files, select the Juggler in the Nodes panel, and from the context menu select 'Juggler Reset SD Contents'.

Network Configuration

Compere Roles: Server and Client

Every Juggler is already running its own instance of Compere. Any PC added to the network, running Compere, provides a user interface (UI) into the network and can join a Project Group, but it does not need to remain present. Every Project Group member shares the same Project file and edits are sent back to the Project Group server to distribute. Any instance of Compere in the Project Group can be assigned the role of server and become the holder and distributor of the Project (as 'stage manager'). Any instance of Compere joining a Project Group will receive the server's Project file. Any edits made to the Project file whilst offline will be lost.

Each network Project Group looks after itself, so that if the server instance of Compere drops out another can be assigned the role. For this reason a control PC running Compere UI should not be assigned as server.

Jugglers or PCs running Compere in the Project Group can also be specifically addressed using external control commands sent from another PC to the Project Group server.

Project Group Server
In this role, Compere can connect to any number of clients but does not connect to other servers directly. A server shares its projects with clients and there can only be one server per Project Group.

Project Group Client
In this role, Compere can connect to any number of servers but does not connect to other clients directly. A client does not share its own projects, so any Project opened on a client will be dropped on joining a Group with a server.

Every instance of Compere can be either online or offline. If online it can be acting as a client or as a server. This toolbar button/switch shows online status:
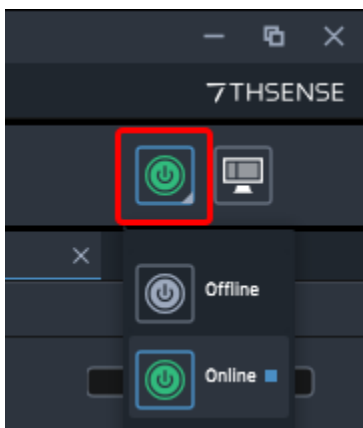
One instance of Compere (only) is assigned as Project Group server, all others are clients. If the server fails, a different client can be configured as the Project Group server. A control PC (e.g. a laptop) is able to assign any Juggler as server. The control PC can therefore be taken away or reinserted once the system is configured and running. Alternatively, redundancy or control bandwidth can be built into a system by dedicating a permanent control PC ready as server.

Configure the Network

Go to *Configuration > Network Configuration* where, in the 'Discovery' tab you should see all Compere devices (Control PC and all Jugglers) on the local network:

Your instance of Compere has the green flash across it marked 'Local'.

Discovery tab

Columns 'Hostname', 'IP Address', and 'Project Group' in this tab can be sorted by clicking on the column heads.

Hostname

This is the name that has been given to Juggler units (or PC). Initially these will be the names assigned when they were built.

App Ver

Different versions of Compere might conflict within a Project Group. This column shows the version on each instance of Compere in grey. If there are different versions of Compere running this may not present a problem, but a blue alert is given indicating that different versions are in play. (Note: if [Debug](50) mode is enabled, the Compere version information is the more specific fingerprint.)

Comment

Edit this for clarification of location, role, etc., for your own use.

IP address

Shows the current IP of each unit.

> **Note**: The internal IP range for all Jugglers is 172.30.n.n so do not use this range for your network of Jugglers and other devices.

Project Group

Which devices are going to share the Project file ('stage manager script')? Jugglers communicate by being assigned to a Project (Network) Group. Initially, all instances of Compere available on the network appear under a Group label 'Default', which is not a communications group. Project Groups are created in the other tab of this panel. Project Group is directly selectable here from the list of Project Groups you set up. Once in a named Group any instance of Compere can be put back into the Default pool.

You must also belong to the same to the same Group as the Jugglers with which you want to share a Project.

Comms Mode

Offline, or online as client or server. The adjacent marker shows if online.

> Click 'Apply' before any other changes, such as adding a Binding NIC. You may see an orange bar with the alert that 'Network Configuration is in a temporary state'.

Each Compere host shown here requires configuring, so click on the three dots to open the 'Edit Config' dialog. Here you need to edit the Comms Mode to assign a role (Off, Server or Client):
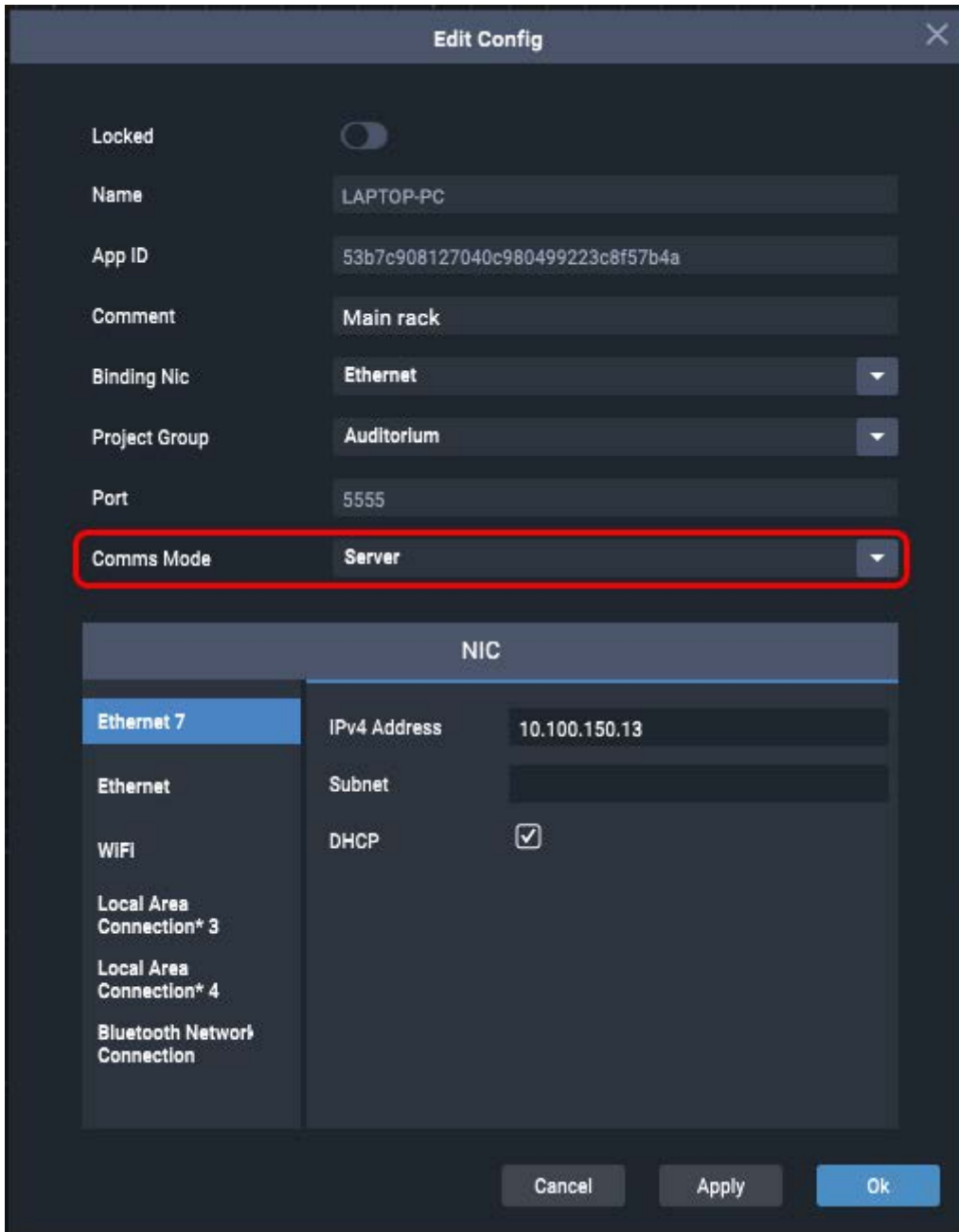
You must have one instance of Compere running as server per Project Group. Where the role is server, there must be a binding NIC: in the lower panel you can see details of available NICs. If joining a Group with no server, there will be a warning message to alert you here, and in the Discovery list:

You cannot have two servers on binding NICs in the same range, and an orange warning will appear to indicate any conflict. You cannot have two servers in the same Project Group. Trying to add a second will be prevented, with a red warning message displayed.

Locked
> Secure your settings by checking this box.

App ID
> Not editable: identifies this instance of Compere.

Comment
> Adds a brief description to the Discover list.

Port
> Default Compere comms port is 5555; edit another free port as required.

Do I need to create Project Groups?

No, if you do not need multiple Project Groups. The initial default is that unassigned devices will communicate with each other in an uneditable group called 'Default'. If you do not need multiple groups of devices, Default will be your Group. If you do create Project Group(s), make sure all Compere instances are assigned to created Groups, with one server role in each Group.

➤ <u>Create a Project Group</u> [75]

Once you have configured one instance of Compere as server, configure all the others as clients. The assigned role will remain with the instance of Compere, wherever it goes, until reconfigured.

The server-assigned Compere will hold the initial configuration for the Project. The Project file can then be saved locally and as server, it will be distributed to all other instances of Compere in the Project Group. In the event of losing a server connection, any Juggler can subsequently be handed the role of server and redistribute the project.

This example shows instances of Compere set up using Project Groups:

The Project Group members, and only the Group members, are shown in the Compere Project panel. Jugglers not in this Group will not be seen, and do not participate in the Project running in this Group.

Changing to another Project Group

Role takes precedence.

- If the instance of Compere being moved to a Project Group is a client, it will adopt the Project being distributed by the Group server.

- If the instance of Compere is a server and is being moved into a Group without a current server, it will distribute the Project that it is currently running.

Before moving an instance of Compere into a Group, check first how you want to join – as server or client.

… as an online client

Your instance of Compere will be given the Project file of the Project Group by the server in that Group, on joining.

… as an online server

If you are joining a Project Group that has no current server (maybe it has failed or been withdrawn), you will be asked if you want to create a new Project. If not, then as a server your current Project will be distributed to the client Group, which will be reconnected.

Lost Network Connection

There must be an instance of Compere running as server in a Juggler system. If the server is disconnected or fails for any reason, the status bar of Compere on an editing PC will be red:

Clients continue to run with the Project but are no longer synchronised and no changes are exchanged.

Create a Project Group

Jugglers and devices on a shared network will be discovered by Compere running on the same network. In Compere on your control PC, go to *Configuration > Network Configuration*. This will bring up the Network Configuration Panel, which has two tabs: 'Discovery', and 'Project Group':
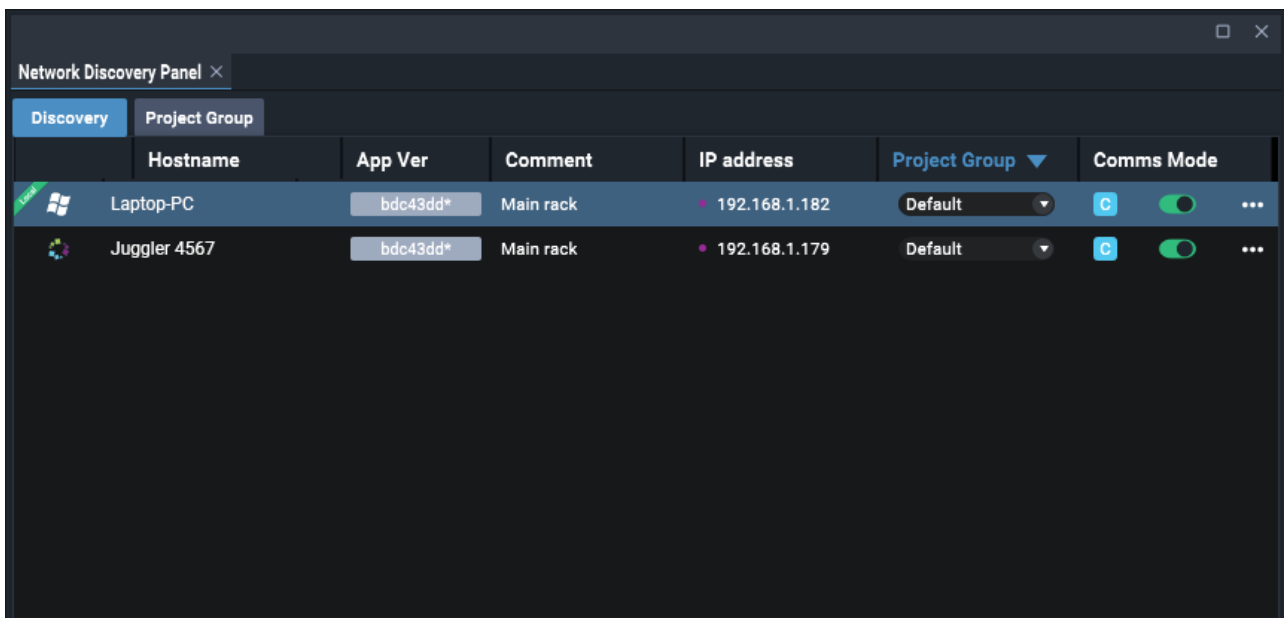
What is a Project Group?

All instances of Jugglers and PCs running Compere that wish to share a Project must belong to the same Project Group.

You do not need to create a new Project Group unless you want more than one, or wish to name it. Each Juggler on the network belongs to the 'Default' Group, or can be assigned to your own Group(s). If you need only two Groups, create two new named Groups, rather than using the Default Group plus one named Group.

Click the 'New Group' button and enter a name and the port on which this Group will communicate (both are required), and click 'Create Group':
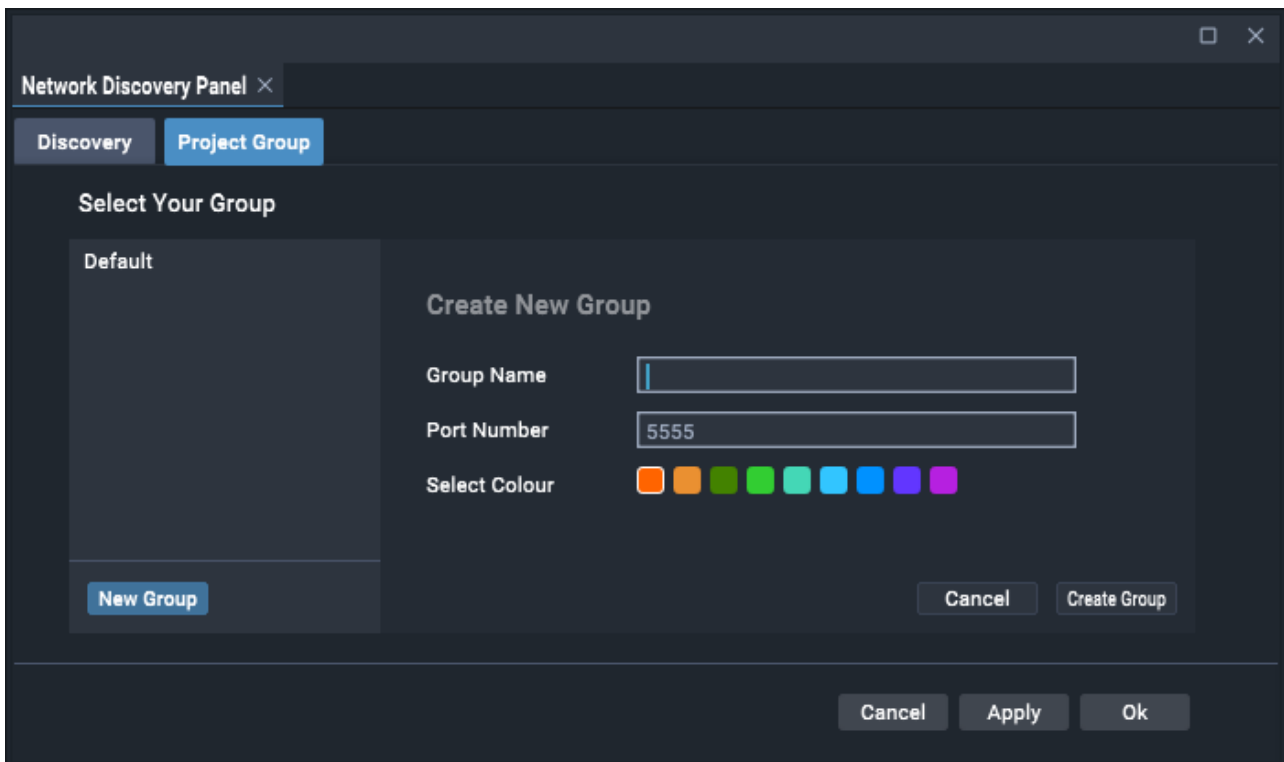
If you go back and edit these details, click 'Apply' and 'OK'.

Group Name

Provide a concise name for this group of Jugglers. Duplicate Group names are disallowed.

Port Number

The port you wish to use for this Group. Groups are separate and do not communicate with each other. The default Compere port is 5555, but can be any available port.

Select Colour

Choose a colour to characterise this Project Group. This will help in the Discovery tab to identify members of different groups.
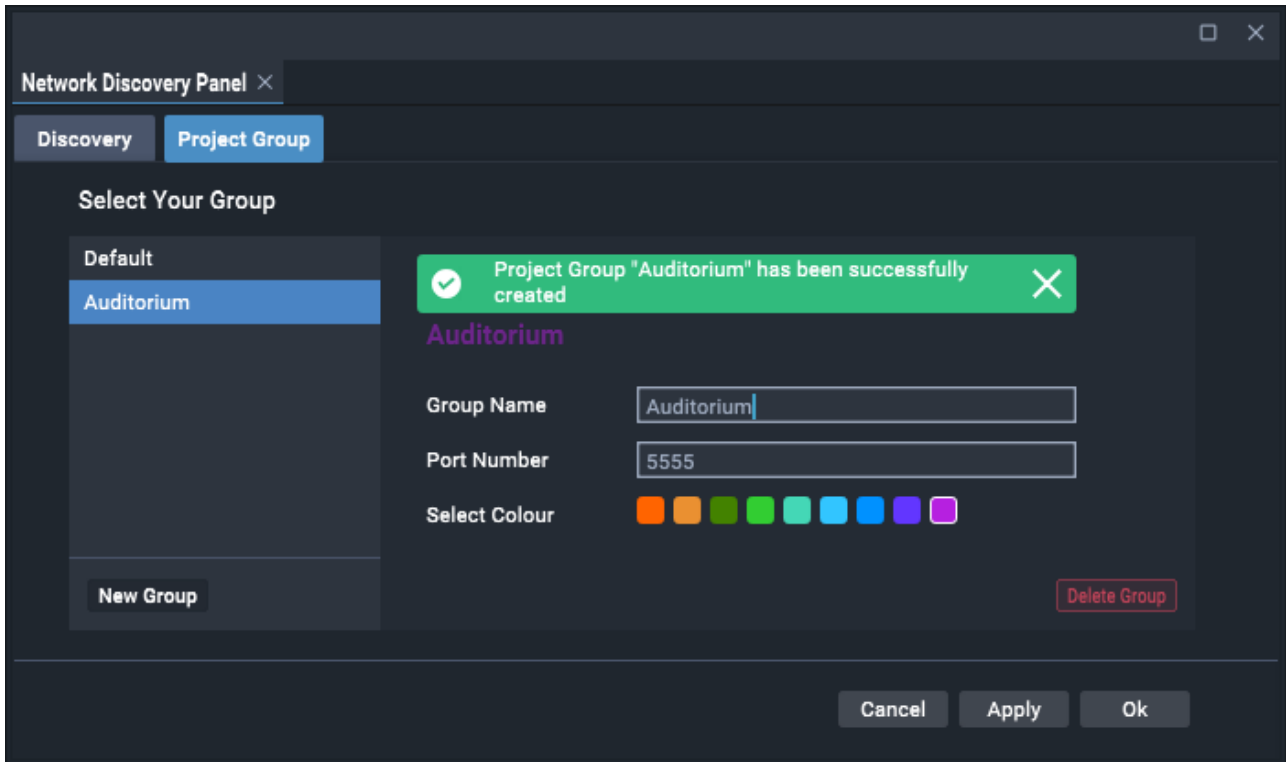
Add as many Groups as you need. Groups can also be deleted and any members will revert to the Default group. The colours are used in the Discovery tab:
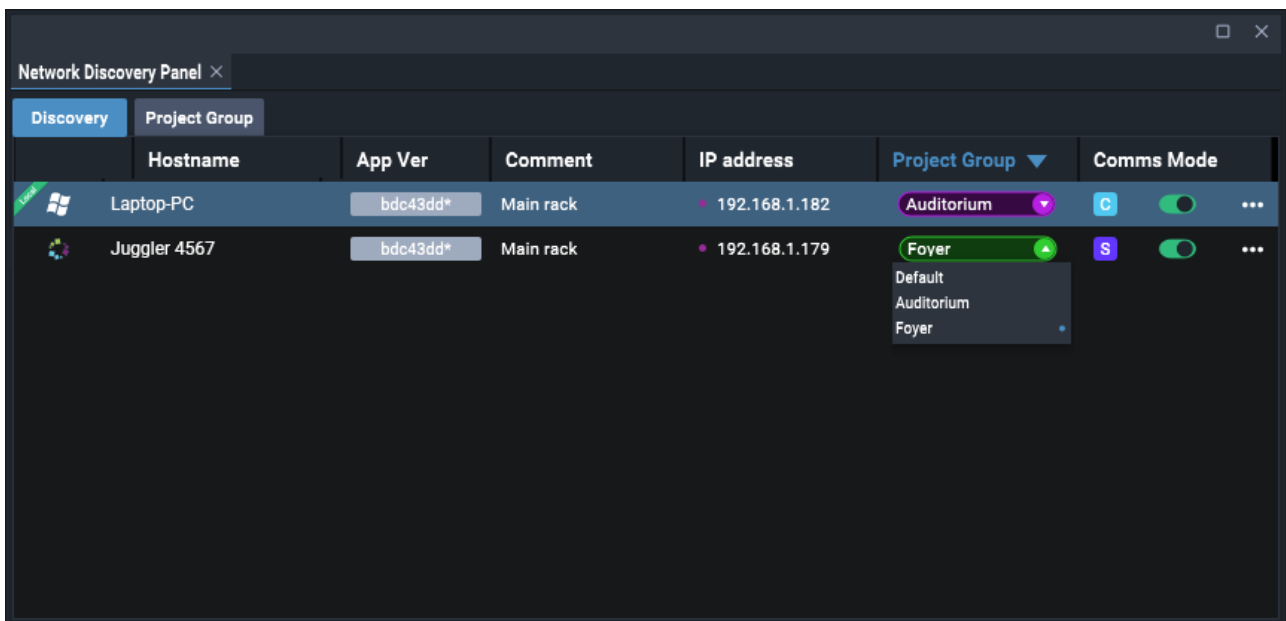
Display Configuration: Canvas Setup

A **Canvas** in a Juggler system comprises all available pixels in a Juggler system as a whole. Within this, **Outputs** to the resolution of available devices are positioned. Depending on the firmware type for your Juggler, direct one-to-one feeds can be made to these Outputs, or to **Windows** as areas anywhere in the Canvas, even across output boundaries. Which type of Juggler you have – Warp/Blend or Windowing – is defined in its factory build (you can see which, in *Juggler > Properties*, under ATOM, Capabilities).

