



# Compere

## Compere 1.0-7

User Guide



## Compere 1.0-7 : 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

All Rights Reserved. This document is copyrighted © by 7thSense Design Ltd and shall not be reproduced or copied without express written authorisation from 7thSense Design Ltd.

The information in this document is subject to change without notice. 7thSense Design Ltd assumes no responsibility for errors, and/or omissions contained in this information.

Printed: November 2024

This edition is for software version Compere 1.0 build 7  
Document ref.: M760-2

E: [info@7thsense.one](mailto:info@7thsense.one)  
W: 7thsense.one

7thSense Design Ltd  
2 The Courtyard, Shoreham Road  
Upper Beeding  
Steyping  
West Sussex  
BN44 3TN  
UK

T: +44 (0) 1903 812299

7thSense Design LLC  
4207 Vineland Rd  
Suite M1  
Orlando, FL 32811  
USA

T: +1 407 505 5200



Introduction	5
Compere UI for Devices	6
Installing Compere on a Control PC	9
Running Multiple Instances of Compere	11
Terms and Definitions in Compere	12
File Locations	15
Keyboard Shortcuts	16
The Default Workspace	20
Using Compere Workspaces	27
File Menu	29
Help Menu	32
Panels Menu	33
Configuration Menu	36
Network Configuration	37
Server Failover Configuration	40
Settings Panel	41
Wizards Menu	53
Task Tracker Panel	54
Status Bar	56
Compere Projects	58
Project Panel	60
Assembly Panel Context Menus	62
System Admin Menu	64
Compere MinGUI	65
Connecting Juggler Systems	66
Juggler Unit Properties	70
Internal Processor	72
Defaults	73
GPIO	73
System Monitor	74
Adding Jugglers as Nodes	75

Video Channels Properties	82
IntelligentSource	85
Video Channel Tx	86
Video Rx / Tx Port Interfaces	89
Video Interface Timing	92
Bus Connections in a Juggler System	93
Bus Properties	96
Bridging Two 8-Lane Buses	99
Output Canvas Bus	103
OCB Wizard	109
Connecting Juggler Inputs and Outputs	113
EDID Spoofing	116
Configure a Display with Juggler	117
Canvas Output Setup	121
Canvas Window Setup	127
Create and Recall Canvas Arrangements	130
Display Setup via MPCDI	137
Nodes: Programming Nodes	139
Presets and Clones	146
Logs and Diagnostics	148
Ports Used by Juggler and Compere	151
Activate a Codemeter USB Dongle	152
Document Information	155
Index	156

## Introduction

---

**compère***noun*

a person who introduces the performers or contestants in a show.

Similar:

host

anchor

master of ceremonies

announcer

presenter

*verb*

act as a compère for (a variety show).

Compere® software conducts the performers of creative media serving (Actor®), generative content streaming (Conjurer®) and dynamic compositing, capture and tracking. It is also the project manager of up to very large systems of Juggler® pixel processors. Compere users may have any combination of these components. This guide addresses the core part of Compere: connecting inputs, outputs and processes between devices, and the common user interface (UI) for all of components.

Other user guides include:

- [M611 Operating Juggler 1 Pixel Processor](#)
- [M512 Compere 1.0 build 5 \(for Juggler 1\)](#)
- [M757 Operating Juggler 2 Pixel Processor](#)
- [M702 Actor User Guide](#)
- [M860 Managing Assets in Compere](#)
- 
- M710 Compere JSON External Control Protocol from [our PDF Library](#)

All user guides are available on the [7thSense user portal](#).

## Compere UI for Devices

---

Compere runs on different 7thSense devices performing different roles. This role might be 'Juggler', 'Actor', 'Conjurer'. The common thread is that they all read from the same project 'stage-management' script, called the project, which instructs them how to act. The script is managed by the project group server, receiving and distributing all edits to the script. The script profiles every element of every connected device, so contains all the connections in, out and between the devices. The script itself is written mainly via the Compere UI, which can be done from any PC with Compere installed. Typically, this will be a dedicated control PC, but it doesn't have to be. Scripts can be edited and projects managed via external control commands.

### The Performers

Compere runs in the Juggler performer role on every Juggler device. Jugglers are first physically connected to each other, to the network and to related devices (see [Juggler operating guides](#) and [Connecting a System](#)<sup>(113)</sup> in this guide) and are then connecting internally, from inputs to outputs via whatever processes are required.

Compere runs on R-Series server devices in the Actor and Conjurer performer roles. These provide media inputs and outputs, run the Actor render engine, interact with generative engines and media sources, managing the asset base.

#### Helpful for new users:

- [Terms and Definitions in Compere](#)<sup>(12)</sup>
- [Keyboard Shortcuts](#)<sup>(16)</sup>

### Project Group, Server, Clients

#### The Project Group

Active instances of Compere, on any device, that have a shared involvement in a project, together form and join a [project group](#)<sup>(58)</sup>. This will include all the Jugglers, as well as Actor or Conjurer devices, plus a control PC that plays a part in managing the project.

#### The Server

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. The group elects a server according to user-assigned scores.

Compere has a mechanism for server failover whereby other devices can be assigned in cascade to take over. When a server failure is detected and internal 'election' is automatically called to assign a new server from the client pool (see [Server Failover Configuration](#)<sup>(40)</sup>).

## The Project and the Group

The server loads the project file and distributes it to all group clients in memory. Any member of the group can make changes to the project – the server listens and shares the changes with all members to update them. The project can in this way be wholly dynamic, receiving changes via the Compere user interface or by external control commands, constantly changing and responding. At any time, the state of the shared project can be saved by the server. If not, the next time the project is loaded by the server, it will revert to the last save and share that again will all group members.

## Server Election

If the server goes absent from the project group for any reason, the project will keep running in memory on each other member, because all clients still hold the script as last edited. Intercommunication pauses until there is a group server again. A client will automatically be elected to become the server, based on the pre-assigned scores.

## The Control PC

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. Any device running Compere can join a project group in order to supply the user interface for editing and control. If more than one member of the project group is using the control UI, this is notified in the [Status Bar](#)<sup>(56)</sup>. Under [Preferences](#)<sup>(42)</sup> a control PC can 'Use Local Render Settings' to avoid conflicts between simultaneous users who may be undertaking different tasks.



**Names:** Almost every element of a Compere 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.

## Network (all devices)

To start, the network that the Juggler system and associated devices live on needs to be set up in Compere, selecting the devices to take part in a project (see [Network Configuration](#)<sup>(37)</sup>).

## Canvas (Jugglers)

The function of 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 (depending on Juggler type) DisplayPort™ 1.2, HDMI® 2.0 and 12G SDI and ST 2110, and when put on a common fibre-optic data bus they can share a single extremely large 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 [Canvas assembly panel](#)<sup>117</sup>. 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.

## Juggler 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](#)<sup>66</sup>. Here you will add representations of all the Jugglers in your system, connect up any data bus that the system shares (see the appropriate *Juggler Operating Guide*), and connect media streams from anywhere in the system, insert media processes, through to display devices and areas anywhere in the system.

## 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, from timelines, direct capture, through outputs and into canvas windows all from within the Compere UI.

## External Control

Everything in a Compere system 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 commands and a show controller.

➤ See M710 Compere JSON External Control Protocol in [our PDF Library](#)



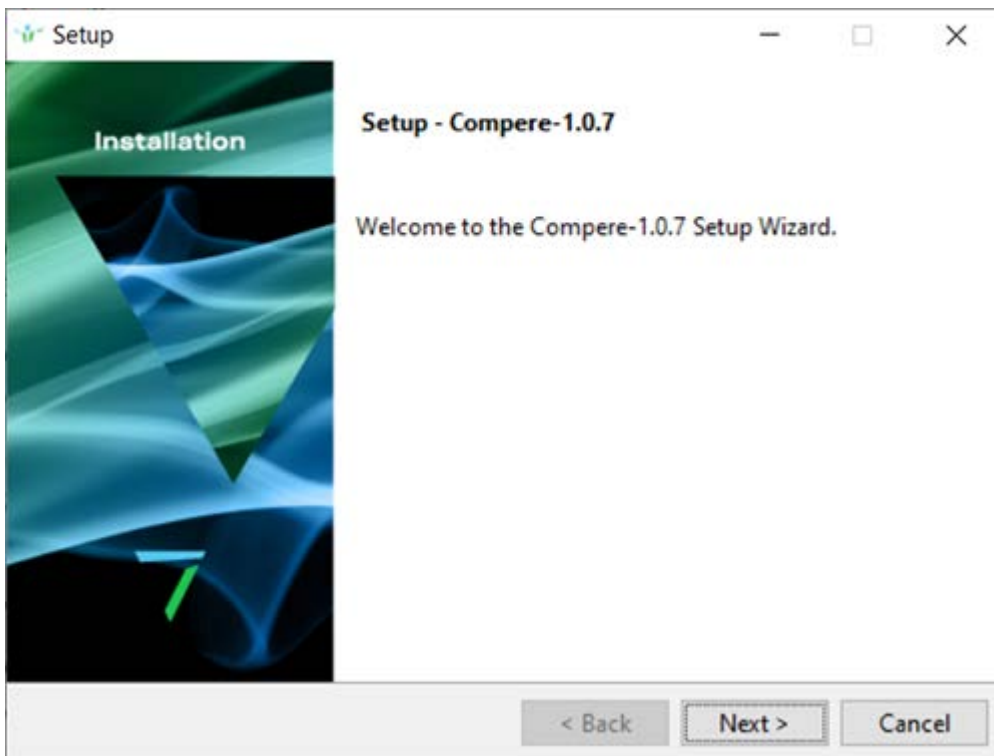
## Installing Compere on a Control PC

---

The installer includes Compere and the Couch DB media asset database app. CouchDB is required for all Actor devices, but not Jugglers.

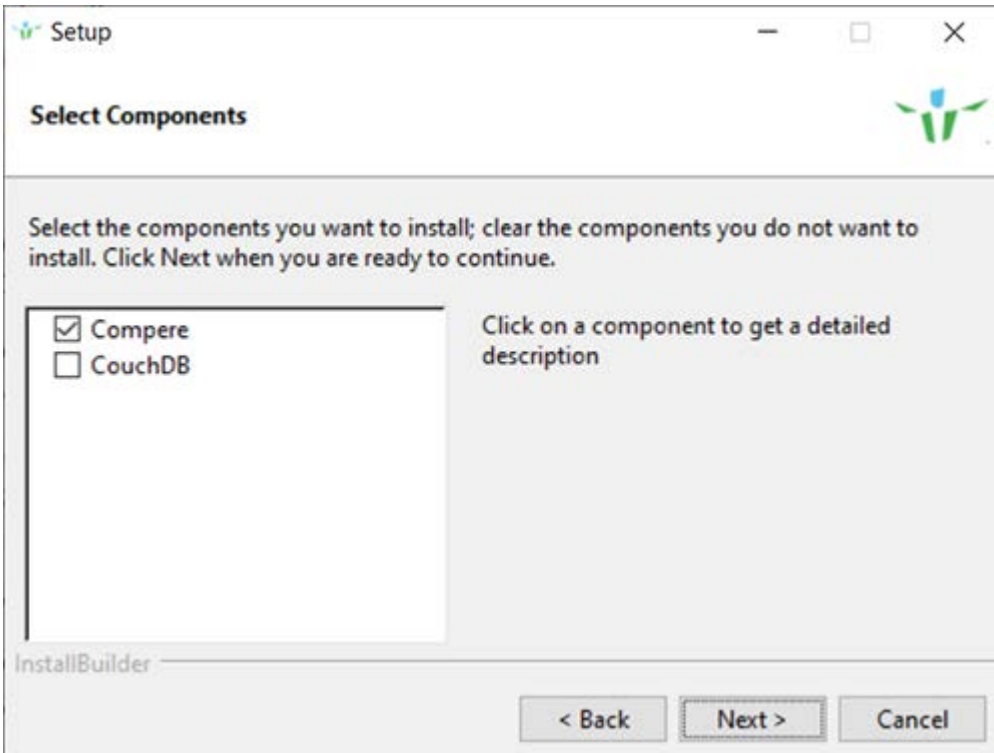
### Windows 10 and above

Compere installs to C:\7thSense\compere. Couch DB installs to C:\Program Files\Apache CouchDB and runs as a service.

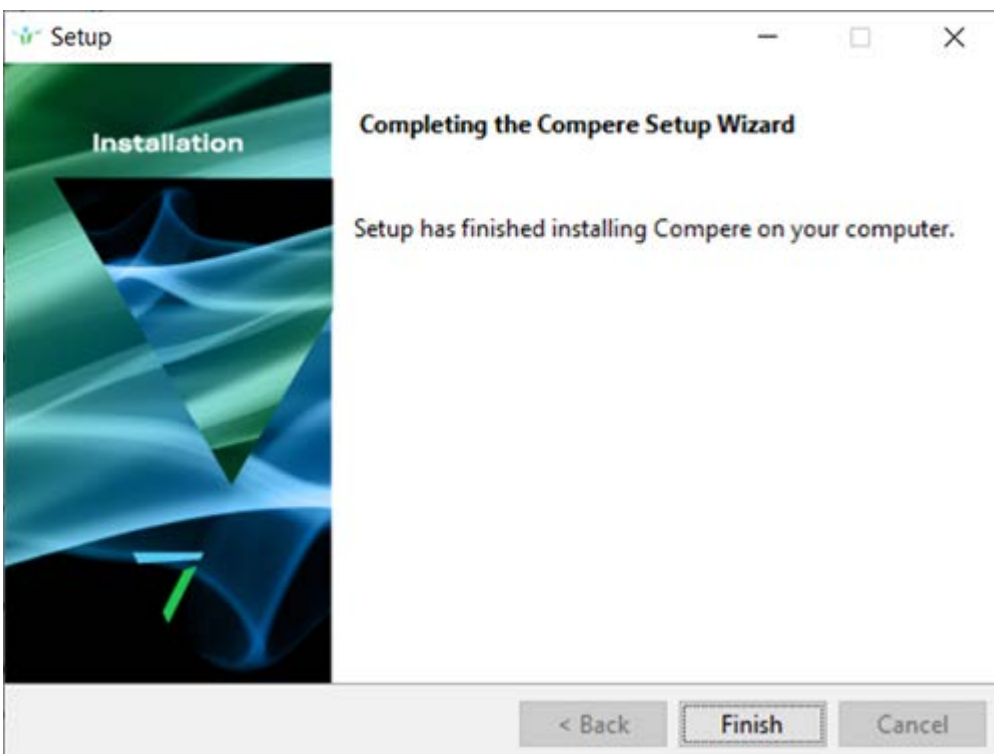


Read and accept the Licence Agreement.

If this is the first time of installing Compere on your device, and you are running in Actor role with timeline assets, you will need to include CouchDB. This is not required if just updating the version of Compere.



Click through 'Next' and Compere will install, showing when complete:



## Minimum requirements

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

You can check if a PC has all the minimum requirements with the help of an app such as GPU-Z, available at <https://www.techpowerup.com/gpuz/>.

For system configuration:

- Minimum quad-core, Intel® Core™ i5 processor
- 16 GB RAM (recommended)
- a dedicated GPU
- OpenGL 3.3 core support
- Vulkan 1.2
- Windows 10

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

Laptops with NVIDIA and also Intel integrated GPUs should give the NVIDIA precedence.

See: <https://gpu.userbenchmark.com/Faq/How-to-force-Optimus-or-Switchable-discrete-GPUs/97>

## Running Multiple Instances of Compere

It is possible to run multiple instances of Compere on the same device, for example if you need to run a server as an Actor device, and also run the Compere UI. To do this, Compere needs to reference separate configuration files, which you can create for multiple 'profiles'.

Since this is a special case, requiring deeper understanding, please contact [support@7thSense](mailto:support@7thSense).

## Terms and Definitions 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
<b>Actor<sup>®</sup></b>	In the 7thSense 'performer' range, Actor is the media server for timeline and asset management, input/output configuration, and output rendering.
Asset	Any piece of file-backed creative material, or modifier of it, that can be used on a timeline. Assets can be copied and will have associated metadata. May be movie, image, audio, MPCDI, mesh object etc. Assets are ingested by Compere's Asset Logistics.
Asset Logistics	The underlying mechanism (app) that ingests assets into protected locations and manages the metadata. It also tracks and distributes assets around the project group.
<b>Asset Management</b>	The Compere component that interacts with the asset database to present metadata of all assets available on the network, and make them findable and accessible to use on timelines.
Bus	Juggler: The 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 (input bus) or the whole system canvas (output canvas bus), and its overall bandwidth is dependent on the unit configuration.
Bypass	Juggler: A media stream input is directed in the same Juggler to a matched-type and resolution output, requiring no processing.
Canvas	Juggler: An array of pixels that is attributed to the combined resolution of number of physical outputs, for displaying media.
<b>Canvas panel</b>	Juggler: The assembly panel containing the total available video space for a Juggler system. This panel can contain multiple display canvases on which Canvas Outputs can be positioned.
Canvas Output	Juggler: A section of a composited canvas that is delivered to an output port via a Juggler.
Canvas Window	Juggler: A source of video that can be positioned on a Juggler Canvas across one or more Canvas Outputs.
Clone	xml descriptor for a project entity, e.g. canvas+outputs+windows; 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.
<b>Compere<sup>®</sup></b>	The project management software and UI that manages the other 'performers': Jugglers, Actors, Conjurers.
<b>Conjurer<sup>®</sup></b>	In the 7thSense 'performer' range, Conjurer introduces generative media as a playback resource.
Connector	A physical signal connection on a device of any type (e.g. DisplayPort, HDMI, SDI, QSFP module or bus), input or output.
Database	Actor: All timeline resources (assets) are indexed in a distributed database, from which the Asset Manager derives all metadata. This allows for searching, filtering and locating assets for use on timelines.
Device	Any item of hardware that can run perform in a Project Group (Actor, Conjurer, Juggler).
drop frame	In order to match video at 29.97 fps with real time clocks at 30 fps, (or 59.94 with 60 fps) this time code

Term	Definition
	drops two frames every minute, on the minute, except every tenth minute. In Compere, drop frame timecode is written with semi-colons HH;MM;SS;FF or HH;MM;SS;ms. See <a href="https://en.wikipedia.org/wiki/SMPTE_timecode">https://en.wikipedia.org/wiki/SMPTE_timecode</a> for a full explanation.
External Control	JSON SDK for command-style control over IP to address and message any element in a Project of networked devices.
Filepath	Refers to a specific file location rather than another item in a Project.
FPGA	Field Programmable Gate Array. The main component for video processing in Juggler.
Grid	A background scaling and positioning guide used in the assembly panels: e.g. Canvas, Nodes and Visualise.
<b>Input Feeds panel</b>	Actor: Select pixel areas from inputs (e.g. timelines) to display across the combined outputs of a media servers.
Ingest (assets)	Bring assets of any kind into a protected storage location and index it in the distributed database, incorporating editable metadata.
IntelligentSource™	7thSense automatic failover system.
Internal processor	Single board computers used in Juggler and R-Series servers managing internal communications and/or front panel touch-control displays.
<b>Juggler®</b>	A Juggler pixel processor unit of any configuration of inputs and outputs. Its external connections and internal processing are all managed by Compere to work with other 'performers'.
Media	Audio or visual creative content saved into a file format. Specific types of assets.
Mini Map	A small preview representing the current viewing position and enabling navigation, amongst all nodes available in the Nodes panel.
MinGUI	Compere running with minimal user interface where the overhead of the full GUI and panels is not needed, for example in client devices in a project group.
Network Client	Any network device running an instance of Compere that is not a Network Server.
<b>Network Discovery panel</b>	Manages all network connections between devices, allocates them to Project Groups and assigns group roles.
Network Server	Network device assigned as Compere 'project coordinator' across all devices. The server receives and distributes all changes to the project.
Node	Generic: UI object with pins that can be connected to other UI objects with pins to create flows and processes.  Type: [device], device interface, source, process node, logic node etc.
<b>Nodes panel</b>	An assembly panel into which collections of nodes are introduced, and connections and processes built.
Output Canvas Bus	(OCB) An alternative bus mode. The canvas with its outputs and windows is carried around the bus, rather than being assembled from independent Juggler inputs that individually can be carried on the bus.
<b>Output Feeds panel</b>	Actor: Add, scale and position viewports across the combined outputs of a media server.
Panel	Any moveable section of a workspace, most of which can be docked with other panels.
Panel Settings	In Configuration> Settings: Properties that define how the contents of an assembly-type panel are seen. Settings are shared by all project users, unless 'use local settings' is set in Preferences.

Term	Definition
Path	In Compere, a path is the route to any Project component, allowing it to be addressed in order to obtain its value or to change it.
Pin	UI representation of a input or output connector, real or virtual.
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. <i>Example of use:</i> multiple arrangements of windows in a canvas or canvases.
Project	A single dataset of information for a system of networked devices, display configurations and show elements, shared and used by all Compere nodes. (Think stage, lighting and cast of performers with various roles and props.) 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 file.
Project Group	A network Project Group comprises all devices on the same network assigned to work together in a specific way.
Project panel	Itemises every element of a Compere project: all devices, inputs, outputs and their connections, everything added to the project, from canvases to all assets in use on timelines.
Properties panel	Properties panels reveal all properties and settings, to read or edit, for selected items. A Properties panel can be locked to the selected item, allowing other properties panels to be opened to show other selections.
Resource	A replicable or reusable entity that can be used as a configurable copy in a project.
Rx / Tx	Rx (receive = an input), Tx (transmit = an output) typically used to identify connectors.
Task Tracker	Dynamic icon and panel displaying progress of internal project operations.
Timeline panel	Assembles assets and other resources, effects and and controls, on multiple layers for show creation and playback.  Actor can have any number of timelines, viewed in separate timeline panels. Each timeline presents a feed which can be viewed in NDI.
Timeline Preview panel	Shows visual media from the combined viewports of a given timeline as arranged in the Output Feeds panel.
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 1 and up to 6 per Juggler 2.
Viewport	An area of a video input resolution selected in the Input Feeds panel and represented in the Output Feeds panel, to appear on a device output. A viewport has an input face and an output face and is assigned to a single video input.
Visualise panel	The assembly panel in which a 3D view of projectors and surfaces in an illuminated theatre space, can be created and manipulated.
Workspace	An arrangement of panels that can be saved.

[^ top of page](#) <sup>(12)</sup>

## File Locations

---

Locations are given here for Windows. For Mac, the root is Users > [UserName] > Library > Compere

### Compere Root

Compere data is stored in: C:\Users\[UserName]\AppData\Local\Compere

Note that %localappdata%\Compere is a more direct route!

### Project

Compere Project (.prj) files are stored by default on the Compere server device in: [Compere Root Dir]\Projects

### Resources

Compere Project Resources are stored by default in: [Compere Root Dir]\Resources. Add subfolders as required.

### Workspaces

Compere Workspace files are the arrangement of panels saved by the user. Their default location is: [Compere Root Dir]\Layouts.

### Presets

Preset (.pre) files are saved to: [Compere Root Dir]\Presets.

### Clones

Clone (.clo) files are saved to: [Compere Root Dir]\Clones.



## Keyboard Shortcuts

Keyboard shortcuts are shown in menus and button dropdown lists, but some that you might use most often may be worth learning.

### General

Mouse / key	Function
Left mouse + drag	Pan a whole grid and everything on it.
Left mouse + Shift + drag	Reposition and resize objects on a grid.
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 <b>Canvas</b> 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 in/out about the cursor point
=/- keys (main keyboard) +/- keys (num pad)	Zoom in/out
V	Bring all objects in an assembly panel into view.
Ctrl+s	Save the project
Ctrl+j	Select all Jugglers
Alt+F4	Quit Compere (standard Windows function)

### Nodes panel

Key	Function
Shift+i	Import an MPCDI file
Ctrl+Alt+d	Clear temporary MPCDI data
Ctrl+j	Select all Jugglers
Ctrl+Shift+Alt+j	Calculate (red) bus links (same as toolbar button to show all bus links). Note that connections require a minimum zoom level to show, and must first be calculated.
h	Cycle display around revealed Juggler bus connections: All; Selected Jugglers only; None.
k	Cycle display around processing node links: All; Selected nodes only; None.
Shift+Alt+b	Reset the bus
Ctrl+x	Show / hide all text in Juggler nodes
Delete/Backspace	Remove link from selected pin

### Canvas panel elements

Key	Function
m	Toggle interaction mode (None/Canvas, Outputs, Windows)
n	Toggle NDI in Windows
Alt+t	Toggle NDI transparent
Alt+m	Toggle the minimap
Shift+drag	<b>Outputs and Windows:</b> snap to edges
Shift+f	Toggle a Window to fill the parent Canvas
Ctrl+cursor left	Move selected Windows to align with left edge of Canvas
Ctrl+cursor right	Move selected Windows to align with right edge of Canvas
Ctrl+cursor up	Move selected Windows to align with top edge of Canvas
Ctrl+cursor down	Move selected Windows to align with bottom edge of Canvas
Shift+c	Centre a Window in the canvas
Cursor left	Move selected Window/Output/Canvas left one unit
Shift+cursor left	Move selected Window/Output/Canvas left 10 units
Ctrl+Shift+cursor left	Move selected Window/Output/Canvas left 100 units
Cursor right	Move selected Window/Output/Canvas right
Shift+cursor right	Move selected Window/Output/Canvas right 10
Ctrl+Shift+cursor right	Move selected Window/Output/Canvas right 100
Cursor up	Move selected Window/Output/Canvas up one unit
Shift+cursor up	Move selected Window/Output/Canvas up 10 units
Ctrl+Shift+cursor up	Move selected Window/Output/Canvas up 100 units
Cursor down	Move selected Window/Output/Canvas down one unit
Shift+cursor down	Move selected Window/Output/Canvas down 10 units
Ctrl+Shift+cursor down	Move selected Window/Output/Canvas down 100 units

### Juggler admin

Key	Function
Shift+c	Show Juggler console window
Ctrl+Alt+c	Upgrade Compere version on a Juggler or Jugglers

Key	Function
Ctrl+Alt+e	Upload an EDID to a Juggler or Jugglers
Ctrl+Alt+f	Update firmware on a Juggler or Jugglers
Ctrl+Shift+e	Upload a script to a Juggler or Jugglers

### Beam control

Key	Function
<b>Individual</b>	<b>Off</b>
Ctrl+r	Red beam off
Ctrl+g	Green beam off
Ctrl+b	Blue beam off
Ctrl+w	RGB beams off
<b>All</b>	<b>Off</b>
Ctrl+Shift+r	All Red beam off
Ctrl+Shift+g	All Green beams off
Ctrl+Shift+b	All Blue beams off
Ctrl+Shift+w	All RGB beams off

Key	Function
<b>Individual</b>	<b>On</b>
Ctrl+Alt+r	Red beam on
Ctrl+Alt+g	Green beam on
Ctrl+Alt+b	Blue beam on
Ctrl+Alt+w	RGB beams on
<b>All</b>	<b>On</b>
Shift+r	All Red beams on
Shift+g	All Green beams on
Shift+b	All Blue beams on
Shift+w	All RGB beams on

### Actor / Timeline

Key	Function
f	Toggle Render window fullscreen
r	Rewind Timeline to start
spacebar	Stop/start toggle for playing a timeline
t	Create a new timeline
Ctrl+c	Copy a selected asset or assets
Ctrl+v	Place copied asset(s) to start at the current playhead position on a selected layer if available (empty and the same media type), or on a new layer if not
Ctrl+Shift+v	Paste copied asset(s) at a specified time on a selected layer if available (empty and the same media type), or at the specified time on a new layer, if not or if required
Ctrl+x	Remove selected asset(s) from the timeline; this does not delete the asset from the project and the asset can be pasted elsewhere
Ctrl+m	Add a timeline marker

Key	Function
Ctrl+Shift+m	Add a timeline marker at a position (raises dialog to specify)
Ctrl+g	Go to frame and stop (raises a dialog to specify)
= and - (or mouse wheel)	Zoom in and out of the timeline horizontally. Add shift for vertical layer zoom.
Ctrl+z	Zoom to fit all timeline assets into current panel view ('zoom extents')
Ctrl+ ◀ ▶	Step playhead to next marker left/right on this timeline
Ctrl+Shift+ ◀ ▶	Step playhead to previous / next asset start frame on this timeline
Cursor keys: ◀ ▶	Nudge timeline asset one frame at a time

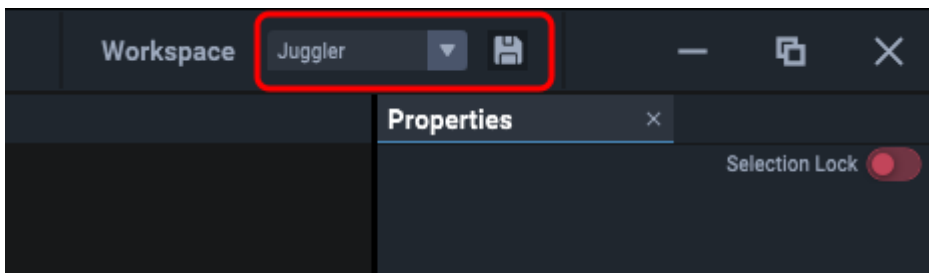
## The Default Workspace

---

In order to see the complete Compere-supported system, switch on all networked devices, inputs, outputs and displays.

Open Compere on a Control PC for the first time, and you will be in a new empty Project. We shall learn more about [Compere Projects](#) <sup>58</sup> later; for now, get to know how Compere is put together.

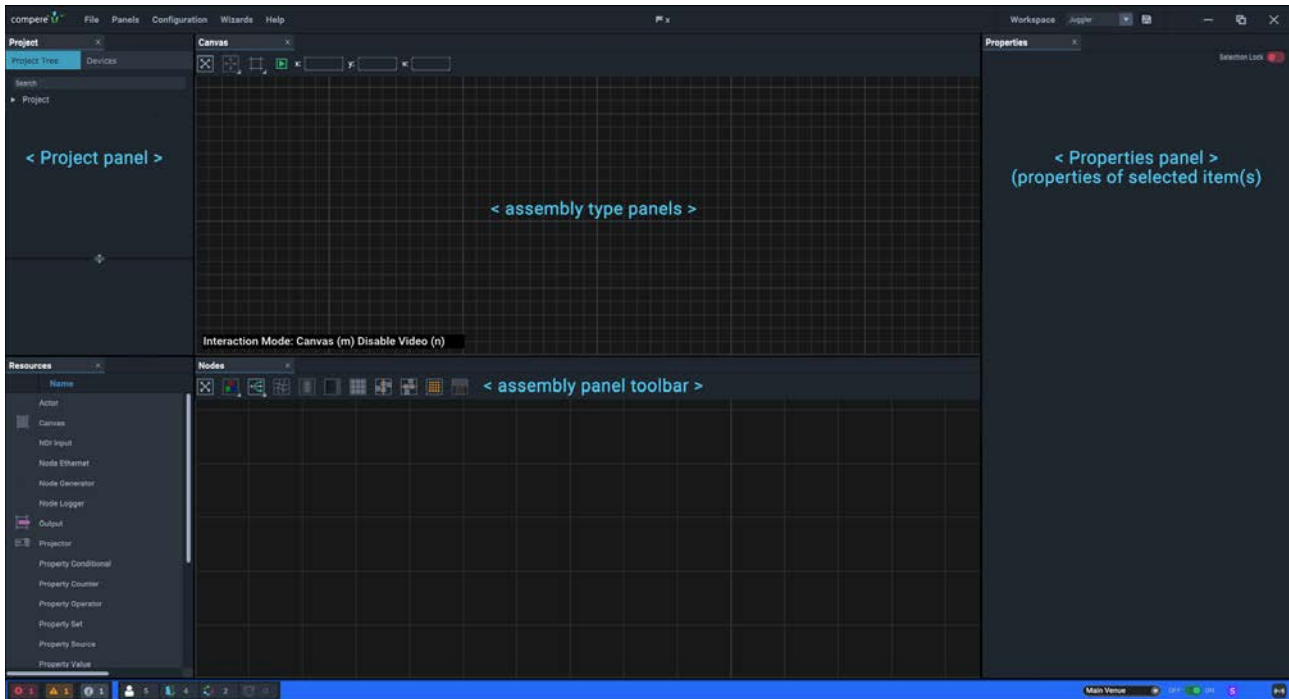
This guide will take you through the default Juggler workspace for a project. If not in the default workspace, select 'Juggler' from the workspace selector:



**An alternative default workspace 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.

## Workspaces

The Juggler workspace is just a default arrangement of panels in the Compere main window. Note the default Nodes and Canvas assembly panels, and the sidebar panels. You may want to maximise the screen:



The initial default 'Juggler' workspace comprises:

- the main (top) menu area
- the 'Project' panel (default: upper left), listing all configurable elements, tabbed as 'Project Tree' or as 'Devices' (Actor, Conjurer servers or Jugglers)
- the 'Properties' panel (default: right) for any selected element(s) 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 type panel' with a tab marked 'Canvas'
- a grid 'assembly type 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.