The following description is for Windowing Jugglers.

Canvas, Canvas Outputs and Canvas Windows

The Canvas, the Outputs on it, and Windows in them are all laid out in the Canvas assembly panel, simply dragged there from the Resources panel. Each of these will be connected by Compere into the Juggler system, using the Nodes assembly panel. As each element is dragged in, it is added to the Projects panel, its properties become available, including how the element is named.

- Drag a canvas from the Resources panel into the Canvas assembly panel.

- You will now have the grey overall canvas maximised to full view in the panel. Since it is selected, it will have an orange bounding box. The initial default dimensions are 3840 × 2160 (4K).

- The Project panel will now contain a branch for this canvas, and the Properties panel is populated for the canvas.

- You may only need one canvas.
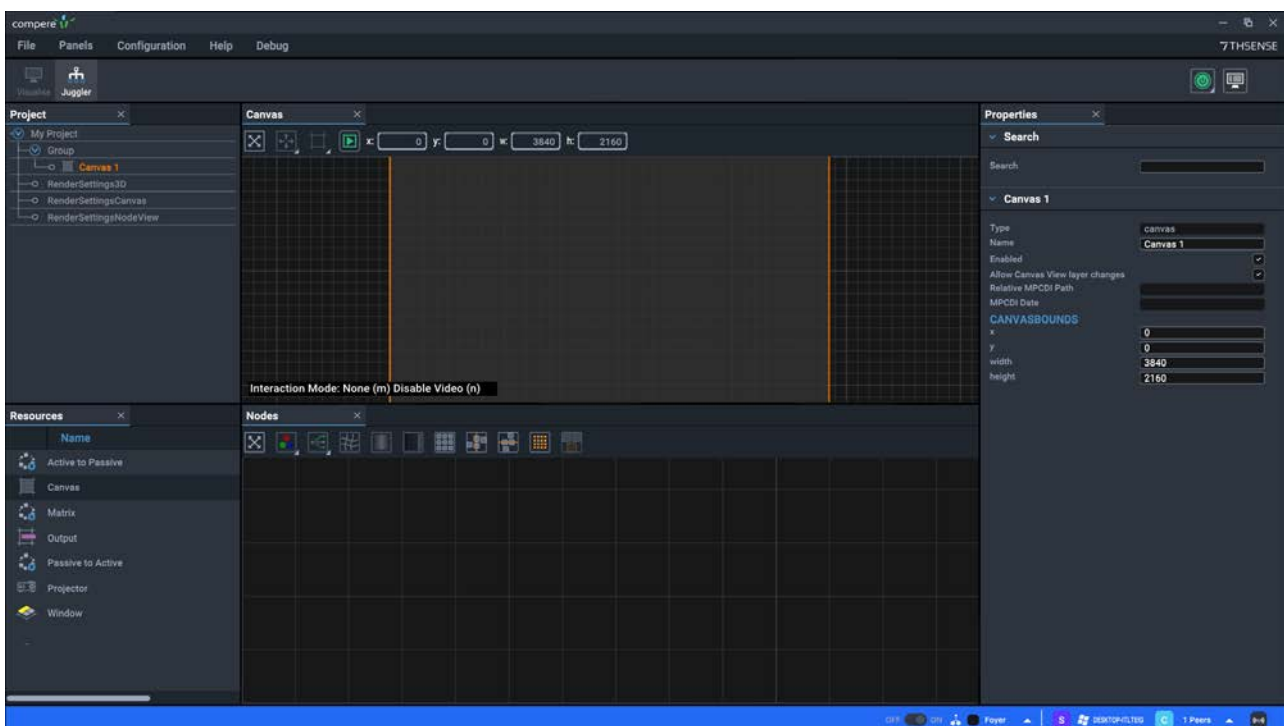
Below the canvas, the **Interaction Mode** is shown:
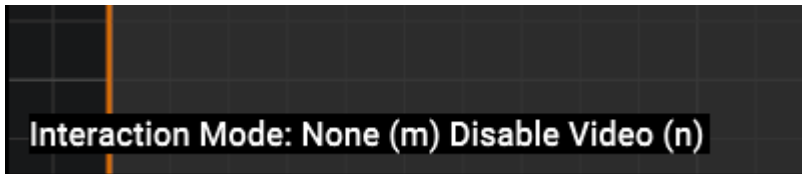
This mode switches the selection layer between the Canvas (none) , Outputs, and Windows.

Pressing 'm' switches between the three types of canvas objects so they can be selected and moved independently. Once set up you will not need to move the canvas, but you will add and position Canvas Outputs and Canvas Windows within it.

In 'None' mode, canvases can be dragged on the grid to position them, but not to resize by dragging.

Mode selection is also available on the toolbar button or on the right-click menu as 'Interaction Mode'.

Canvas Panel Toolbar
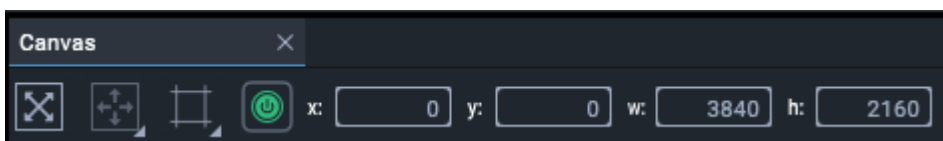
- The first button operates the same as pressing the 'v' key and will bring into scope everything in the panel.

- The second button refers only to Canvas Windows, and provides a list of common positions (top, bottom, left, right, centre and fullscreen).

- The third icon operates the same as pressing the 'm' key, switching between object types for selection.

- The green / grey button icon operates the same as pressing the 'n' key, to enable or disable NDI preview in the Canvas panel.

With the canvas selected in the Project panel, configure its properties in the Properties panel.

Canvas Properties

Name
> Name this canvas clearly. Multiple Canvases are by default numbered sequentially. It will be listed in your project and selectable for connection to Jugglers. Names can also be changed in the Project panel by double-clicking the name there.

Allow Canvas View layer changes

> Unchecked, the layering will be fixed.

Relative MPCDI Path

> If using MPCDI data, enter the relative folder path where you choose to keep MPCDI data files.

Canvas bounds :X and Y, width and height

> Specify the dimensions of a Canvas.* If using more than one Canvas, these are all inserted at 0,0 X,Y coordinates, so set the X and Y offsets to displace a second Canvas.

* **The resolution** (Width and Height in properties) of the Canvas for a Project is set to at least the maximum overall requirement. It does not necessarily represent contiguous physical space, only the number of pixels available to be claimed by outputs (see Output Setup ⁸⁰).

Canvas assembly panel shortcuts

| Key	Action
M	Cycles the focus to move the Canvas grid as a whole (Mode: None), to move and size a media Window (Mode: Window), or move and size an Output channel (Mode: Output).
N	Toggles any current media NDI feed on and off, instead of showing elements as colour blocks.
V	View all objects (i.e. scale to fit everything in the current assembly panel).
+ / −	Zoom the grid in and out (not num pad keys)
mouse wheel	Zoom the grid in or out.
Left mouse hold	Move the Canvas grid around as a whole, or if a Canvas Output or Canvas Window is selected, move this around.
Shift+drag	Hold the shift key to snap to the grid when positioning or resizing using the mouse.

Canvas Output Setup

Canvas Outputs are the channels driven from Juggler transmit (Tx) ports that together will comprise your display area(s). They are sized to the physical pixel resolution of the output device. The EDID of Tx ports are set in the Juggler using Nodes View.

- Drag an Output from the Resources panel onto the Canvas.
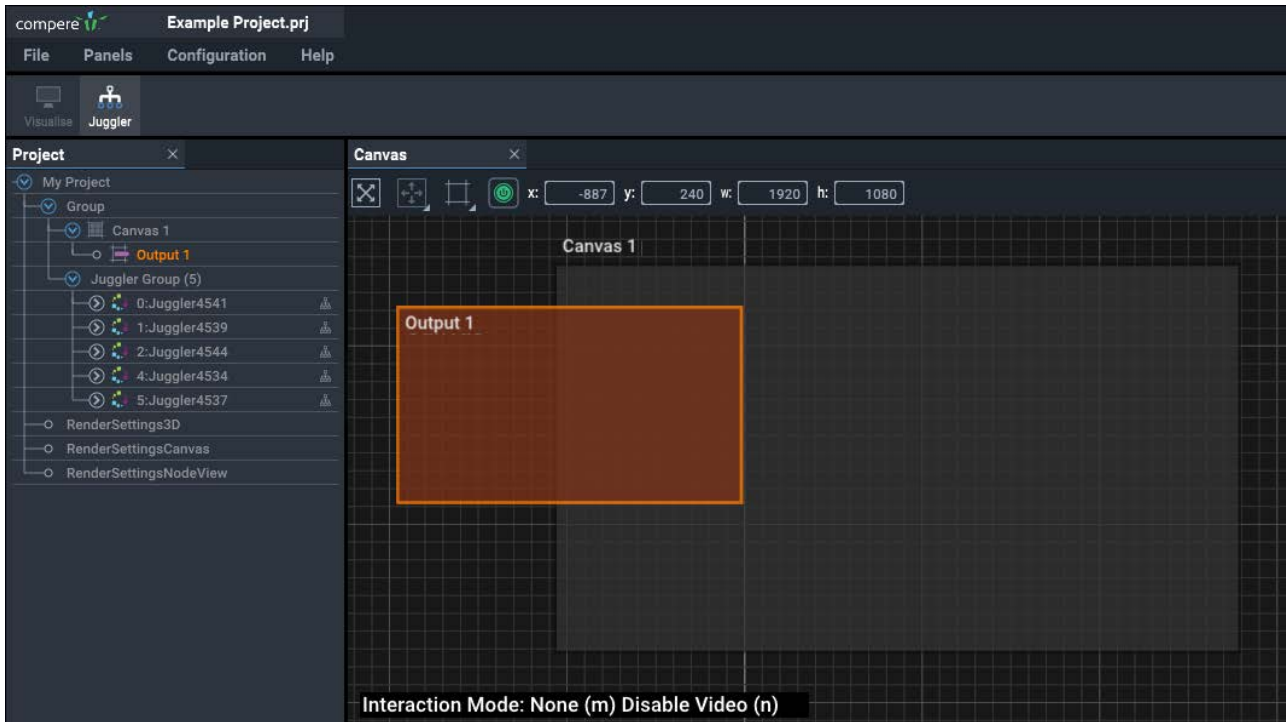
Again, since it is selected it will have an orange bounding box.

The Output is now visible as a branch in the Project panel, and by selecting an Output in the Project panel, you can configure it in the Properties panel. Add all your Outputs in the same way:
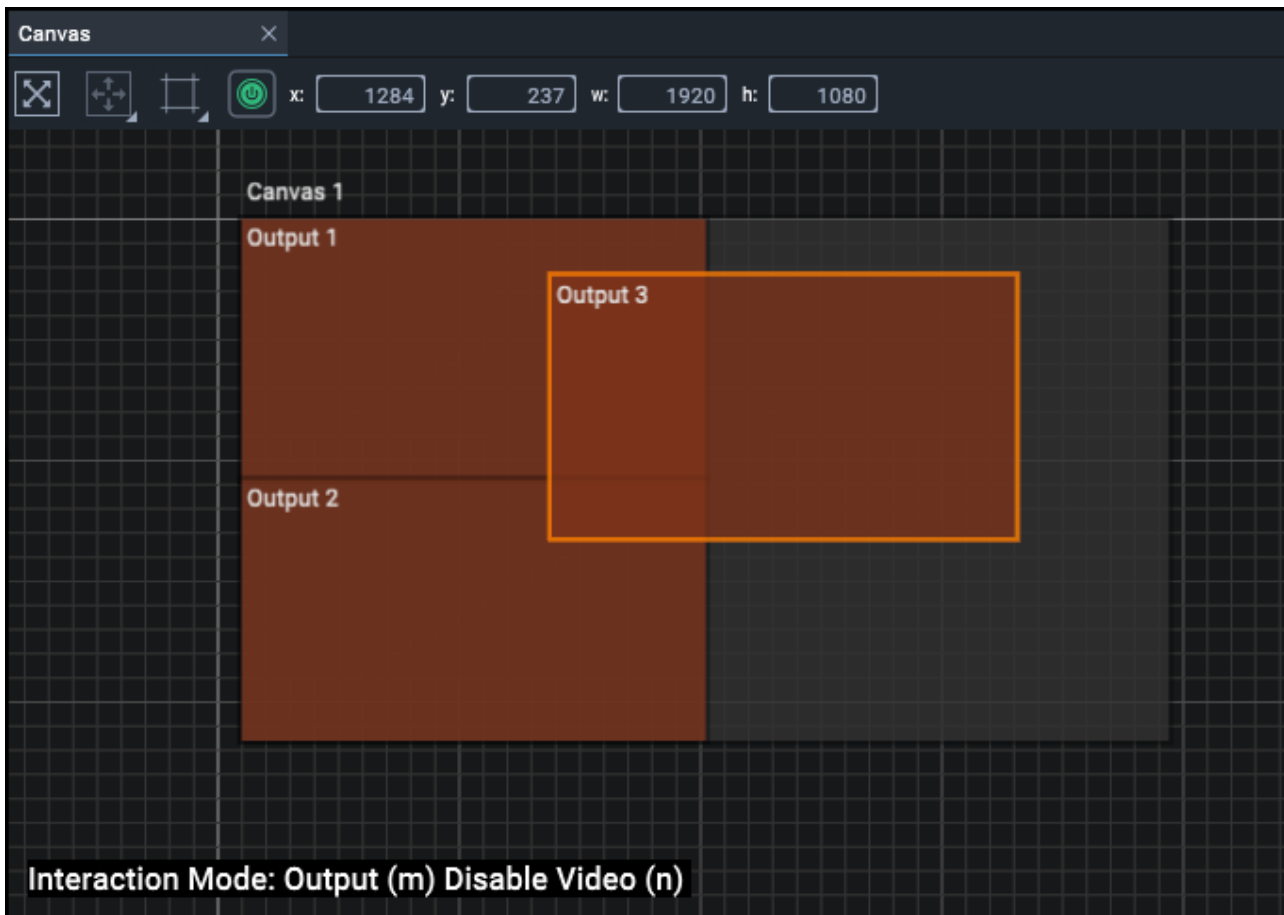
Note how the position and dimensions of the selected object is shown in the toolbar.

- Change the **Interaction Mode** to 'Output' by pressing the 'm' key while in the Canvas assembly panel until it reads 'Output'. The same toggle is available on the toolbar button or right-click menu as 'Interaction Mode'. This enables direct selection by double-clicking, otherwise Outputs are selected in the Project panel as branches of the Canvas.

Move an Output around the assembly panel by selecting and dragging it. Hold Shift to snap the Output to Canvas edges or to other items in the panel. Alternatively, type in the position, with the dimensions, into the Output Properties or the toolbar fields.

Right click Canvas objects for the shortcut menu item 'Move', offering predefined positions and incremental adjustments:

- The Up/Down/Left/Right arrow keys move the object towards that boundary of the canvas, 1 px at a time.

- Add Shift to move in 10 px increments.

- Add Ctrl+Shift to move in 100 px increments.

- Just add Ctrl to move straight to the boundary.

Window objects only:

- Shift+C will centre the item in the canvas.

- Shift+F will make the window fill the canvas.

Apply an Output EDID. The properties of an output will adopt the EDID of the physical Juggler port to which it is assigned. When, in the <u>Nodes assembly panel</u> ⑨⁴, you select a Canvas Output for a video channel output that has been assigned to a physical Juggler port, the Output *here in the Canvas* will be dynamically resized for you. In this next example in the Nodes panel, the **Tx** (physical output port) for channel 2 has been assigned to DisplayPort DpTx:2. By selecting 'Output 1' in the **Output** selector, the EDID for DpTx:2 will be applied to Canvas 'Output 1', and the Output in the Canvas panel will adopt the correct dimensions.
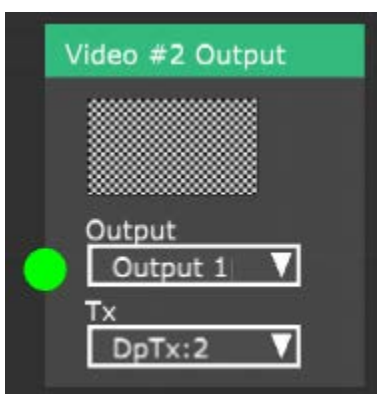
Edit the properties of an Output by selecting it in the Project panel to populate the Properties panel. Outputs and Windows are branches of the Canvas, which can be expanded or collapsed.

Output Properties

Name
Name the output logically and clearly. Multiple outputs are by default numbered sequentially. Names can also be changed in the Project panel by double-clicking the name there.

Can Scale
Allow feeds to this Output to be scaled to fit the Output display resolution. The default dimensions of Outputs are defined by the EDID of the node to which they are docked.

Power of Two Scaling
Optimise rescaling of frames to the Output display resolution.

NDI Available
Indicates whether NDI is available on this Canvas Output.

XResolution / YResolution
Populated when using MPCDI.

Canvas Position
XY origin width and height. Precisely position and size this Output in the Canvas.

Canvas Window Setup

Having set up the output areas, you can now place input sources (media feeds) on them. The placed input source is called a Canvas Window and can be positioned and scaled anywhere in the Canvas area, on or off the Canvas itself.

- Drag a yellow Window from the Resources panel into the Canvas assembly panel:
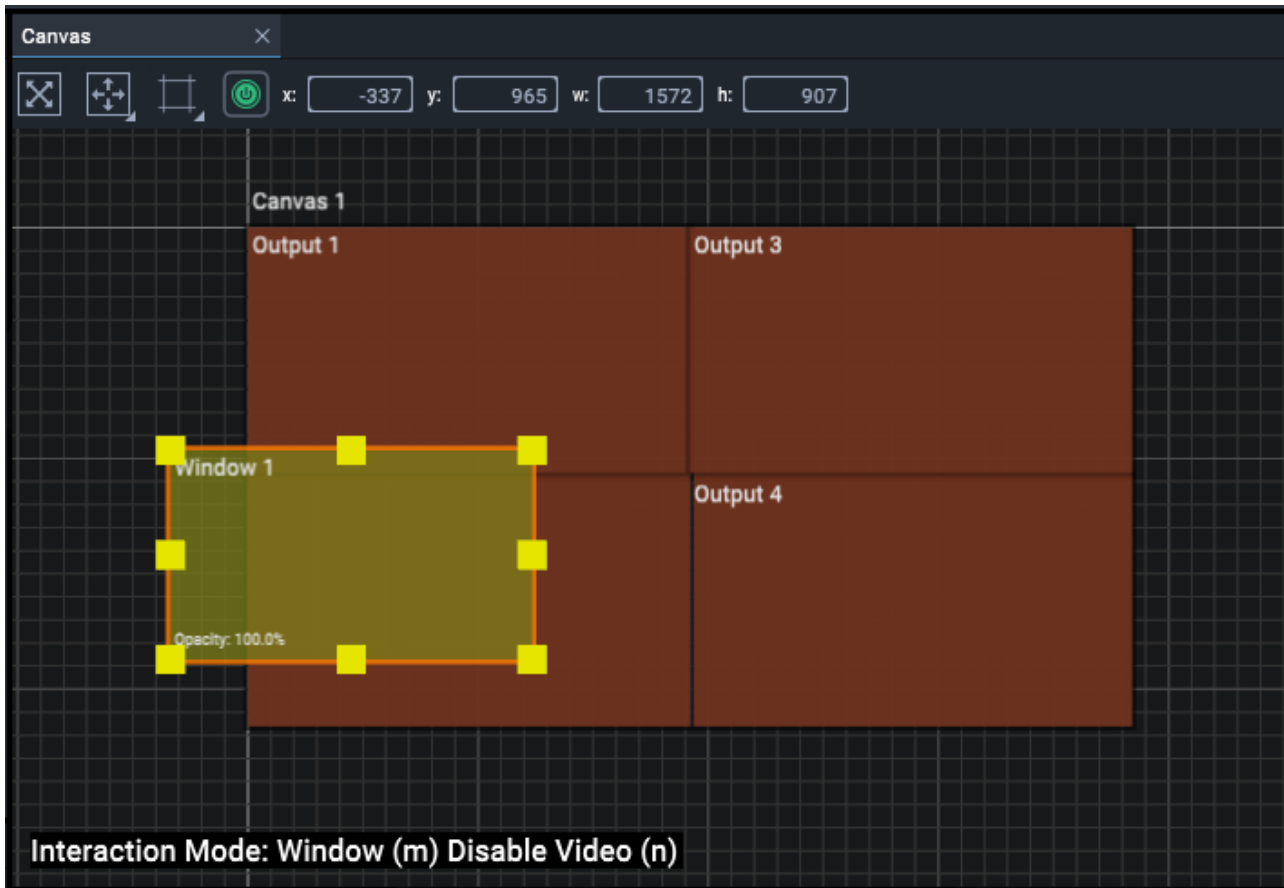
As with the Canvas Output, an unassigned asset adopts a default resolution. The Canvas Window has been added as a branch to the Canvas in the Project panel. Select it there and its properties become available in the Properties panel.

Interaction Mode: Use the 'm' key toggle to 'Window', to enable dragging and also sizing of Windows. Select Windows by double-clicking, or in the Project panel as a branch to the Canvas.

If any window appears to be inaccessible for dragging and resizing, for example, obscured by other windows, select it by name in the Project panel.

Position the Window anywhere in the Canvas assembly panel. Its size reflects a media source sent to it, scaled as required, and is not bound by any Canvas Output dimensions.

When within the Canvas bounds, the area of a Window that overlaps an Output on the Canvas will be displayed/projected. Any part of the media feed to this Canvas Window that lies outside an Output will not be seen. In this way streaming feeds can be moved around, on or off the Canvas display dynamically, either by interacting directly in this Compere interface, or by External Control [115].

Resize a Canvas Window with the mouse using the grab handles. Add Shift to snap to other object edges. Add Ctrl when sizing from a corner to maintain aspect ratio.

The other toolbar button is now active to move selected Windows around to common positions. The Window can also be toggled to fill the Canvas and return to its set size. Shortcuts are listed in the toolbar menu.
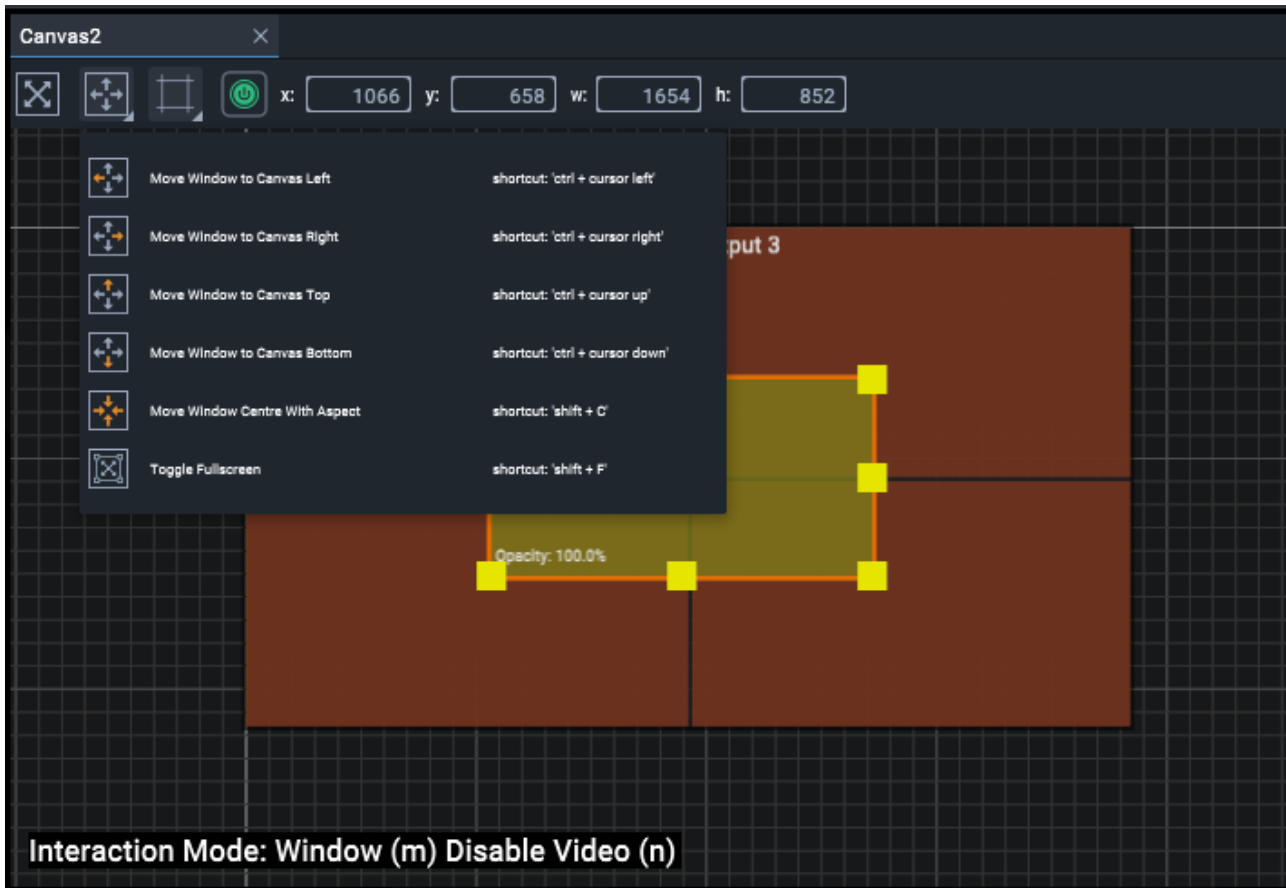
Window Properties

Name
Name Canvas Windows logically and clearly. Names can also be changed in the Project panel by double-clicking the name there.

Opacity
WIndows are on top of Outputs and lie over any full-channel media. Change their opacity here.

Fullscreen
A toggle switch, to make this Window fit and fill the Canvas. Shift+F also has this function, which is on the Window's Move toolbar menu.

Canvas position
X and Y are positions in the Canvas. Set Height and Width. The input (media feed) will be scaled to this size. Unlike Canvas Outputs, nothing forces the size of a Canvas Window. Like position, resizing can be performed dynamically and by [External Control](115).

## Create and Recall Canvas Arrangements

How can you quickly rearrange your canvas setup? For many venues, there will be a requirement for a number of flat-screen scenarios for different times of days or events, or just to reset at the end of a session. It is, of course, quite possible to create multiple Compere projects, but setting up, closing and reopening these would be a slow and cumbersome way of managing the total assets of a venue. In fact we don't actually *need* to replace or rebuild everything.

Presets are designed to capture the arrangements and settings for components in the same project. When recalled, dimensions and locations etc. of existing canvas objects will be set again to those of the saved preset.

Example of use

1: Assemble all canvas elements

In this example we create a display with 6 × UHD outputs, and video channels running to 15 canvas windows. So to begin, the canvas [78] is added to the project, followed by the outputs [80], sized and positioned as described above:
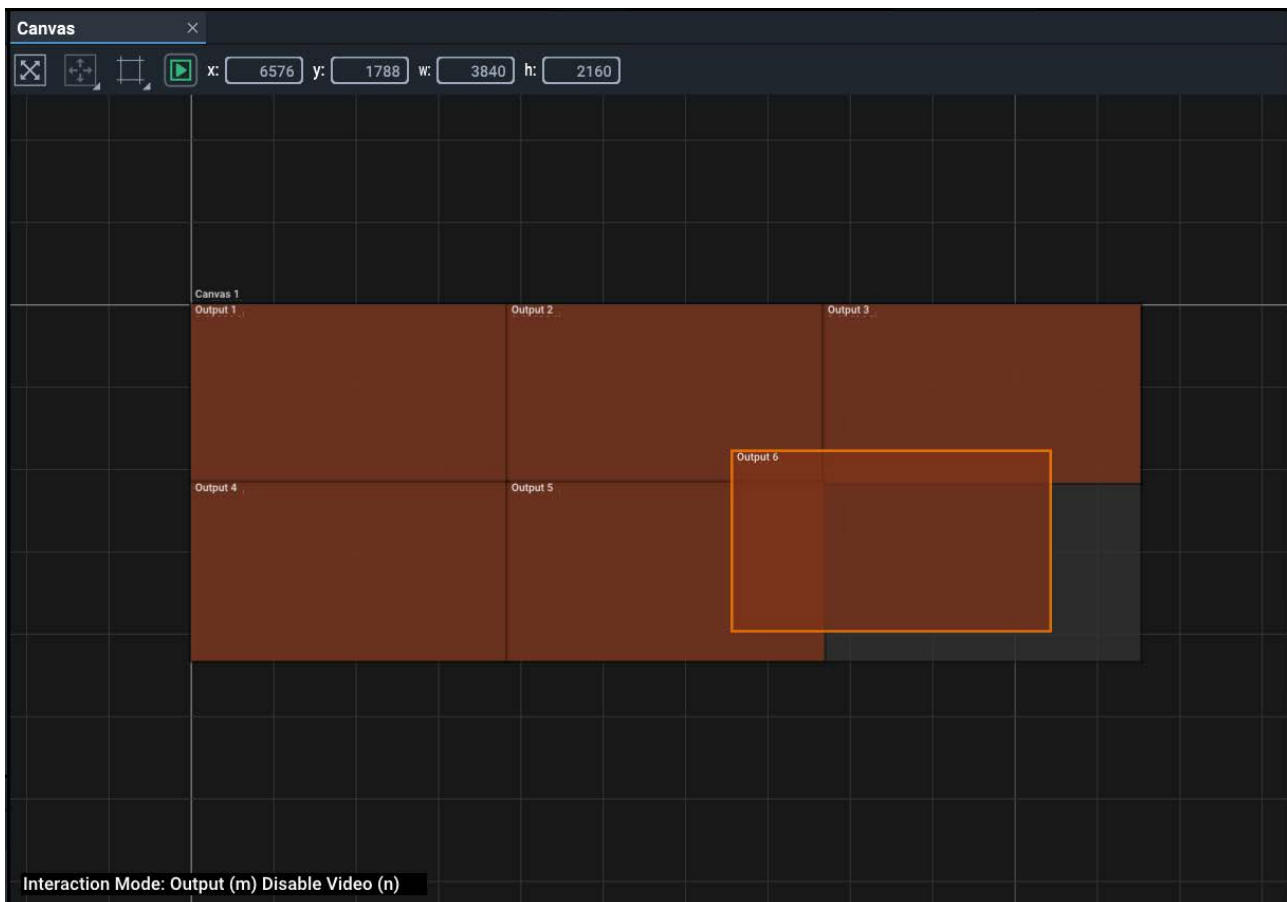
Into this, all the windows [84] can be added and given dimensions. Once released within the canvas bounds, windows can be moved off into the space outside the canvas bounds, where they will not appear on the actual display. Only those required for a particular arrangement are placed inside the canvas:
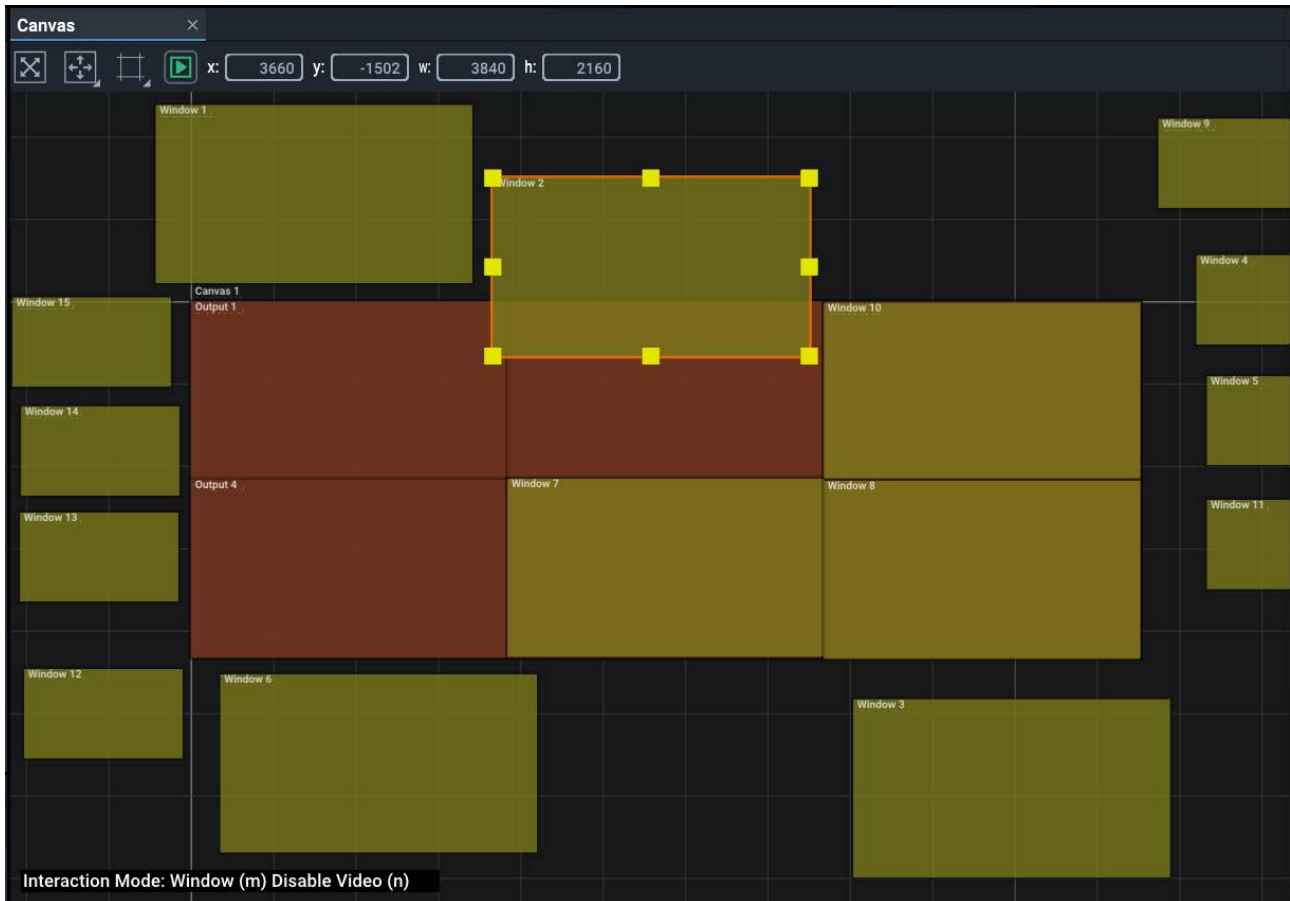

2: Create window arrangement Presets

Once set up, the arrangement can be captured as a Preset. It does not matter where you park windows that are not required for an arrangement. However, with many windows it can be easy to lose track of them.

> While you are assembling different arrangements, keep associated windows together. NDI feeds may provide sufficient identifiers, but clear naming of windows is important. For example, where a group of windows forms a 'side panel', or a position is used as a 'news feed'.