Once you have saved a workspace, do not press Reset UI because this will delete your workspace and revert to a default.

## Project panel

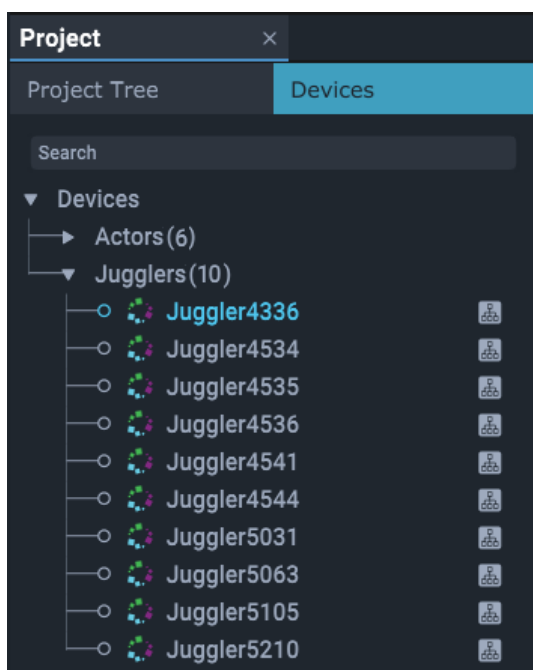
**The Project panel** is a branching structure of every elements and detail in a Compere project. These can be dragged into assembly panels as appropriate, examined for properties and address through external control.

The two tabs reflect everything being used in the project, and the devices that run it (Jugglers, Actors, Conjurers).

When you start Compere for the first time, you will find a new empty Project. Once saved, projects are held as extensive xml files, by default stored on the server in C:\Users\[username]\AppData\Local\Compere\Projects, with the extension .prj.

The Project is where you will assemble everything required to 'set the stage'. Your devices 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.

When you join a network with the Compere UI on a device, all Devices 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.



This is just the list of what is available. Each can be named, and examined for the properties of all its connections.

From here, you will select and drag Juggler names into Nodes panel to further configure and connect them (see [Nodes: Connecting Juggler Systems](#)<sup>66</sup>). Only Jugglers in the same [project group](#)<sup>37</sup> can be added to the Nodes panel, and those that have then been added are identified by an icon to their right (as shown above).

When a Juggler is physically connected to display outputs via DisplayPort or HDMI, 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.



Note that systems employing ST 2110 require SDP files rather than EDIDs.

### Server maintains the Project

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

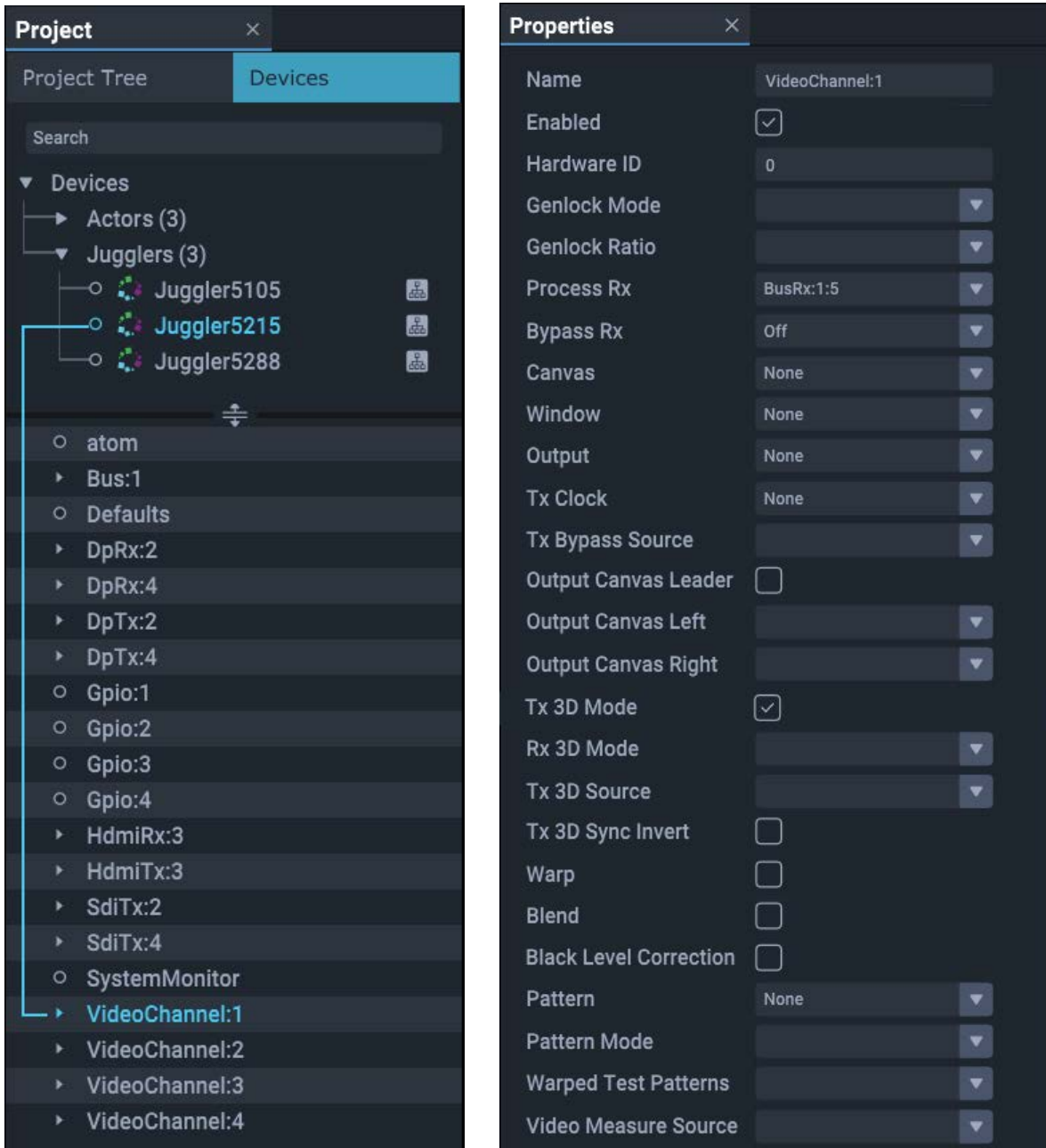
➤ For full details, see [Compere Projects](#) <sup>58</sup>

### Properties panel

**The Properties panel** reveals all the features for any element in the Project. The Properties panel will always be showing the properties of the item or items highlighted as selected in the Project panel.

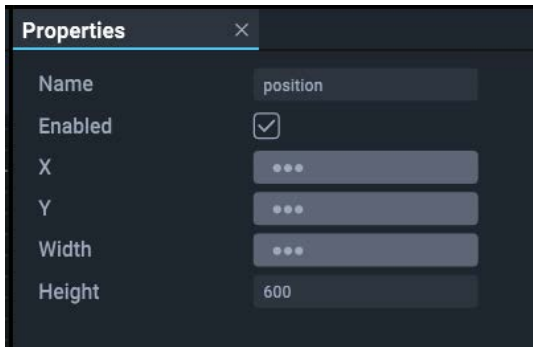
**Note** that illustrations in these user guides are examples of standard objects. The flexible nature of Compere allows for dynamic objects to be inserted into projects with their own dynamic properties. This allows for external devices and processes brought into a project to be controlled via Compere.

Members in the Project panel have their properties broken down by elements, for example a Juggler is broken down to each of its components and connections. Select a branch to see and edit the properties in the properties panel. For example, the properties of Video Channel:1.



**Note:** multiple items, such as Jugglers or Canvas Outputs, can be selected in the Project panel, but only shared common properties and values will be shown. Unshared properties will be greyed out and unshared values will show ellipsis (...).

Example: multiple windows that only have a shared height value will only show that value, which can be edited for all:

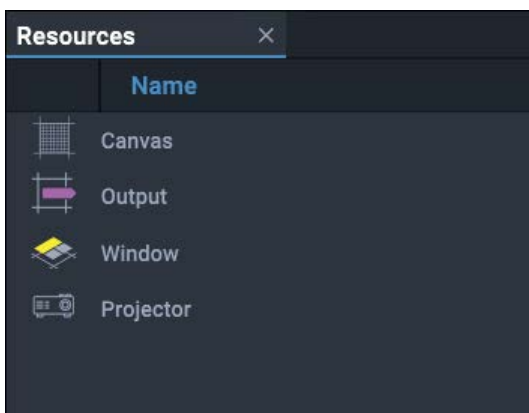


## Addressing properties remotely

Hover over a property label in the Properties panel to see its full address. This will appear as a tooltip but also in the bottom left of the status bar. Shift+Right-click to copy this path string for use in External Control commands.

## Resources panel

**The Resources panel** contains internal Compere objects that can be inserted into the project, including Canvas, Window, Juggler node components, programming nodes and default projectors.



## Using Resources: Add, Select, Edit, Remove, Delete

- Resources can be dragged onto an assembly type panel (e.g. Timeline, Input feeds, Visualise model etc.). This simultaneously adds them as branches of the Project panel.
- Resources and Devices shown in the Project panel, such as Jugglers and imported objects, can be dragged into an assembly panel.
- Objects are selected within an assembly panel to position, size or edit them, either 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 a light blue border, and its name highlighted in blue in the Project panel. Its properties are now shown in the Properties panel.
- Right-click objects for contextual options: the selected object will be bordered in blue as well.

- 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 without deleting it from the project, select 'Remove' from its right-click context menu.
- Links between nodes and/or pins are deleted by selecting the receiving connector, and pressing delete.

### Summary

- ❖ **Assembly** panels are a visual representation to aid system assembly and configuration.
- ❖ The **Project** panel holds all the members of the Project Tree with their individual configurations and links, whether displayed in other panels or not, and devices (Jugglers, Actors or other device sources).
- ❖ Resources are **added** to a project by being dragged into an assembly panel.
- ❖ Resources are **taken out** of a project by being deleted from the Project Tree.
- ❖ When resources are **removed** from an assembly panel, they remain fully **configured and interactive** within the project.
- ❖ Resources that have been removed can be dragged back from the Project Tree into an assembly panel, where their configuration, links, etc. will be correctly displayed.

## Canvas assembly panel (Jugglers)

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

➤ [Display Configuration: Canvas Setup](#) <sup>(117)</sup>

## Nodes assembly panel

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.

The Nodes panel is also used for building [Programming Nodes](#) <sup>(139)</sup> into dynamic functions.

➤ [Nodes: Connecting Juggler Systems](#) <sup>(66)</sup>

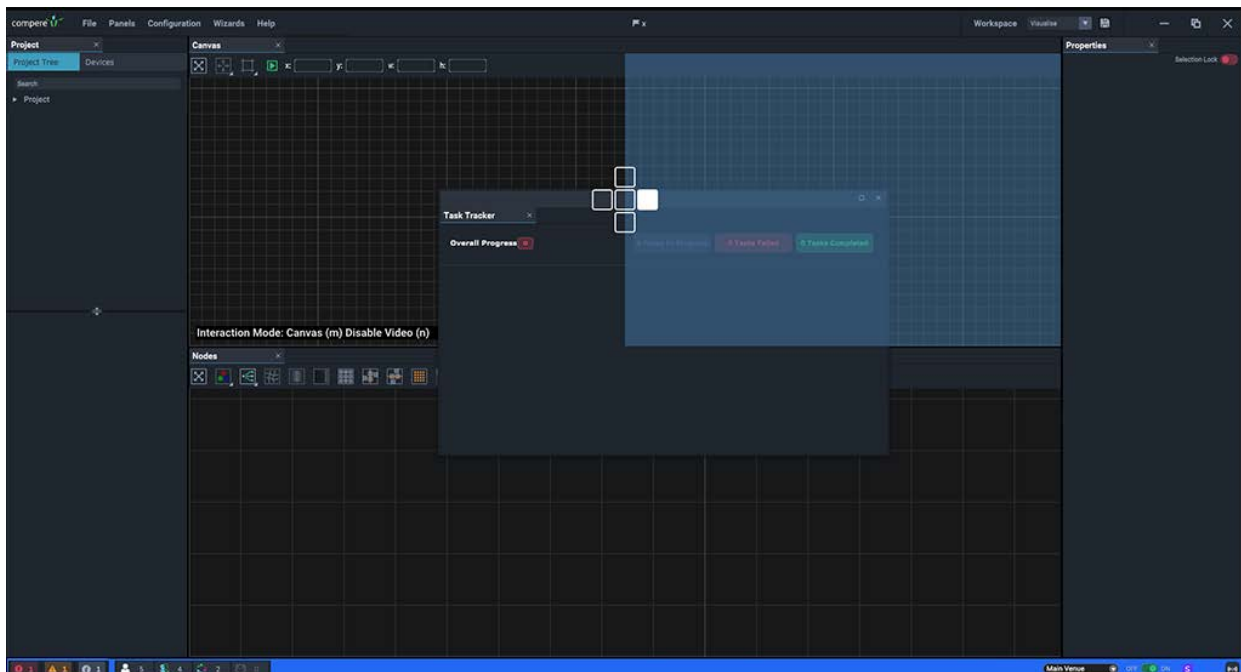
## Using Compere Workspaces

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 in the same space). Any item can be a docking space for others.

Any element of a workspace can be resized. When the whole space is resized, panels except for Project and Properties, resize proportionately. When more than one panel is in a row or column, they resize together.

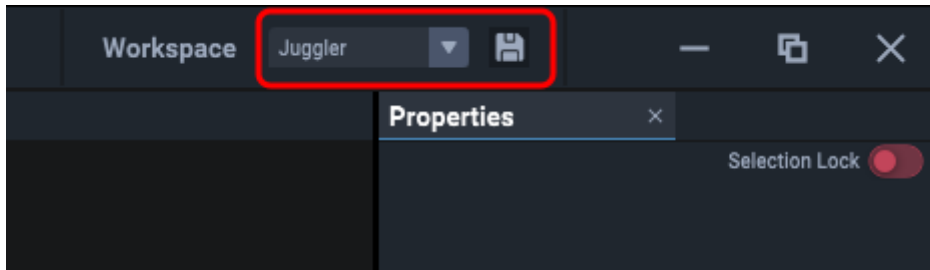
Elements are positioned by dragging their title tabs.

- 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 showing the drop positions relative to the panel you are over.
- 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 docked is highlighted.
- Drag free of the squares (no highlight) to float a panel. Multiple floating panels can be docked together.
- Here the Task Tracker panel is being moved to dock to the right of the Canvas panel:



- 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.

- Multiple Properties panels can be displayed but 'locked' to display for specific items.
- Menu *Panels > Reset UI* will restore the default fully-docked arrangement of panels to a default workspace relevant to either Juggler or to Visualise View.
- To save your own workspace 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

- Resources can be **added** to assembly panels in several ways: by dragging from panel lists, selecting from context menus, using shortcut keys or selecting from dropdown lists.
- 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.
- To **delete** items from the Project altogether, select and delete them from the project tree (there is no undo).
- 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, hover over 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.

## 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.

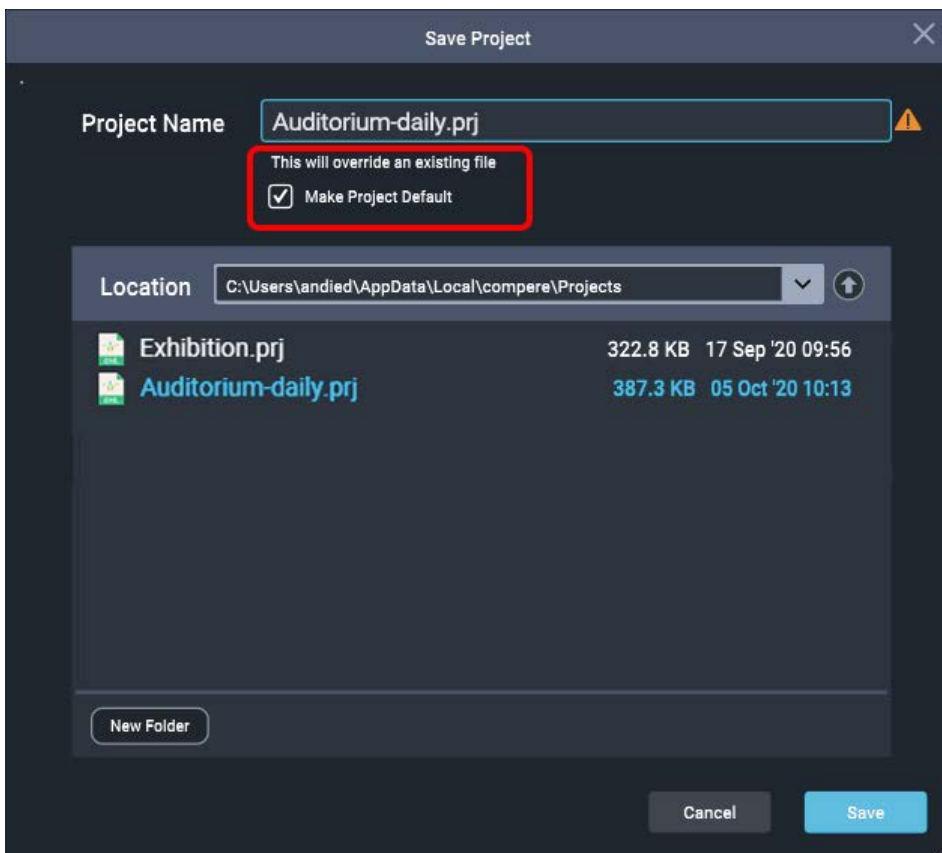
#### Open

Open a saved project file (file extension: .prj). Project files are constructed in relation to project groups, so will anticipate all members, their roles and contents. When the server opens a file (or a default project opens on a server start-up) the server will distribute the project to all clients.

#### Save / Save As

Enter a filename for your project, and/or as the server, override the default location. For a client, the save instruction will go to the project group server for saving and 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 an overwrite 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).

When a client is a member of a project group with a server, the 'Save' will always be in favour of the server's project folder, to redistribute the current project around the group.

If offline, the project file will be saved locally to the client, but on rejoining as a client to a project group running the same-name project, the client will receive the current project file fresh from the server.

If, as a live member of a project group, you want to keep a local copy of the project, then use 'Export' instead. If you want this to run on the group server, you must transfer the file to the server separately.

### **Export**

For a client, or a client not attached to a server, it can be useful to save a copy of a project to a client-local 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.

### **Apply Preset to Selection**

Typically for configuring multiple Jugglers of the same kind from the saved properties of just one.

➤ See [Clones and Presets](#) <sup>146</sup>

## 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). This will raise a confirmatory dialog if set in *Preferences > Settings > 'Confirm Close'*.

## Help Menu

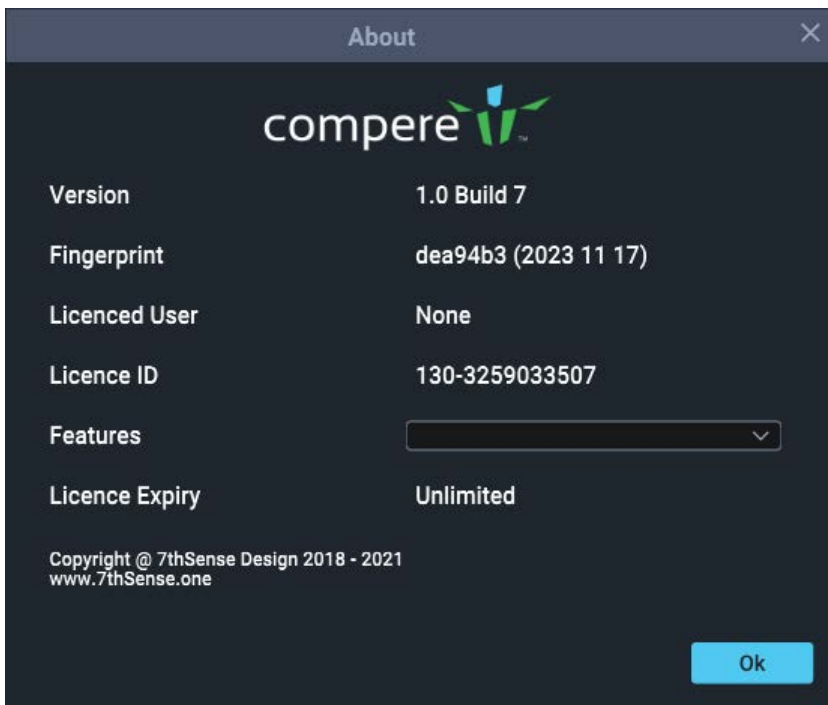
---

### Generate System Report

See [Logs and Diagnostics](#) <sup>148</sup> for how this is used to gather comprehensive logged information, for the local device, for diagnostic purposes.

### About

Shows the version of Compere on this device, included features, and licence details.



## Panels Menu

---

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

➤ See [Compere Workspaces](#) <sup>(27)</sup>

### Nodes

Add a Nodes assembly panel into the workspace.

A node is any device or processing block. They are introduced into a Nodes panel to create the connections between them.

### Canvas

Add a Canvas assembly panel into the workspace.

A canvas is an assembly of Juggler canvas outputs and canvas windows into a map of pixels being processed by Juggler systems.

➤ See [Configure a Display with Jugglers](#) <sup>(117)</sup>

### Project

Add a Project panel into the workspace.

This panel itemises every element of a Compere project: all devices, inputs, outputs and their connections, everything added to the project, from canvases to input and output feeds, and all assets in use on [timelines](#).

➤ See [Project Panel](#) <sup>(60)</sup>

### Properties

Add a Properties Panel to the workspace.

Properties panels reveal all properties and settings, to read or edit, for selected items. A Properties panel can be locked to the selected item, allowing other properties panels to be opened to show other selections.

### Logs

Open live log messages. See [Logs and Diagnostics](#) <sup>(148)</sup>.

### Resources

Add a Resources Panel to the Layout

### Task Tracker

Open a process progress [Task Tracker Panel](#) <sup>(54)</sup>.

### Warp Mapping

Two tabs list Region Sets and Virtual Canvas Presets. Used with ST maps in Actor, to create picture-in-picture (PiP) regions over an overall canvas (contiguous display area). See [Actor User Guide](#).

### Visualise

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

### Timeline

An Actor panel for constructing timelines. See the [Actor User Guide](#).

### Timeline Preview

A timeline preview can be achieved locally by running two instances of Compere: one for the UI and the other as a server (minimal playback interface only). See [Running Multiple Instances of Compere](#)<sup>11</sup>.

### Input Feeds

An Actor panel for applying viewports to timelines.

### Output Feeds

An Actor panel for positioning viewports on device outputs.

### Asset Management

Interacts with the Asset Logistics database to organise timeline assets and make them available. See the [Actor User Guide](#).

### Reset UI

Restore the default Compere workspace. **Note: this will overwrite any workspace you have created and saved.** under this name.

## Custom Web Panels

Browser control interfaces can be created (for example in Medialon) using web panels on remote devices. These can be replicated in Compere as custom web panels, connected to the page URLs.

Access to Custom Web Panels is provided by creating a JSON file 'CustomWebPanels.json' for them, and locating it in the local (typically C:\) %localappdata%/Compere older. This file is read once at start-up, so Compere will need to be restarted to refresh the Panels menu.

Where this file exists, the panel name(s) listed in the file will be reflected in the panels menu.

Here is an example of the file contents, containing 3 groups, to deliver: a "Show Control" set containing 7thSense website pages", a "Search Engines" set of typical SE pages, and a set containing any number of other useful URLs. Note the **comma** separators between elements.

```
[
  {
    "name": "Show Control",
    "panels": [
      {
        "name": "7thSense",
        "url": "https://www.7thSense.one",
        "show-status-bar": "true",
        "allowed-navigation-regexes": [
          "https://(www.)*7thsense.one/",
          ".*7thsense\\.one/discover/.*",
          ".*7thsense\\.one/product/.*"
        ]
      }
    ]
  },
  {
    "name": "Search Engines",
    "panels": [
      {
        "name": "Google",
        "url": "https://google.com",
        "show-navigation-bar": "true",
        "show-status-bar": "true"
      },
      {
        "name": "Duck Duck Go",
        "url": "https://duckduckgo.com",
        "show-navigation-bar": "true"
      }
    ]
  },
  {
    "name": "Test",
    "panels": [
      {
        "name": "Test 1"
      },
      {
        "name": "Test 2"
      },
      {
        "name": "Test 3"
      },
      {
        "name": "Test 4"
      },
      {
        "name": "Test 5"
      }
    ]
  }
]
```

For further information and advice, contact [support@7thSense.one](mailto:support@7thSense.one).

## Configuration Menu

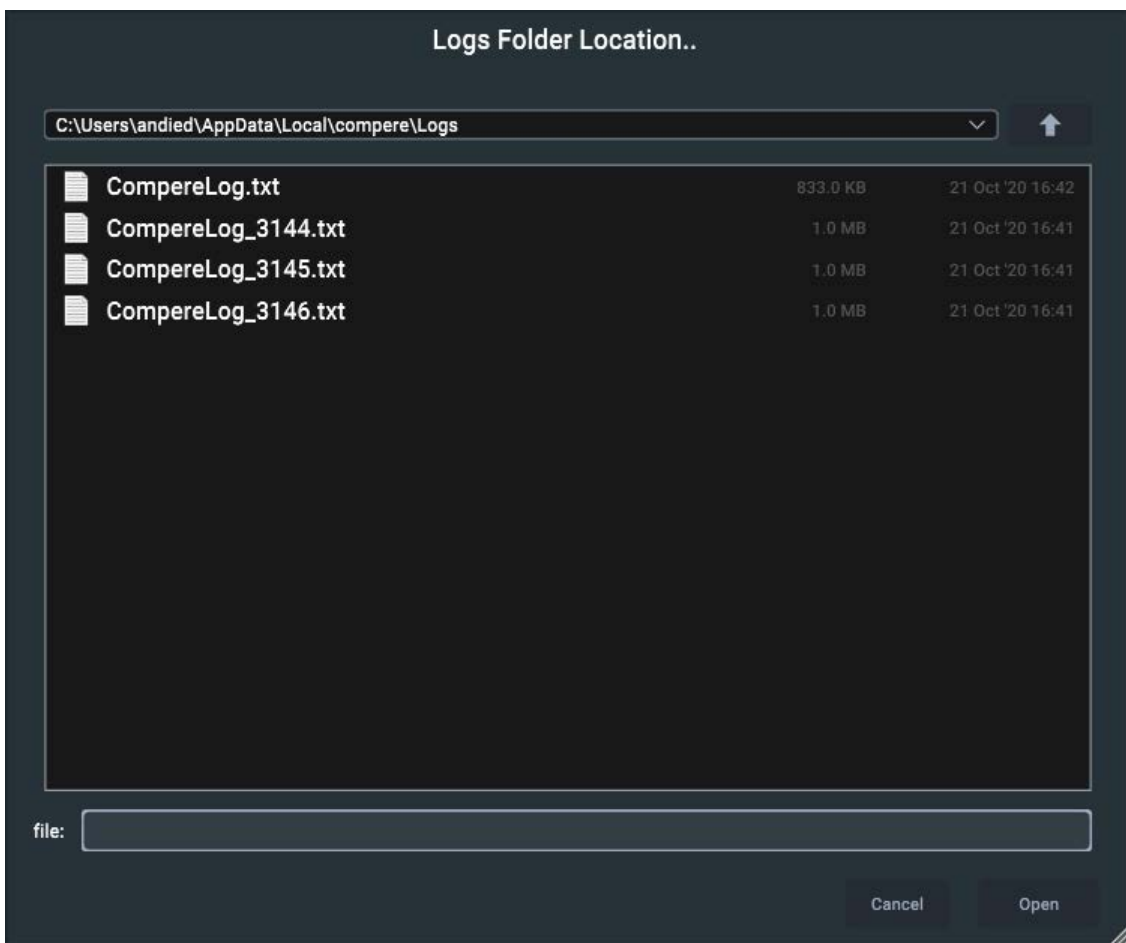
---

### System Config Table

Displays a table of all **Jugglers** found on the network, 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 panel, for configuring a network of all Compere devices.

➤ See [Network Configuration](#) <sup>37</sup>

### Settings

Opens the Settings panel, containing Preferences for the current instance of Compere, and render settings for the appearance of each assembly panel (Canvas, Nodes and 3D Visualise).

➤ See [Settings Panel](#) <sup>41</sup>



## Network Configuration

### Project Group Roles: Server and Client

Every Juggler, Actor or Conjurer is running its own instance of Compere. Any device added to the network, running Compere, can also provide a user interface (UI) into the network and can join a project group.

- A project group comprises a system of devices running instances of Compere on a common network that operate together from a single Project file.
- The group has one server to lead the project group and one or more clients following.
- Any device can see project groups represented by other devices live on the same network.
- When a device joins a project group it retains this membership setting when switched off, or offline.
- If all members, clients and server leave a project group, the project group no longer exists.

#### Project Group Server

In this role, Compere can connect to any number of clients. A server shares its projects with clients and there can only be one server per project group. The server is automatically assigned.

#### Project Group Client

In this role, Compere 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.

*Example system:*

4 Actors as pre-rendered media servers, 1 Conjurer for generative content, 2 Jugglers handling a large canvas, a dedicated PC as project user interface, all in a single project group. The project group server role is preferably given to a device with least project load.

The project group **server** is automatically elected within the project group, based on a score that the user assigns to each device. If the server instance of Compere drops out, another will be reassigned to the role. (See [Server Failover Configuration](#)<sup>(40)</sup> below.)

The instance of Compere elected as server holds the Project file, and distributes it to all other instances of Compere in the project group. Every project group member then shares the same Project file. Any new device or instance of Compere joining a project group will receive the server's Project file.

All edits made to the project file by group members are sent back to the project group server to redistribute. Any edits made whilst a device is offline will be lost (but see [Save, Save As, and Export](#)<sup>(29)</sup>).

## Changing to another project group

If a device elected as server is moved into a different group, it does so as a client. This does not immediately close any project file it retains.

The device being moved will then adopt the project file of the server in a new group, displacing any project file it previously may have held as server.

If the new group has no server, the device that has moved may or may not be elected into the role according to its failover score as assigned in Network Configuration.

If the moving device was moved to a group without a server, and is then elected as server, it will distribute the project file it retained to the new project group of clients.

## Configure Devices in the Network

All instances of Compere on devices on the local network are listed under *Configuration > Network Configuration*, in a panel titled 'Network Discovery'. This includes Jugglers, Actors and Conjurers:

Hostname	Type	Version	Instance	Colour	Selected NIC	IP Address	Project Group	Role	Comms	Score
Actor_01	Actor	v1.0.7 B (39a533d)	4b0c95ba-f472-41...	Light Blue	Ethernet	10.100.106.15	Main Venue	C	OFF ON	8
Actor_02	Actor	v1.0.7 B (5a319f5)	f9c69e50-3566-4a...	Light Blue	Ethernet	10.100.106.16	Main Venue	S	OFF ON	10
Actor_03	Actor	v1.0.7 B (5a319f5)	56b4f46f-5d2c-4e...	Light Blue	Ethernet	10.100.106.10	Main Venue	C	OFF ON	9
Control_PC	Actor	v1.0.7 B (39a533d)	7035818e-c524-4...	Orange	Ethernet	10.100.106.11	Main Venue	C	OFF ON	Client only
JUG01-04	Juggler 2	v1.0.7 B (a14d1de)	50d428b2-5733-4...	Green	Ethernet	10.100.106.24	Main Venue	C	OFF ON	Client only
JUG01-05	Juggler 2	v1.0.7 B (a14d1de)	6eea20b7-2ee8-4...	Green	Ethernet	10.100.106.25	Main Venue	C	OFF ON	Client only

Your local instance of Compere has the green flash across it marked 'Local'. The list can be sorted by clicking on any column head.

Here, you select a NIC (expand the dropdown for details) and a project group (editable dropdown - type to add a new Project Group), set yourself online/offline and set your score for server failover.

### Hostname

This is the name that has been given to a device running an instance of Compere: Juggler units, control PC(s), Actors and Conjurers. Initially these will be the names assigned when the devices were built.

### Type

Compere instances have an instance type – determined by startup command line parameters – which is one of:

GUI, Juggler 1, Juggler 2, Actor, GUI (Actor), Daemon, or Unknown. The default is Unknown.

**Version**

Different versions of Compere might conflict within a project group. This column shows the version of each instance of Compere.

**Instance**

The UUID of the instance of each Compere running on this device, mainly useful for External Control.

**Colour**

User choice. Just used as a visual identifier for each instance of Compere.

**Selected NIC**

This dropdown selector shows all available NICs for the device. Jugglers will be defined and show the label, which cannot be changed.

**IP address**

Shows the current IP of each unit. Mouseover the coloured dot for a tooltip revealing DHCP/Static status (magenta = DHCP, blue- static IP).

**Note:** The *internal* IP range for all Juggler 1 devices is 172.30.n.n so do not use this range for your network of Jugglers and other devices. For Juggler 2 it is 127.0.0.255.

**Project Group**

Which devices share the Project file ('stage manager script')? Devices communicate by joining a project 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 by typing a new name into the field here, or select to join from listed groups.

**You must also belong to the same group as the devices with which you want to share a Project.**

**Role**

Shows cyan C icon for Client or magenta S icon for Server (elected). An adjacent yellow warning triangle shows that this project group currently has no server (and has so far been unable to elect a server), or that an individual device has a Compere version different from that of the Project Group server. All members of a Project Group should be running the same build version.

**Comms**

Toggle switch for offline or online, but stay within the project group.

**Score**

Assign a relative score for [Server Failover](#)<sup>(40)</sup> election. 0 = remain as client only, 10 = strongest candidate as server.

Although any instance of Compere can act as group server, we advise giving preferential voting scores to a separate control PC or to a Juggler because of the additional communications load.

## Server Failover Configuration

Every instance of Compere in a project group can be either online or offline. If online, each will be acting as a client or as a server. Network information is shown in the main panel [Status Bar](#)<sup>56</sup>, whose colour indicates the online/offline status:

**grey** = offline. Since it is disconnected from any group, any information may be residual from the last online state.

**blue** = online and in a project group that has a server.

**red** = this device should be online but there is a problem. You can mouse-over the warning triangle in the Status Bar or Network Discovery window to find out what the problem is (e.g.: 'Project group has no server', or not receiving peer data from one or more peers – the tooltip will show expected versus actual).

In the event of loss of server connection, clients continue to run with the project but no project file changes are exchanged until a server is re-elected.

## Server Election

Server failover is an automatic process for reassigning the project group server role in case of server failure. If a server failure is detected and persists, Compere holds an election for a replacement, i.e. a client automatically becomes the server. Election is on the basis of preassigned scores, where each device in a project group is assigned a relative score, with highest numbers given highest priority to take over.

Failover for a project group is configured from the Network Discovery dialog, by assigning relative preference scores. Highest score wins, in the event of a tie-breaker, the MAC address sequence is used.

Some of these controls are available in the [Status Bar](#)<sup>56</sup> for the local instance:



- Elections are held whenever something upsets the status quo of the system.
- There is a grace period at start-up of a system, to allow things to get up and running and scores to be compared.
- When there are no viable candidates (all score 0), an election does not run.

- The election score property is available to JSON external control. This means an election can be deliberately triggered, and even rigged if required.

The Status Bar shows you a list of counts of Compere instances (updated even when you are offline). Use this to detect missing instances in your Project Group and/or potential other GUI instances configuring your system.



## Manual Server Assignment

If you do not want automatic failover, just configure a single device with a non-zero score and the rest as 'Client only' (0). If that device fails, adjust the score of another device to be non-zero to trigger a manual failover (reassignment).

## Settings Panel

This non-docking panel is where local and system settings and preferences are stored. Panel settings are the display properties of the Compere assembly type panels themselves. This might be to see, or to set, how or from where the panel contents are viewed, what detail is included, or the zoom level. You can also choose whether to see the grid and text labels of objects or not, to declutter a complex view.

### Shared and local panel (render) settings

These settings are by default shared by all project users, so if one user changes the view, it changes for all observers. Adopting local panel settings enables the user to choose and keep an independent view of all the assembly type panels *for this particular project*.

Shared settings are stored with the project, local settings are stored in the local Compere root (Windows: C:\Users\[UserName]\AppData\Local\Compere\LocalRenderSettings).

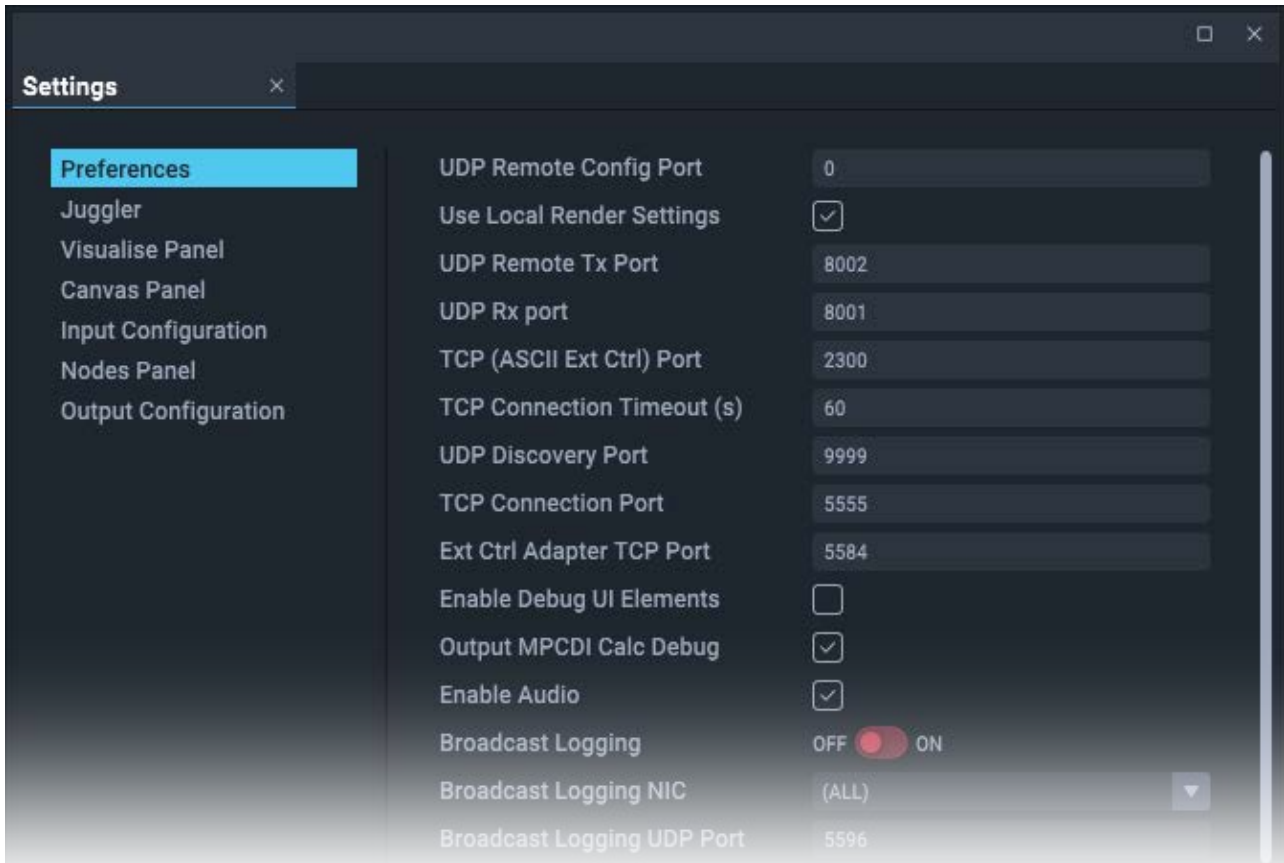
To use local render settings, first check the option in *Configuration > Settings > Preferences* and then assert your own settings. Uncheck to revert to shared render settings, and check again to pick up your local settings.

- [Preferences](#) <sup>42</sup>
- [Juggler Settings](#) <sup>46</sup>
- [Visualise Panel Settings](#) <sup>46</sup>
- [Canvas Panel Settings](#) <sup>49</sup>
- [Nodes Panel Settings](#) <sup>50</sup>

➤ [Input / Output Configuration](#) <sup>51</sup>

## Preferences

Menu: *Configuration > Settings > Preferences*



This extensive list covers many areas of Compere operations.

### UDP Remote Config Port

Remote discovery port: default 9103, configurable.

### Use Local Render Settings

Compere allows users to retain their own settings for how assembly panels are rendered, identified by the project, and stored in their Compere root folder, under `\LocalRenderSettings`.

### UDP Remote Tx Port

UDP port number to which (JSON) external control will transmit.

### UDP Rx Port

UDP port number the (JSON) external control socket will listen on.

**TCP (ASCII Ext Ctrl) Port**

TCP port number that the ASCII external control socket will listen on.

**TCP connection timeout (s)**

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

**Network binding NIC**

Shows the network connector of the current instance of Compere.

**UDP discovery port**

UDP port used when discovering other Compere applications on the network (default 9999).

**TCP connection port**

TCP port on which connections will be made between instances of Compere for Project updates (default 5555).

**Ext Ctrl Adapter TCP Port**

TCP port to use for external control communication (default 5584).

**Locked**

Check to avoid accidental changes in the panel.

**Enable Debug UI Elements**

Implements an additional Debug Menu for support diagnostics.

**Enable MPCDI Calc Debug**

Enables a diagnostic debug feature for users of MPCDI.

**Broadcast Logging**

Enable/disable ...

**Broadcast Logging NIC**

Select a NIC on which to broadcast logging data.

**Broadcast Logging UDP Port**

Set the port for broadcast logging (default 5596)

**Asset Logistics Watch Folder**

Folder receiving distributed assets.

**Asset Logistics Vault Folder**

Folder into which assets are ingested.

**Asset Logistics IP Address**

Local IP address of server receiving distributed assets.

**Asset Vault Location**

Choose between Local and NAS as the source for Actor asset ingest.

See [M860-managing-assets-in-compere](#).

**Lock to VSync**

Synchronise to display refresh rate.

**Display Actor Rendering Stats**

Show or hide GPU performance and sync displays on each output.

**Layer Opacity Indicators**

Set to 'On' to display the opacity levels for visual layers, in the layer name section of the Actor timeline panel.

**ST 2110 Card IP Address**

IP to connect to, for ST 2110, e.g. a BlueField/Rivermax card.

**ST 2110 Timing**

Switch on to use PTP (Precision Timing Protocol)

**ST 2110 Frame Rate (Fraction)**

For ST 2110 systems, enter the non-drop frame rate per second (e.g. 60 in the first field and 1 in the second), or for drop-frame timing, use the second field as a fraction (e.g. 60000 over 1001 for 59.94 fps).

**ST 2110 Packet Burst**

Controls burst packet duration size. Typically in the order of 20  $\mu$ S and 40  $\mu$ S, adjusted for timing requirements.

**ST 2110 Output Default SDP**

Enter the complete path and filename for the required SDP file

**External Control Path Format**

Legacy (ASCII/XML) or JSON. Default is for legacy ASCII to be disabled to protect the system from accidental usage of the ASCII interface.

**Show Direct Addresses**

Display direct addresses using UUIDs rather than paths to project objects, when using .

*Example:*

```
"path": "4e017f49-adc2-46a1-a0d1-ae92ed207443.enabled"
```

instead of:

```
"path": "tree.children[5].ResourceSet.children[2]. juggler.children[1].enabled"
```



**Enable Renderer NatNet Rx**

Enable instances of Compere to receive NatNet tracking data directly rather than by distribution over the network, in order to reduce latency.

**Enable waitForPresent**

Use the Nvidia-specific extension ("waitForPresent") to make Compere wait until a frame image is actually displayed.

**Framebuffer Format**

Actor can render at up to 16-bit, but if content is not provided at, required at, or deliverable to 16-bit, optimise processing by selecting the appropriate RGBA format.

**Playback Background R**

Background red colour element where the timeline playback is not occupied by video pixels.

**Playback Background G**

Background green colour element where the timeline playback is not occupied by video pixels.

**Playback Background B**

Background blue colour element where the timeline playback is not occupied by video pixels.

**Playback Background A**

Background transparency of the above colours where the timeline playback is not occupied by video pixels.

**Timecode Format**

Select from Microseconds, SMPTE, ISO8601 or Frame Count. This is the format presented throughout the Compere UI time properties.

**Note** that drop frame rates (23.976, 29.97, 56.94 fps) are expressed using semicolons (00;00;00;00) whilst non-drop expressions use the usual colons (00:00:00:00).

**Confirm Close**

This will raise a confirmatory dialog if switched on.

**Collated System Reports Location**

Leave empty unless you wish to direct System Reports to a specific alternative location. See [Logs and Diagnostics](#) <sup>(148)</sup>.

The default location for media servers is C:\Users\[name]\AppData\Local\compere\System Reports.

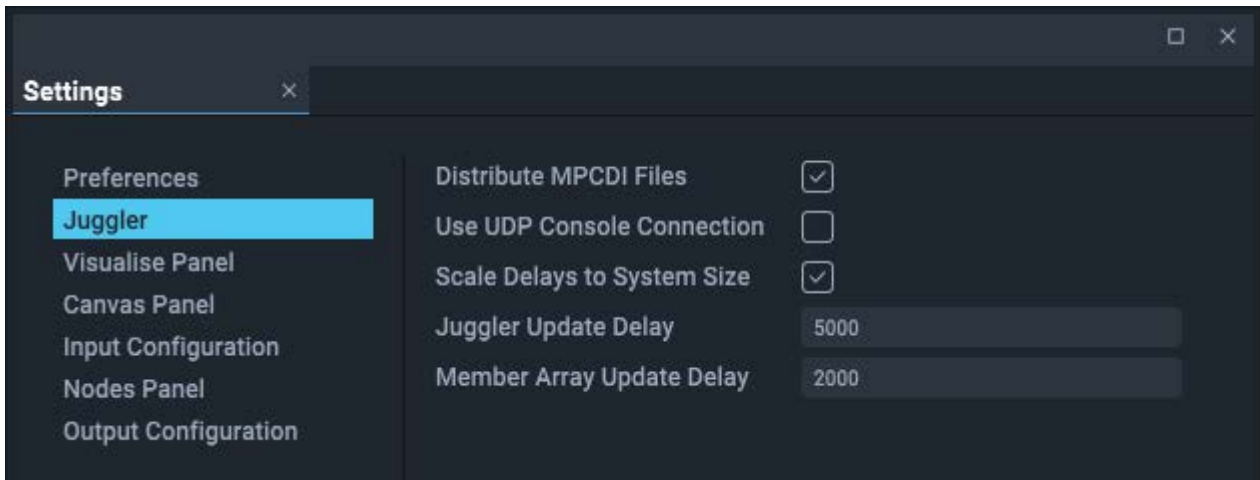
For Juggler 2 it is root@[JugglerName]:/7thApps/compere/bin/SystemReports#

An alternative file location setting can be set in *Configuration > Settings > Preferences*.

➤ See also: [Ports Used by Juggler and Compere](#) <sup>(151)</sup>

## Juggler Settings

Menu: *Configuration > Settings > Juggler*



### Distribute MPCDI Files

uncheck to prevent distribution of .mpcdi compressed source files by default (see note in [Display Setup via MPCDI](#))<sup>(137)</sup>. Content extracted from the file will be distributed, just not the original single source.

### Use UDP Console Connection

Juggler Console is by default TCP on port 9013. Check this to use a UDP connection instead.

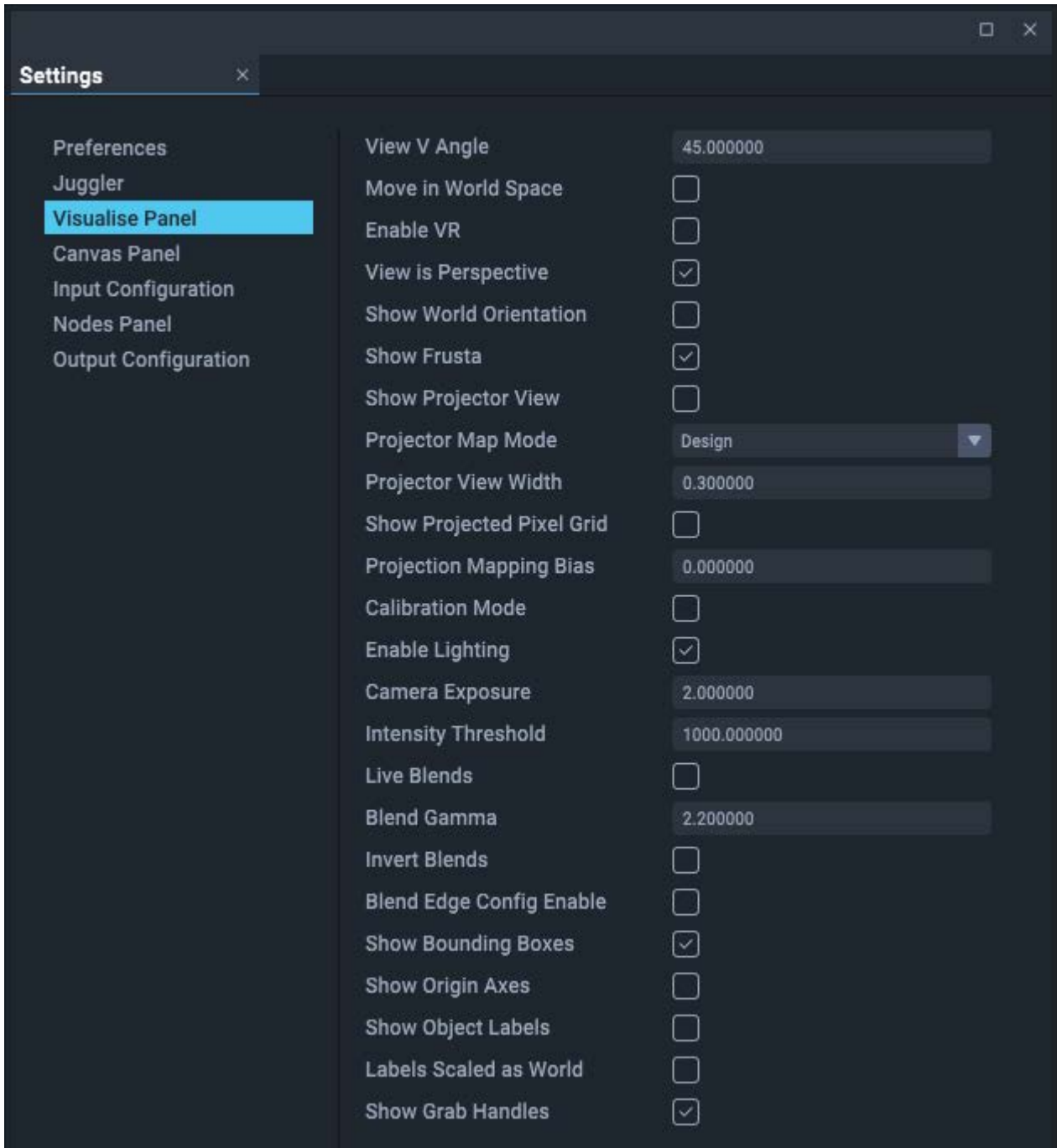
### Scale Delays to System Size

When creating an [Output Canvas Bus](#)<sup>(103)</sup> using the wizard, it takes a few seconds to propagate all the settings. This increases with the size of the Juggler system. For large systems, scaling the delay introduces a more appropriate wait time.

## Visualise Panel Settings

Menu: *Configuration > Settings > Visualise Panel*

These settings relate to Compere Visualise, Compere's 3D theatre view, which is not covered in this guide.



**View V Angle (degrees)**

Field of view on the scene. Fit more into view without getting closer and changing the perspective. The default is 45 degrees. This is equivalent to zooming the view of the scene rather than zooming within the scene.

**Move in World Space**

move to axes of 3D scene, not of a selected object

**Enable VR**

Used for Virtual Reality headsets

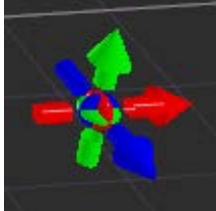
**View is Perspective (chk)**

Switch between perspective and orthographic views

**Show World Orientation**

displays interactive world axes (click the icons to align view with an axis)

X = red, Y = green, Z = blue

**Show Frusta**

show projector frusta

**Show Projector View**

enables display, in a smaller area of the scene, of what is in scope of a projector's lens, as seen from the viewpoint of the lens. Interacts with the Visualise toolbar button that also toggles view for a selected projector.

**Projector Map Mode**

Select from the dropdown list, this is also selectable from the Visualise panel toolbar mode button.

**Projector View Width**

Scales the Projector View display, 0 (smallest) to 1 (largest, fills 95% of the scene width), default is 0.3. Holds the aspect ratio of the projector. This view scales with the scene window.

**Show Projected Pixel Grid**

Shows a black border around every pixel. This works by reducing the footprint of each pixel, to give a better idea of resolution on a real-life surface.

**Projector Mapping Bias**

Adjusts light intensity for all projectors.

**Calibration Mode**

For real-life adjustment of vertex points. Select a vertex on a mesh model in 3D view, then switch to cal mode, observe the real life point and move around on the green square to position the point accurately.

**Enable Lighting**

Enable or disable the artificial background, ambient and directional lighting in the scene.

**Camera Exposure**

Overall exposure control higher value dims **when working in Design mode**.

**Intensity Threshold**

For use in illumination threshold mode.

**Live Blends**

Dynamically create autoblend files. Can be observed as objects move in relation to each other.

**Blend Gamma**

Adjust projector response curves.

**Invert Blends**

Switch 'black to white' to 'white to black'.

**Blend Edge Config Enable**

Enables/disables manual configuration of blend edges.

**Show Bounding Boxes**

enable the display of a box around objects. Interacts with the Visualise toolbar button that also toggles box display for a selected object

**Show Origin Axes**

display axis lines for all objects, bright for +ve, dim for -ve

**Show Object Labels**

display names of objects in the scene

**Labels Scaled as World**

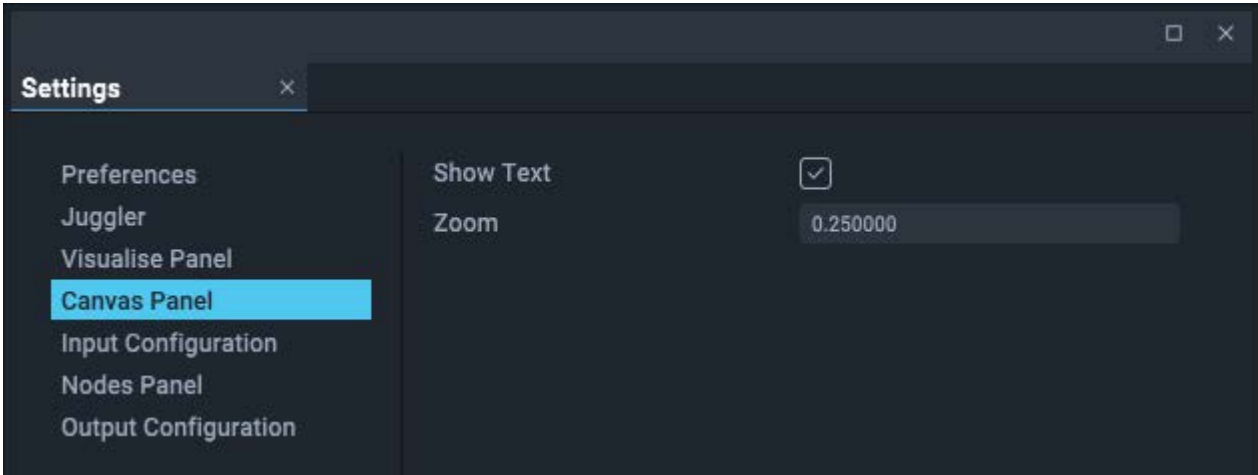
text of object labels zooms with view

**Show Grab Handles**

display the gimbal with grab handles when an object is selected

**Canvas Panel Settings**

Panel settings determine the preferred appearance and/or behaviours when opened in the local instance of Compere.



**Show Text**

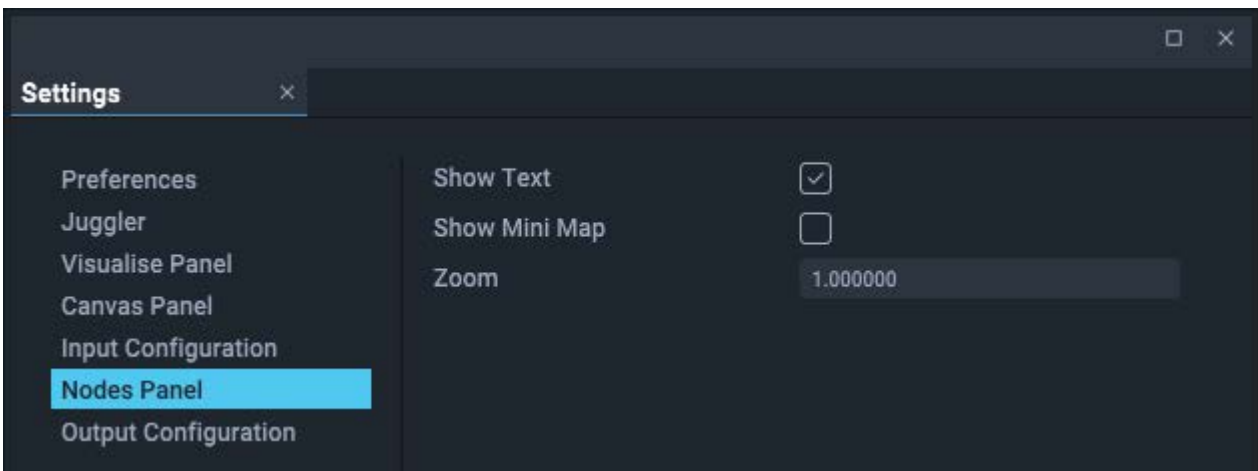
Show or hide the text names of all Canvas objects

**Zoom**

Shows the zoom ratio. Editable.

**Nodes Panel Settings**

Panel settings determine the preferred appearance and/or behaviours when opened in the local instance of Compere.



**Show Text**

Show or hide the text names of all Canvas objects

**Zoom**

Shows the zoom ratio. Editable.

**Show Mini Map**

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:

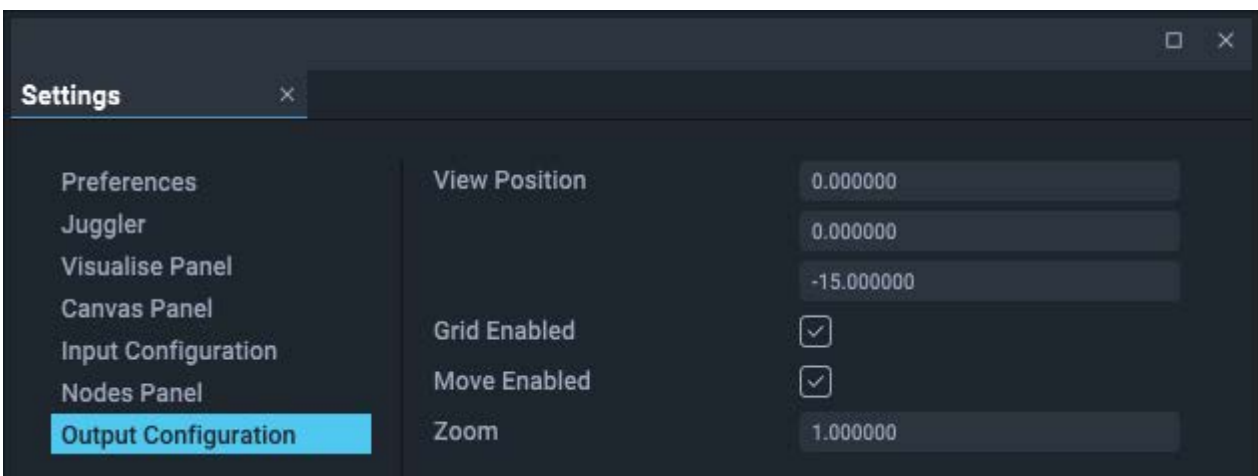


The orange box contains all objects. The lighter area is the current view scope and can be dragged around to view the required Juggler(s) without having to zoom out or in to find your bearings.

**Input / Output Configuration**

Menu: *Configuration > Settings > Input Configuration / Output Configuration*

Settings for these two panels relate to applying timeline media to outputs. This is a function of the Actor role.



➤ See the [Actor User Guide](#)



## Wizards Menu

---

### I/O Feed Wizard

This wizard provides a quick and efficient setup for large systems with regular and simple display configurations, based on rows and columns of output feeds.

➤ See the [Actor User Guide](#)

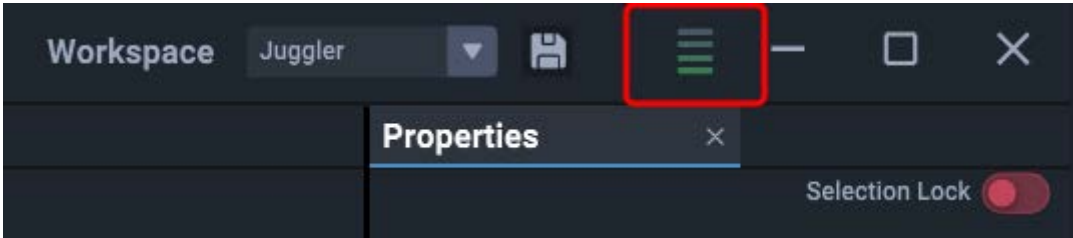
### Output Canvas Bus Wizard

The step-through assistant for creating all the essential settings for an Output Canvas Bus system.

➤ See [OCB Wizard](#)<sup>(109)</sup>

## 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 side of the main menu bar. This will only appear when tasks are in progress.



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'.

## Status Bar

The status bar has three colour states indicating network status:

**grey** : offline (will be a client). Since it is disconnected, any information may be residual from the last online state.

**blue** : online and in a project group that has a server.

**red** : you should be online, but there is a problem. A warning triangle on the right will also show in the status bar (and the Network Discovery panel): hover over it to see what the problem is (e.g.: 'Project group has no server', or not receiving peer data from one or more peers – the tooltip will show expected versus actual).

### Device status

The right hand side of the status bar carries information about the current instance of Compere on the network.



Here, the device can be switched to offline, and by doing so will make the device a client and the status bar will go grey. The device remains in the project group, but will not receive or share any project file updates. Switching back on, the device will join as a client and receive the current project file.

### Group members

Project group peer information is on the left side of the status bar:



This shows other devices by type.

### Alerts

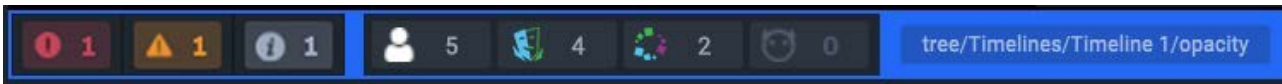
On the left side of the status bar, alerts, warnings and information can be found:



Hover over an alert to see its detail.