The Preset will register the positions of windows off the canvas as well as those that are on. Once set up, Presets are recalled by either menu selection, or a single external control command.
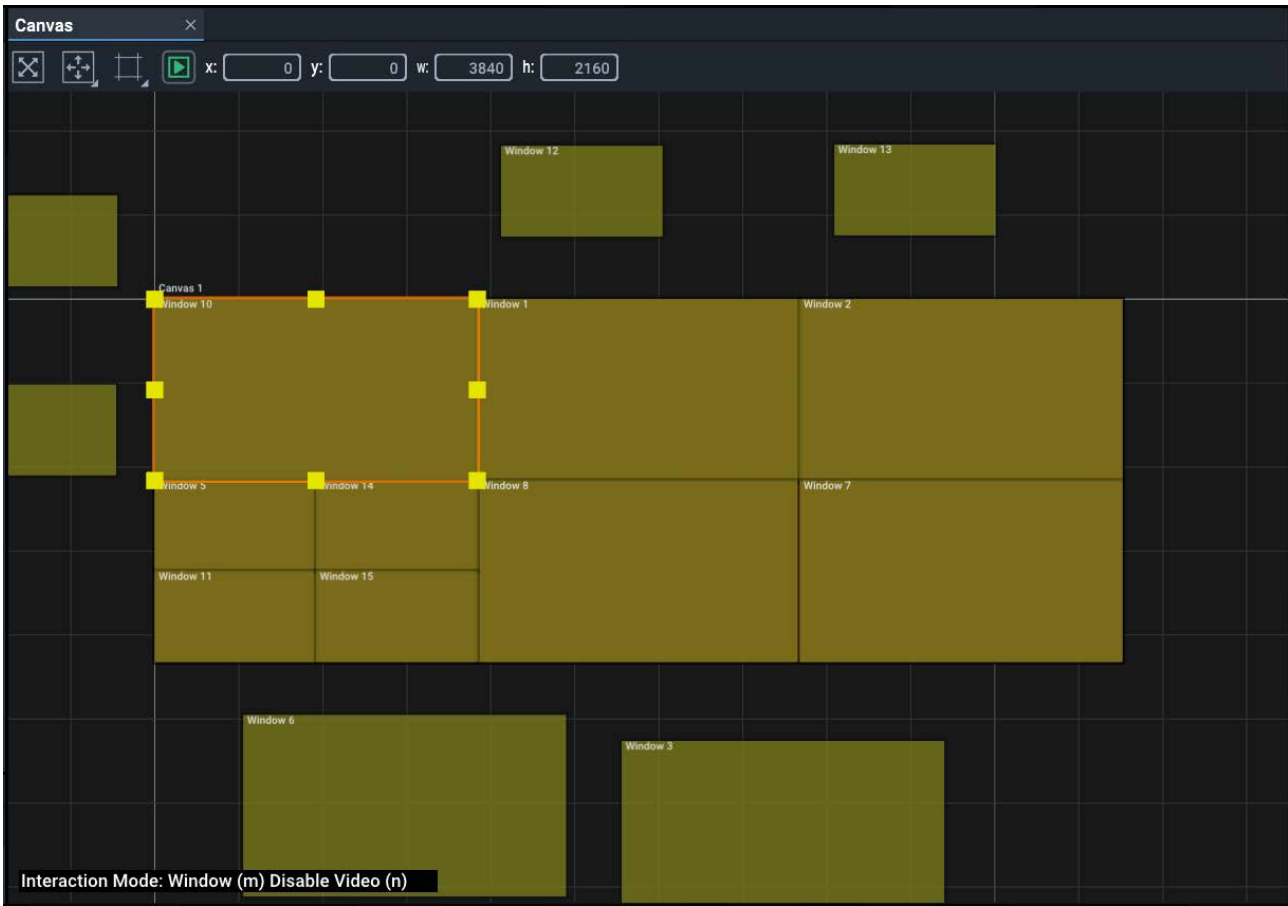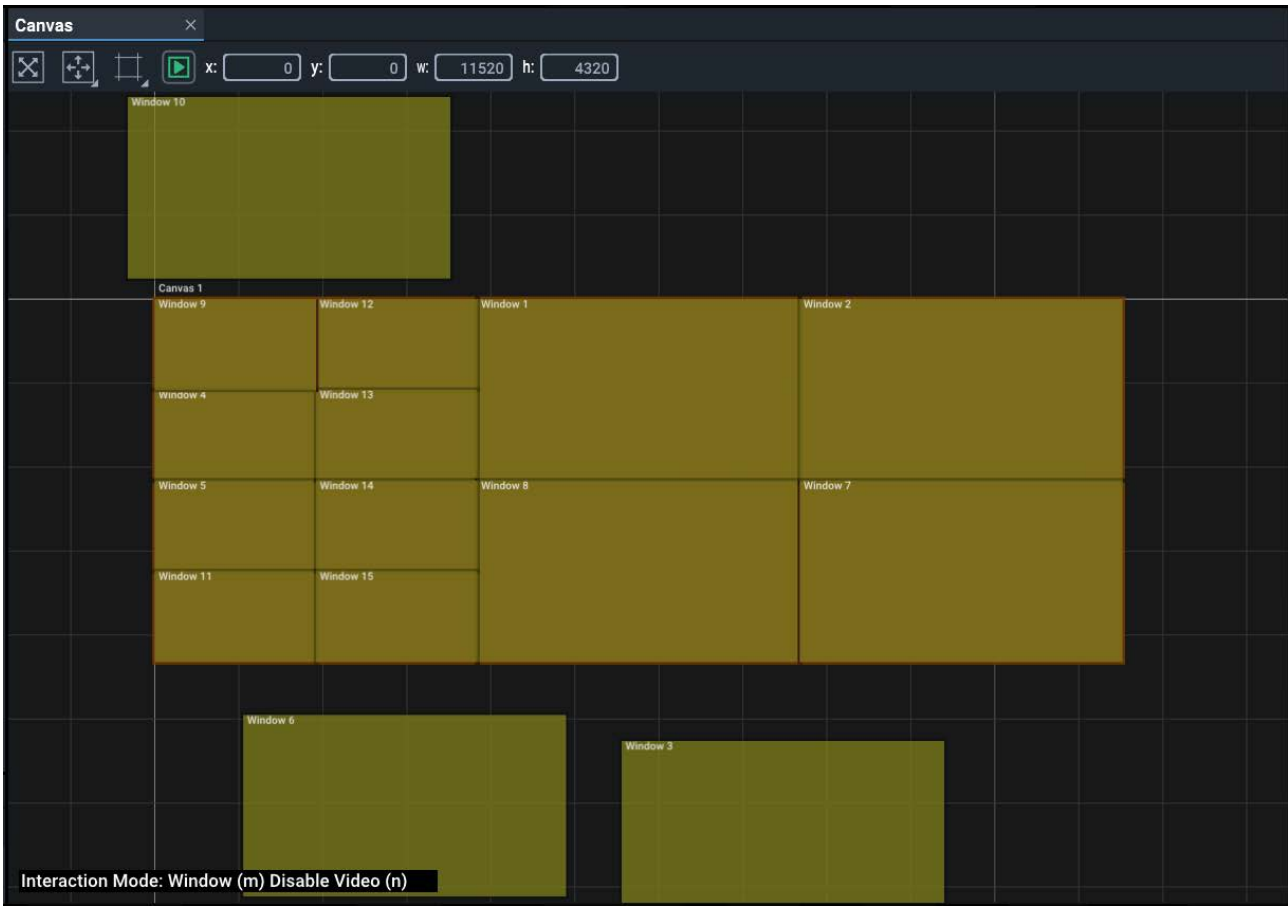
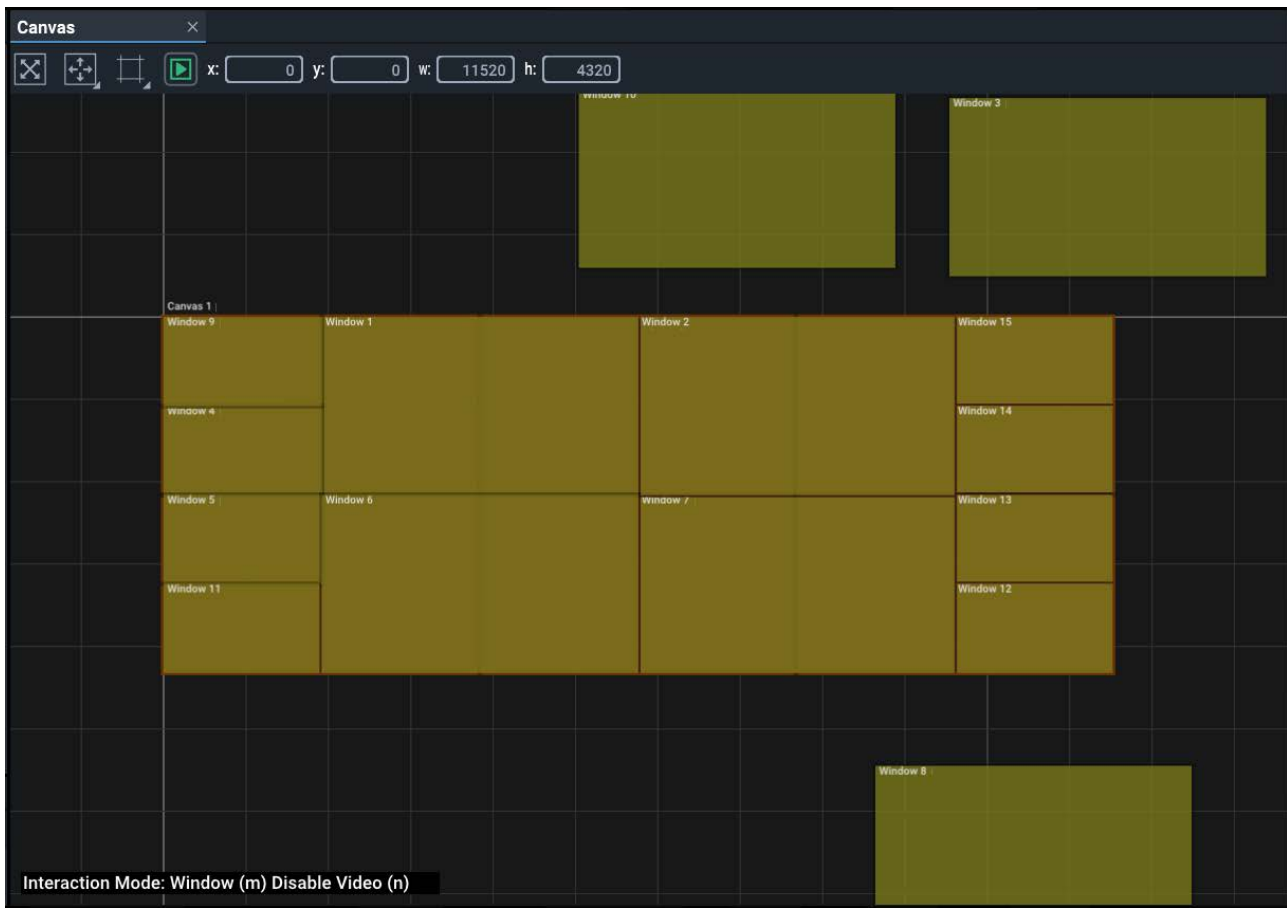Select *File > Presets > Save As > Canvases*
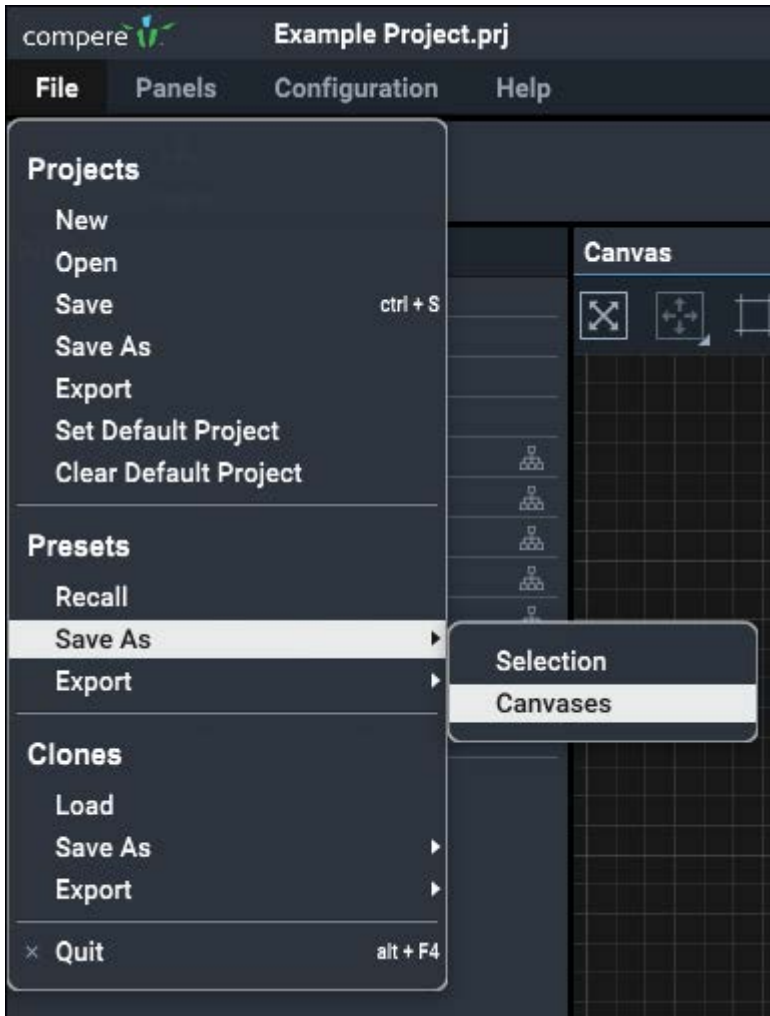
The default file location for Presets is set in *Configuration > Preferences*. Name the preset.

Now rearrange the windows again as required for the next Preset:

Save this as another Preset with a new name. Create further arrangements as required.

3: Recall Presets

From this example, the only changes will be the window positions and/or sizes.

Method 1: From *File > Presets > Recall*, select and apply the arrangement required. As the values for each window are inserted, they will be repositioned in the Canvas panel and the actual display.

Method 2: Send an External Control [115] command to recall the Preset. A single command will select the Preset file and apply all its values. Example:

<command cmd="recallpreset" file="C:\[Compere Root Path]\Presets\[file].pre"**/>**

**Point of comparison: Clones**

There are two ways to capture parts of a project setup such as a display configuration: Clones and Presets. Unlike Presets, Clones are designed to create reusable parts for other configurations or projects. When loaded into a project they are instead added as new branches of the project. *Example*: a clone would be useful for multiple LED walls of 6 × UHD panels, or very different projects using the same display.

DIsplay Setup via MPCDI

The non-Windowing (Warp/Blend) version of Juggler firmware instead allows for the placement of MPCDI data to define media location, warp and blend for a projection system.

In File Explorer, browse to the MPCDI file (*.mpcdi format) and drag it into the upper Canvas assembly panel. It may make take a few moments to ingest the data, depending on its complexity, before a complete populated canvas appears with the data.

The MPCDI name will now appear as a branch in the Project panel.
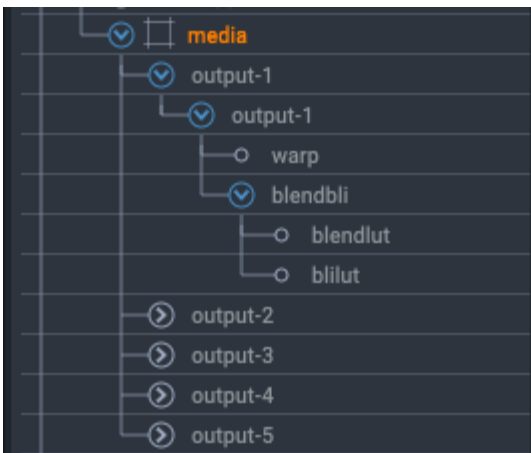
All properties appear on the right as in the windowing setup, but since these are from the MPCDI file, there is no user interaction as when a canvas is assembled from scratch in Compere. All the required canvas outputs will be represented, and connections to outputs are made in the Nodes panel.

The MPCDI information is directed to the project group server, which distributes it to reside on every Juggler in the group.

Nodes: Connecting Juggler Systems

The Nodes assembly panel is where representations of Jugglers and other parts of the system can be connected: to each other, to outputs, to other devices, and to processes. Here you decide how the receive (Rx) ports of Jugglers – whichever type is present – accept media sources, transform them, and put them out to display via the transmit (Tx) ports – whichever type is present – or elsewhere in the system via the system bus.

➢ See also How Juggler is Seen in Compere [62]

Adding a Juggler node

When Jugglers are added into the network and assigned to Network Project Groups, they are recognised and added into the Project running for that Network Project Group, becoming visible in the Project panel.

- Drag a Juggler from the Project panel into the Nodes assembly panel. If necessary expand the Nodes assembly panel and press V to view all, or zoom in to reveal more detail and text. Drag further Jugglers in, and position them on the grid as best fits your purpose.
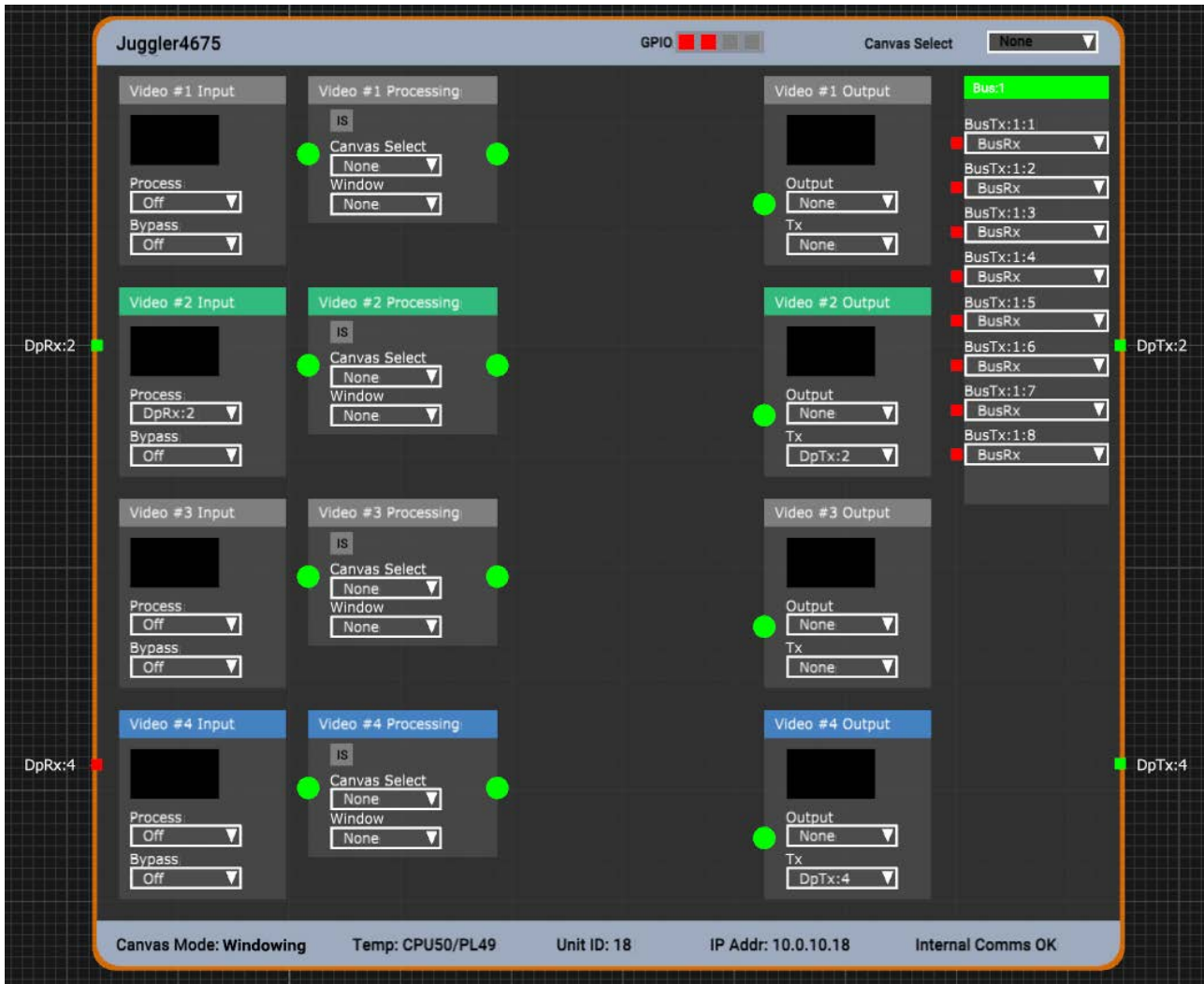
Title bar

Identifies the Juggler name and allows the Canvas to be selected for all video channels in the Juggler.

GPIO

These four indicators correspond to the GPIO HD-BNC ports on the back panel 1-4 left to right.
Grey: not enabled (in Juggler properties)
Red: enabled but no signal present
Green: enabled and connected

Node status bar information

Canvas Mode

Reflects the type of Juggler firmware being used: Windowing, or Warp and Blend (MPCDI).

Temperature

> For internal monitoring of the Juggler.

Unit ID

> Last 2 digits of the Juggler module's IP address, to provide an identifier within the Project Group.

IP Address

> Full IP address of the Juggler module.

Internal Comms

> Indicates the status of communication within the Juggler module.

The four Juggler channels

Every Juggler has four channels of processing: 4 × Rx, 4 × Tx. Available processing channels are arranged left to right and coloured; unavailable channels (e.g. being utilised for the bus) are grey. Here are the essential parts of channel 2:

On the left is the square green Juggler module input pin, here DisplayPort #2 (DpRx : 2). It is showing green because a valid media feed has been detected on it.

Hint: Click any input or output pin to show its properties in the Properties panel.

Video Input

This block is used to connect the video channel to an input (shown here with DpRx : 2 selected). This has connected it automatically to the Video Processing block, adding the green connecting link. You might equally want to take a media stream from the bus as an input for this channel to process.

Process: this selector chooses the source for this channel to process.

Bypass: media can, if required, bypass all processing and go straight to the same-numbered channel output. This direct routing reduces bandwidth if no processing is required. A warning is given if the resolutions do not match.

Video Processing

Canvas Select: select a named Canvas within which this channel will be displayed. The list is populated from Canvases added and named in the Canvas assembly panel. This can be set from the title bar for all available video channels per Juggler.

Window: select a Window of the selected Canvas, in which the channel media will be displayed (if required). The list is populated when the Canvas Windows are added and named.

Video Output

Output: select a Canvas Output for this channel. The list is populated when the Canvas Outputs are added and named.

Tx: select the port out from Juggler module to a display (square pin on the right-hand side of the Juggler node in Compere).

> **Note** that when **Tx** here is assigned here to a Juggler output, the EDID of that output port will pass to the Canvas Output selected in the **Output** selector. The Canvas Output will adopt size and resolution of the EDID. This is a quick way of setting the properties of a Canvas Output, which can then be positioned in the Canvas panel.

See also:

➢ Media connectors: <u>Connecting Inputs and Outputs</u> [108]

Colours, pins and links

Channel headers, pins and links are grey if unavailable (for example channels dedicated in a Juggler module to a bus). Available channel headers will be coloured when available. Pins and links will be red if disconnected, and green if connected.

The small square pins on the left and right edges of a Juggler in Compere, are labelled as channel inputs (Rx, left) and outputs (Tx, right).

The larger round pins inside a Juggler in Compere represent connection points for processing media through a channel.

Links are drawn between pins to show these connections, and naturally can become difficult to trace and distinguish with many units. The Nodes toolbar button allows some clarification:
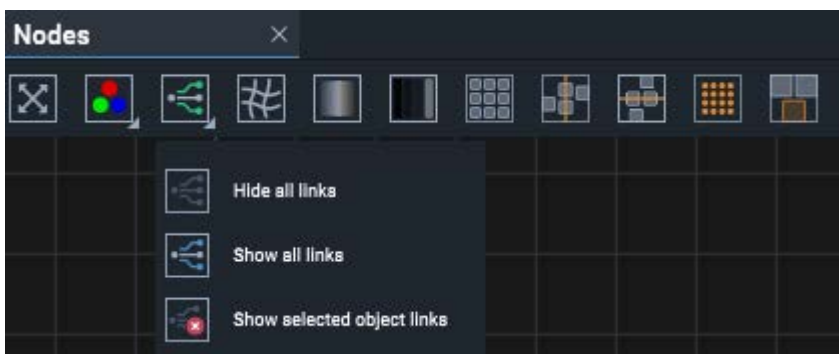
'Show selected object links', for example, will show links to, from and inside a selected Juggler only.

The square pins in the Bus block (upper right of each Juggler node) are the bus connections. These pins will show green when valid media is available on the bus lane.

Multiple selection

Whilst properties cannot be displayed for more than one object, you may want to move or arrange multiple Jugglers. Select a node by double-clicking on it, or click on its name in the Project panel. To select several nodes at once use Ctrl+select.

Nodes panel toolbar
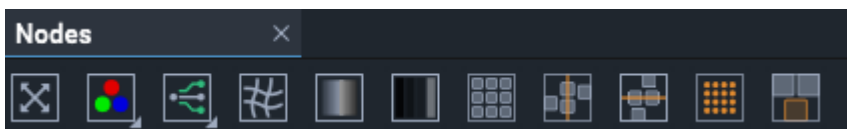
	Bring all objects into view (shortcut key: 'V').
	Select colour beams on or off for all outputs, all Jugglers in the Project. The context menu (right-click on a Juggler in Nodes) also offers beam controls for the outputs of the specific Juggler.
	Display links between nodes selectively.
	Enable/disable warp.
	Enable/disable blends.
	Enable/disable black-level offset.
	Arrange selected nodes (e.g. Jugglers) in a grid pattern.
	Arrange selected nodes in a single column.
	Arrange selected nodes in a single row.

Show a miniature map of all nodes and the position of the current view.

Make the position of a selected node the viewpoint for any node then selected in the Project panel.
This toggle button is not enabled until one and only one node is selected. When the position of that node is pinned, any node then selected centres all nodes around that point. If more than one node is selected, the viewpoint is disabled again.

The Juggler bus options

On the side of the Juggler node is the bus block. Jugglers can have one of two optical fibre bus options: 4-lane (2 × Rx and 2 × Tx video channels) and 8-lane (4 × Rx and 4 × Tx video channels). The pins on the left show green when the bus pin is connected. This example shows a fully connected 8-lane Juggler. Its green title bar shows it is active. An inactive bus is red.

Wider bus options with 12 and 16 lanes are being developed.

For how the bus connections correspond to the physical cabling, see .

Input / Output Connector Properties

Each Juggler module is delivered with firmware capable of the hardware functionality purchased. The firmware determines how each module will be seen in Compere and can be programmed. The Compere software manages what functionality can be configured, per Juggler module.

➢ For some typical configurations, see: Input and Output Options [17]

Select a Juggler in the Project and expanded its branch to list its individual elements. Each can be selected for its properties. Alternatively, select the Juggler and expand and collapse its parts in the Properties panel.

Channel Input/Output Properties

There are a total of four Channels, numbered 1-4, in every Juggler. For example, two of these may be allocated as an 8-lane bus, and two as HDMI. The number and type of inputs and outputs on each Juggler module is detected and listed in the Juggler's properties in Compere.

- DisplayPort connectors marked DPRx and DPTx, numbered 1 to 4

- HDMI connectors marked HDMIRx and HDMITx, numbered 1 to 4

- SDI connectors marked SDIRx and SDITx, numbered 1 to 4.

- Data bus connectors marked BusRx and BusTx, numbered 1 to 4

Example: DisplayPort Rx and Tx properties

DpRx:2	
Type	videointerface
Name	DpRx:2
Enabled	☑
Signal Name	
Hardware Id	0
Genlock Mode	▼
Genlock Ratio	▼
Edid	UHD60.BIN ▼
NDI source name	
Is Display Port MST Active	☐
Force Hot Plug Detect	☐
Enable Red Beams	☑
Enable Green Beams	☑
Enable Blue Beams	☑
Identify	☐
Pixel Format	▼
Pixel Bpc	▼

VIDEOTIMING

Name	VideoTiming
Enabled	☑
ACTIVEPIXELS	3840
ACTIVELINES	2369
TOTALPIXELS	3920
TOTALLINES	2399
HSYNCWIDTH	16
VSYNCWIDTH	5
HACTIVESTART	56
VACTIVESTART	27
PIXELCLOCKMHZ	564.26
BITSPERPIXEL	8
COLOURSPACE	RGB-4:4:4
INTERLACED	☐
STEREO3D	☐
VRATE	60.0

The combination of Name, Hardware Id, and Channel Identify the Connector within the Juggler. Use the Identify checkbox to see in in the media display.

Name

Type, Tx or Rx and number (1-4).

Hardware Id

The internal recognition identifier in the respective Juggler. Be sure to understand the correlation with the physical interface panel identification.

Connectors in Compere project are identified by type and number, as follows:

DisplayPort: base 000 (so numbered 001, 002, 003, 004)
SDI: base 100 (so numbered 101, 102, 103, 104)
HDMI: base 200 (so numbered 201, 202, 203, 204)

Genlock Mode (Tx only)

Internal mode for genlock: the pin proving the genlock signal (see: GPIO Genlock Example [21]).

Genlock Ratio (Tx only)

This is ratio of input to output frame rate:
1:1 For normal operation (e.g. 60 fps in to 60 fps out)
1:2 (e.g. 60 fps in to 120 fps out) for passive to active stereo
0 = 1:1, 1 = 1:2, 2 = 2:1

EDID

Select the EDID from the available list (see EDID Spoofing [110]).

NDI Source Name

If NDI confidence monitoring is enabled, the name of the source.

Is Display Port MST Active

Relates to DisplayPort Tx: check if using DisplayPort multi stream transport on this AV connection.

Force Hot Plug Detect

For DisplayPort and HDMI connectors: bring a connection back to life if lost, e.g. a monitor or a capture source.

Enable Beams Red / Green / Blue

Enable / disable colour beams on this connector (can help in identification).

Identify

Check this to throw a sequence of RGB colours in turn onto an input or an output to identify it.

Pixel Format

Select from: RGB-4:4:4, YCbCr-4:4:4, YCbCr-4:2:2

Pixel BPC

Select 8 or 10-bit

Videotiming

This part of the properties relates to the EDID being used on this input or output. It is not editable.

Bus Connections in a Juggler System

The Juggler bus can be configured in two ways: as a 4-lane 'half' bus via SFP modules, or a an 8-lane 'full' bus with MPO connectors (see: Connect a Juggler Data Bus ²⁶). In Compere this makes no difference to configuration, and the bus is automatically detected.

A further option is a high-bandwidth configuration of the 8-lane bus, in which lanes 4 and 8 share their capacity with the other six.

Juggler bus bandwidth is UHD (3840 × 2160) at 60fps 10bpc 444 capable (or equivalent bandwidth) per bus lane. The high-bandwidth option (6-lane) is capable of 4K.

A juggler System is a group of 2 or more Jugglers on a data bus. In Compere, any input can be sent onto the bus and redirected to another Juggler module to be picked up there as an input, to process and direct onto a display. This configuration is stored in the Project file.

The Bus connection block is represented inside the Juggler in the Nodes panel of Compere.

➤ See also: Physical bus connectors, Connecting a System in Compere ¹⁰³

8-Lane Bus

24 or 30 bpp − (30 bpp limited to UHD at 60 fps or equivalent)

There will typically be only one bus in a Juggler, but two are possible. The number of pins in the bus are those available lanes on the installed bus. Any Juggler can then place up to four video channels on the bus, and receive up to four as inputs. See in the diagram below how Jugglers interconnect, and the bidirectional flow of the bus itself, indicated by the pointers of the green pins.
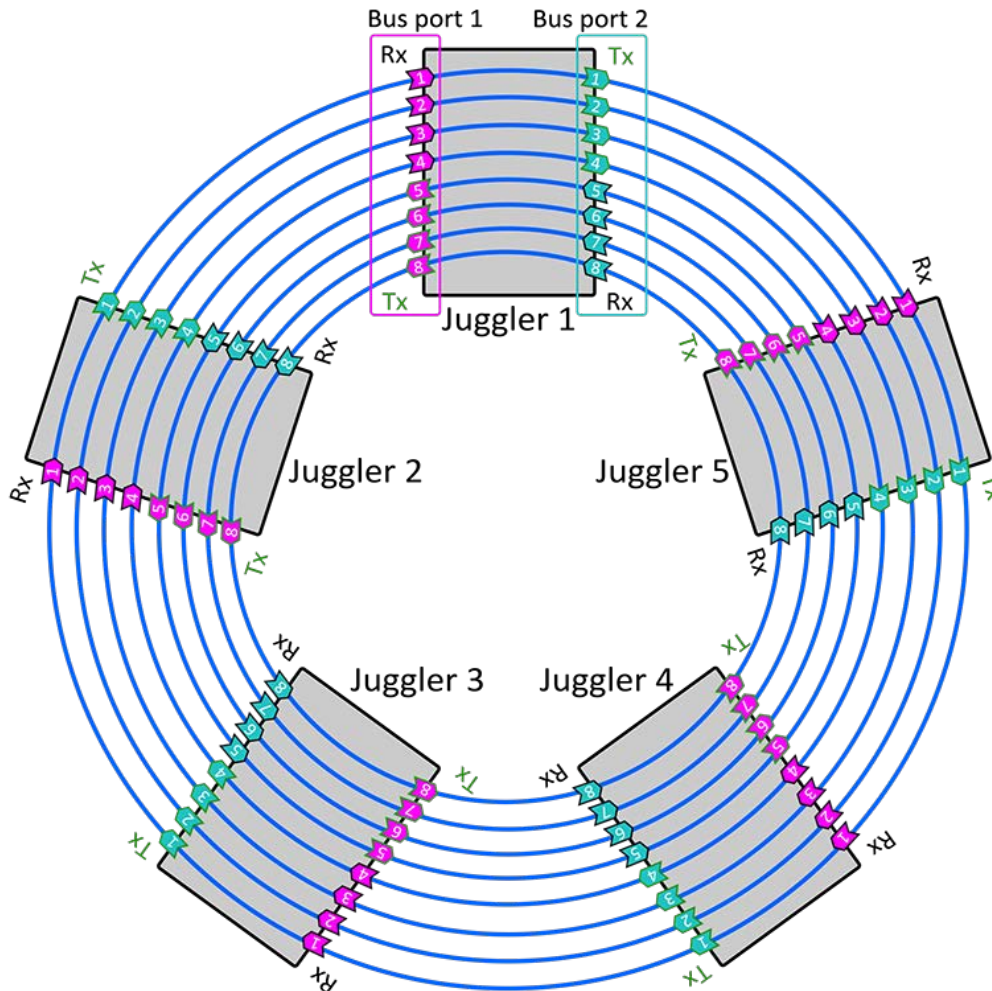
Any Juggler can therefore place media onto the bus, from where any other(s) can stream it off as an input.

- The red/green Rx marker on the left of the Bus nodes indicate whether a bus input exists (green).

- The drop-down list indicates what this bus node is to send out. This could be nothing (Off), a video channel, or whatever is being received on the bus from the previous Juggler in line (Rx in the list).

The drop-down selector in the Compere Juggler Bus, offers 5 options:

BusRx

Selecting 'BusRx' here will simply pass on this Juggler's received bus, so Juggler 1 BusRx:1:1 will connect internally to Juggler 1 BusTx:1:1 and be passed on to Juggler 2 BusRx:1:1 and so on, as in the diagram above.

Video Channels

The 4 video channels listed are the media channels of this Juggler. By selecting an available channel that has an input, this input will be placed on the bus and can be streamed off as an input anywhere else along the bus.

OtherBusRx

A single Juggler can have two 8-lane bus interfaces. One example of use would be to link two racks of Jugglers in an input failover (IntelligentSource) configuration.

Example:

For Juggler 1 in the rack, select an input source for its Video #2 Input. Now, in the Bus block, select Video Channel 2 for BusTx:1:1. On Juggler 2 (next on the rack bus), the pin for BusTx:1:1 will change to green to show a valid media source is now present on that pin.

This media is now available on this lane of the data bus, from where the channel can be selected as an input by another Juggler to process.

Example continued: On Juggler 2, select BusTx:1:1 as the input to Video #4 Input. Media from Juggler 1 channel #2, will now pass to Juggler 2 channel #4, from where it can go to a Canvas, an Output, and a Window. If NDI is enabled, the media will show present on this video channel in the Nodes panel.

Connect the bus as required for your data. You do not need to make unnecessary connections.
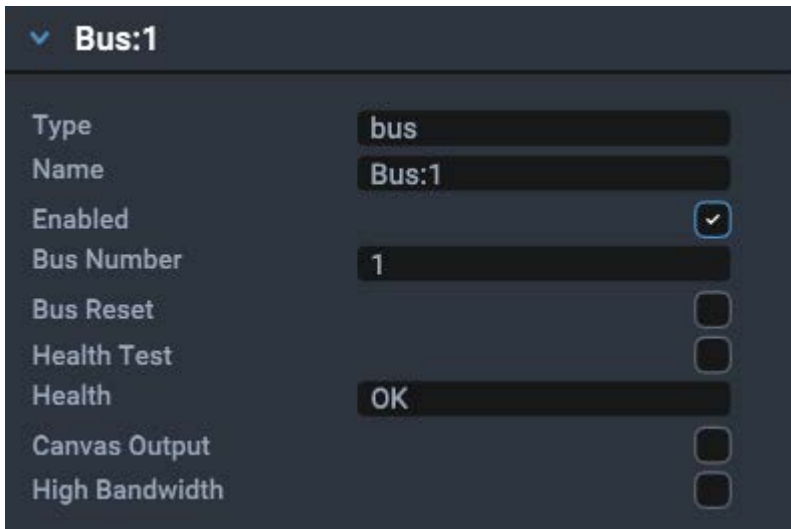
4-Lane Bus

This option has half the capacity of the 8-Lane bus, operating through Juggler SFP modules. The arrangement is similar to the 8-Lane diagram, but can be imagined as pins 1 to 4 only.

High Bandwidth Bus

To increase bandwidth, lanes 4 and 8 can share their capacity with lanes 1-3 and 5-7. Effectively this becomes a 6-Lane bus, with lanes 4 and 8 unavailable to separate video channels. This is enabled in the main bus properties (see below).

Bus Properties

Bus properties can be shown by selecting the Bus from a Juggler in the Project panel, or by expanding the Bus section from all properties of selected Juggler in the Properties panel. These are the common properties for the bus as a whole:

Name: If you wish, for clarity with your own naming convention, rename your bus pins clearly and consistently throughout.

Bus Number: For most systems there will only be one bus.

Health: shows the status per bus lane, or 'OK' when all well. Health can also be seen in general in the green or red colour of the Bus block in the Juggler Node.

Canvas Output
Enable output (rather than video channel) busing mode.

Bus Rx and TX pins

Below this are the properties for each Rx and Tx bus pin available on the Juggler:

Name:

BusRx 1:1 indicates bus 1, receiving lane 1.

Signal Name

NDI source name.

Hardware Id

The internal recognition identifier in the respective Juggler. Be sure to understand the correlation with the physical interface panel identification.

Connectors in Compere project are identified by type and number:
Bus: base 300 (so numbered 301, 302, 303, 304)

Is Display Port MST Active

Check this if using DisplayPort multi stream transport on this AV connection.

Rx Unit Id

The 2-digit Id of the Juggler providing the connection

Rx Bus Id

The number of the Bus being received from (normally there is only one but two are possible)

Rx Lane Id

The number of the Bus lane being used to receive.

Tx Bus Id

> The number of the Bus being transmitted on (normally there is only one but two are possible)

Tx Lane Id

> The number of the Bus lane being used to transmit.

Tx Src

> Video channel from this Juggler, added to this bus pin. The selector is populated the same as the selector in the Juggler node of the Nodes panel.

Video Timing

This part of the properties relates to the EDID being used on this input or output. For information; it is not editable.

Connecting Inputs and Outputs

In a windowing Juggler system, you will have set up your <u>display Canvas</u> ⁷⁸, with a number of Canvas Outputs and Canvas Windows. The Outputs can be named or numbered to identify the area of Canvas on which you may want to display media. Any number of Windows can be positioned and scaled dynamically anywhere on the Canvas, across the boundaries of the Canvas Outputs, to which media streams can be assigned.

Throughput for a single Juggler

In the top bar of the Juggler, select the Canvas name that this Juggler will address. This list is populated when Canvases are created and named. Selecting here will populate 'Canvas Select' in all available video channels for the Juggler.

Juggler input pins, on the left edge of the Juggler, are named by type, e.g. **DpRx : 2**, meaning a DisplayPort input on channel 2 of the Juggler. If no valid media are present on that physical port, it will be red. If valid media is present, it will be green.

From the list of available feeds in the **Video #2 Input** 'Process' selector, add in input. The Process block will now be linked to the **Video #2 Processing** block. If valid media is present, pins and links will be green. If you have 'Enable NDI Confidence Monitors' selected in the Properties of each Juggler, this will help verify that a media stream is connecting correctly. NDI monitors can be seen in the video channel input and output blocks of a Juggler in Compere, and in the Canvas assembly panel.
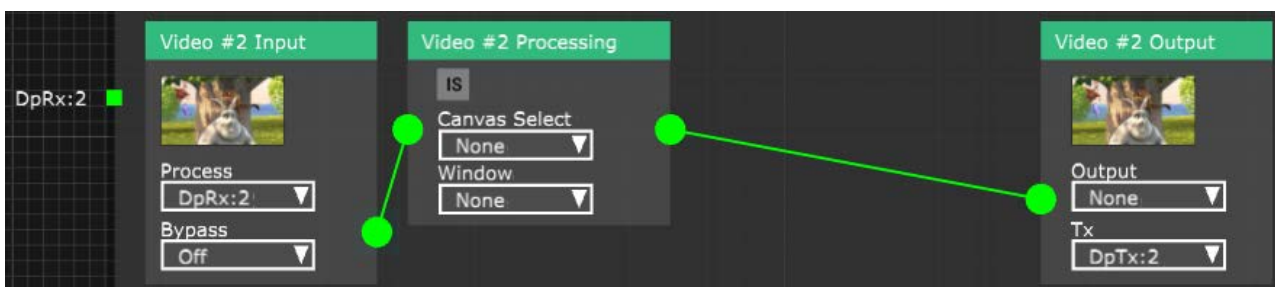
In the Processing block, you decide where this is to go next. There are other processes you can add, but for now, just ensure you have the canvas you want to use.

In the Video #2 Output block for this channel, ensure the **Tx** has the appropriate destination port selected. Using the Output selector, choose which Canvas Output area is required. The lists is populated as Canvases are added to the Canvas assembly panel. This will remove the direct link to the Output block because in order to display the media we need to say where on the overall Canvas it is to appear, i.e. in a Canvas Window. You can use a Juggler test pattern as a background of the output when media is not present:


Reminder: You can identify a Juggler output in the real world by checking the Identify box in the Juggler output properties:
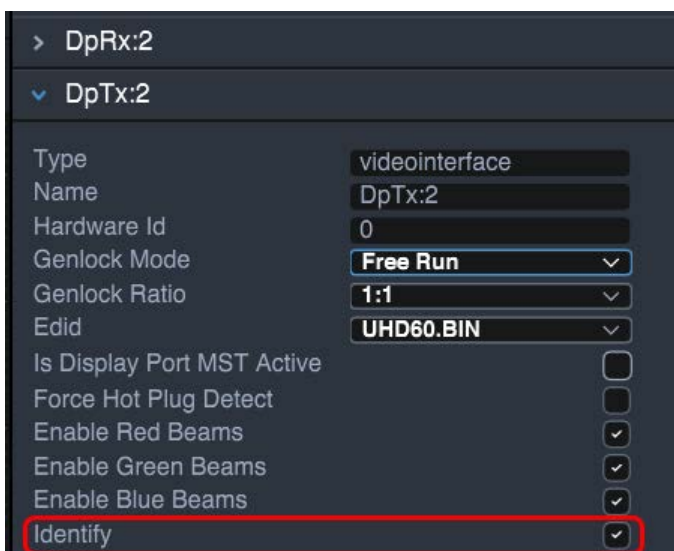
This checkbox is 'push-to-show', and will temporarily cause the output device to flash red and then green.