## Property paths

Property paths will be displayed to the right of status information when any properties panel label is hovered over.



Paths are an essential part of writing external controls. These can become quite long, and will be truncated (...) from the middle so that stem and final segment are identifiable.

## Compere Projects

---

### There is always a Project

Once Compere runs, on a control PC or as a Juggler, Actor or Conjurer, 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 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

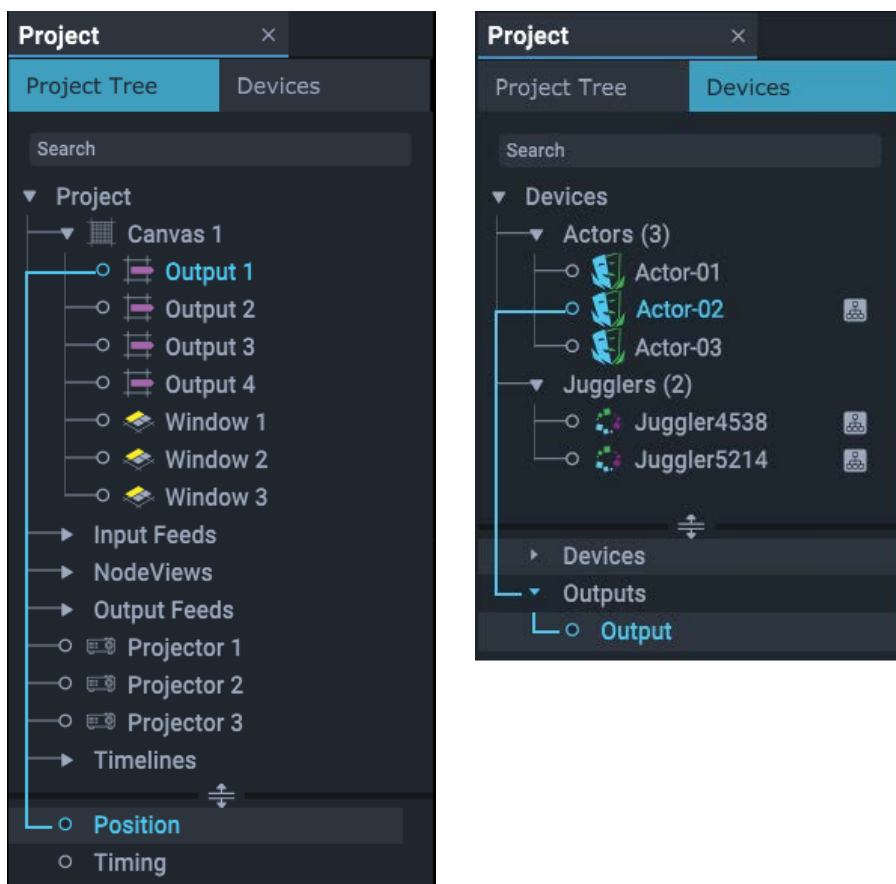
Distributed files reside on all instances of Compere on the network (this includes every Juggler and control PC), so that every one has its own directions, and any one can be assigned the role of server. The server coordinates the distribution of the current Project to all clients, receiving edits from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

## Project Panel

The Project panel on the left lists everything that takes part in a Project. There are two tabs, the Project Tree and the hardware Devices (Jugglers, Media Servers etc.) that run the Project.

Many items will have elements with their own properties, for example a Juggler can be expanded into its channels, inputs and output ports, and a Juggler Canvas can be expanded into its Canvas Outputs and Canvas Windows. When any branch of the Project is selected, it is highlighted blue and its top-level properties are shown in the Properties panel. All the internal parts of the selected object will be expanded below the sliding divider, from where they too can be selected for their properties.



Top-level clusters (e.g. all Canvases in 'Project' or all Actors under 'Devices') and the items in them, are arranged alphabetically, so they can be re-ordered by editing the names (above by adding numbers before Juggler names). The name is a property of any object.

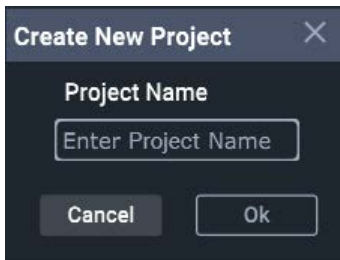
Multiple objects can be selected in the usual way using Shift or Ctrl, and the properties panel will only show shared properties of common values. Unshared properties will be greyed out and unshared values will show ellipsis (...). However, other operations can be performed on multiple objects at once.



## Project Properties: Name

The Name property is the only editable property of a Project and represents how the user knows the Project. This 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:



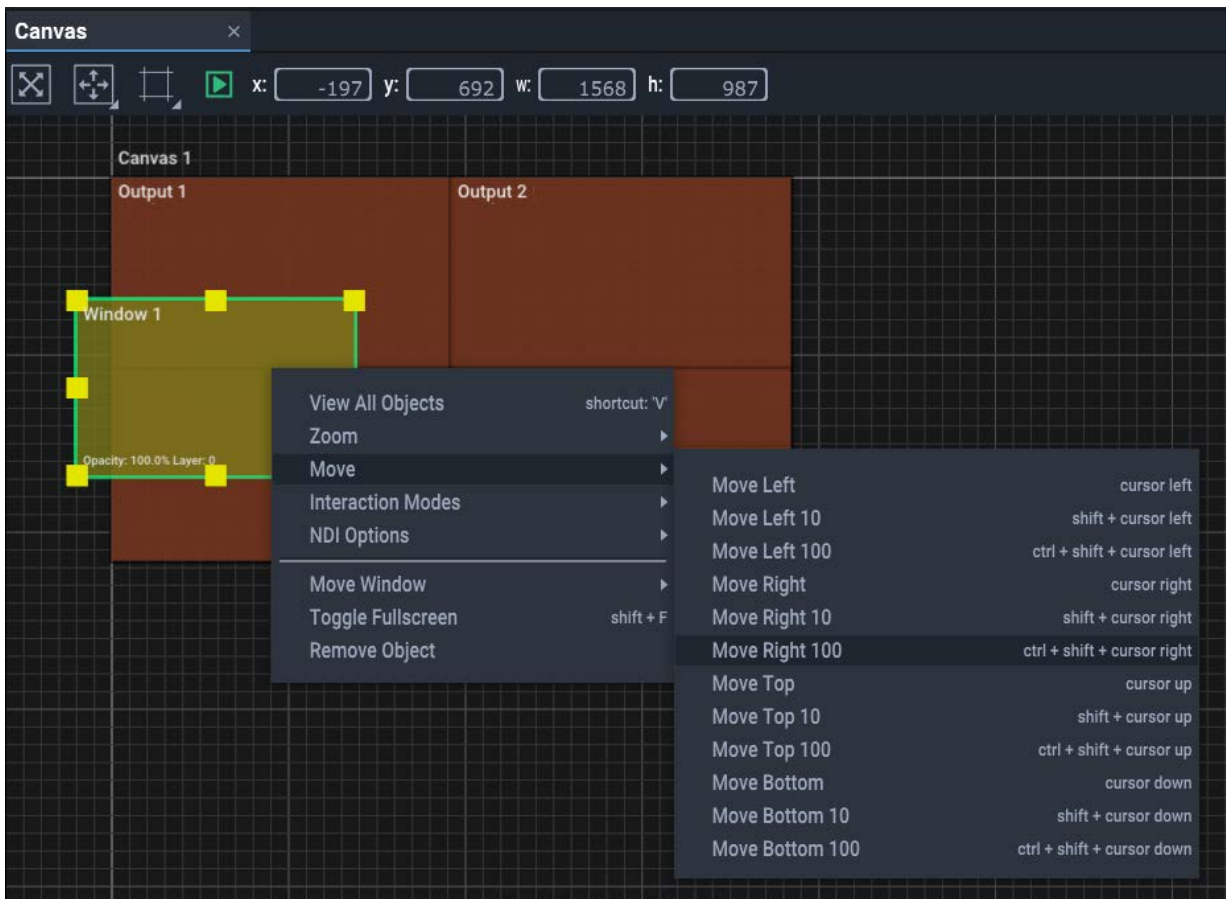
This will then show as the Project name in the Project panel.

The name can be subsequently edited in the Properties panel of the Project.

## Assembly Panel Context Menus

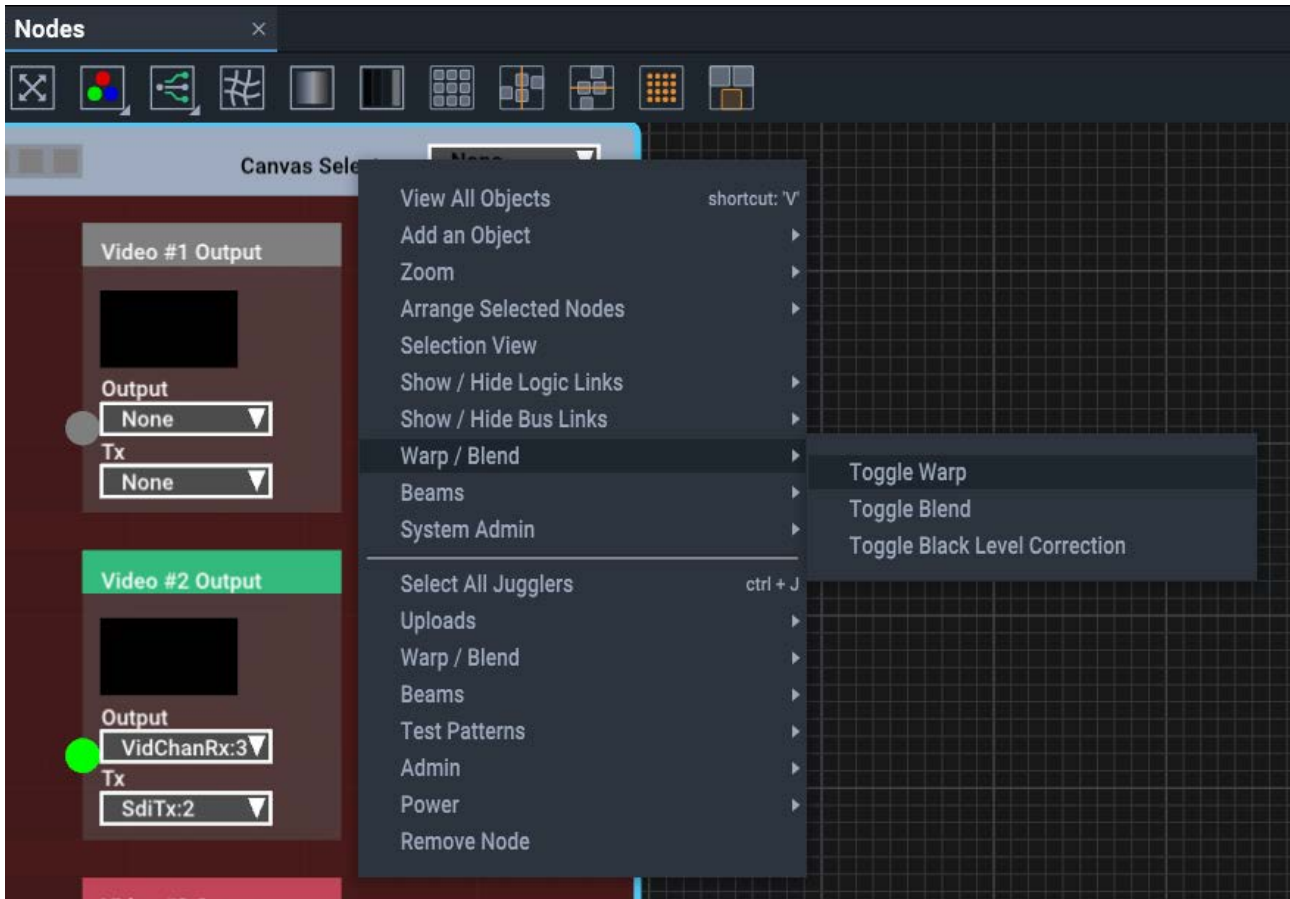
Right-clicking on an assembly type panel background, or on a resource in such a panel, will open a contextual menu, many with shortcut hints. The object selected will change its border to green.

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. Context menus are also sensitive to which part of an item you right-click. Here is another example – Right-click a selected Juggler in the Nodes panel:



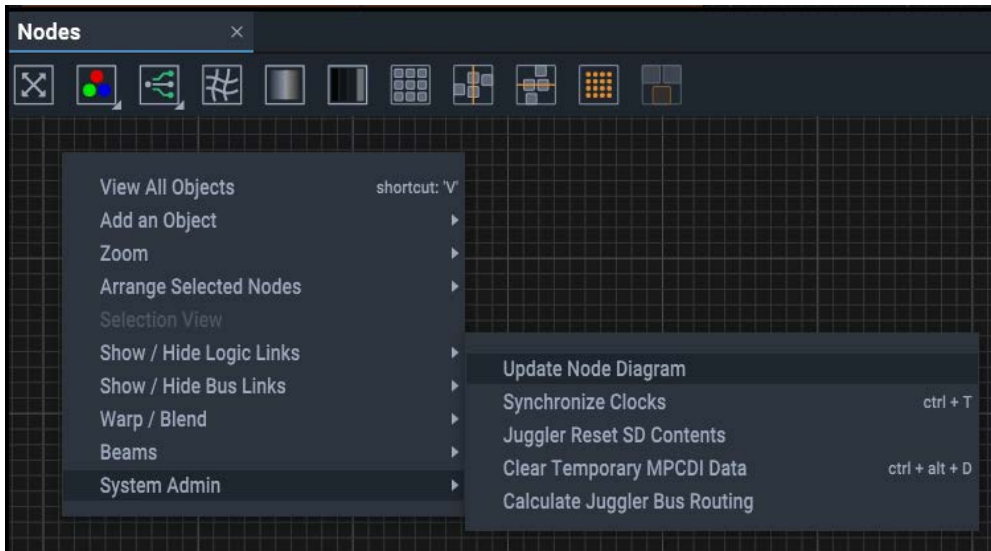
Some context menu items repeat the toolbar button actions. When the menu is divided, the upper section applies to **all** Jugglers in the assembly panel, and the lower to the **selected** Juggler only.

All objects (nodes) and the Nodes Panel itself, have a context menu.

## System Admin Menu

---

The Nodes assembly panel context menu has a system admin option. This is more for troubleshooting use.



### Synchronize Clocks

Reset Juggler real-time clocks to that of the Control PC.

### Juggler Reset SD Contents (Juggler 1)

The SD card inside a Juggler 1 holds essential configuration files, including EDID, MPCDI, warp, blend and colour correction files. Reset will remove all files added to the SD card since initial configuration.

### Clear Temporary MPCDI Data

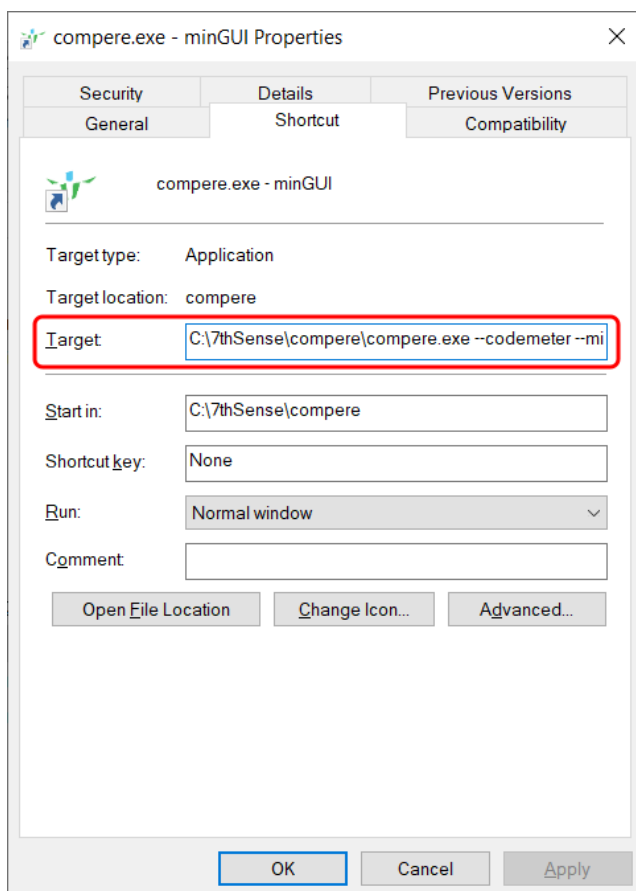
Large files can accumulate in the upload process of zip files. This will just clear these out.

## Compere MinGUI

Running Compere with the full graphical user interface and controls is CPU intensive, and there is no need beyond a control PC. Devices in a Compere project, whether Juggler, Actor, Conjurer, run Compere without the full UI. A reduced UI runs on a separate internal processor to provide the front panel display for information and touch controls.

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

`C:\7thSense\compere\compere.exe --codemeter --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](#) <sup>9</sup>

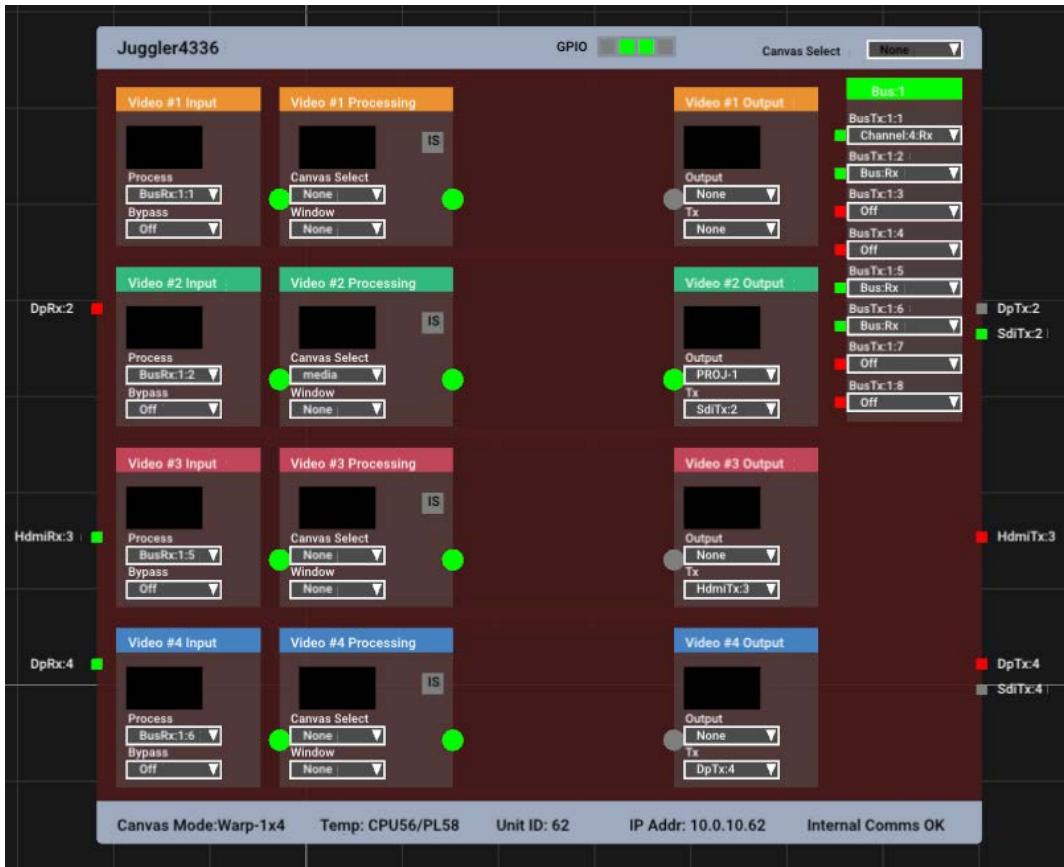
## Connecting Juggler Systems

---

Large Compere systems will most likely make use of Juggler pixel processors, to achieve very large canvases. This section relates to connecting Jugglers together for the Juggler Canvas. With Jugglers connected, media sources, including Actors are brought into the same project group and configured. For this, see the [Actor User Guide](#).

Each Juggler in a Project has its build configuration of available ports recognised by Compere, and this is exposed in the Properties and Project [Devices] panels. When a Juggler is added from the Project [Devices] panel into a [Nodes](#)<sup>66</sup> 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 (Actors, Conjurers, media capture etc.) 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. This is a typical Juggler 1 node:



Selected parts of a Juggler note are bordered blue:



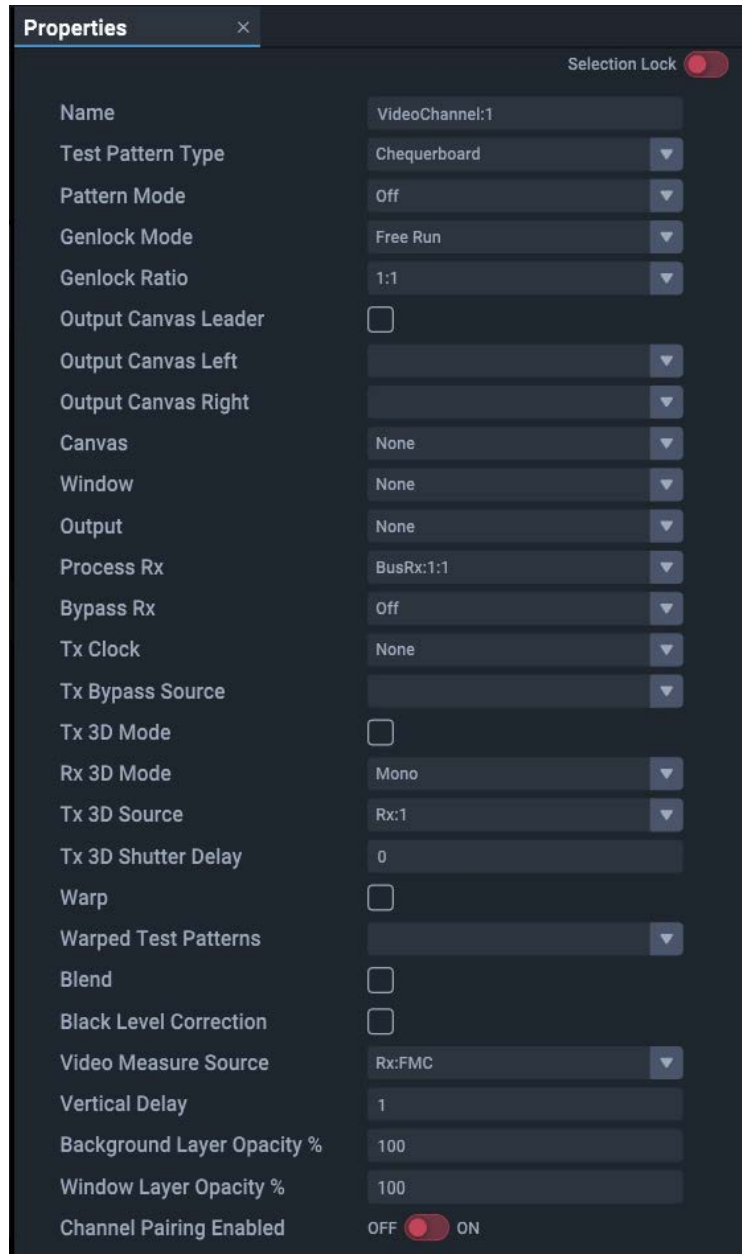
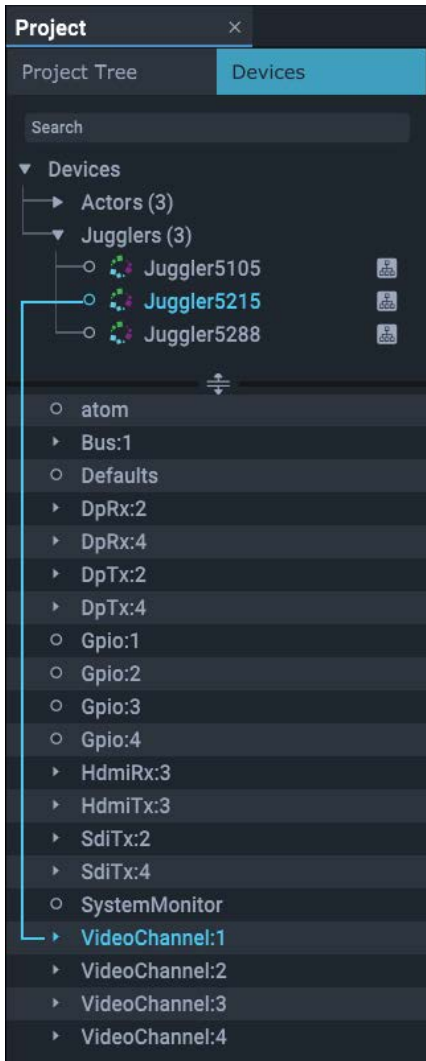
The square pins on the left and right margins represent the physical input and output connectors. When a signal is present they are green, and when disconnected they are red.

An unexpected red pin is a good indicator of a bad physical connector or cable.

## Properties

All Jugglers in the network are visible together in the Project [Devices] panel. An icon against a device 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 (left, below – highlights in blue) to examine and edit its properties in the Properties panel (right, below).



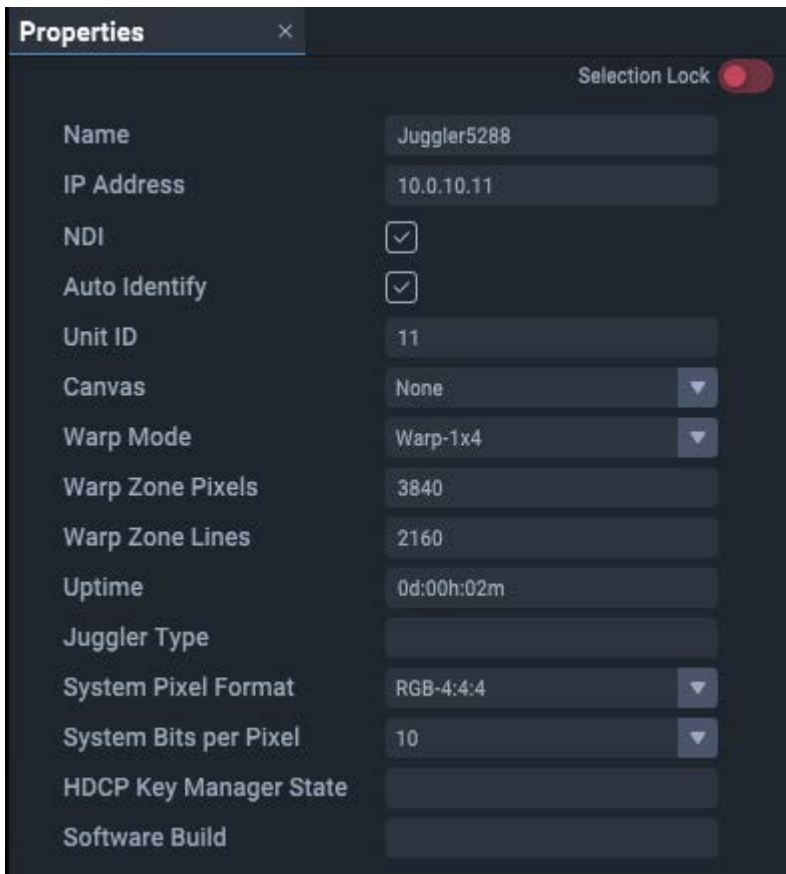


The Nodes assembly panel is where representations of Jugglers and other parts of the system can be connected: to each other, to outputs, to other devices, and to processes. Here you decide how the receive (Rx, input) ports of Jugglers – whichever type is present – accept media sources, transform them,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

## Juggler Unit Properties

In the Project panel, under 'Devices', select the part of a Juggler for which you want to see the properties. Some of these properties relate to the unit as a whole.

Select an individual Juggler name in the Project panel for its module level properties.



Note the **Selection lock** top right. Since you can open multiple Properties panels simultaneously it can be useful to retain sight of properties for one element whilst editing those of another. Switch the lock on to keep these properties active while looking at another set of properties.

### Name

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

### IP Address

This unit's IPv4 Address (see [Network Configuration](#)<sup>37</sup>).

### NDI

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

**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** (warp and blend firmware only)

Depending on the firmware of the Juggler, the unit may be configured for warp and blend rather than flat display 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.

**Warp Zone Pixels / Warp Zone Lines** (warp and blend firmware 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.

**Warped Test Patterns** (warp and blend firmware only)

Select from: Grid, None, X-Ramp, Y-Ramp

**Uptime**

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

**System Pixel Format**

Select as appropriate, e.g. RGB 4:4:4.

**System Bits Per Pixel**

Select as appropriate 8- or 10-bit.

**HDCP Key Manager State**

For users of HDCP (content protection). See also [Video Interface Properties](#) <sup>92</sup>.

Values:

Initialising

Read Key File

Failed to Read Key File

Key File Decrypted

Failed To Decrypt Key File  
 Keys Written to Flash  
 Failed to write keys to flash  
 Read Keys From Flash  
 Failed to read keys from flash  
 Ready  
 Disabled

## Properties of the component parts of a Juggler

- [Internal Processor](#) <sup>72</sup>
- [Defaults](#) <sup>73</sup>
- [GPIO](#) <sup>73</sup>
- [System Monitor](#) <sup>74</sup>

Properties of the connections and channels are described in context.

- [Video Channels Properties](#) <sup>82</sup>
- [Video Interface Properties](#) <sup>92</sup>
- [Bus Properties](#) <sup>96</sup>

You can select the same feature in more than one Juggler, and edit shared properties as one. See [Project Panel](#) <sup>60</sup>.

## Internal Processor

This is the single-board computer (SBC) in each Juggler that runs a reduced Linux version of Compere, and communicates with the main FPGA board. The properties listed under this are largely for reference only, including software and firmware configuration versions installed on the Juggler unit.

### Name

Internal Processor

### Juggler Connected

Should be checked.

The other properties are for reference:

- Juggler low level SW version
- Juggler FW version
- DisplayPort version (for DP configured units)

- HDMI version (for HDMI configured units)
- SDI version (for SDI configured units)
- Bus version
- Capabilities (e.g Warp & Blend; Scaling/Windowing)
- Compere version
- PIC version
- Front panel default (Info, Rx video input channels, Tx video output channels)

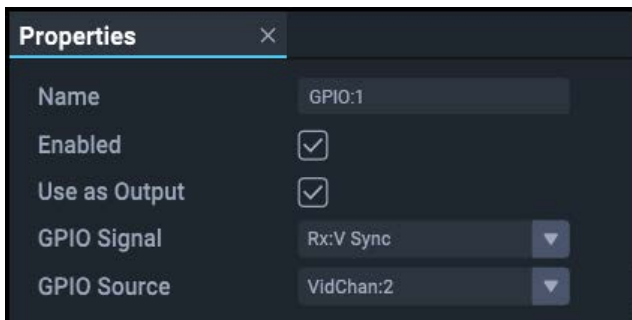
## Defaults

Default properties relate to Juggler properties held not in the Project, but in the Juggler module’s firmware on its internal SD card. They include internal IP and MAC addresses, port numbers, the Tx/Rx EDID on each connector, and pixel formats.

The connectors shown for a Juggler node are detected by Compere, and include DisplayPort, HDMI and ST 2110.

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

## GPIO



### Name

Identifies the physical GPIO.

### Enabled

For GPIO ports to be live, they must be enabled. If you really 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 Signal** (when used as an output)

Select an available signal from the selector. Example: an input (Rx) as a sync 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.

## System Monitor

Includes internal temperature monitoring of a Juggler module.

**Name**

There is no need to edit this.

**CPU Temp**

Shows the current temperature of the internal Linux processor.

**FPGA Temp**

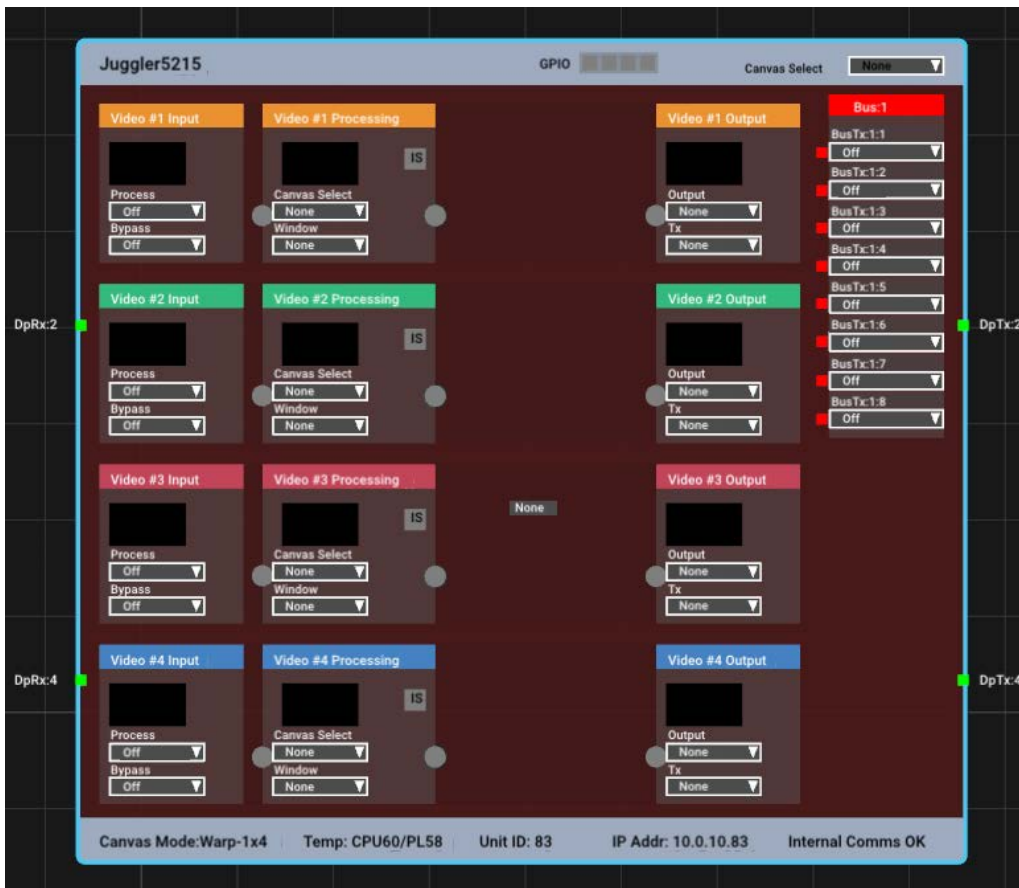
Shows the current temperature of the FPGA processor.

**Temperatures** are also shown in each Juggler status bar in the Nodes panel, and on the front panel display of each unit. The text will turn orange and then red as temperatures rise to levels requiring attention. Check Juggler filters regularly to optimise cooling fan performance.

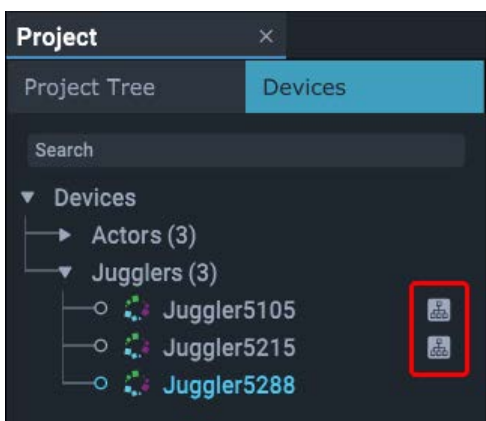
## Adding Jugglers as Nodes

When Jugglers are added into the network and assigned to project groups, they are recognised and added into the Project running for that project group, becoming visible in the Project panel.

- Drag a Juggler or Jugglers from the Project panel into a 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.



- These Jugglers are now nodes. A Juggler that has been added to the Nodes panel has an indicator against it in the Project panel:



## Naming and arrangement

With many Jugglers in various roles, you are probably working to a diagram of how they are connected and their functions. It is useful to name and arrange your Jugglers in the Nodes panel (and/or use the alignment tools – see below) to best represent your diagram.

## Juggler 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

## Juggler 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.

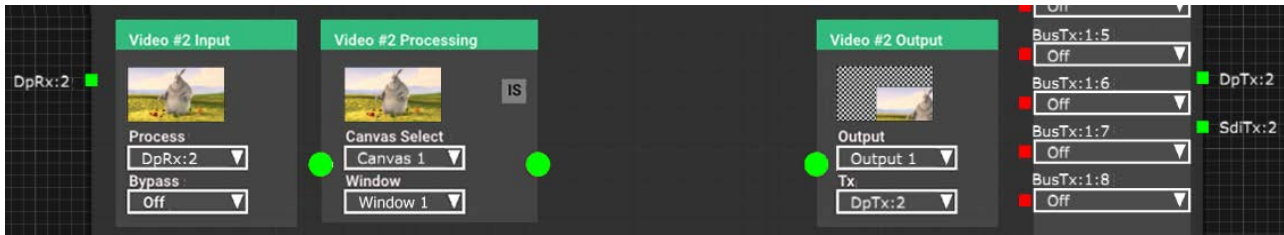
### Internal Comms

Indicates the status of communication within the Juggler module between the Linux processor running Compere and the main board processing the pixels.

## The four Juggler video 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.

➤ See [Video Channels Properties](#) <sup>(82)</sup>

## Video Input

- This block accepts either the input to this video channel, and/or an input from the bus.
- With NDI enabled, the input video for the channel will appear here.
- It can connect an input directly to a matched output, **bypassing** any further processing, or it can **process** an input in order to position it on the output canvas (i.e. in a named canvas window).
- An adjacent input pin does not have to be handled by this video channel. It could instead be **selected in the bus** for this Juggler, in order to pass it on to a different Juggler to handle (process to a window, or bypass to a matched output).

**Process:** this selector chooses the source for this channel to process. Select 'Off' if nothing to process. In the example shown here, DpRx : 2 has been selected as an input to process. 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.

**Bypass:** media can, if required, bypass all processing and go straight to the same-numbered channel output. Select 'Off' if nothing to bypass. This direct routing reduces bandwidth if no processing is required. A warning is given if the resolutions do not match and the pin will be yellow.

You can bypass one input and simultaneously process another in the same video channel.

## Video Processing

If you *process* an input, you are locating it in a window on a canvas.

**Canvas Select:** select a named Canvas within which this channel will be displayed. The list is populated from Canvases added to the Project, named and placed in the Canvas assembly panel. This can be set from the title bar to apply to 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: [Connecting Inputs and Outputs](#) <sup>113</sup>

## 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. A yellow pin indicates an incompatibility.


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, and their colour, together with the status bar indicator. the background colour of the Juggler node and of its name in the project tree, is diagnostic:

### Summary:

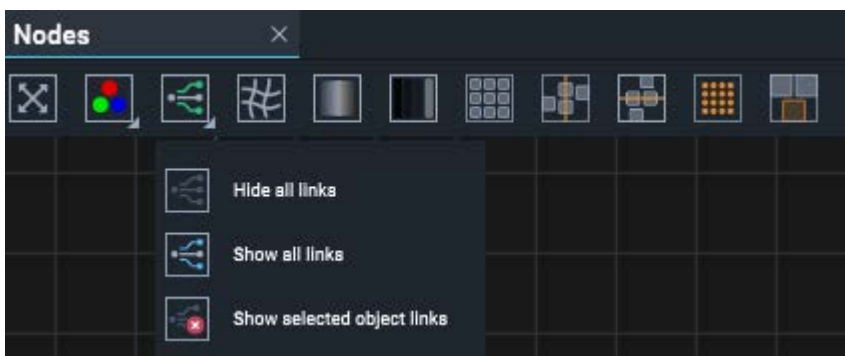
- The [status bar](#) <sup>56</sup> **internal comms indicator** for a Juggler in the node view is **green – 'OK'** when Compere and the Juggler are communication correctly.
- The status bar **internal comms indicator** for a Juggler in the node view is **red – 'Internal Comms Fault'** when Compere and the Juggler are not communicating. Reboot the Juggler to correct this.
- The status bar **internal comms indicator** for a Juggler in the node view displays the last known indicator state if Compere is not running on the Juggler.
- The **Juggler name** in the project tree is **dark grey** if Compere on the Juggler is not connected to Juggler SW

- The **background colour** of a Juggler in the node view is **red** if the Juggler is not present in the current [failover peer list](#) <sup>(40)</sup>
- The **background colour** of a Juggler in the node view is **grey** when the Juggler is present in the current failover peer list
- The **Juggler name** in the project tree is dark grey if the Juggler is not present in the current failover peer list
- The Juggler name in the project tree is **light grey** if the Juggler is present in the current failover peer list and Compere on the Juggler is connected to Juggler SW



You may later need to link, address and see other dynamic properties of a Juggler or a Juggler channel, as pins. These can be added via a shortcut menu: Shift+Right click in context and select the required item from the list.

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:



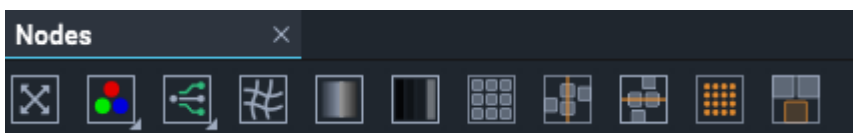
'Show selected object links', for example, will show links to, from and inside a selected Juggler only.












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.
-  Mini Map: Show a miniature map of all nodes and the position of the current view.
-  Selection 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.

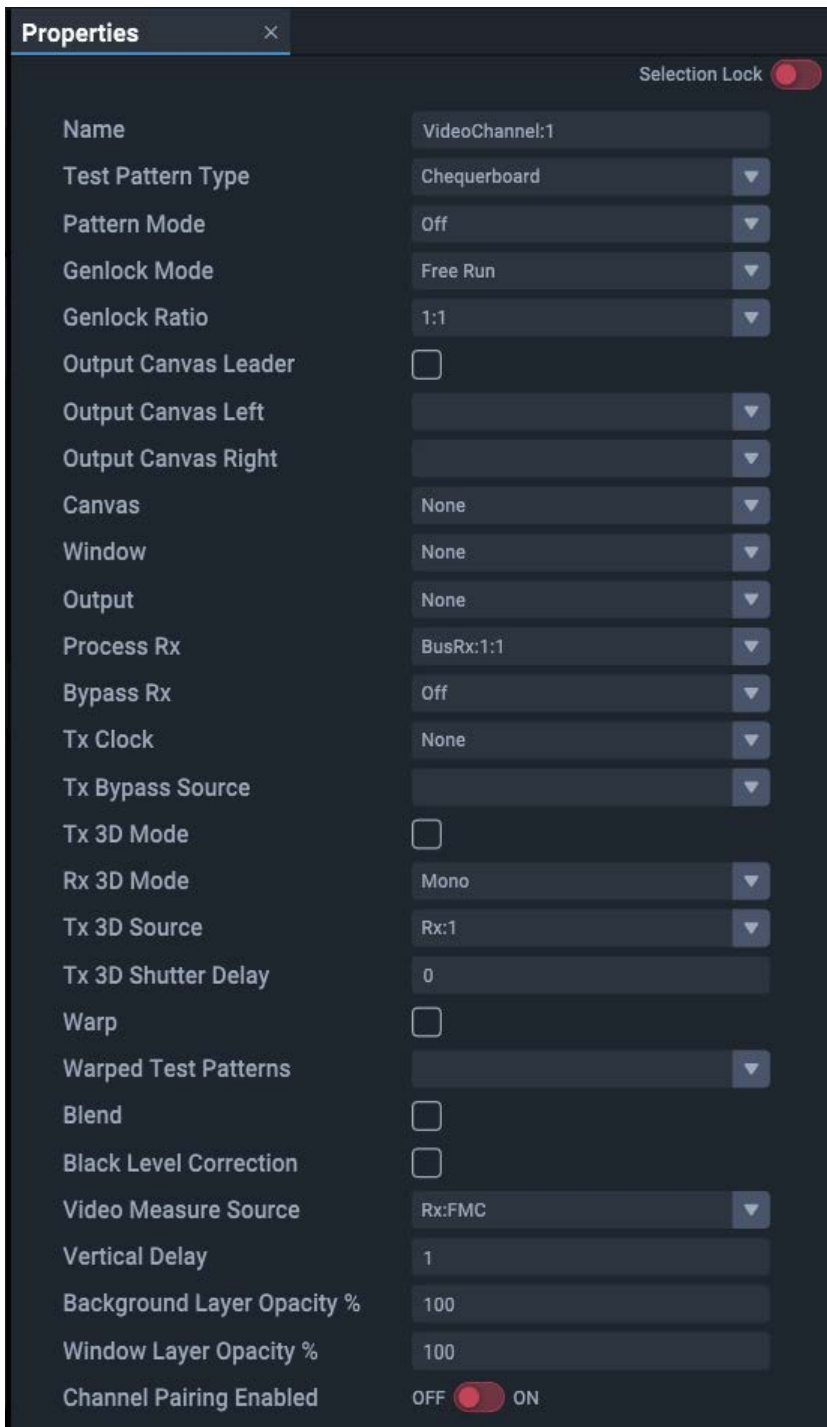


Wider bus options with 12 and 16 lanes are being developed.

For how the bus connections correspond to the physical cabling, see [Bus Connections in a Juggler System](#)<sup>(93)</sup>.

## Video Channels Properties

Video channels available per Juggler depend on its configuration. Select each video channel branch for its properties.



### Name

The default name of video channels is VideoChannel:#

**Test Pattern Type**

Select a test pattern to display (None; 4 quadrant colour Y ramp; Black; Chequerboard; Colour Chequerboard; Flat; Grid; X ramp; Y ramp).

**Pattern Mode**

Select On, Off, or Mix to overlay the pattern on media.

**Genlock Mode**

Free run or lock to a source (for an example used in Juggler 1, see: [GPIO Genlock Example](#)).

**Genlock Ratio**

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

**Output Canvas Leader**

Applies to [Output Canvas Bus](#)<sup>103</sup> mode only. Check for this Juggler to adopt the lead *for this channel* (i.e. send this video channel to a display device).

**Output Canvas Left** (if Leader for this channel)

Applies to Output Canvas Bus mode only. Select the Tx pin on this Juggler to carry the left-side part of the output pair.

**Output Canvas Right** (if Leader for this channel)

Applies to Output Canvas Bus mode only. Select the Tx pin on this Juggler to carry the right-side part of the output pair.

**Canvas**

Name of the canvas containing the output that this channel feeds.

**Window**

Name of a window on this canvas to which media is sent via this video channel.

**Output**

Name of the output on this canvas to which the Tx port connects.

**Process Rx**

Select a Bus channel to receive media in or place media out, or none.

**Bypass Rx**

Select to direct a signal from the Bus to the corresponding channel Tx (must be matched resolution).

**Tx Clock**

Sync this output port to an available option.

**Tx 3D Mode**

Select to enable 3D out.

**Rx 3D Mode**

Select from: Mono, Left Eye only, Left Eye (Right=Other), Right Eye (Left=Other), or Right Eye, with Left Eye from Channel 2, 3 or 4)

**Tx 3D Source**

Select the required video source.

**Tx 3D Shutter Delay**

Select angle in degrees (0 to 359).

**Tx Bypass Source**

Select the input to send through bypass.

**Warp** (Warp & Blend configurations only)

Enable/disable warp.

**Warped Test Patterns** (Warp & Blend configurations only)

Apply warps to test patterns.

**Warped Highlight Enabled** (Warp & Blend configurations only)

Display green fill where video is warped away from rectangular bounds (required Blends to be disabled).

**Blend** (Warp & Blend configurations only)

Enable/disable blends

**Black Level Correction** (Warp & Blend configurations only)

Enable/disable black level correction

**Video Measure Source**

Select the video source for which you want to see details, listed under the Juggler in the Project panel. See [Video Measure](#) <sup>(87)</sup>.

**Vertical Delay**

Adds a fractional amount of vertical delay to an output, specifically with the aim to minimize throughput delay to the bare minimum.

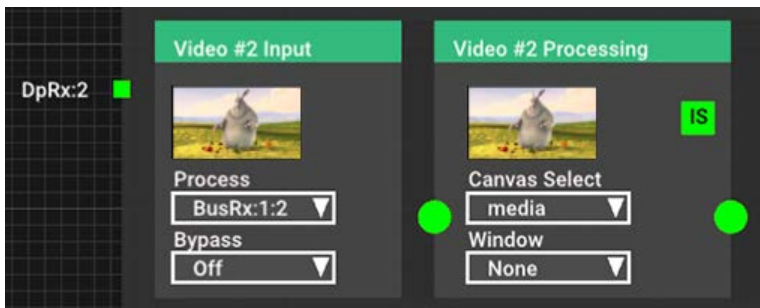


## IntelligentSource

7thSense IntelligentSource™ is a method of inserting data into the top line of every source frame in order to detect frame sequence breaks and errors, and monitor performance drops. It does this by comparing the data delivered with each frame against expected values such as frame number and frame timing.

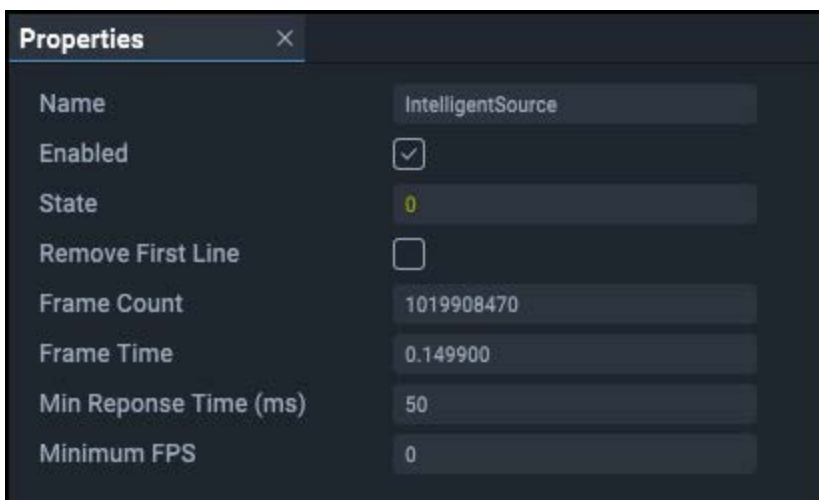
IntelligentSource is therefore used to trigger same-frame failover to a backup stream. This can work across Juggler modules to provide either an automatic or manual loop option for source failover. Use of the 7thSense IntelligentSource API in conjunction with this allows both sync and video data monitoring.

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 VideoChannel Properties panel for that channel of the Juggler.

## Properties



### State

IntelligentSource presents a state that can be evaluated and used as a trigger.  
 0 = OK, 1= Warning, 2 = Error, 3 = Disabled

### Remove First Line

If using a stream with IntelligentSource data, check this to strip off the additional row of pixels used to convey the frame data.

### Frame Count and Frame Time

these will be incrementing when all is well. They are just an indicator.

### Min Response Time

This is the duration in milliseconds within which performance can drop before IntelligentSource presents a change of status.

### Minimum FPS

Integer. A value of zero will be ignored.

This figure is a proxy for the actual time taken to process a frame, so that if, for example, generative content input is running slow, this will trigger an IntelligentSource change of state.

*Example:* 50 is used to compare the frame time for 50 fps against the correct frame time for 60 fps.

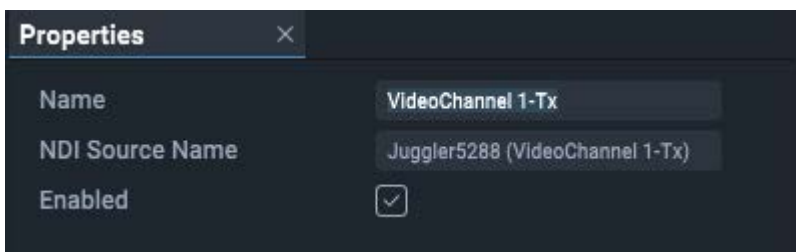
The IS icon will be green when enabled and good IntelligentSource data is flowing, and red if enabled but no IntelligentSource data is being received. If the IntelligentSource status changes, the IS icon for the channel will be yellow.

In the event of signal failure, a double-banked Juggler system can switch instantly to the second system.

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

## Video Channel Tx

For each Video Channel of each Juggler, the Tx NDI source name is shown (read only):

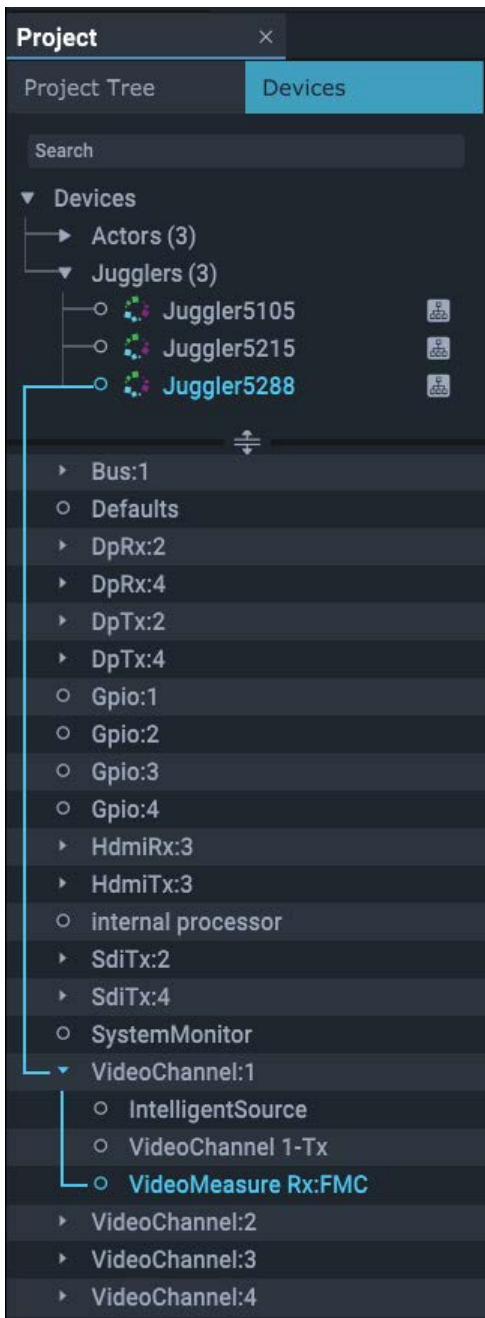


The Rx NDI source belongs to the adjacent Juggler connector on the left, and shows in Video # Input.

## Video Measure

Video Measure is a feature that profiles video sources coming into and emerging from a Juggler video channel. These can be via rear panel Interface 1 and 2 (listed here as FMC), or the Unit HDMI (listed here as OnBoard) or via the modules (listed here as SFP).

Select which you want to see the properties for, by selecting from the dropdown in the main Video Channel property, 'Video Measure Source'. This will then be added to the Video Channel branch of the Juggler in the Project panel:



All properties except 'Refresh' are read-only, and give a complete profile of the source.

**Refresh Video Measure**

This checkbox triggers a refresh of the video measure data when checked. You need to check it after selecting the source in the main Video Channel properties. In response, the Juggler will immediately reset the refresh control after the data is refreshed.

## Video Rx / Tx Port Interfaces

---

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

Select a Juggler in the Project and expand 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.

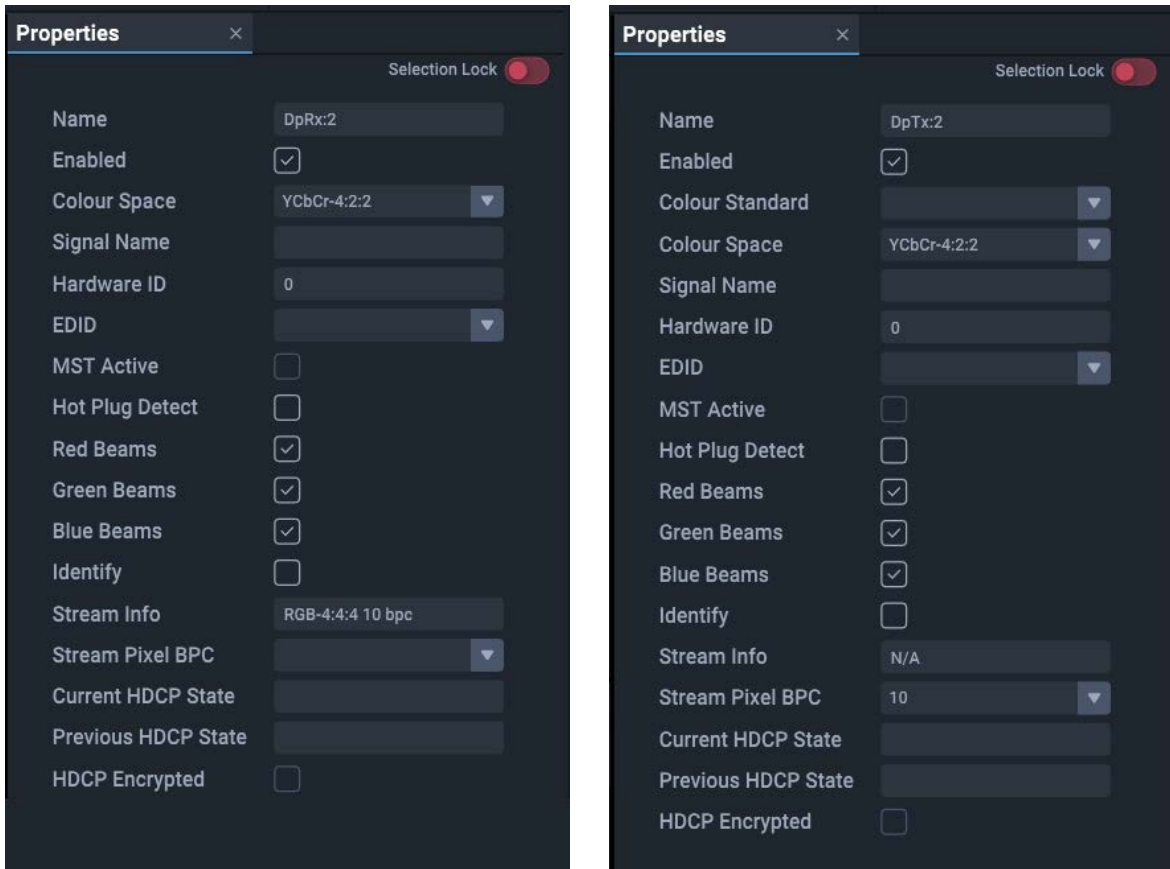
### Channel Input/Output Properties

There are a total of four Channels, numbered 1-4, in every Juggler 1. 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

Note that in the case of a 4-channel HDMI unit, video channel 3 occupies the Unit HDMI on the rear left, not an interface panel on the rear right.

## Example: DisplayPort Rx and Tx properties



### Name

Type and number. Example default: DP Rx:2, meaning DisplayPort input, video channel 2.

### Colour Standard (Tx)

Colour gamuts such as BT.709/Rec. 709 may require scaling the full colour range to a limited range. Options are BT.709 or BT.2020, with BT.709 being the default.

### Colour Space

Select from: RGB-4:4:4, YCbCr-4:2:0, YCbCr-4:2:2, YCbCr-4:4:4.

### Signal Name

Optional user-friendly name for the signal.

### 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)

**EDID**

Select the EDID from the available list (see [EDID Spoofing](#)<sup>116</sup>).

**MST Active**

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

**Hot Plug Detect**

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

**Red / Green / Blue Beams**

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. Self-cancelling.

**Stream Info**

Shows the active video format on the Rx or Tx, for example, RGB 8/10 or YUV 4:2:2 8/10 etc. On the Tx side the format is defined in the Tx itself (or the defaults) This directly impacts the format which is negotiated with the display. Notice the options for 'Stream Pixel BPC and Stream Pixel format' when clicking on a Tx pin.

**Stream Pixel BPC**

Set your colour bit depth: 8- or 10-bit.

While SDI 12G is always 12-bit for the channel pipe, the video data might only be 8- or 10- bit, and the end device needs to know this.

**Current HDCP State**

For HDMI HDCP-enabled systems only.

**Previous HDCP State**

For HDMI HDCP-enabled systems only.

**HDCP Encrypted**

For HDMI HDCP-enabled systems only.

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

## Video Interface Timing

These properties are common to all video interfaces. This area of the properties of a video interface relates to the EDID being used on this input or output. Values are not editable.

- Active Pixels
- Active Lines
- Total Pixels
- Total Lines
- H Sync Width
- V Syn Width
- H Start
- V Start
- Pixel Clock
- Bits per Pixel
- Colour Space
- Stereo 3D
- Framerate



## Bus Connections in a Juggler System

---

There are two modes of use for the Juggler bus: the 'Input Bus' and the 'Output Canvas Bus'. The mode of your system will be established in its firmware. This section relates mainly to the former, and describes the bus in general.

The physical Juggler 1 bus can be configured in two ways: as a 4-lane 'half' bus via SFP modules, or as an 8-lane 'full' bus with MPO connectors (see: [Connect a Juggler Data Bus](#)). 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.

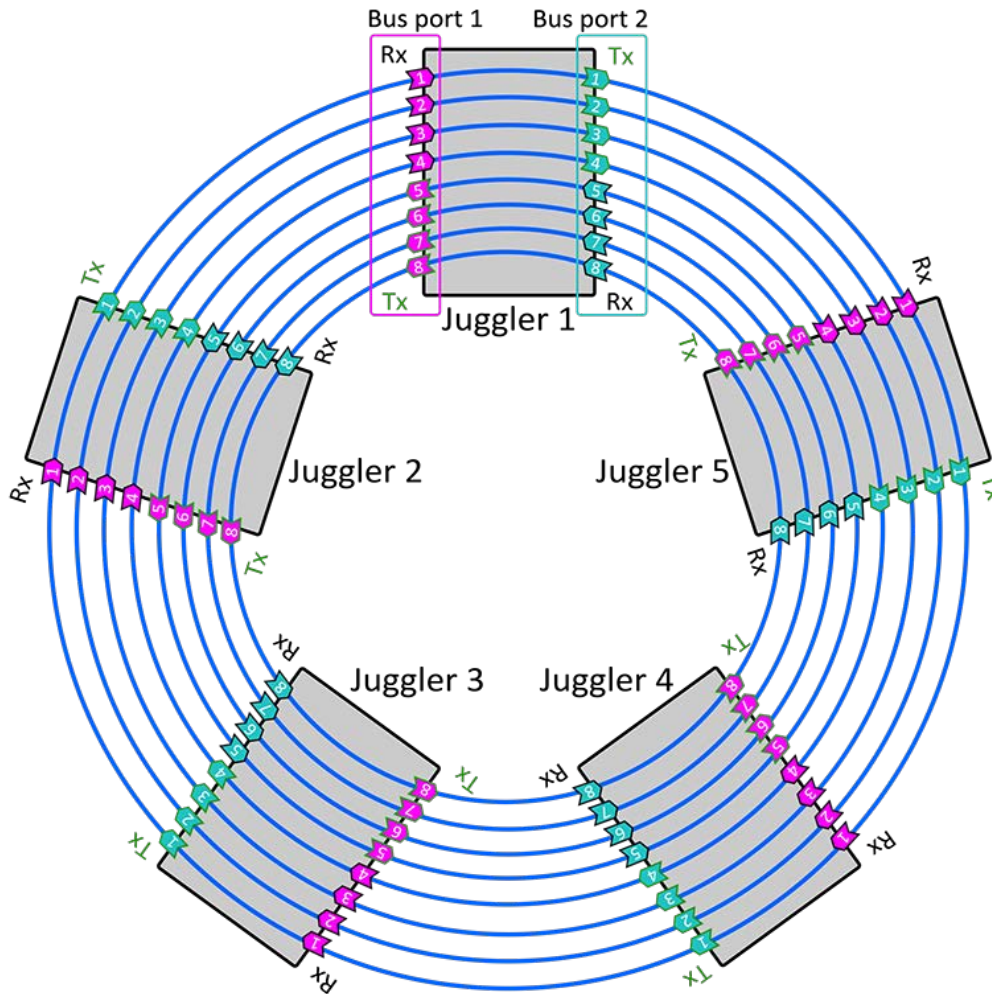
The Bus connection block is represented inside the Juggler in the Nodes panel of Compere.