Now assign the video channel to a <u>Canvas Window</u> [84] in the Video Processing block. The selector list is populated by the addition of Windows in the Canvas assembly panel. Canvas windows can be positioned anywhere in the Canvas, so what you see on the output display will require this Window to be positioned somewhere over the selected Canvas Output for this video channel, as here:
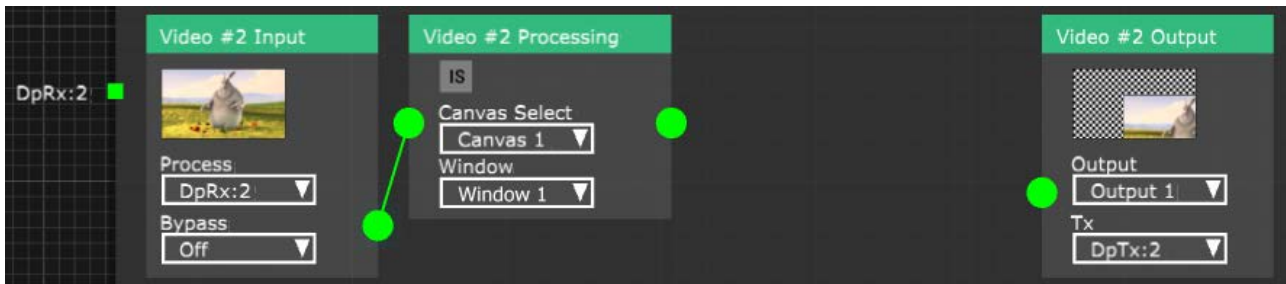

EDID Spoofing

Juggler supports spoofing of EDIDs onto DisplayPort and HDMI inputs and outputs via the 7thSense Compere software.

Each Juggler comes with a list of standard EDIDs. Additional EDIDs can be loaded into the Juggler Module(s) via the Compere interface.

For each Juggler output in the Nodes assembly panel (described 'AVout#' per Juggler) select the appropriate EDID from The Properties panel > edid list:
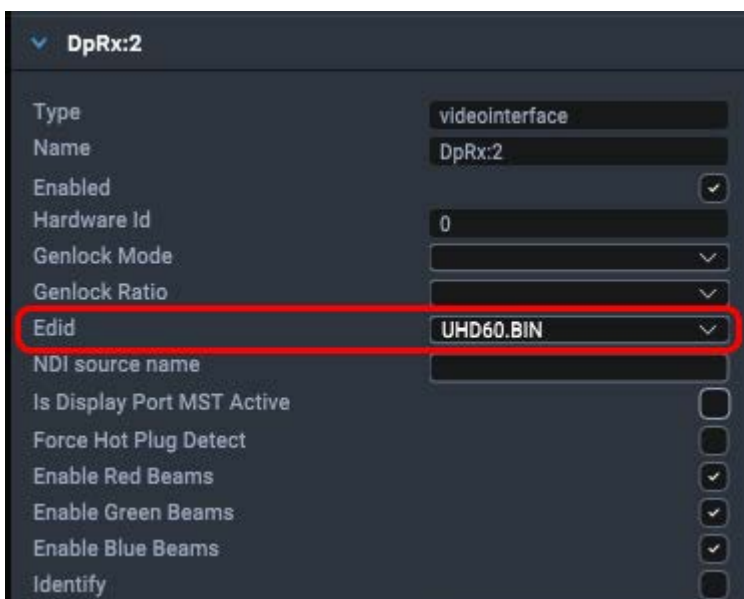
When an Input or Output is assigned to a Juggler video connector, it adopts the EDID of that connector.

3D Passive to Active Conversion

To convert a 3D passive input to 3D active, drag the Passive to Active Node from the Resources panel onto the required Juggler:
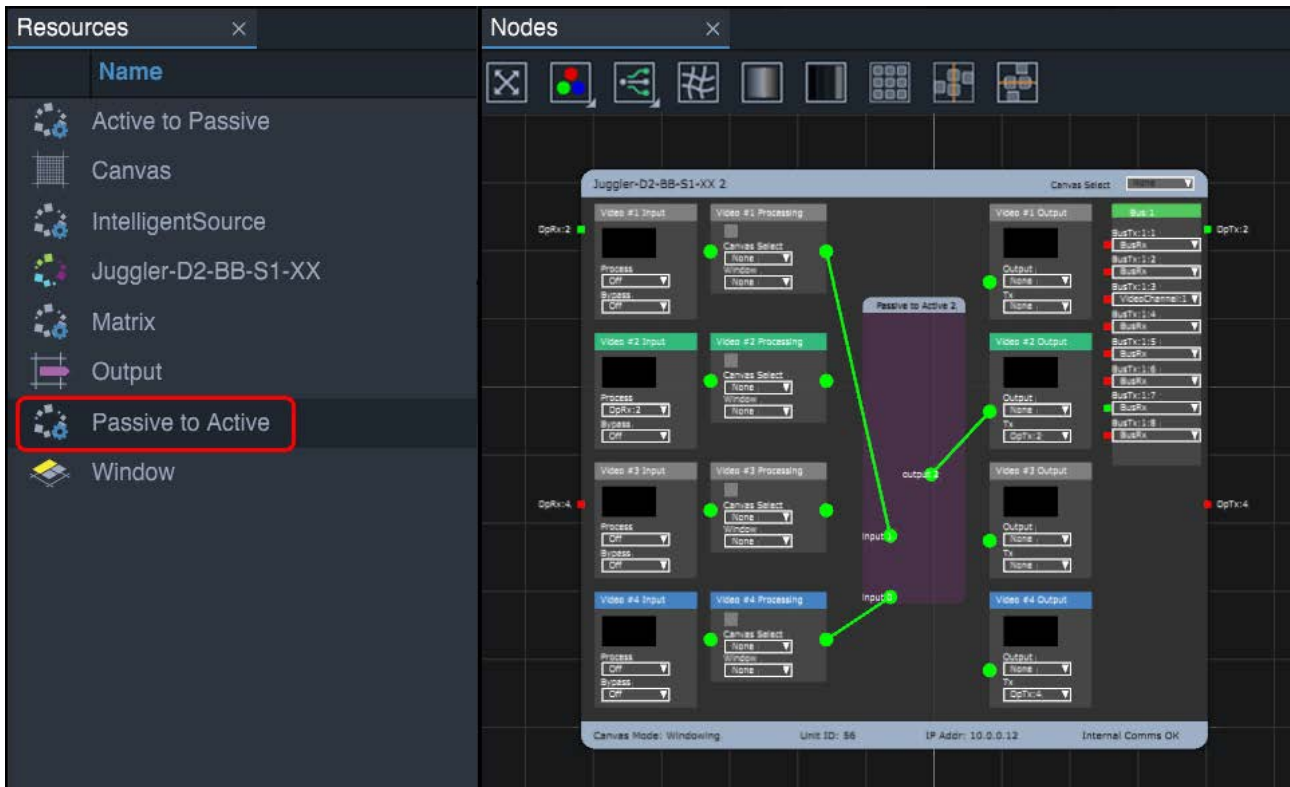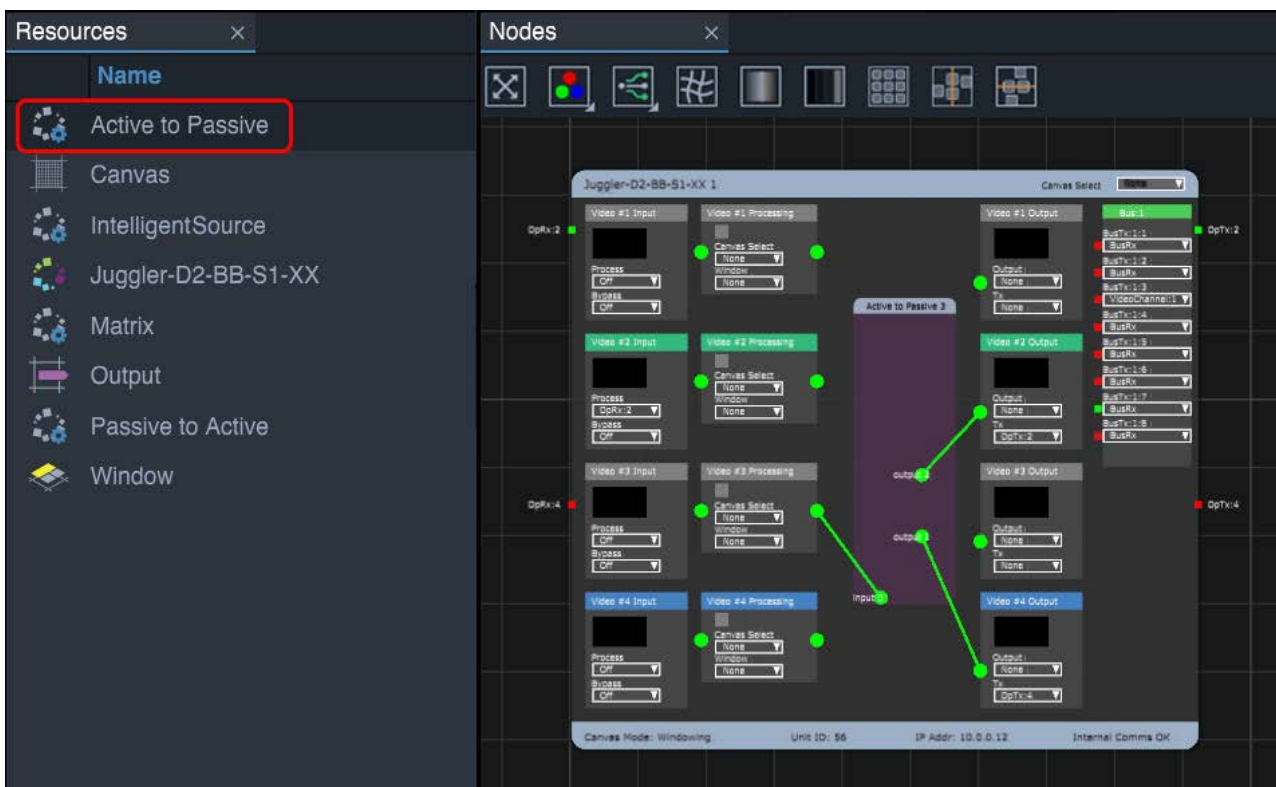

The resource is added to the Juggler in the Project tree, from where it can also be renamed or deleted. Connect the appropriate processing input pins and the required video output pin.

This node has the effect of directing two active frame channels to a single interleaved active 3D channel with an associated 3D sync output signal.

3D Active to Passive Conversion

To convert a 3D active input to 3D passive, drag the Active to Passive Node from the Resources panel onto the required Juggler:

The resource is added to the Juggler in the Project tree, from where it can also be renamed or deleted. Connect the appropriate processing input pin and the required video output pins.

This node has the effect of directing interleaved frames to two independent 2D output channels. To use this 3D converter, a 3D sync input is required.
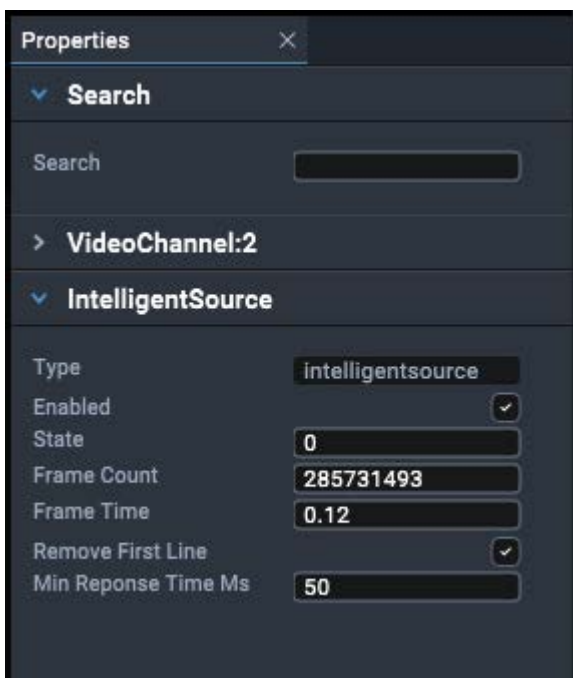
IntelligentSource

7thSense IntelligentSource™ is a method of inserting data into the top line of source frames in order to detect frame sequence breaks and trigger same-frame failover to a backup stream. This can work across Juggler modules to provide either an automatic or manual loop option for source failover. Use of the 7thSense IntelligentSource API in conjunction with this allows both sync and video data monitoring.

IntelligentSource detection is seen in the selected video channel as an 'IS' icon:
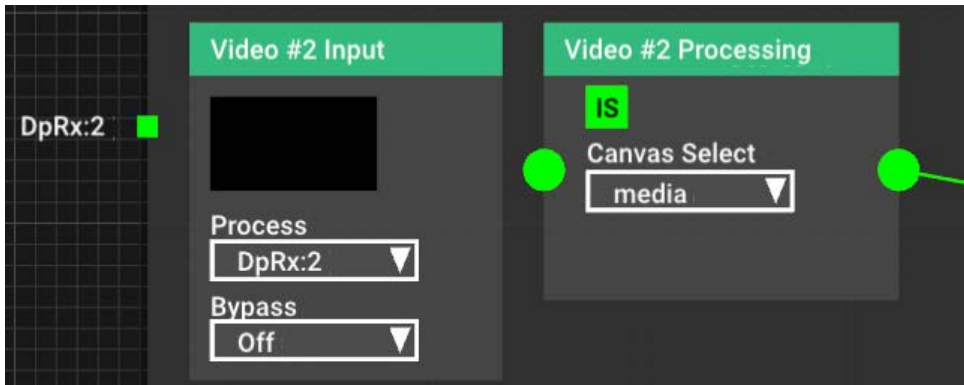
The 'IS' icon will be grey until IntelligentSource is enabled. It is enabled, per required video channels for a Juggler, in the Properties panel for that channel:

- Frame Count and Frame Time will be incrementing when all is well.

- If using a stream with IntelligentSource data, check 'Remove First Line' to strip off the additional row of pixels used to convey the frame data.

The IS icon will be green when enabled and good IntelligentSource data is flowing, and red if enabled but no IntelligentSource data is being received. If IntelligentSource framecount/time data stops incrementing, the IS icon for the channel will be yellow.

Currently, system scripts in Python create the IntelligentSource signal redirect to a backup system, and in the event of signal failure will switch instantly to the second system.

➢ To implement IntelligentSource with Delta 2.7 software, document **M450: IntelligentSource API** is available from 7thSense.

Ports Used by Juggler and Compere

Function	Configurable	Port
Compere UDP	Default, editable via comms.ini or in Preferences	9999
Compere TCP	Default, editable via comms.ini or in Preferences	5555 5553 (early versions of Compere)
Multicast discovery UDP	Fixed	5353
External control comms, default UDP receive port	Yes, in Preferences	8001
External control comms, default UDP transmit port	Yes, in Preferences	8002

It is important to make sure the UDP and TCP ports match between all instance of Compere in a system that need to communicate with one another.

External Control

Principle

Every element of a Compere Project (the branching tree you see in the Project panel) has a unique identifier (UUID) and a common text name that you give to it. The same is true for every parameter that you can see, and may want to change, in the Properties panel for that element.

- Use a **remote terminal client** e.g. PuTTY, or Tera Term, on a raw (or raw ASCII over TCP) connection, to address Compere on the **server** for a project group.

- The **IP of a target Juggler** can be found in *Configuration > Network Configuration >* [Discovery].

- Use **Port 2300**.

Protocol

In the Project Panel you see the structure of the Project file that contains all the instructions for the entire setup of a Juggler system.

From this information you can see that there is a route to the value of everything. In the same way as we use paths to files in folders on devices on a network, you can send commands, to find or change any value. As an example we might want, in rough terms, to do this:

set the value to '10' for the x-axis position of a projector in this project

which as a command would look something like this:

set path="/GroupSet/Group/Projector 1/position/x" value="10" (in plain text)

<command cmd="set" path="/GroupSet/Group/Projector 1/position/x" value="10"**/>** (in xml)

The message format (plain text or xml) is detected and the response is returned in the same format. All recognised commands respond immediately with: success=true/false.

For commands that do not have an instant response (e.g. open, or "Import MPCDI") the command must be followed by a system status request.

These commands or controls can be sent in either plain text or XML to the IP address of the **server Juggler** in a project group, or to a Windows control PC as client *or* server.

Compere supports commands sent via TCP. Use the Compere Preferences Panel to configure IP addresses and transmit/receive ports, including the connection timeout.

Commands and their parameters are case insensitive. Each parameter's actual *value* is case sensitive. Individual commands are activated in text strings by line ending characters \n, or in xml, by opening and closing tags **<command> </command>** or, for empty tags, **<command />**.

Grouping Commands

It can be more efficient to encapsulate many SET commands in a single group using the **<commands></commands>** tag.

For example to position two projectors in a single control message:

```
<commands>
<command cmd="set" path="/GroupSet/Group/proj1/position/x" value="0.2"/>
<command cmd="set" path="/GroupSet/Group/proj1/position/y" value="0.3"/>
<command cmd="set" path="/GroupSet/Group/proj1/position/z" value="0.4"/>
<command cmd="set" path="/GroupSet/Group/proj1/heading/az" value="0.5"/>
<command cmd="set" path="/GroupSet/Group/proj1/heading/el" value="0.6"/>
<command cmd="set" path="/GroupSet/Group/proj1/heading/roll" value="0.7"/>
<command cmd="set" path="/GroupSet/Group/proj2/position/x" value="0.2"/>
<command cmd="set" path="/GroupSet/Group/proj2/position/y" value="0.3"/>
<command cmd="set" path="/GroupSet/Group/proj2/position/z" value="0.4"/>
<command cmd="set" path="/GroupSet/Group/proj2/heading/az" value="0.5"/>
<command cmd="set" path="/GroupSet/Group/proj2/heading/el" value="0.6"/>
<command cmd="set" path="/GroupSet/Group/proj2/heading/roll" value="0.7"/>
</commands>
```

The reply would be:

```
<group success="true" numberOfCommands="12" atomic="true" commandDuration="8111us"/>
```

Note: 'atomic' in this reply indicates that the **commands** group was not to be distributed to the other members of the Compere Project Group until all commands had been received by the instance of Compere addressed.

In this example, projector movement will therefore be smoother. The attribute atomic="true" is implicit, but atomic="false" can be used if you want commands to be distributed one at a time as received, i.e. while other commands are still being received.

Explicit example:

```
<commands atomic="true">
<command cmd="set" path="/GroupSet/Group/proj1/position/x" value="0.2"/>
...
</commands>
```

The plain text equivalent would be like this:

```
group commands="GroupSet/Group/proj1/position/x=0.2;
[...];GroupSet/Group/proj2/heading/roll=0.2"
```

Scope of external control

With complex multi-Juggler systems, External Control is the route to both systems management, and dynamic interaction with all the elements. Whether used directly or through show controllers, it removes the need to use Compere for complex daily interaction with devices, models, projectors, sources, processes and interconnections.