➤ See also: Physical bus connectors, [Connecting a System in Compere](#) <sup>93</sup>

### 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 (see [Bridging Two 8-Lane Buses](#) <sup>99</sup>). 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.

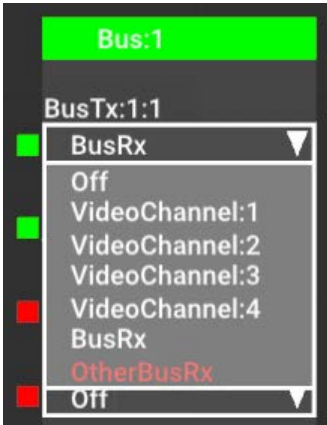


Any Juggler can therefore place media onto the bus, from where any other(s) can stream it off as an input.

### In Compere Nodes panel

- The red/green markers on the left of the Bus nodes indicate where a bus input (Rx) exists (green).
- The drop-down list indicates what this bus node is to send out (Tx). This could be nothing (Off), an unprocessed video channel input, or whatever is being received on the bus from the previous Juggler in line (an Rx in the list).

The drop-down selector in the Compere Juggler Bus offers these 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 (green), this input will be sent out onto 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. See [Bridging Two 8-Lane Buses](#)<sup>99</sup>.

#### 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 availability and type will vary according to the Juggler build.

### Name

It can be useful to name a bus by function if there is more than one.

### Bus Number

In most cases there will only be one bus in a system.

### Bus Reset

A debug function only, checking this will switch the bus off and on again, and will therefore interrupt any video flow around the bus.

### Health Test

A debug function only, checking this will request a health check be performed on the Bus. The box will be instantly unchecked again and the **'Health'** parameter below will report some text about the health. If all is well, the text will be 'OK', otherwise it will be one of a number of things that to assist with debugging the Bus.

### Health

Should be reading OK. Anything else indicates a possible problem with bus connectivity.

### High Bandwidth

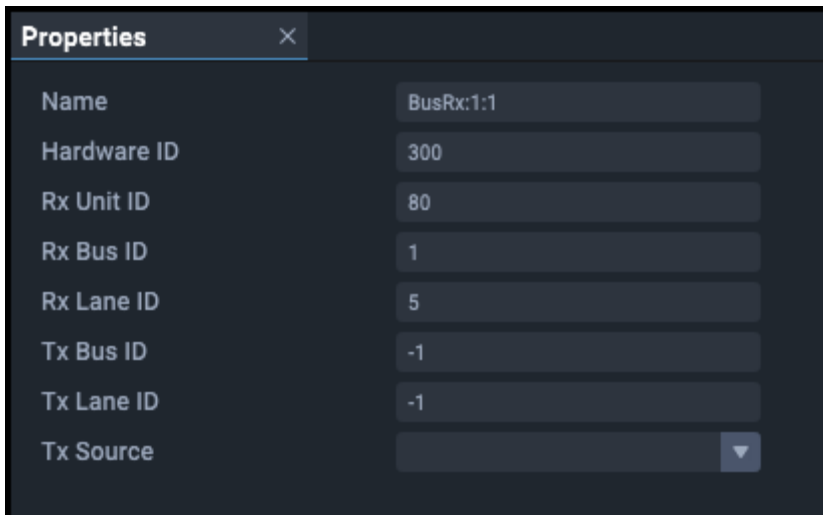
Check if a 6-lane high bandwidth bus is implemented. See [Bus Connections in a Juggler System](#)<sup>93</sup>.

### Output Canvas Bus

Enable output (rather than video channel) busing mode. See [Output Canvas Bus](#)<sup>103</sup>.

## Bus Rx and TX pins

Each Bus element of a Juggler can be further expanded in the Project panel, into its available receive (Rx) and transmit (Tx) pins. The Bus is a Video Interface and shares some of the same property types as Video Rx/Tx pins.



### Name:

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

### 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)

Other properties here identify the bus, the type, and the bus lanes in use.

### Rx Unit ID

Identifies the Juggler from its IP address last digits.

### Rx Bus ID

Identifies the bus interface.

### Rx Lane ID

Identifies the bus lane being used.

### Tx Bus ID

Identifies the bus interface.

**Tx Lane ID**

Identifies the bus lane being used.

**Tx Source**

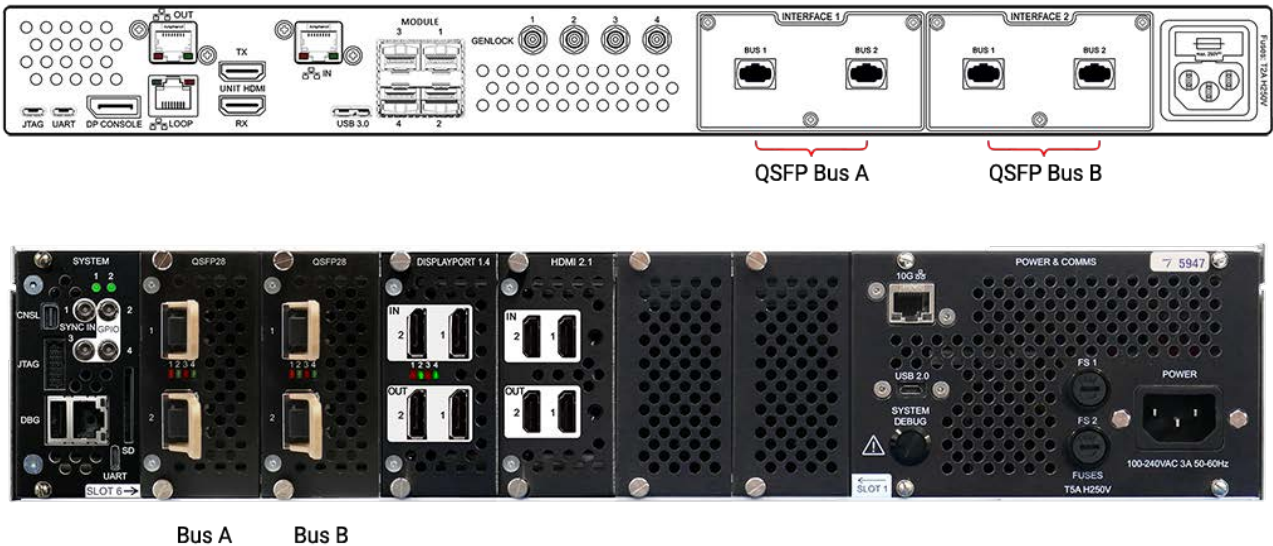
As selected in the bus selector in Nodes panel.

**Video Timing**

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

## Bridging Two 8-Lane Buses

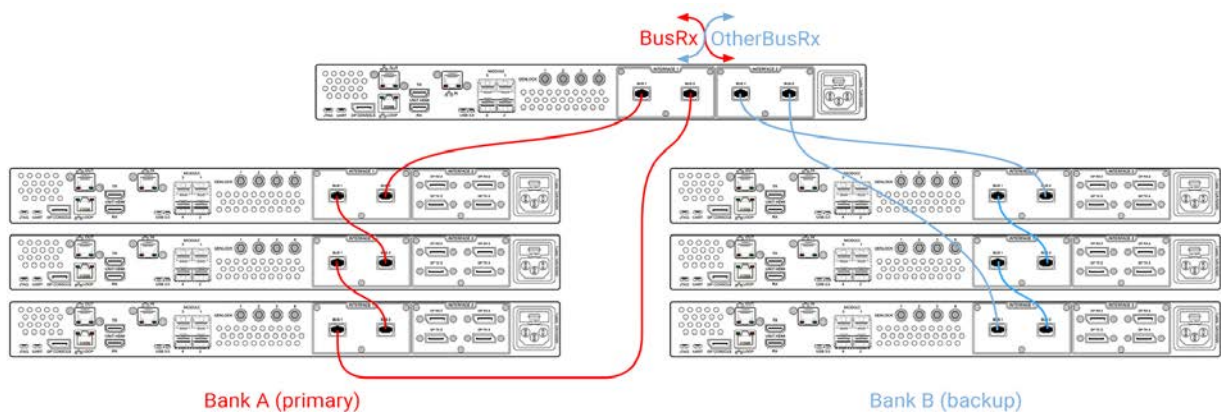
Juggler systems can process many inputs and drive complex displays. Compere is in control, but what happens if a video input fails? Compere has the IntelligentSource failover feature, which can be used to instantly redirect synchronised inputs from a second system into the primary bus instead. Such a system uses a specific Juggler configuration with two bus interfaces (dual bus) instead of media inputs of its own.



This allows for two separate banks of channels to be routed through the same Juggler, acting as a bridge between the two. Compere is used to assign and distribute any combination of the available 16 channels\* to any of the two 8-channel bus banks, to be processed for outputs. (\*A high bandwidth configuration dual bus would comprise 2 × 6 channels.)

### Example of use: redundancy failover

In this example, a primary bank (A) of Juggler 1 units can assign 8 input channels to 8 bus channels and use those bus lanes to support specific outputs. A second (or backup) bank of Jugglers (B) with matched inputs is placed on a second set of bus lanes. If input to primary bank A fails, the backup bank's inputs take the place of those on the primary.



This switch would happen within the dual bus juggler where the 'BusRx' assignment would be switched to 'OtherBusRx':



'BusRx' refers to the channels placed on the local bus, and the 'OtherBusRx' refers to the channels from the second bus. This switch would reassign the backup channels to the bus, in place of the primary channels. After this switch, within each juggler that is supporting a physical output, the processing channel is selected from the bus that is now carrying the corresponding backup input channel.

**This is the first Juggler in Bank A.**

Incoming video from a local DP input (e.g. VideoChannel:2) is placed onto BusTx:1:1.

**This is the second Juggler in Bank A.**

Incoming video from a local DP input (e.g. VideoChannel:2) is placed onto BusTx:1:2.





**This is the dual-bus Juggler.**

On the first bus (Bus:1) this Juggler will receive the inputs that were assigned within the Bank A Jugglers. It also receives the inputs on the second bus (Bus:2) from the Bank B Jugglers.

Note that it simply receives each BusRx, per bus lane, and routes to the equivalent OtherBusRx.

Bank A Jugglers are the primary units, whilst Bank B is connected as backup. IntelligentSource can be used to trigger instant switching.

The Bank A Juggler inputs are being assigned to the Bank B bus by the assignment 'OtherBusRX'. This will allow the inputs from Bank A to be available in Bank B.

The routing can be switched, so that the Bank B inputs (as BusRx) instead become available to the Bank A bus (as OtherBusRx).

This will allow the inputs from Bank B to be available in Bank A.



This applies to whatever inputs are placed on the bus of each bank, per channel process inputs, bypass streams or output canvas.

## Output Canvas Bus

So far we have looked at using the bus as a transport system for taking an **input** from one Juggler and sending it to another to handle the display output. The canvas is effectively assembled at the end of the process. This has its limits as display sizes increase and the number of inserted windows multiplies.

Whilst every Juggler has four video channels, and up to four input ports and four output ports, there isn't always a match between the number of inputs required and number of outputs. For example, you may well want many input options to many simultaneous canvas windows, on a very large overall canvas, comprising a much smaller number of display devices. This will require many input signals to be processed and sent to many windows, each of which can be **anywhere** on the overall canvas, on any canvas output, at any time, whereas the physical displays need the output connectors of only a few of the Jugglers.

### The OCB Difference

This is done by operating the Juggler bus system in a different way. On an [Input bus](#)<sup>93</sup>, Jugglers are described as selectively processing 'Input > Process > (Bus) > Output' – *across each Juggler video channel*. (left to right through the Juggler in the Nodes panel). In other words, inputs can be selectively and individually placed on and taken from the bus before deciding their display destination. It also means there can only be a maximum four Windows on a Canvas.

In an output canvas bus (OCB) setup, the four Juggler 'Video # Processes' (including their canvas location) are diverted automatically onto four *bus video channels*, which carry the whole canvas assembly (Canvas, with any number of Outputs and Windows) around. In an 8-lane bus, two bus lanes are assigned to each of these bus video channels\*, which then feed two output connectors each. One Juggler is assigned to each bus video channel to send these outputs to its Tx ports to display.

\* On a 4-lane bus, these can do the same by enabling 'dual-stream' channels.

In this mode, any number of windows (media input) can be moved anywhere on the canvas (overall display) regardless of which Juggler hosts the input media, and which is driving the actual display output.

It is essential to have an understanding of how an output canvas bus works, but to make it easier to set up, there is the [OCB Wizard](#)<sup>109</sup>.

### Video Channel Leaders

- Allocated Jugglers take one bus video channel (only) onto two of their Tx ports.
- A juggler that takes a bus video channel output and places it on its Tx ports 'takes the lead' for that video channel. In that sense, a Juggler that provides the output connection for video channel 1 is a 'leader', one that provides the output for video channel 2 is also a 'leader', and so on.

- Any Juggler that provides no physical output is referred to as a 'follower'. In reality it just has no channel lead role.
- Every 'Video # Output' in the system uses the same canvas output as named by the leader of that channel.  
*Example:* Juggler A has its VideoChannel:1 property set to 'Output Canvas Leader' and its Video #1 Output is set to an Output named 'Top Left'. Juggler B, C, D ... will all set their Video #1 Output also to Output 'Top Left'.

## Populating the Canvas (Canvas Panel)

A canvas comprises up to four video channels, carried over the 8 bus lanes of a 'full bus' system (or 4 dual streams of a 4-lane system). Each bus video channel is mapped onto two lanes, providing left and right parts for that canvas output to be delivered by a Juggler taking the lead for that bus video channel.

These paired elements of the overall display can be arranged as required. Here are three common linear arrangements, but the pairs don't have to be aligned to each other. Names for the canvas outputs are just suggestions that will help in constructing the flows.

8 × 1

Output 'Far Left'		Output 'Mid Left'		Output 'Mid Right'		Output 'Far Right'	
VC1:Left	VC1:Right	VC2:Left	VC2:Right	VC3:Left	VC3:Right	VC4:Left	VC4:Right
B1:1	B1:2	B1:3	B1:4	B1:5	B1:6	B1:7	B1:8

4 × 2

Output 'Top Left'		Output 'Top Right'	
VC1:Left	VC1:Right	VC2:Left	VC2:Right
B1:1	B1:2	B1:3	B1:4
VC3:Left	VC3:Right	VC4:Left	VC4:Right
B1:5	B1:6	B1:7	B1:8
Output 'Bottom Left'		Output 'Bottom Right'	

2 × 4

VC1:Left	VC1:Right	Output 'Row 1'
B1:1	B1:2	
VC2:Left	VC2:Right	Output 'Row 2'
B1:3	B1:4	
VC3:Left	VC3:Right	Output 'Row 3'
B1:5	B1:6	
VC4:Left	VC4:Right	Output 'Row 4'
B1:7	B1:8	

When a canvas output is now added onto the canvas in the Compere Canvas panel, and then linked to one of these dual video channels in a Juggler node, it will represent the two bus lanes – i.e. as wide as two outputs. The left half of this will go to one Tx port of the lead for the video channel, the right half to another Tx port.

*Example:* the 4 × 2 arrangement above:



A similar output canvas mode is available to 4-lane bus systems, but instead of each video channel connecting to two bus lanes, they each connect to one. In effect this is as above, but with only a single output. By applying a further 'dual stream' mode, this also can be split into two outputs.

## Output Canvas Bus Workflow (manual method)

To set up an OCB system in this way, requires a different workflow from the [Input bus](#)<sup>93</sup>. A schematic diagram is important to work from, with systematic naming of Jugglers, canvas elements and output devices. In this way, between diagram, physical racking and Compere, you will more easily see the sequential arrangement of all the components.

### Add the Jugglers to the Nodes panel

- Drag all required Jugglers in the network from the Project panel into the Nodes panel.
- Name the Jugglers for their roles, e.g. Juggler3456 Ldr1 (this Juggler is leader for video channel 1), Juggler3457 F (this Juggler is a follower only).
- Arrange the Jugglers logically. For example, you may want to lay out the leader Jugglers at the top or group them separately.

### Connect all inputs

- Connect each input pin on every Juggler to its corresponding video channel. *Example:* DpRx1 to Video #1 Input, Process = DpRx1:





### Establish the Canvas

- In Compere, drag a Canvas from Resources into the Canvas panel.
- Size the Canvas in its 'Canvas Bounds' properties for the overall pixel resolution of the whole display area.
- In the Canvas main properties, check 'Output Canvas Bus'. This identifies all its child Outputs and Windows as coming from the bus.
- For 4-lane bus configurations with the same left/right output pairing, select 'OCB Dual Stream'. Otherwise all the bus video channel will go to a single rather than paired output.
- From Resources, add the required number of Outputs onto the Canvas to fill/fit the Canvas, bearing in mind that each bus video channel will provide 2 Tx feeds, 'Left' and 'Right' (i.e. double width).

Do not manually size these outputs, but it can help to name them for their positions (e.g. 'Output top left' rather than default 'Output 1' – see the illustration, above).

- Paired outputs? An odd number of display devices means that one half of one of the outputs will be used. This unused part will simply lie off the canvas.

### Set each bus mode to 'Output Canvas'

- For all these Jugglers, and every bus on each (you can multi-select them all in the Project panel) set the bus properties to 'Output Canvas Bus'. This will change the way the whole bus is connected and operates.
- The bus drop-down menus will now be **auto-populated** with their set channels. *Example:* for an 8-lane bus system:

BusTx:1:1 and BusTx:1:2 carry Video Channel 1 Output

BusTx:1:3 and BusTx:1:4 carry Video Channel 2 Output

BusTx:1:5 and BusTx:1:6 carry Video Channel 3 Output

BusTx:1:7 and BusTx:1:8 carry Video Channel 4 Output

Do not change any of these auto settings.

## Which Jugglers will be providing the physical output connections?

One Juggler must be assigned to lead each bus video channel. *Example:* select VideoChannel:1 from Juggler leader1 in the project panel. In its properties, check 'Output Canvas Leader'.

- Refer to your schematic drawing for which Leader Juggler is feeding each display (e.g. from left to right)
- On the Leader Jugglers, assign one video channel that will be connected to an output device, to 'Output Canvas Leader' in the properties.

*Example:*

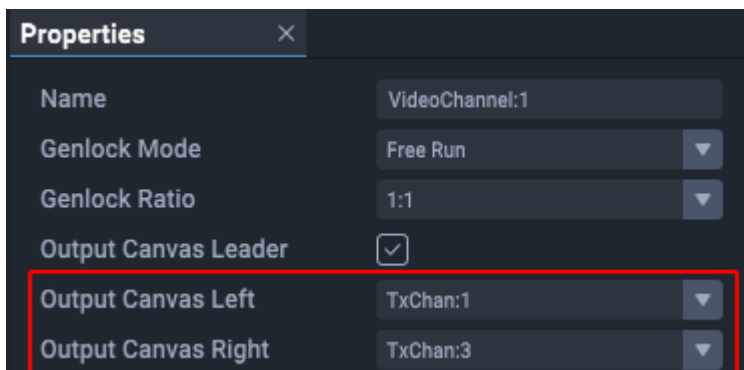
Juggler Ldr1 claims Video Channel 1 as Leader, taking Bus lanes 1 & 2 (other channels available as Follower)

Juggler Ldr2 claims Video Channel 2 as Leader, taking Bus lanes 3 & 4 (other channels available as Follower)

Juggler Ldr3 claims Video Channel 3 as Leader, taking Bus lanes 5 & 6 (other channels available as Follower)

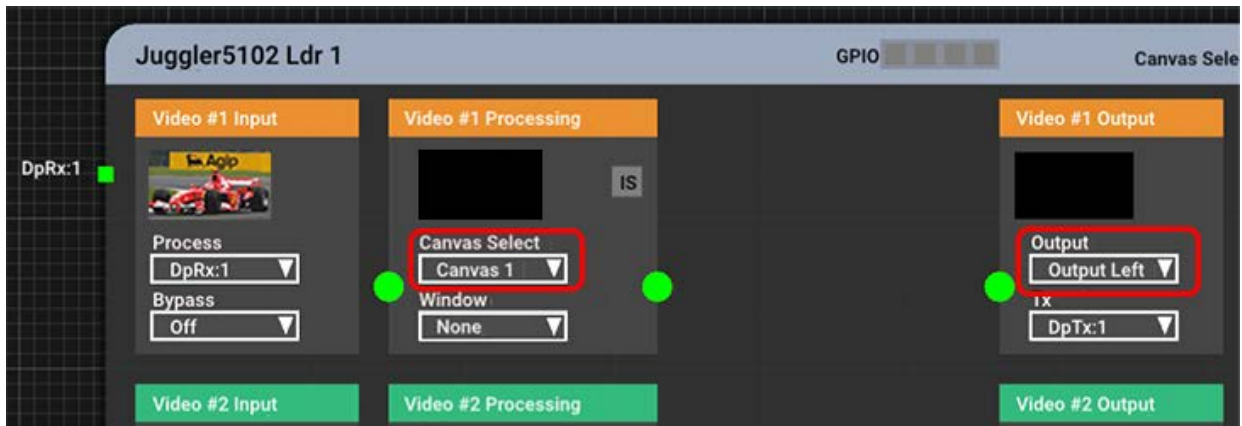
Juggler Ldr4 claims Video Channel 4 as Leader, taking Bus lanes 7 & 8 (other channels available as Follower)

- Set up/check the EDID of all these Juggler leader output connectors. These may be custom-supplied to suit the particular configuration.
- In the properties for each of these channel leads, assign 'Output Canvas Left' and 'Output Canvas Right' in their drop-downs, to the output connectors for that Juggler (e.g. TxChan:1 and TxChan:3):



## Route the lead Juggler outputs to the canvas outputs

- Find Juggler video channel 1 leader in Nodes, and in Video #1 Processing assign the Canvas, and select the required canvas output name in Video #1 Output. (Ignore the Tx box.)



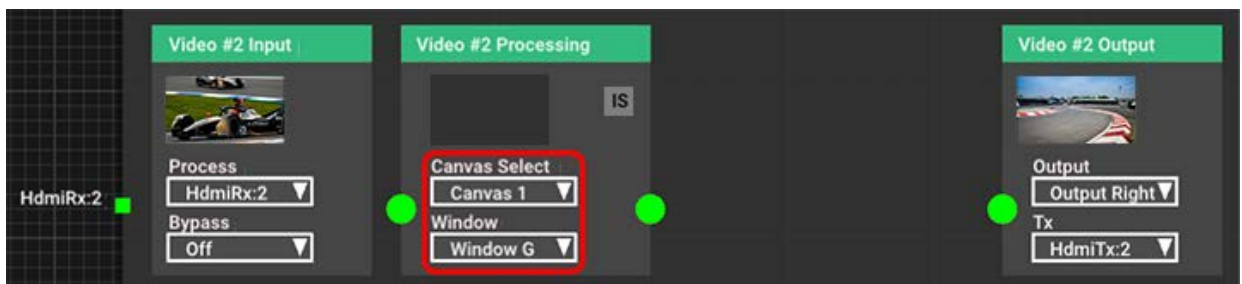
- The canvas output in the Canvas panel will now have assumed its dual (left/right) dimensions, so now locate it correctly in the canvas.
- Do the same for the Juggler leading video channel 2, and assign Video #2 Processing to the Canvas and Video #2 Output to its required output name.
- The canvas output will now have assumed its dual (left/right) dimensions, so now located this correctly in the canvas.
- Repeat through all leader Jugglers until all video channels are covered.

### Route all other Juggler non-lead outputs

- In all Jugglers, each remaining video channel needs to be assigned to the canvas. Do this irrespective of any active input on that Juggler’s video channels.
- In every Juggler where the Juggler is not managing (leading) that channel, set ‘Video # Output’ in the output drop-down to the same output name that the leader for that channel number is managing. *Example:* channel lead 1 is addressing ‘Output Top Left’, so in the ‘Video #1 Output’ block for all other Jugglers, also select ‘Output Top Left’ from the Output drop-down.

### Add the media windows to the canvas

- Add Windows to the canvas as required. Name, size and position these as in any Compere system. These reflect the all the inputs you wish to display. They may have preset positions or groups, but each can now be moved seamlessly anywhere on the entire canvas.
- Assign these named windows as required to Juggler inputs (under Video # Processing):



- Windows can be moved on or off the Canvas; when off, their content will not appear on the display. (See also: [Create and Recall Canvas Arrangements](#)<sup>130</sup>.)



Since the whole canvas is on the bus, all window information is carried around with it, so unlike the Input bus mode, the media is not going to the same Juggler's output (on the right in the node), but straight onto the bus.

## OCB Wizard

When you have understood how and why the [Output Canvas Bus](#)<sup>103</sup> works, the Wizard can take you through all required settings for a new project.

The workflow for an OCB system is similar to that for an input bus system but the wizard will set up the canvas and make the bus and output connections for you. It will also automatically switch you into OCB mode.

➤ For OCB in Juggler 1 systems, see: [Workflow: Juggler 1 in Compere](#)

## Preparation Checklist

- ✓ The Juggler system, inputs, outputs and bus, is fully wired.
- ✓ You will already have configured your network, project group membership, server election scores and project group server.
- ✓ All required Jugglers in the project group appear in the Project panel.
- ✓ You have added all these Jugglers to the Nodes panel, set up their Rx and Tx EDIDs, connected video inputs to sources, and (optionally) renamed them in some way to easily identify their roles, especially those you want to be video channel leaders.

## Step 1: The Canvas

**Context:** see [Canvas: Configure a Display](#)<sup>117</sup>.

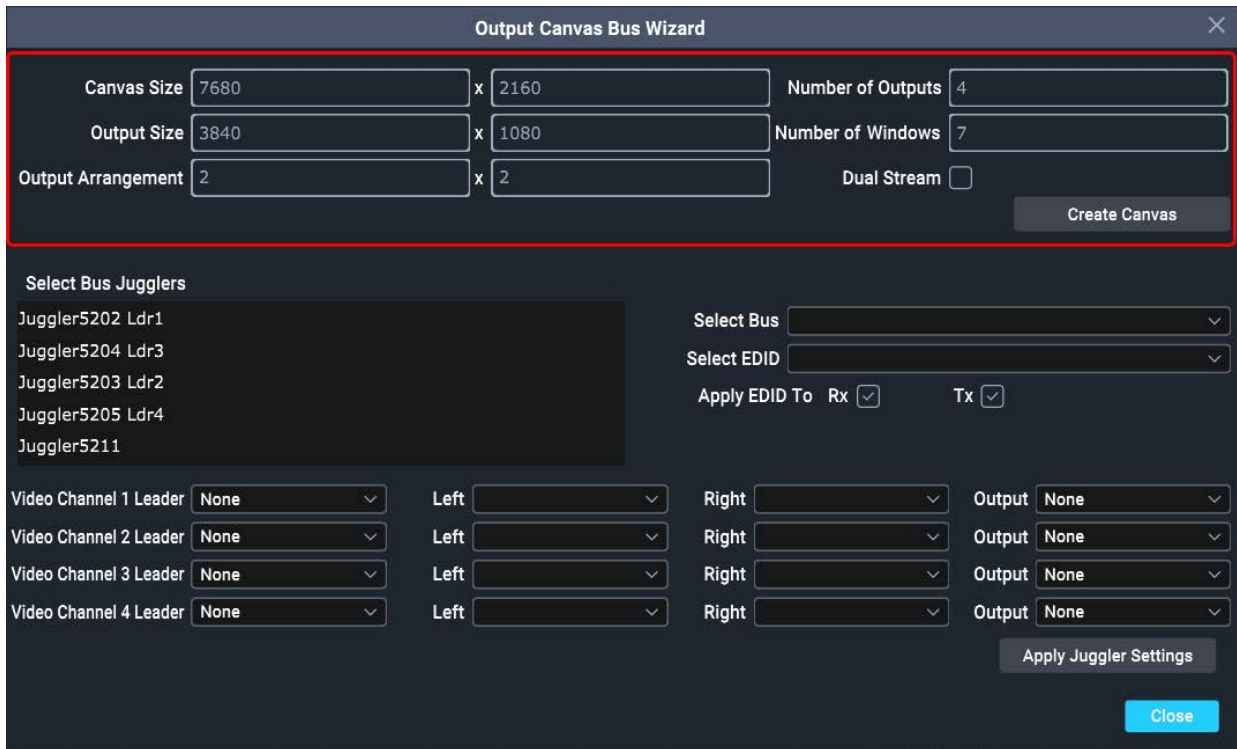
This part of the wizard will create your basic display configuration in the Project, and requires that there is no Canvas already in the Project panel. It will not delete, reset or edit an existing canvas.

Once created, the normal advantages of naming your canvas outputs and windows apply, so that you can easily identify all the elements and match them to any system schematic drawing.

Enter the overall canvas size, the number of canvas outputs (paired, so double width) and their arrangement (e.g. 3 × 3, 4 × 2, 1 × 4).

- Enter the common resolution for all the canvas outputs (double the width to accommodate the pairing on the bus).
- The number of windows is less important; these can be added and changed afterwards. This will simply add a number of unconfigured canvas windows, or none.

- If your bus is a 4-lane ('thin') bus, check 'Dual Stream' to create the paired outputs. This is unnecessary for 8-lane ('full') bus systems.
- Click 'Create Canvas'.



**Notes:**

Each OCB output is *per video channel* and comprises a pair of 'Left' and 'Right' parts. The above example has a 2 x 2 arrangement.

- You can close the Wizard for now.
- Drag the new **Canvas** into the Canvas panel.



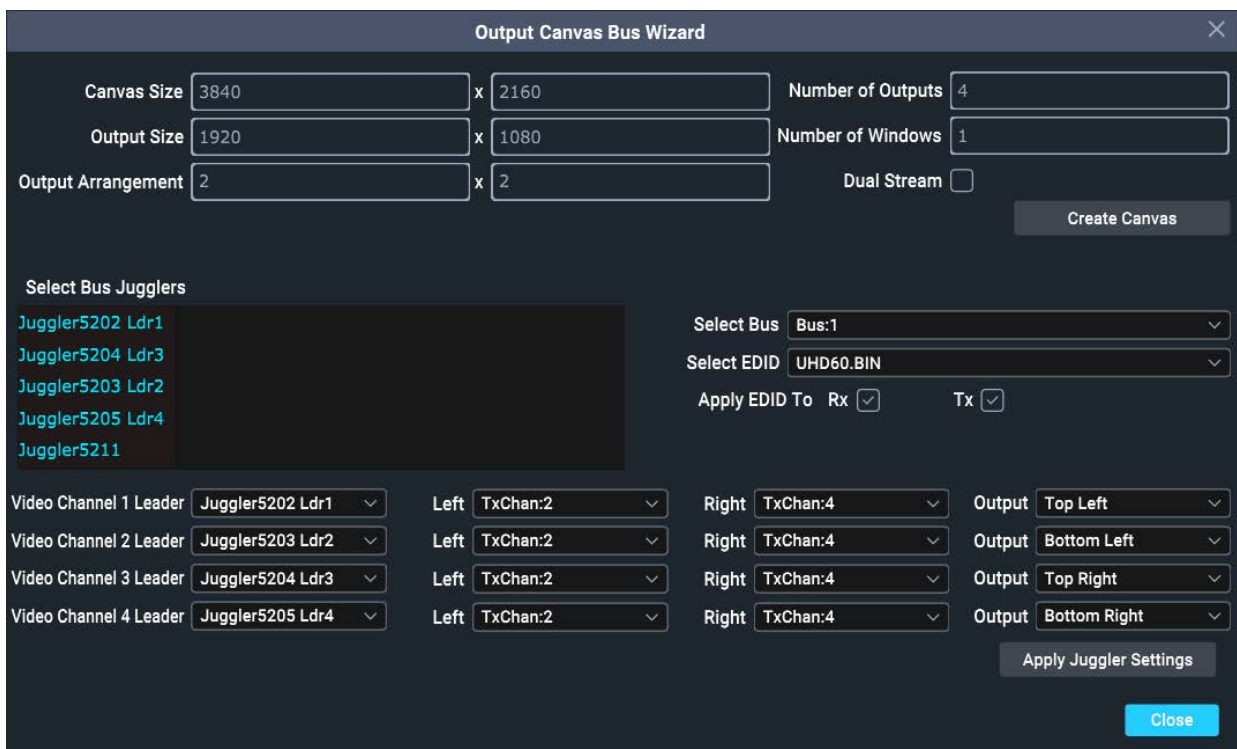
- At this point, it can be very helpful to rename your **canvas outputs**, for example by their location (Top Left, Top Middle etc.).