Static and Dynamic commands

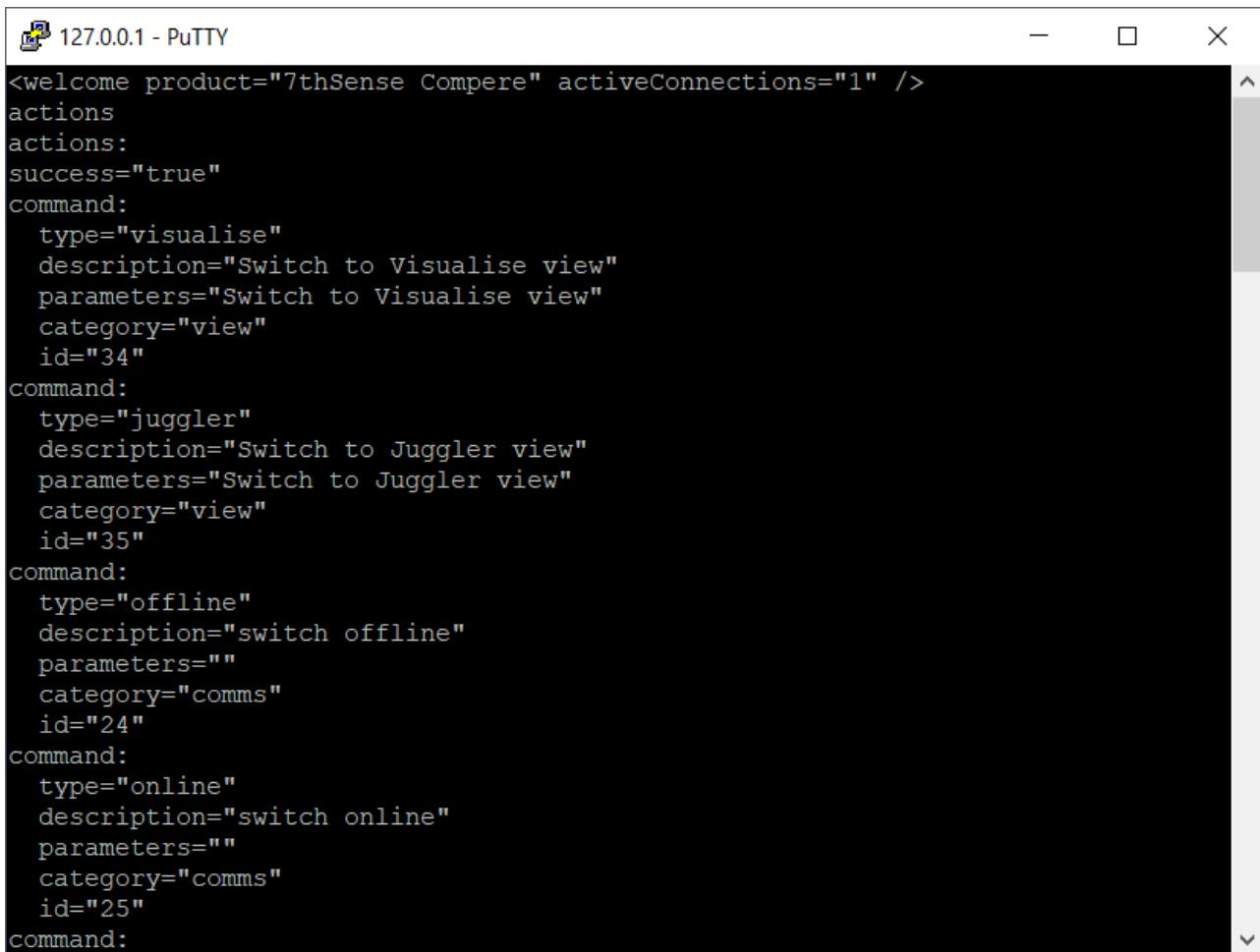There are two types of external control commands, **static** and **dynamic**.

Static, or core, commands interact with a Compere Project, load and save Project files, and request specific notifications (registers). For example to `set` a value of an object property.

➢ The full list of core commands is available within Compere, *Help > External Control Protocol*.

➢ For the latest version, on this Portal, see ExternalControlHelp.html

Dynamic (action) commands are less generic and go beyond changing properties in the Project in order to do something. These are detailed with examples in the following pages of this guide.

Available actions

Type "actions" to list all actions available for the target instance of Compere in its current state. This will return a series of actions of name type= under headings of command:

To use an action:

actions type="[name]" enacts a specific action of this name

Example

Import an MPCDI file from a given location:

actions type="import MPCDI" file="C:\[MyPath]\MPCDI\example.mpcdi"

Real use case example:

actions type="Import MPCDI" file="C:\Users\[UserName]\Downloads\MyFile01.mpcdi"

Options

Any of the commands below could be used to set the x position of a Canvas Window value to 800:

Using names

<command cmd="set" path="/GroupSet/Group/Canvas 1/Window 1/canvasposition/x" value="800"**/>**

Using class names (i.e. non-specific)

The element path="juggler", for example, addresses all objects of type 'juggler'.

<command cmd="set" path= "/GroupSet/Group/juggler"**/>**

Using a mixture of wildcards [119] **(*), names (specific) and class names (non-specific)**

<command cmd="set" path="*/*/*/*/Window 1/canvasposition/x" value="800" **/>**

Wildcards

Matching a name between slashes

Command

<command cmd="get" path="JugglerName/*/health"**/>**

Reply

<replies success="true" path="JugglerName/*/health" numberofreplies="1"
	commandDuration="123us"**>**
<health value="~RxLane:1-LaneDown~RxLane:2-LaneDown~RxLane:3-
	LaneDown~RxLane:4-LaneDown~"
	fullpath="GroupSet/Group/JugglerName/Bus:1/health"**/>**
</replies>

Absolute versus relative paths

Relative : **<command** cmd="get" path="JugglerName/DpRx:2/VideoTiming"**/>**

Absolute: **<command** cmd="get" path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming"**/>**

It is possible to use a relative path, offering a shortcut for an absolute path. Compere will perform an exhaustive search so there is a performance overhead depending on the extent of the Juggler system. Additionally, multiple results may be returned: searching for "name" will return every absolute path that matches.

The full path is returned in replies, and in commands, the leading / (/GroupSet) is preferred.

Command

<command cmd="get" path="health"**/>**

Reply

<replies success="true" path="health" numberofreplies="3" commandDuration="123us"**>**
<health value="~RxLane:1-LaneDown~RxLane:2-LaneDown~RxLane:3-
LaneDown~RxLane:4-LaneDown~"
fullpath="GroupSet/Group/JugglerName1/Bus:1/health"**/>**
<health value="~RxLane:1-LaneDown~RxLane:2-LaneDown~RxLane:3-
LaneDown~RxLane:4-LaneDown~"
fullpath="GroupSet/Group/JugglerName2/Bus:1/health"**/>**
<health value="~RxLane:1-LaneDown~RxLane:2-LaneDown~RxLane:3-
LaneDown~RxLane:4-LaneDown~"
fullpath="GroupSet/Group/JugglerName3/Bus:1/health"**/>**
</replies>

Message Formats

Message formats: core and extended

Messages may be sent in either plain text or XML format – see below for the plain text summary – and Compere will respond in the same format as the incoming message. All XML attributes must be quoted.

Messages are defined by a 'command' tag with various attributes depending on the message. Typical attributes include: cmd, path, type, value, makeactive and fullpath.

The core message set – as described in the HTML API – must have at a minimum a 'cmd' attribute.

<command cmd="status" **/>**

The extended message set is accessible via the 'action' command. The availability of these commands depends on the current system configuration: e.g., adding a Juggler to the system will make additional commands available to the external control user.

Apps that use backslash as an escape character

Where the backslash \ is used as an escape character (e.g. C++, or a showcontroller app) paths to a file must themselves be escaped:
e.g., file="C:\\Images\\foldername".

Example (Windows o/s):

```
<command cmd="actions" type="Import MPCDI" file="C:\Users\[UserName]\Downloads\mpcdi-file-name.mpcdi"/>
```

and in plain text:

```
actions type="Import MPCDI" file="C:\Users\[UserName]\Downloads\mpcdi-file-name.mpcdi"
```

Response format

All valid messages are acknowledged immediately with 'success=true/false'. The XML response is a well-formed XML document which may be parsed by any established third-party library.

Note: the Compere version example below includes a date string and a build hash which can be used for build tracking/reporting.

Request

```
<command cmd="get" path="compereversion" />
```

Response

```
<replies success="true" path="compereversion"  numberofreplies="1"
commandDuration="123us">
<compereversion value="1.0.0.1 (20200924 50473f3)"
fullpath="GroupSet/Group/JugglerName/atom/compereversion"/>
</replies>
```

Replies include the number of replies (relative paths especially are likely to elicit multiple responses) and how long the reply took (commandDuration), in microseconds.

System configuration

The complete system configuration can be read using the 'get' command with an additional 'path' (keyword) attribute.

Get current project and default project

```
<command cmd="status" />
```

Get info for all Jugglers in a project

```
<command cmd="get" path="juggler" />
```

Get info for a specific Juggler

```
<command cmd="get" path="JugglerName" />
```

Get all versions from all Jugglers in a system

```
<command cmd="get" path="fwversion" />
<command cmd="get" path="swversion" />
<command cmd="get" path="compereversion" />
```

Get versions for a specific Juggler

```
<command cmd="get" path="JugglerName/atom/fwversion" />
<command cmd="get" path="JugglerName/atom/swversion" />
<command cmd="get" path="JugglerName/atom/compereversion" />
```

Unsolicited notifications: registers

Typically, commands elicit an immediate response, but an external control user can add callbacks via registers to be notified asynchronously when a parameter changes: e.g., when the status of an interface is updated.

Get all CPU temperatures in the system

```
<command cmd="get" path="temperatureCPU" />
```

Receive a notification when any CPU temperature changes

```
<command cmd="register" path="temperatureCPU" />
```

Plain text messages

For convenience, when typing at a console, all XML messages have a shorter, plain text equivalent. This can be constructed by removing the outer command tag and dropping the quotes on the 'cmd' attribute.

Example: get all IP addresses in the system

XML

```
<command cmd="get" path="externalipaddress" />
```

Plain text

get path="externalipaddress"

Example: MPCDI import

XML

```
<command cmd="actions" type="Import MPCDI" file="C:\\path\\to\\system.mpcdi" />
```

Plain text

```
actions type="Import MPCDI" file="C:\\path\\to\\system.mpcdi"
```

Further examples

Get all temperatures for all Jugglers in a system

```
<command cmd="get" path="temperaturePL" />
<command cmd="get" path="temperatureCPU" />
```

Get CPU temperature for a specific Juggler

```
<command cmd="get" path="/GroupSet/Group/JugglerName/SystemMonitor/temperatureCPU" />
```

Disable red beams system-wide

```
<command cmd="actions" type="Red Beam Off" />
```

Enable blue beams for a specific Juggler

```
<command cmd="actions" type="Blue Beam On" path="/GroupSet/Group/Juggler4540" />
```

Get active pixels and lines for a specific Juggler's video input

```
<command cmd="get" path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming/activepixels" />
<command cmd="get" path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming/activelines" />
```

Heartbeats

A heartbeat could be implemented by periodically requesting 'status' from each Juggler.

System Status

Contains status elements:

- Timestamp of Compere

- Task manager

- Current project

- Default project

- Comms mode (client/server/off)

- Peer data

Command

```
<command cmd="status" />
```

Reply

```
<status success="true" time="Thu Jan 21 15:32:21 2021">
 <taskSummary total="0" failed="0" pending="0" complete="0"
      percentComplete="0.000000"/>
 <projectStatus ProjectPath="C:\Users\[UserName]
      \AppData\Local\compere\Projects\myproject.prj" DefaultProject="C:
      \Users\[UserName]\AppData\Local\compere\Projects\myproject.prj"/>
 <networkInfo commsMode="server"/>
 <peers>
  <peer uniqueId="79833e07d0e64ffe9a874c49e4516b5e" auxData="Host Name=BootCamp-0263E
      (Windows);MemUse=85% DiskFree=57.6GiB CPU idle=239 kernel=18 user=34;Curr Tree=My
      Project (8c5202e4b1174e4ea6055df9279ca500);Project Group: MainGroup;machineID=00-
      60-73-06-86-16;10.100.150.20;10.0.10.207;192.168.0.11;172.29.144.1;172.28.80.1;"
      userData=""/>
 </peers>
</status>
```

Juggler Status

Uptime

Command

```
<command cmd="get" path="JugglerName/juggleruptime"/>
```

Reply

```
<replies success="true" path="JugglerName/juggleruptime" numberofreplies="1"
      commandDuration="123us">
```

```
<juggleruptime value="0d:01h:52m"
        fullpath="GroupSet/Group/JugglerName/juggleruptime"/>
</replies>
```

Bus Status

Get bus status

Command

```
<command cmd="get" path="/GroupSet/GroupJugglerName/Bus:1/health"/>
```

Reply

```
<replies success="true" path="/GroupSet/Group/JugglerName/Bus:1/health"
        numberofreplies="1" commandDuration="123us">
  <health value="~RxLane:1-LaneDown~RxLane:1-CtrlBad~RxLane:2-
        LaneDown~RxLane:3-LaneDown~RxLane:3-CtrlBad~RxLane:4-
        LaneDown~RxLane:4-CtrlBad~RxLane:5-LaneDown~RxLane:6-
        LaneDown~RxLane:7-LaneDown~RxLane:8-LaneDown~"
        fullpath="GroupSet/Group/JugglerName/Bus:1/health"/>
</replies>
```

Bus Reset

Flash the bus

Command

```
<command cmd="set" path="/GroupSet/Group/JugglerName/Bus:1/reset" value="1"/>
```

Reply

```
<set success="true"/>
```

Genlock

Get genlock status

Command

```
<command cmd="get"
        path="/GroupSet/Group/JugglerName/VideoChannel:2/outputgenlocked"/>
```

Reply

```
<replies success="true"
        path="/GroupSet/Group/JugglerName/VideoChannel:2/outputgenlocked"
        numberofreplies="1" commandDuration="123us">
 <outputgenlocked value="0"
        fullpath="GroupSet/Group/JugglerName/VideoChannel:2/outputgenlocked"/>
</replies>
```

Get current genlock ratio and available options

Command

```
<command cmd="get" path="/GroupSet/Group/JugglerName/HdmiTx:3/genlockratio"/>
```

Reply

```
<replies success="true" path="/GroupSet/Group/JugglerName/HdmiTx:3/genlockratio"
        numberofreplies="1" commandDuration="234us">
 <genlockratio typeName="MemberArray" value="1:1" enabled="1"
        fullpath="GroupSet/Group/JugglerName/HdmiTx:3/genlockratio">
  <memberarrayitem UUID="04e120c50000000000400000000000178" value="1:1"
        enabled="1"/>
  <memberarrayitem UUID="04e120c50000000000400000000000179" value="1:2"
        enabled="1"/>
  <memberarrayitem UUID="04e120c50000000000400000000000180" value="2:1"
        enabled="1"/>
 </genlockratio>
</replies>
```

Get current genlock configuration and available options

Command

```
<command cmd="get" path="/GroupSet/Group/JugglerName/HdmiTx:3/genlockconfig"/>
```

Reply

```
<replies success="true" path="/GroupSet/Group/JugglerName/HdmiTx:3/genlockconfig"
        numberofreplies="1" commandDuration="123us">
 <genlockconfig typeName="MemberArray" value="Free Run" enabled="1"
        fullpath="GroupSet/Group/JugglerName/HdmiTx:3/genlockconfig">
  <memberarrayitem UUID="04e120c50000000000400000000000165" value="Free Run"
        enabled="1"/>
  <memberarrayitem UUID="04e120c50000000000400000000000166" value="Lock to Rx:1"
        enabled="1"/>
  <memberarrayitem UUID="04e120c50000000000400000000000167" value="Lock to Rx:2"
        enabled="1"/>
```

```
<memberarrayitem UUID="04e120c50000000000400000000000168" value="Lock to Rx:3"
    enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000169" value="Lock to Rx:4"
    enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000170" value="Lock to Tx:1"
    enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000171" value="Lock to Tx:2"
    enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000172" value="Lock to Tx:3"
    enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000173" value="Lock to Tx:4"
    enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000174" value="Lock to
    GPIO:1" enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000175" value="Lock to
    GPIO:2" enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000176" value="Lock to
    GPIO:3" enabled="1"/>
<memberarrayitem UUID="04e120c50000000000400000000000177" value="Lock to
    GPIO:4" enabled="1"/>
 </genlockconfig>
</replies>
```

Hotplug Detect

Simulate removal and reinsertion of HDMI output

Command

```
<command cmd="set" path="/GroupSet/Group/JugglerName/HdmiTx:3/triggerhpd"
    value="1"/>
```

Reply

```
<set success="true"/>
```

EDID Lists

Get current EDID selection and EDID list

Command

```
<command cmd="get" path="/GroupSet/Group/JugglerName/DpRx:2/edidlist"/>
```

Reply

```
<replies success="true" path="/GroupSet/Group/JugglerName/DpRx:2/edidlist"
        numberofreplies="1" commandDuration="567us">
  <edidlist typeName="MemberArray" value="HD60-Internal" enabled="1"
        fullpath="GroupSet/Group/JugglerName/DpRx:2/edidlist">
    <memberarrayitem UUID="04e120c500000000400000000000081" value="HD60-
        Internal" enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000082" value="2xHD120.bin"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000083" value="2xHD60.bin"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000084" value="4k30.bin"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000085" value="4K60.BIN"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000086" value="720p60.bin"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000087" value="HD120-
        10.bin" enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000088" value="HD120.BIN"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000089" value="HD60.BIN"
        enabled="1"/>
    <memberarrayitem UUID="04e120c500000000400000000000090" value="UHD60.BIN"
        enabled="1"/>
  </edidlist>
</replies>
```

Resolution

Get active lines for an input

Command

```
<command cmd="get"
        path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming/activelines"/>
```

Reply

```
<replies success="true"
        path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming/activelines"
        numberofreplies="1" commandDuration="123us">
    <activelines value="2160"
```

```
        fullpath="GroupSet/Group/JugglerName/DpRx:2/VideoTiming/activelines"/>
    </replies>
```

Get all input info

Command

```
<command cmd="get" path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming"/>
```

Reply

```
<replies success="true" path="/GroupSet/Group/JugglerName/DpRx:2/VideoTiming"
        numberofreplies="1" commandDuration="789us">
  <videotiming UUID="04e120c5000000000000000000000010"
        fullpath="GroupSet/Group/JugglerName/DpRx:2/VideoTiming" value="">
    <name value="VideoTiming"/>
    <enabled value="1"/>
    <state value="0"/>
    <activepixels value="4096"/>
    <activelines value="2160"/>
    <totalpixels value="4256"/>
    <totallines value="2222"/>
    <hsyncwidth value="32"/>
    <vsyncwidth value="10"/>
    <hactivestart value="112"/>
    <vactivestart value="59"/>
    <vrate value="59.999523"/>
    <pixelclockmhz value="567.40999999999996816769"/>
    <bpp value="8"/>
    <colourspace value="RGB-4:4:4"/>
    <interlaced value="0"/>
    <stereo3d value="0"/>
    <direction value="3"/>
    <icon value=""/>
    <dynamicicon value="0"/>
    <bitsperpixel value="8"/>
  </videotiming>
</replies>
```

MPCDI

Get import progress

Command

```
<command cmd="status"/>
```

Reply

```
<status success="true">
  <taskSummary total="7" failed="0" pending="1" complete="6"
        percentComplete="85.714286" MPCDI-Processing="85.714286%"/>
  <projectStatus ProjectPath="C:\Users\[UserName]
        \AppData\Local\compere\Projects\primary.prj" DefaultProject="C:
        \Users\[UserName]\AppData\Local\compere\Projects/myproject.prj"/>
  <networkInfo commsMode="server" peerData="64-4b-f0-12-75-fd "/>
</status>
```

Project Controls

Note: Saving and Loading is to the server and the path must use the correct slashes for the server OS.

Open a project

Command

```
<command cmd="open" file="C:\Users\[UserName]
        \AppData\Local\compere\Projects\primary.prj"/> (Windows)
<command cmd="open" file="/share/compere/bin/Projects/primary.prj"/> (Linux)
```

Reply

```
<open success="true" commandDuration="123us"/>
```

Save a project

Command

```
<command cmd="save" file="C:\Users\[UserName]
        \AppData\Local\compere\Projects\secondary.prj"/> (Windows)
<command cmd="save" file="/share/compere/bin/Projects/secondary.prj"/> (Linux)
```

Reply

```
<save success="true" commandDuration="123us"/>
```

Presets and Clones

Use a Preset or Clone resource

(Recalling a Preset imports values to matching UUIDs. Loading a Clone imports saved copies of objects with new UUIDs.)

Command (Presets)

<command cmd="recallpreset" file="C:\[Compere Root Path]\Presets\[file].pre"**/>**

or (implied file location)

<command cmd="recallpreset" file="[file].pre"**/>**

Reply

<recallpreset success="true" commandDuration="504us"**/>**

Command (Clones)

<command cmd="loadclone" file="C:\[Compere Root Path]\Clones\[file].clo"**/>**

or (implied file location)

<command cmd="loadclone" file="[file].clo"**/>**

Reply

<loadclone success="true" commandDuration="652us"**/>**

Logs and Diagnostics

Compere generates a log file (CompereLog.txt) which can be very useful for system debugging and troubleshooting. On a Windows machine this can be found by default in: C:\Users\[USERNAME]\AppData\Local\Compere\logs.