- Add, name, position and resize any **canvas windows** now if you wish, but this can be done at any time later. Windows are not invoked by the OCB wizard.

## Step 2: Connecting Jugglers to the bus

This part of the wizard connects all Jugglers connected to a canvas, and assigns up to four Jugglers to provide the Tx ports out for each canvas output.

- Reopen the wizard.
- Ignore any default numbers appearing in the canvas section – you will not be using this again.
- Select all Jugglers that are on a common bus.
- Select the bus they belong to (typically this will be Bus:1).
- If necessary, select the EDID for the Tx and Rx ports that the Leader Jugglers use.
- Now assign a Leader Juggler to each active video channel, up to four. If you have named these Jugglers already (as in this example) this will help.
- Select the named canvas output to attach to each of the bus video channels. Every non-lead Juggler will be given the same association between canvas output name and Video # Output. (In the illustration below, every Juggler will have 'Top Left' in its Video#1 Output, 'Bottom Left' in its Video#2 Output, and so on.)
- Select the two Tx ports on each Video Channel Leader that connect the left/right pair of the canvas output to output devices:



- Click 'Apply Juggler Settings'. Updating Jugglers is not instant: you will see an alert to give a moment before proceeding further.

- Close the wizard.
- All Jugglers are now addressing the same Canvas, and their video channel outputs correspond to those of the Leaders.

## Step 2: Add, configure, assign windows

This step is the same as in the final manual workflow step in [Output Canvas Bus](#)<sup>(103)</sup>, above.

- Whether you added windows via the wizard or not, now make sure you have all the windows you need for the project, name them, size and locate them.
- For each Juggler video channel with an input, in Video # Processing select the window on which to display the media:



## Connecting Juggler Inputs and Outputs

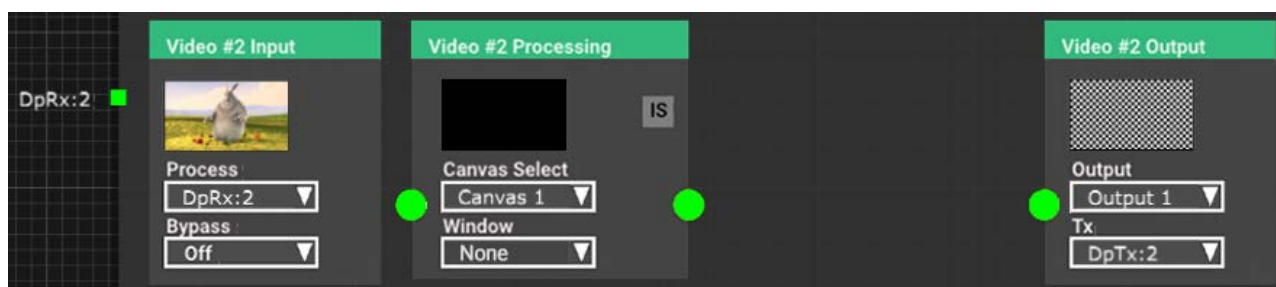
In a windowing Juggler system, you will have set up your [display Canvas](#)<sup>(117)</sup>, with a number of Canvas Outputs and Canvas Windows. The Outputs can be named or numbered to identify the area of Canvas on which you may want to display media. Any number of Windows can be positioned and scaled 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 an 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. This illustration confirms that video is being received from the named input and video is being presented to the listed Tx pin – in this case the canvas as a whole has a test pattern.

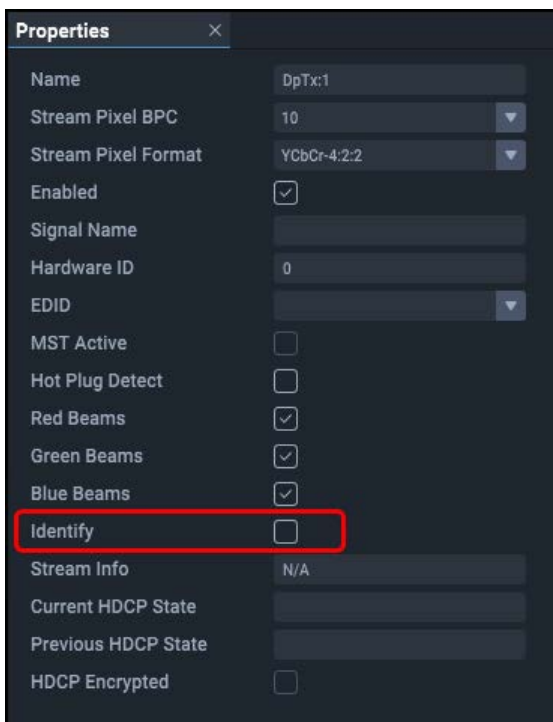


In the Processing block, you decide where this is to go next. There are other processes you can add, but for now, just ensure you have the canvas you want to use.

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 list 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:

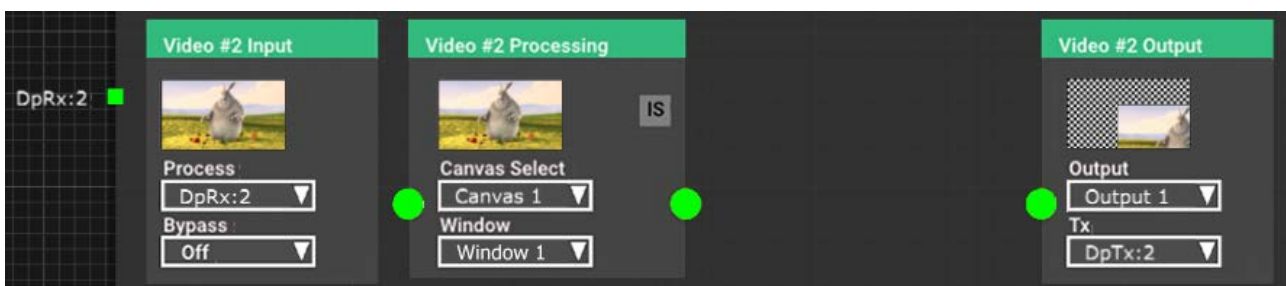


**Reminder:** You can identify a Juggler output in the real world by checking the Identify box in the Juggler output properties:



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 [Canvas Window](#)<sup>(127)</sup> 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:



Now we have NDI showing an input signal is being received, that it is being presented to a named window, and where it is being displayed on the named output.

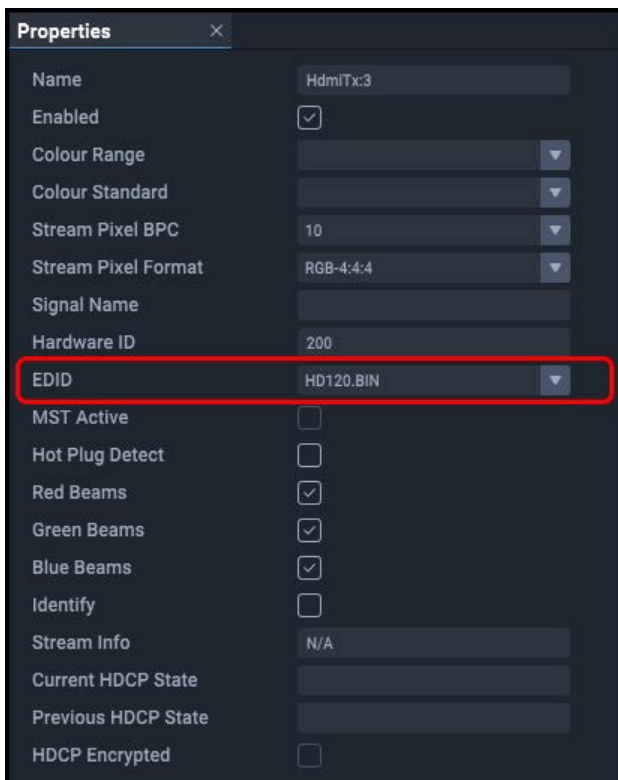
## EDID Spoofing

When an Input or Output is assigned to a Juggler video connector, it adopts the EDID of that connector.

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 Tx EDID from The Properties panel > EDID list:





## Configure a Display with Juggler

---

This section describes the **Juggler Canvas**: the arrangement of video pixels on outputs assembled into a display directly from Juggler connectors. In comparison, when Actor outputs are assembled in the Input and Output Feed panels, this selects source pixels and places them on the arrangement of Actor device outputs.

In addition, Compere can achieve **virtual canvases** operating picture-in-picture regions over the display (this is covered in the Actor user guide).

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 Internal Processor, Capabilities).

The following description is for Windowing Jugglers.

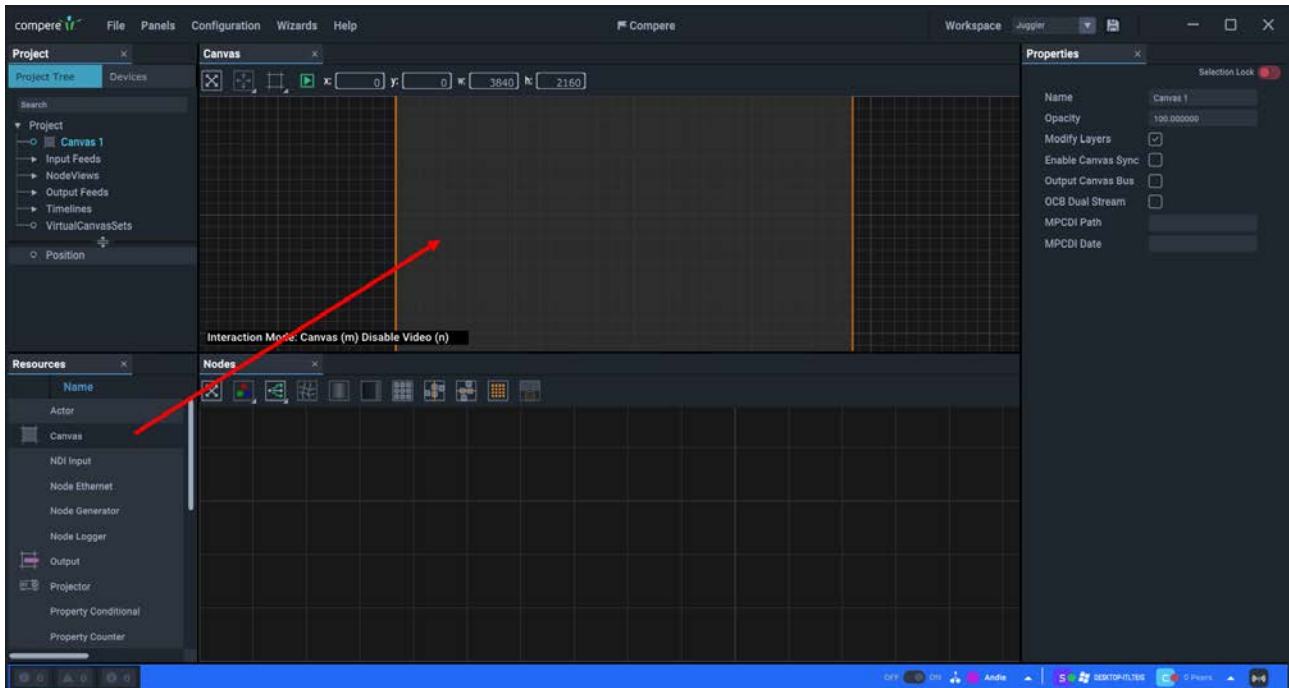
See also:

- [Display Setup via MPCDI](#) <sup>(137)</sup> – canvas and outputs are pre-configured, defined and uploaded along with warp, blend and black level files
- [Output Canvas Bus](#) <sup>(103)</sup> – a configuration for maximising numbers of inputs directed to windows moved anywhere on very large canvases

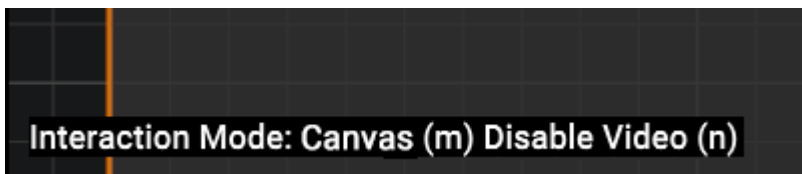
### Canvas, Canvas Outputs and Canvas Windows

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. This will be added at default coordinates (0,0) but can be moved.
- 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, 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 within the grid to position them using Shift+left mouse, but not to resize by dragging.

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

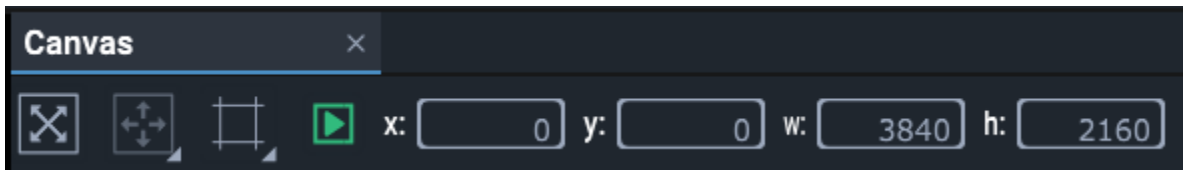
**Disable Video:** press 'n' to toggle between the outlines of Outputs and Windows and any connected video input.

## Menus, moving, sizing

Right-click context menus are available everywhere in the Canvas assembly panel. Many of these options also have keyboard shortcuts and mouse operations as in other panels.

➤ See [Keyboard Shortcuts](#) <sup>16</sup>

## 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 'frame' icon operates the same as pressing the 'm' key, switching between object types for selection.
- The green / grey arrow 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. Use unique names.

### Opacity

Set the opacity of all Windows on this Canvas. This will not be applied to any bypassed video channels.

### Modify Layers

Uncheck to fix the layering of Canvas Windows. Check so that selecting any window brings it to the top.

### Enable Canvas Sync

For cross-Juggler synchronisation of Juggler Canvas elements. (Not yet implemented.)

### Output Canvas Bus

On OCB firmware-enabled systems: switch bus mode to place the canvas onto the bus. See [Output Canvas Bus](#)<sup>(103)</sup>.

### OCB Dual Stream

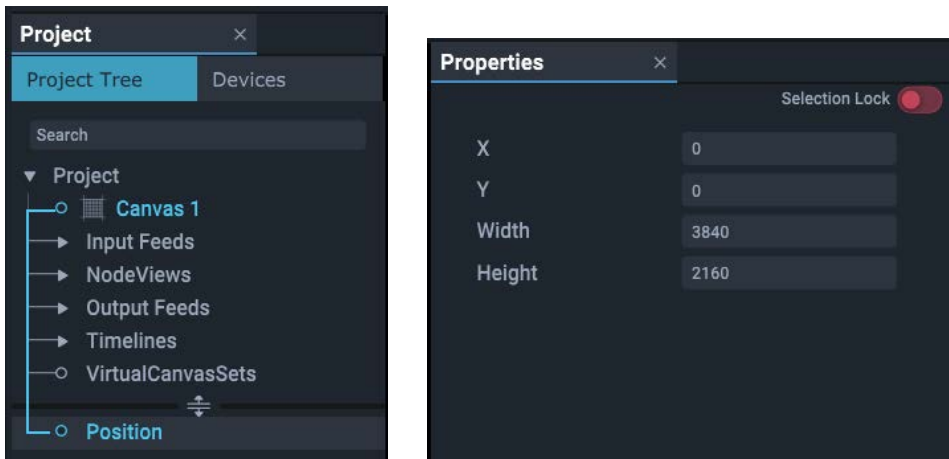
Split inputs in 4-lane output canvas bus systems.

**MPCDI Path**

On warp and blend firmware systems: if using MPCDI data, enter the relative folder path where you choose to keep MPCDI data files.

**MPCDI Date**

Date of the MPCDI source being used.

**Canvas > Position****X and Y, Width and Height**

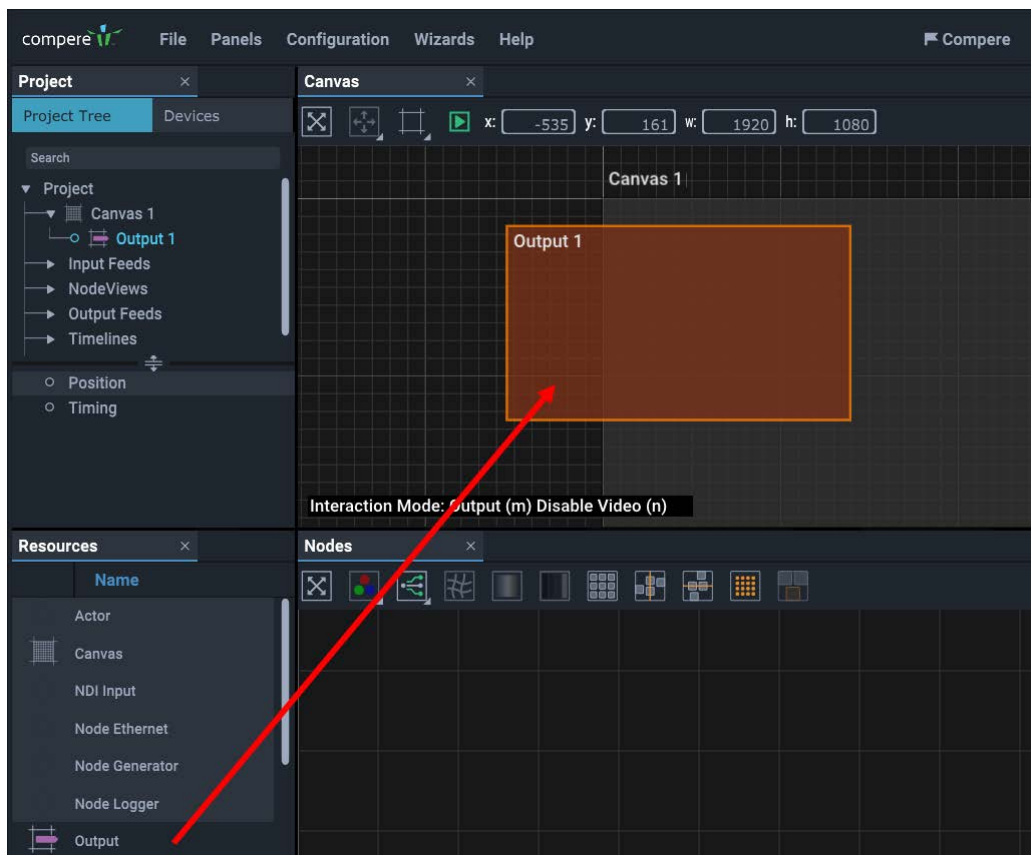
Specify the dimensions of a Canvas.\* If using more than one Canvas, these are all inserted at 0,0 X,Y coordinates, so set the X and Y offsets to displace a second Canvas.

\* **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](#)<sup>121</sup>).

## Canvas Output Setup

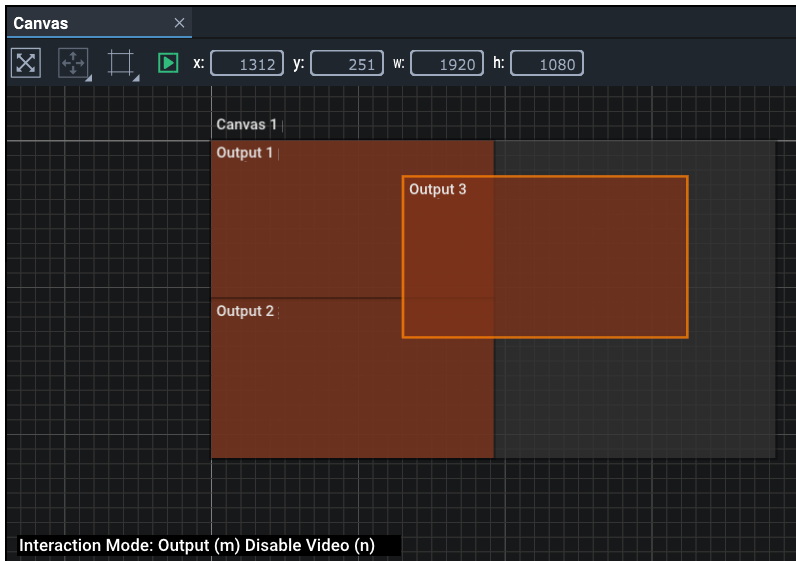
**Canvas Outputs** are the channels driven from Juggler transmit (Tx) ports that together will comprise your display area(s). They are sized to the physical pixel resolution of the output device. The EDID of Tx ports are set in the Juggler using Nodes View.

- 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 as a child to the canvas it was dragged onto. By selecting an Output in the Project panel, you can configure it in the Properties panel. Add all your Outputs in the same way:

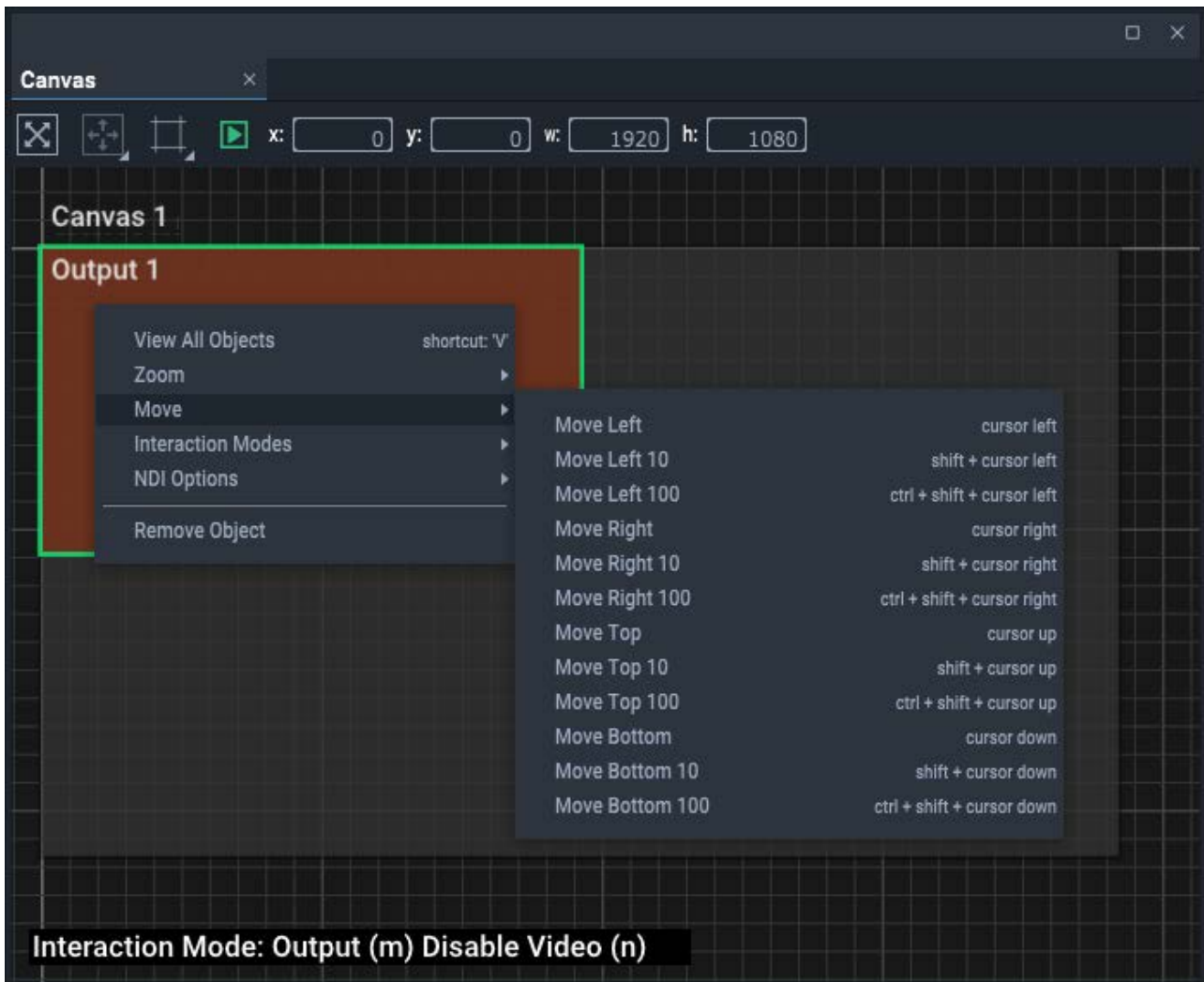


Note how the position and dimensions of the selected object is shown in the toolbar.

- 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 it, and holding the Shift key while dragging it. Hold Shift+Alt 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 > position 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.

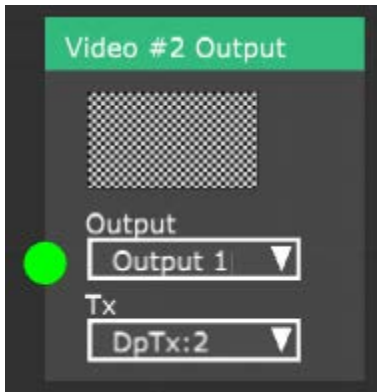
Window objects only:

- Shift+C will centre the item in the canvas.
- Shift+F will make the window fill the canvas.

**When you move a canvas around** in this panel (you can have more than one canvas), all the child outputs and windows move with it.

**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 [Nodes assembly panel](#)<sup>66</sup>, 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

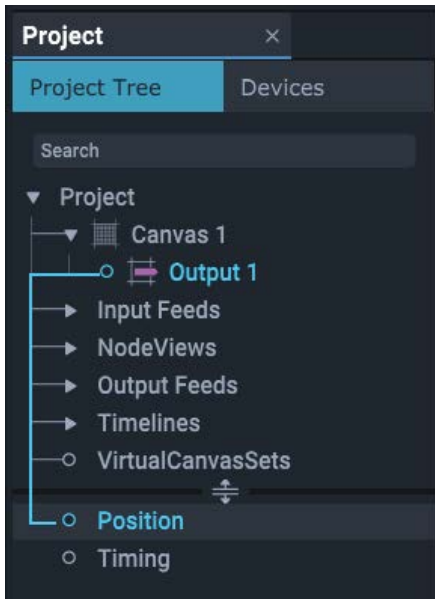
Name the output logically and clearly. Multiple outputs are by default numbered sequentially. Names can also be changed in the Project panel by double-clicking the name there. If using multiple canvases, name Canvas Outputs for the canvas to which they belong.

### X Resolution / Y Resolution

Populated when using MPCDI (this is not the width and height of the output – see below).

For many objects in a Project, there will be parts of that object that have their own property sets. These internal parts are selected first to see their properties. For example, select an Output, and then its Position (they will connect with a blue link line):





## Output > Position

### X and Y, Width and Height

X and Y are the top-left co-ordinates in the Canvas. Width and height relate to the physical output device. Precisely position and size this Output in the Canvas.

## Output > Timing

These properties are read only, and provide information about this Canvas Output.

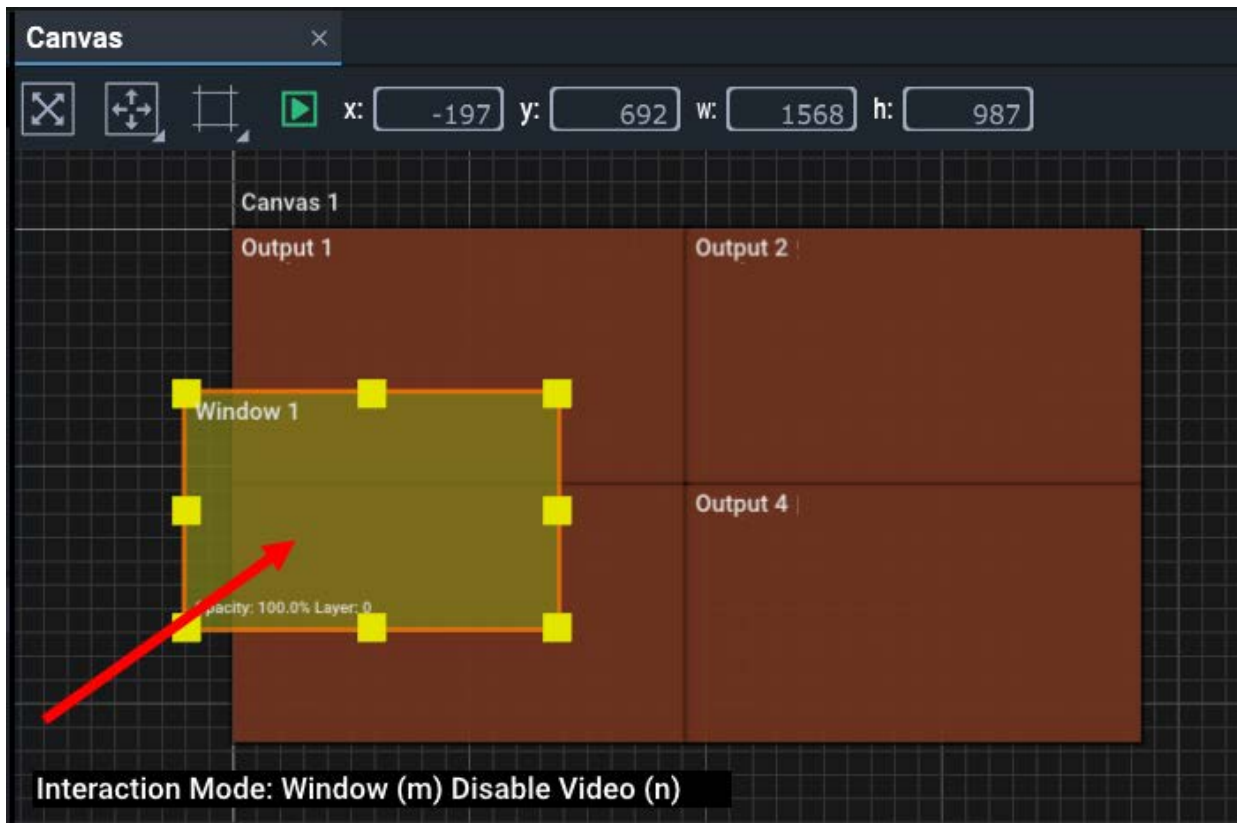
**Properties** ✕

Name	Timing
Active Pixels	0
Active Lines	0
Total Pixels	0
Total Lines	0
H Sync Width	0
V Sync Width	0
H Start	0
V Start	0
Pixel Clock	0.000000
Bits per Pixel	8
Colour Space	4:4:4
Stereo 3D	<input type="checkbox"/>
Framerate	0.000000

## 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 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 and it comes into focus.