➢ Default file locations can be edited. See File Locations in [Preferences Panel](#) ⁵⁵ .

These files inflate, so when they get to a certain size they are archived with a sequential file suffix: CompereLog_nn.txt.

On Linux, this will be in a directory one up from the executable directory and then in a folder called logs.

Juggler's internal Linux Atom control board also runs a reduced version of Compere, running from /share/compere/bin. Its logs can be found in /share/compere/logs.

On Mac these should be found in ~/Library/Compere/logs.

Logs can be viewed dynamically, from the Panels > Logs menu:
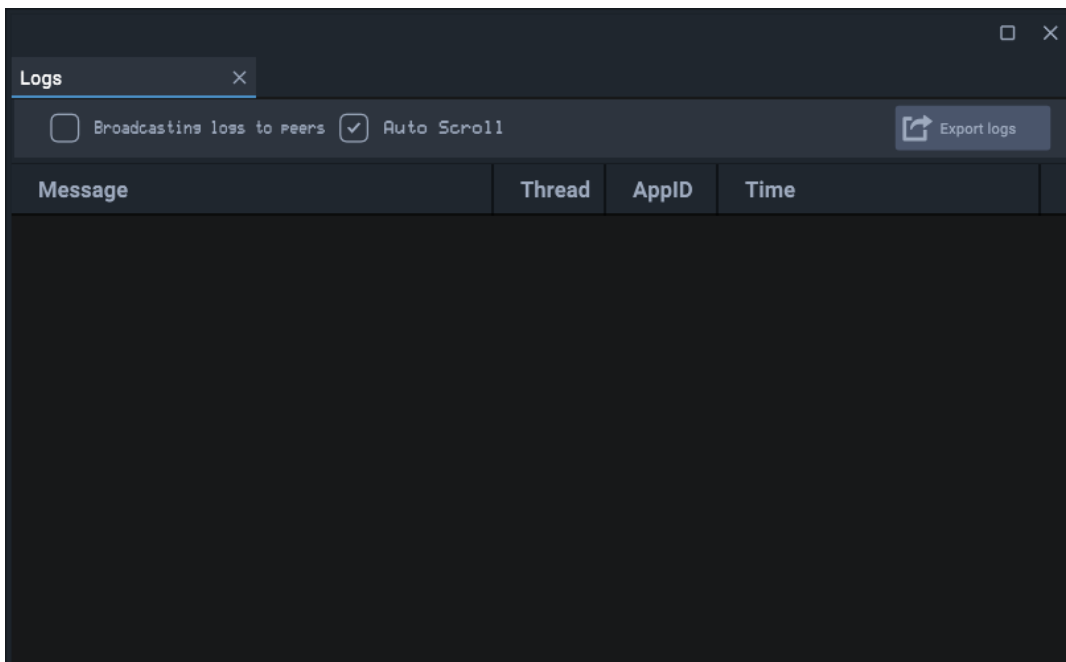
- Choose whether to broadcast the log to Compere peers in the Group.

- Check Autoscroll to keep latest logs in view.

- Export logs to a LogData.txt file (location defaults to the location set for logs in Preferences, or choose your filename and location).

Gather Project Data

Lists logs per Juggler in the Project Group. Click 'Open' to create a zip file in this location of all logs and the local Compere Preference.pref file.
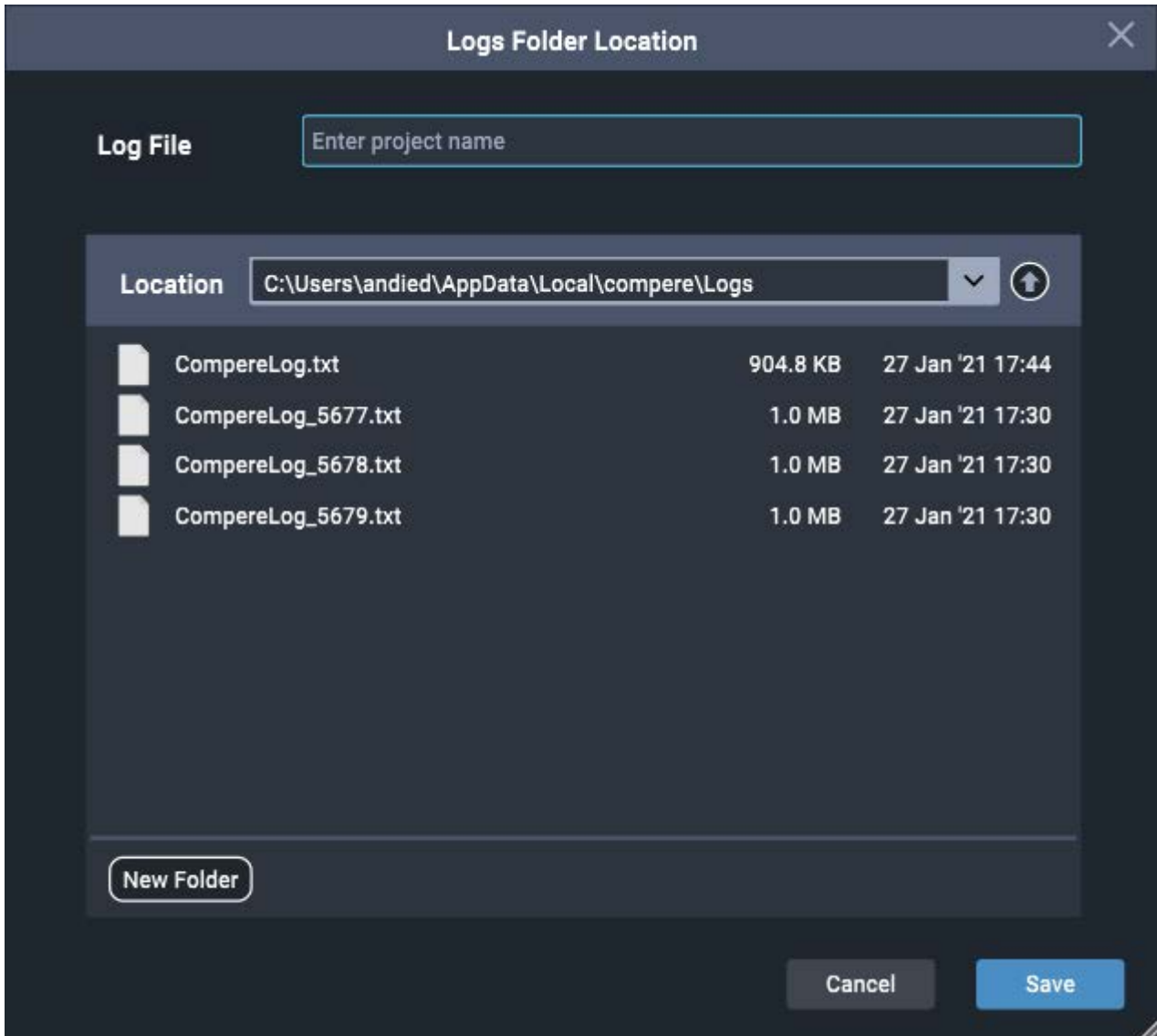
Reporting diagnostic information

Send the following information to us to help us understand what you are seeing:

- Save server Project File locally (XML)

- Screen shot of Nodes View

- Current log file (no number suffix)
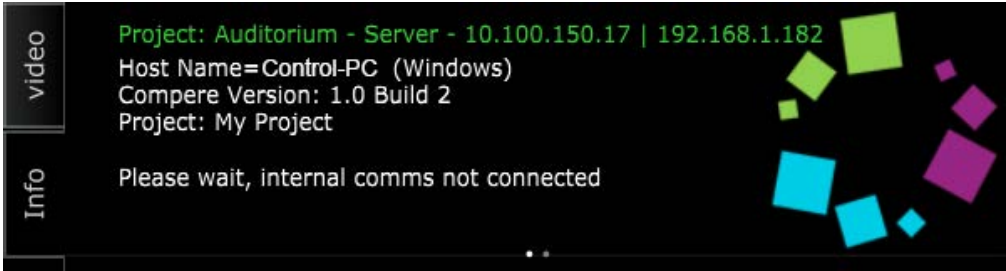
For technical support, please contact support@7thsense.one.

Command Line Arguments

Occasionally it can be useful to run Compere with a command line switch, to force it to run in a particular way.

For example, different Windows shortcuts can be created to do this, by adding the switches into the shortcut target:

C:\7thSense\compere\compere.exe networkserver

Switch	Function
networkserver	opens Compere as a network server on startup
networkclient	opens Compere as a network client on startup
mingui	opens Compere in a minimal form, as it does on a Juggler front panel:
mingui systemtray	runs Compere in the Windows System Tray, with the option to Quit (see Compere MinGUI) [32]

Troubleshooting a Juggler System

Bus faults

Faults in the bus will be indicated by the Rx bus pins in a Juggler in the Nodes panel being red when they should be green (not set to Off, but cabled). The majority of faults in the bus system will be caused by connector contamination. See Connect a Juggler Optical Data Bus [26] for cable care and cleaning.

Media connection faults

When disconnected or valid media is not present, Juggler input and output pins in the Nodes panel will show red.

NDI continuity

NDI preview in Jugglers in the Nodes panel should be showing the continuity of media flow. When a feed is unavailable but NDI preview is enabled, these previews will show a grey NDI icon only:

Network faults

In the Nodes panel, disconnection of a Juggler is indicated by the node going red. Loss of communications by a control PC will be indicated by the Compere UI going red.

Check in *Configuration > Network Configuration* that the Project Group is correct and that the affected instances of Compere (control PC or Jugglers) are in a Group with a server. Reconnecting a Juggler as server will replace a lost server.

Juggler Physical Specifications

Environmental Characteristics	Operating	Non-operating
Temperature	+15 to +30 °C	-10 to +50 °C
Humidity (non-condensing)	10 to 90%	5 to 95%
Altitude	≤2000 m	≤10000 m

Specification	Rating/Description	Notes
Rackmount Dimensions (H × W × D)	(1U) 44.3 × 424 × 500 mm	Width including mounting ears: 482.6 mm
Rackmount weight (approx.)	7.2 kg (approx)	
Power Supply	100-240 VAC 50-60 Hz	Autoranging, IEC C14 inlet
Power	100 W	
Current (max.)	1.5/0.9 A	
Cooling	Forced air	Inlets at front, exhaust at rear
BTU Rate	341	

Front panel dimensions, mm [in]
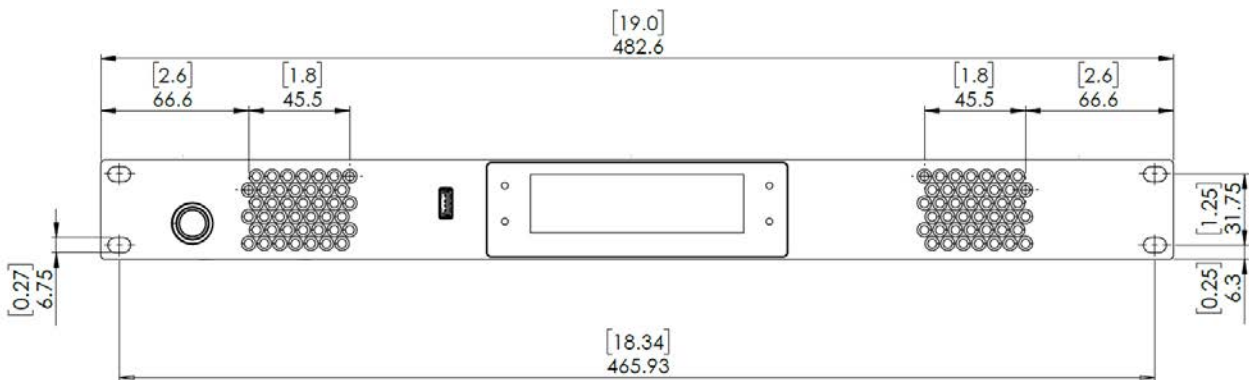
Side panel exhaust vents, mm [in]
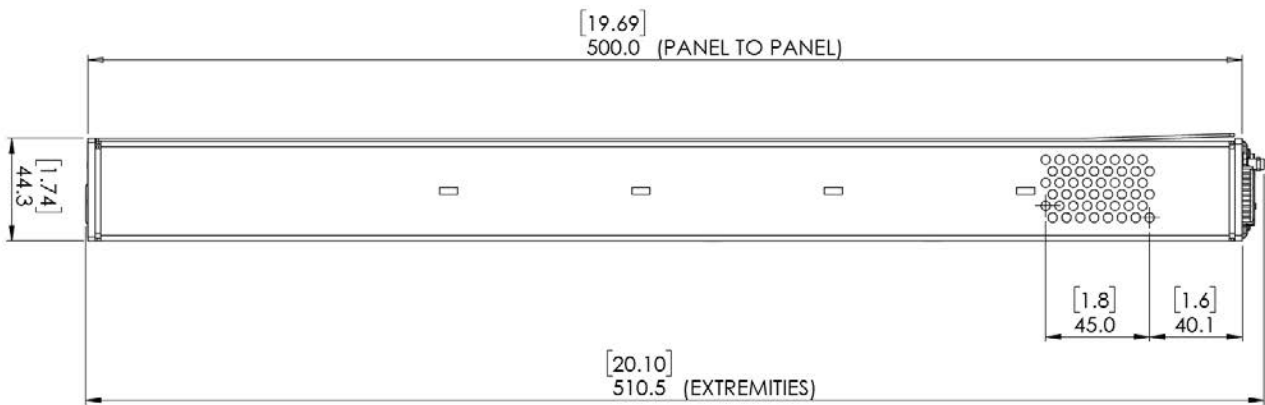
Rear panel exhaust vents, mm [in]
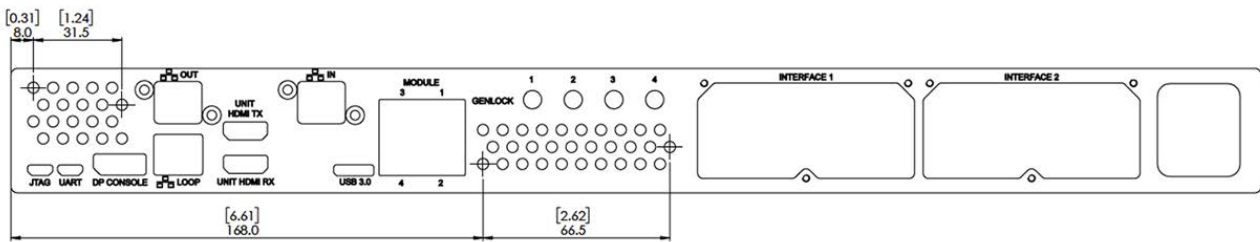
Rack Mounting Jugglers

Juggler units must be installed in a rack with load-bearing support along the sides, under the rear, or on a shelf. The rear and side exhaust vents must be kept clear. The front panel should be secured using an M6 rack mount bolt in each of the top and bottom mounting holes on each side of the unit's front panel. These ears alone will not support the weight of the unit.

Mount Jugglers with a minimum one-third rack-unit spacing:

➢ See Juggler Physical Specifications [136] for dimensions.

Airflow and cooling

Airflow intake is through the filtered vents in the front panel, and exhaust is out through the rear panel and the rear right-hand side. These vents must all have a clearance of at least 50 mm.

Juggler Care and Maintenance

Environment

Juggler is intended for use in a commercial (not domestic) environment and should be operated in a clean, dust free area (e.g. a server room). It is intended for operation in benign environments (not subject to corrosive or explosive atmospheres, moisture or flammable gases). Units are not protected against ingress of liquid (IPx0) or designed to be subject to excessive mechanical shock or vibration.

Only used at altitude not exceeding 2000 m

Only used in not-tropical climate regions

Thermal

Do not block any of the unit's air vents and ensure the ambient temperature around the Juggler is compatible with the environmental characteristics stated in the [Physical Specifications](#) [136].

Particularly when mounted in a rack cabinet with front or rear doors the ambient temperature around the unit may be higher than the ambient room temperature.

Electrical

Before connecting power to the unit, ensure the electricity supply is compatible with the electrical ratings shown in this document and on the unit's product label. The unit shall not be connected to an IT (unearthed neutral) power distribution system and the mains socket-outlet used shall connect to the building installation ground.

Juggler is connected to and disconnected from the mains supply with the IEC 60320 power connector on the rear of the unit.

The Juggler is protected by twin (live and neutral) fuses in the IEC mains connector block. These are the only user-serviceable part of the Juggler. If either malfunctions, they must both be replaced by a qualified person, and only by **F2A H250V, 5 × 20 mm ceramic cartridge fuses**. Disconnect the Juggler from the mains supply before releasing the clip with a suitable tool to withdraw the fuse drawer, and replace both fuses. Push the drawer back in securely before reconnecting.

Cleaning

External casing and the front panel should be carefully wiped with a soft cloth and antistatic solution.

Air filter

A user-serviceable air filter is located behind each of two vents on the unit's front panel. It is important to remove and clean or replace the filters periodically to avoid overheating.

Spare filter pads for Juggler units can be provided by 7thSense. Juggler should not be run without filters, since accumulations of dust inside the chassis can also cause overheating.

Replacing filters

Withdraw the Juggler from the rack far enough to access the retaining screws above the front panel. There are two per filter allowing a small carrier to be lifted vertically out. The Juggler lid does not need to be unscrewed or raised.

Internal maintenance

Other than the mains fuses, Juggler contains no user-serviceable parts.

Warning: Do not operate the Juggler with any covers removed as the air flow through the unit will be compromised and may lead to overheating. The lid of the Juggler forms an integral part of the cooling circuit. Operation of the unit with the lid removed can cause overheating.

Any alterations to the product's specification, e.g. additional/alternative components, may invalidate the warranty or impact the unit's performance. Additionally, such changes or modifications could cause equipment no longer to comply with the relevant EMC regulations, and void the user's authority to operate the equipment. If in doubt, please contact your supplier.

Internal batteries

The Juggler has a service-only replaceable CMOS battery on the Atom CPU.

End of service life

Dispose of the equipment and any components in accordance with all local and national safety and environmental requirements.

Warranty, Support and Service

Standard warranty is 24 months, return-to-distributor. Please contact 7thSense Design if you require an extended warranty.

Please ask for instructions and request authorisation before returning a Juggler unit to your distributor.

Server Internal Maintenance

> Internal maintenance must only be undertaken by suitable qualified service personnel.

Warning: Do not operate Juggler units with any covers removed as the air flow through the unit may be compromised leading to overheating.

Any alterations to the product's specification, e.g. additional/alternative components, may invalidate the warranty or impact the unit's performance. Additionally, such changes or modifications could cause equipment no longer to comply with the relevant EMC regulations, and void the user's authority to operate the equipment. If in doubt, please contact your supplier.

Electrostatic discharge (ESD)

When handling electronic assemblies, take precautions to avoid electrostatic discharge (ESD) causing damage. Precautions should include the wearing of an ESD protective wrist strap and using antistatic bags for electronic assemblies when not installed in any equipment.

➢ Visit our Support Portal for support and service

Regulatory Compliance

FCC

Juggler Pixel Processor complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

Note: The equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with this user guide, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at their own expense.

European Union

Juggler Pixel Processers are a Class A product. In a domestic environment, this product may cause radio interference, in which case the user may be required to take adequate measures.

End of Service Life

Dispose of the equipment and any components in accordance with all local and national safety and environmental requirements.

Document Information

Date	Document edition	Software version	Revision Details	Author/Editor
December 2020	1	Compere 1.0-1	New release	Andie Davidson
February 2021	2	Compere 1.0-2	Compere version updates	Andie Davidson
April 2021	3	Compere 1.0-2	Updates to clones and presets; external control	Andie Davidson

The Adopted Trademarks HDMI, HDMI High-Definition Multimedia Interface, and the HDMI Logo are trademarks or registered trademarks of HDMI Licensing Administrator, Inc. in the United States and other countries.

E: info@7thsense.one
W: 7thsense.one

7thSense Design Ltd

2 The Courtyard, Shoreham Road
Upper Beeding
Steyning
West Sussex
BN44 3TN
UK

T: +44 (0) 1903 812299

7thSense LLC, Michigan

332 E Lincoln Ave
Suite 100
Royal Oak, MI 48067
USA

T: +1 248 599 2717

7thSense LLC, Orlando

4207 Vineland Rd
Suite M1
Orlando, FL 32811
USA

T: +1 407 505 5200