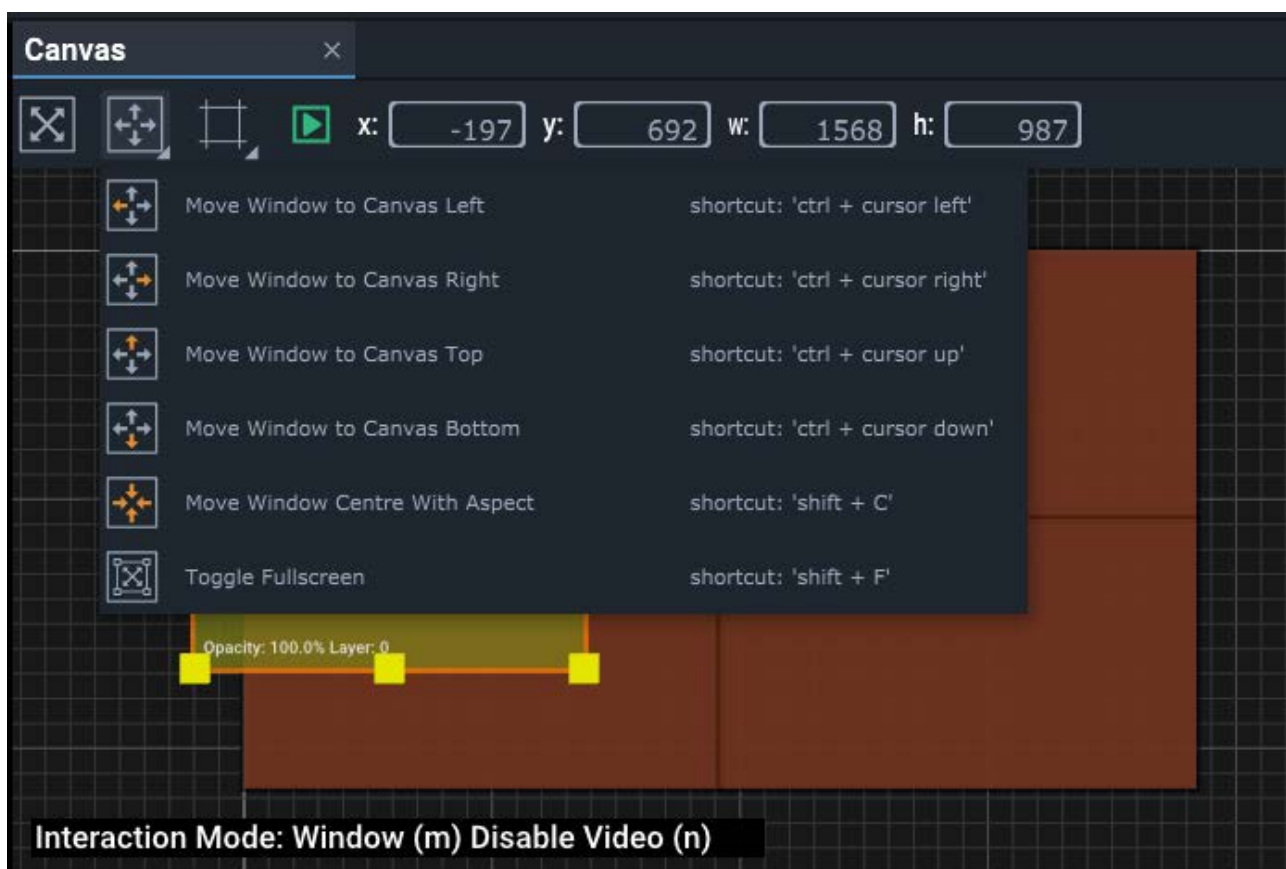
**Position** the Window anywhere in the Canvas assembly panel using Shift+left mouse. Add Alt+ to drag to snap to other edges. Its size reflects a media source sent to it, scaled as required, and is not bound by any Canvas Output dimensions.

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.

**Resize** a Canvas Window with the mouse using Shift+ the grab handles. Add Shift+Alt+ to snap to other object edges. Add Shift+Ctrl when sizing from a corner to maintain aspect ratio. Alternatively enter values in the Canvas toolbar or the Window properties.

The other toolbar button is now active to move selected Canvas Windows around to common positions. Incremental 'Move' positions are also available in the right-click context menu for a Window. 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 for their position or purpose. Names can also be changed in the Project panel by double-clicking the name there. If using multiple canvases, name Canvas Windows for the canvas to which they belong.

### Enabled

Media will only be processed through a Canvas Window when it is enabled. When this is unchecked, the Window object remains in the Canvas panel shown at 25% opacity. If you do not

want a Window to display, and do not want to remove it, it can either be moved off the Canvas, or un-enabled.

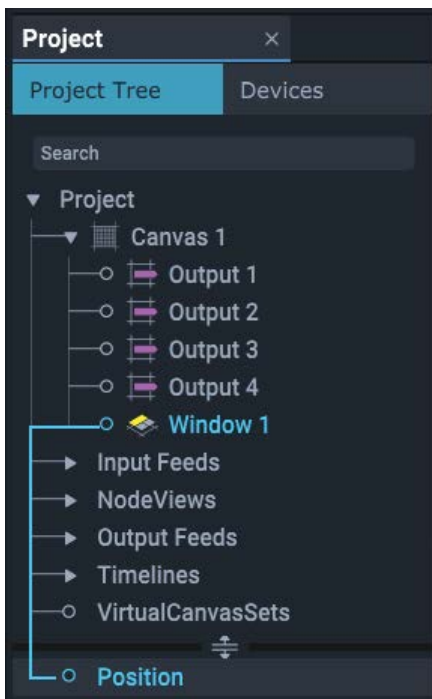
### Fullscreen

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.

### Layer

The Canvas property 'Modify Layers' allows selection of any single Canvas Window to be brought to the top (front). Use this to achieve the correct layering of Windows and then fix the layering by unchecking the Canvas 'Modify Layers' property.

### Window > Position



### X and Y, Width and Height

X and Y are the top-left co-ordinates 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.

## 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.

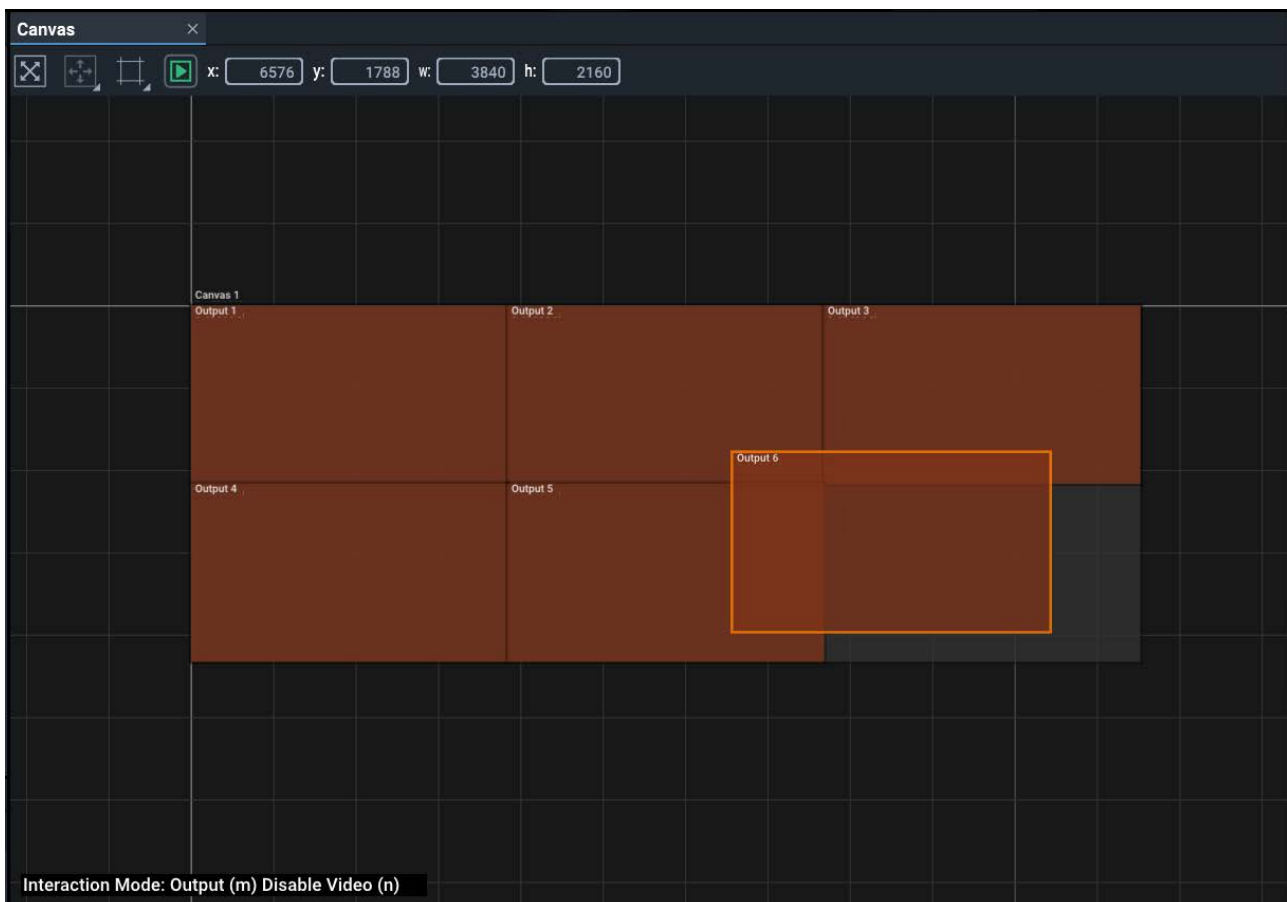
Presets are designed to capture the arrangements and settings for components in the same project. When recalled, dimensions and locations etc. of existing canvas objects will be set again to those of the saved preset.

➤ See [Clones and Presets](#) <sup>(146)</sup>

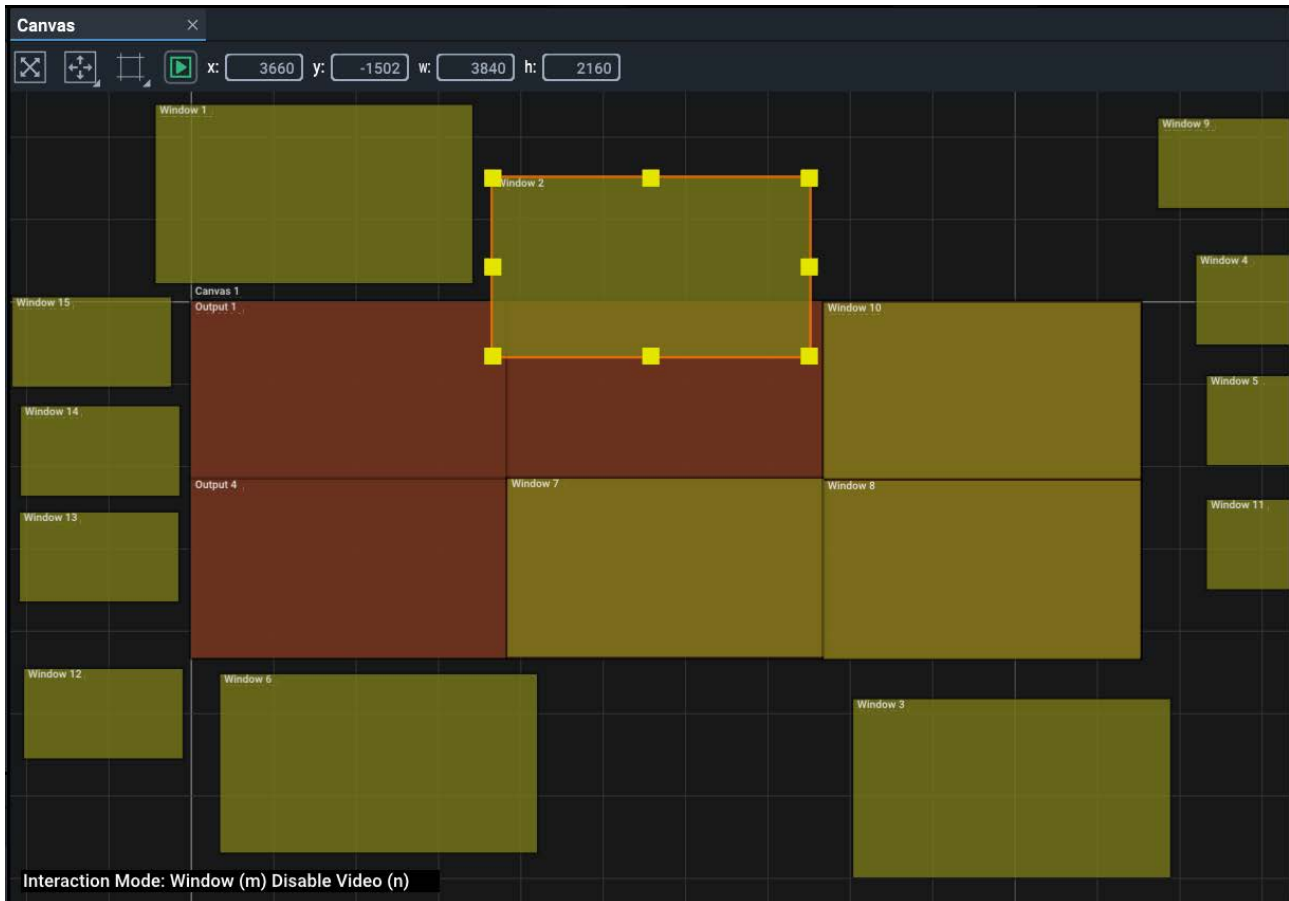
### Example of use

#### 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](#) <sup>(117)</sup> is added to the project, followed by the [outputs](#) <sup>(121)</sup>, sized and positioned as described above:



Into this, all the [windows](#) <sup>(127)</sup> 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.



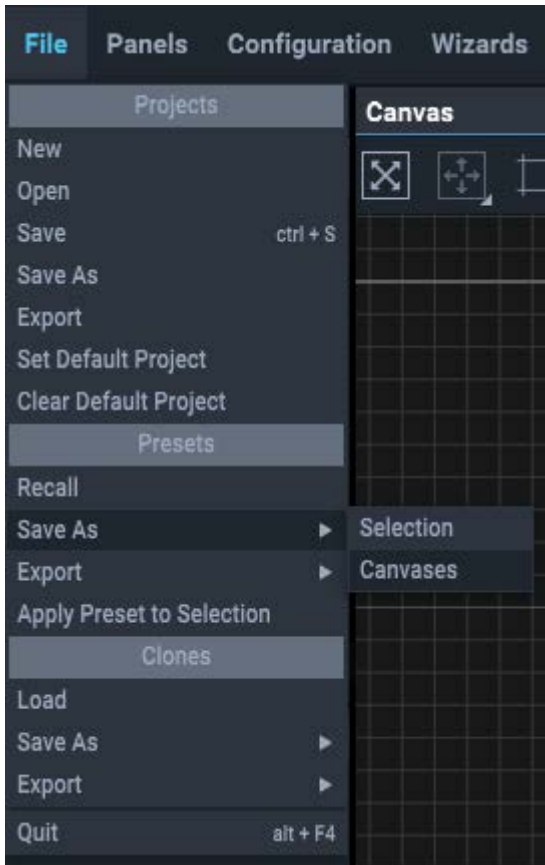
While you are assembling different arrangements, keep associated windows together. NDI feeds may provide sufficient identifiers, but clear naming of windows is important. For example, where a group of windows forms a 'side panel', or a position is used as a 'news feed'.

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*



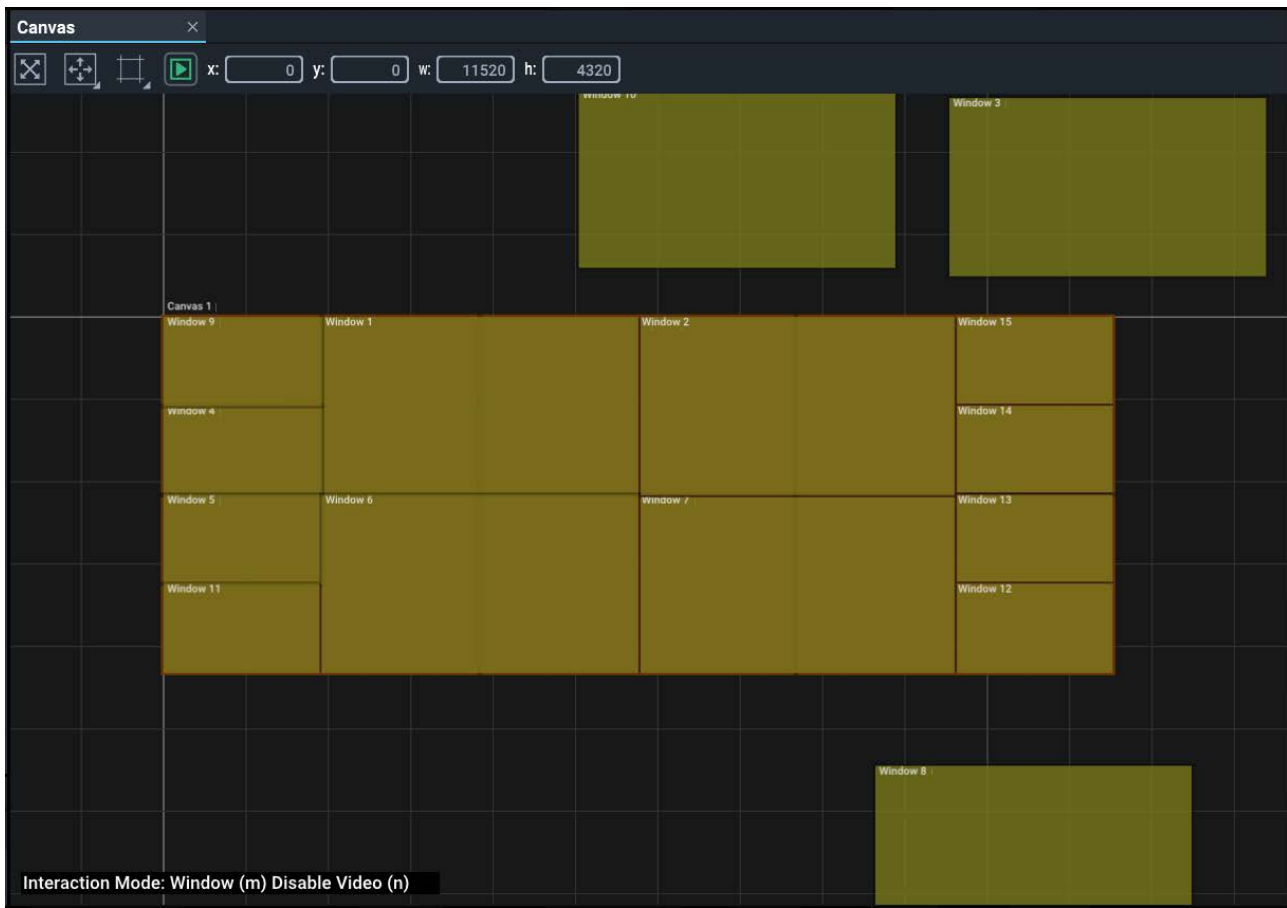


The default file location for Presets is set in *Configuration > Preferences*. Name the preset.

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 command to recall the Preset.

➤ See User Guide [M710 for Compere JSON External Control](#).

#### A note on Canvas Window layering (z-order)

Presets are not aware of any other elements on a canvas and so cannot relate to overall arrangements that have changed since they were saved. For this reason, recalling a preset or loading a clone will not necessarily assert the z-order of windows that you require. These can only be changed with 'Modify Layers' enabled in the Canvas, and by selecting Windows in turn to bring them over each other or to the top.

**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

---

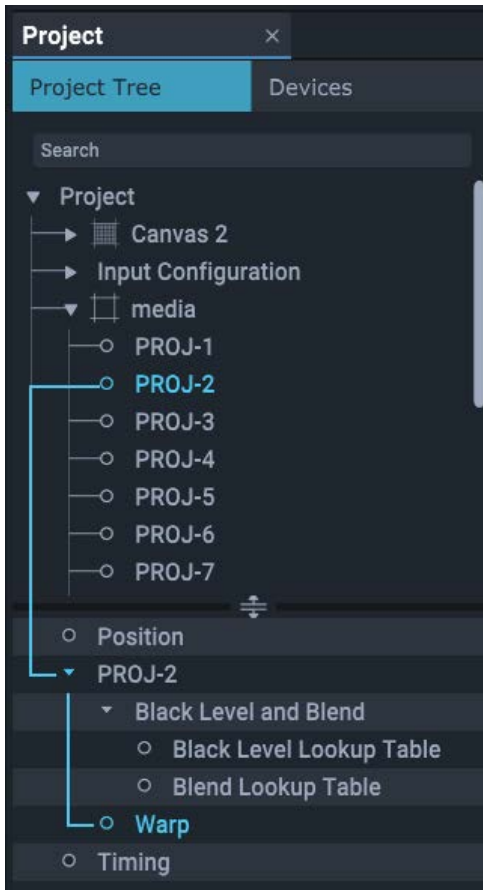
The non-Windowing (Warp/Blend) version of Juggler firmware instead allows for the placement of MPCDI data to define media location, warp, blend and black level correction for a projection system.

In File Explorer, browse to the MPCDI file (\*.mpcdi format) and simply drag it into the upper Canvas assembly panel. It may take a few moments to ingest the data, depending on its complexity, before a complete populated canvas appears with the data. .mpcdi files compress and collate a number of other files, including warp, blend and black level correction files, and the mpcdi.xml file itself.

By default, when MPCDI files are imported, Compere automatically distributes the imported (unprocessed) file to all Juggler devices within the project. This can block UI operation for an unacceptable time, and is not necessary. To avoid this, in the Project panel, make sure Juggler Settings > Distribute MPCDI Files is unchecked. The imported file will still be processed and incorporated into the project, which will be distributed.

The MPCDI name will now appear as a branch in the Project panel. The name of the source (here 'MPCDI media') equates to the Canvas. Branches below these (here named 'PROJ-#') equate to Canvas Outputs. These carry the properties for black level, blends and warps.

Names for the Canvas and Canvas Outputs are derived from the imported MPCDI file.



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.

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

Black Level, Blend and Warp filenames will be shown in the properties for these branches.

### A note on MPCDI 'Canvas' naming

Where an imported MPCDI file lies in the Project panel depends on its name (the xml display id) because the panel is alpha sorted. Its name can be edited, but if the MPCDI is reimported, it will not be identified as the same file and will duplicate the data.

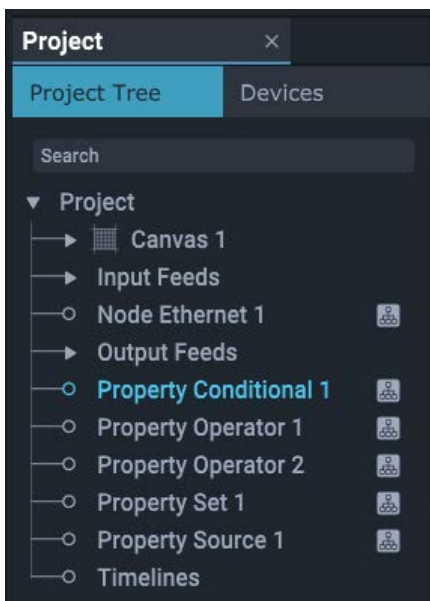
## Nodes: Programming Nodes

Programming node types include sources, values, and logic processes. Flexible and powerful, they can be combined to create more complex interactions in Juggler systems, including automated events such as IntelligentSource™ failover.

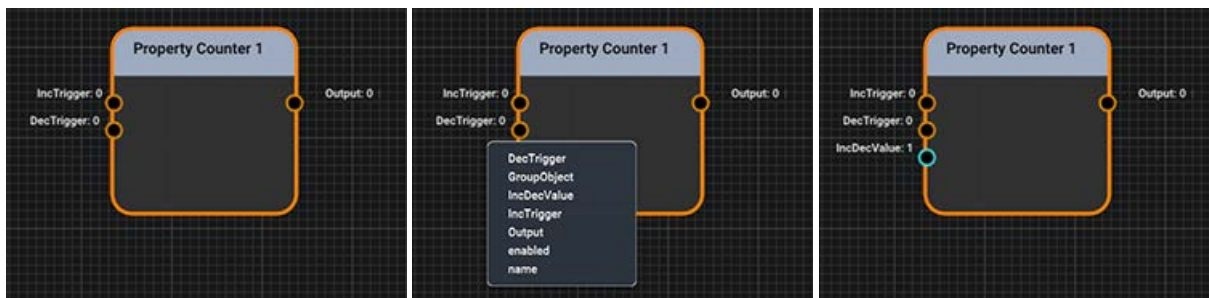
Programming nodes can be added and manipulated by external controls, but are connected in the Nodes panel.

Drag nodes from the Resources panel into a Nodes panel and arrange or select them in the same way as Juggler nodes. Once added to a Project, they are listed in the Project panel, and when selected, their properties are shown for edit in the Properties panel.

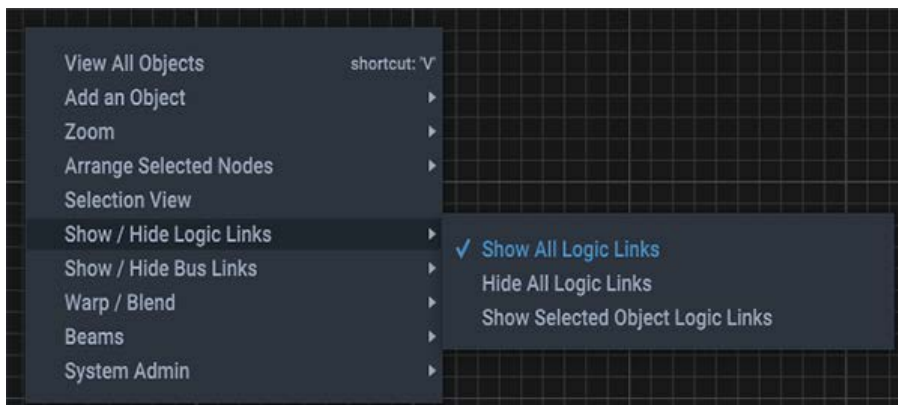
Nodes that have been added into the Nodes panel are marked, just as Jugglers are, by an icon. Absence of this icon does not mean they are inactive, since a node removed from the Nodes panel remains in the project with all its properties.



- Nodes can be **removed** (Right-click menu) from the Nodes panel, but remain an interactive part of the Project, losing the 'displayed' icon in the Project panel. They can only be **deleted** from the Project panel.
- **Pins** available to a processing node can be added to what is otherwise a minimalist representation, as required. Select the node, Shift+Right click for properties to add, and select the pin to add:



- Nodes have default **names** denoting their function but can be named for their purpose. The name appears in the title bar of the node.
- Those **prefixed** 'Node' initiate things, those prefixed 'Property' relate things via a process.
- Pins shown in orange take **integer** values, those taking **strings** are green, and blue indicated **double** values.
- Pins in a Nodes panel are **linked** by Shift+Left mouse dragging from source pin to target pin. In this way, values are communicated, transformed, used as triggers for other events, and so on.
- Link lines can be hidden, shown or shown for only selected nodes. Right-click in the Nodes panel for the context menu:



- Links are **deleted** by hovering over the target pin and pressing Delete.
- Pin value **colour**: pins that exert control show yellow in the Properties panel, those that are being controlled show green.
- Additional dynamic properties can be added to nodes from the shortcut menu (Shift+Right click).
- [Presets and Clones](#)<sup>146</sup> can be created from selected nodes.

## Nodes that do things

### Node Ethernet

#### Description

An Ethernet receiver or transmitter. Used for external control messaging within a Compere system, over UDP or TCP.



In client mode this is a message sender, in server mode it is a message receiver.

### Properties

- **Name:** defaults to node type, rename for function as required
- **Connected:** can be switched on or off by giving the value 1 = yes.0 = no
- **Host name:** name or IP address of source to use
- **Input:** as server, string message received
- **Output:** as client, string to send by UDP/TCP
- **Port:** default: 23
- **Type:** UDP, or TCP Client (send a string) or TCP Server (receive a string)
- **Input** – remove end of line chars from input strings
- **Output** – remove end of line chars from output strings
- **UDP read port:** default 7781

## Node Generator

### Description

Generates a waveform output. Constantly changing property values might include a position, colour or intensity, for example. The timing source is that of the server Juggler.

### Properties

- **Name:** defaults to node type, rename for function as required
- **Amplitude** (relative units)
- **Frequency** (Hz)
- **Mode:** sine or square
- **Phase:** shift in degrees (180 is inverted)
- **Time:**
- **Output:** amplitude value

## Node Logger

### Description

Receives an input to append events to a named log file in a named location.

## Properties

- **Name:** defaults to node type, rename for function as required
- **Directory:** log file destination
- **Filename:** log file name
- **Input:** link to a node output being tracked for changes

## Nodes that relate and process values of other things

### Property Conditional

#### Description

Connects to a configurable number of integer inputs. (The integer part of a double input will be read.)  
This node has two functions:

- A. To **switch** an output to be one of several inputs.  
Another process or an external control could provide the switch index.

OR

- B. To scan a series of inputs to give **priority** to the lowest-index positive (>0) input.  
The output is the index number of the respective input.

#### Properties

- **Name:** defaults to node type, rename for function as required
- **Number of Inputs:** add number required
- **Output:** Input pin number that meets the conditions.  
Default is *value* of Input 1  
Default in Priority Mode is the number of the first input (i.e. = 1).

#### Switch mode

- **Switch Index:** the number of the input to pass through (**output** becomes the value of this input).  
(Index: the index of Input\_n is n. The index of Input\_2 is lower than that of Input\_3.)

#### Priority mode

- **Priority Mode:** when enabled, this scans instead for the lowest-index positive (>0) input.  
The output **value** is the **pin number** of the lowest-index input that has a positive value.  
*Example:* Input\_1 == 0, Input\_2 == 50, Input\_3 == 20. Output == 2
- **+ Priority Latch:** the **pin number** of the input to latch onto (stop scanning when matched).  
When this pin has the lowest-index positive input, the output stays latched onto this input, even if

lower-index inputs become positive again. Unlatch (Priority Latch = 0) to restart the input scanning.

## Property Counter

### Description

Receives triggers that match set values to increment or decrement the output by a step value.

### Properties

- **Name:** defaults to node type, rename for function as required
- **Increment Trigger:** set the value to compare with an input, and when it matches, increment the output
- **Decrement Trigger:** set the value to compare with an input, and when it matches, decrement the output
- **Increment / Decrement Value:** set the value by which to increment or decrement the current output when triggered.

## Property Operator

### Description

Configurable integer logic operator based on a flexible number of inputs with selectable mode to generate an integer output value.

### Properties

- **Name:** defaults to node type, rename for function as required
- **Number of Inputs:** add number required
- **Input:** auto-labelled Input\_# as generated
- **Mode:** AND, OR, NOR, NOT, INVERT, ADD, SUBTRACT
- **Output:** integer value to be output

### Mode output values

Boolean:	No. of inputs evaluated	Condition
AND	Any	True (output =1) when all inputs are non-zero
OR	Any	True (output =1) if any input is non-zero

Boolean:	No. of inputs evaluated	Condition
NOR	Any	True (output =1) when all inputs are zero
NOT	1	False (output =0) when value of input 1 is non-zero
Non-Boolean:		
INVERT	1	changes the sign of any value in input 1
ADD	Any	output = sum of all inputs
SUBTRACT	Any	output = sum of all other inputs subtracted from the value of input 1

## Property Set

### Description

Apply values to a number of set destinations (paths) when triggered by a defined value.

### Properties

- **Name:** defaults to node type, rename for function as required
- **Input Trigger:** input value received.
- **Trigger Value:** when the input matches this value, all outputs are triggered.
- **Number of Outputs:** add number required. Generates a [Path] and [Output value] pair for each.
- **Path #:** auto-labelled Path\_# as generated. Enter the property paths required. Paired with:
- **Output #:** auto-labelled Output\_# as generated. Sends this value to the paired path when trigger value is met.

## Property Source

### Description

Creates string outputs with values derived from paths. The path of any property in Compere can be copied using Shift+Right click on the property label in the Properties panel. Paths can be read in the left of the status bar when hovered over in the Properties panel.

### Properties

- **Name:** defaults to node type, rename for function as required

- **Number of Outputs:** add number required. Generates a Path and Output pair.
- **Path #:** auto-labelled Path\_# as generated, paired with:
- **Output #:** auto-labelled Output\_# as generated; adopts the value from the paired path.

## Presets and Clones

---

All Compere project information is held in blocks of xml. This means that you can save, recall and distribute a complete show setup with all its players and all their parts. It also means that you can save just the parts, which might mean the way you set up a canvas populated with just outputs, or just a set of windows with their sizes and positions, or a group of Jugglers configured in a particular way, and so on.

Elements in presets and clones are selected copied from the Project panel, not from what is placed and visible in the assembly panels.

### Presets: file type .pre

If you want to use these same things in the same project, you can use these xml blocks to recall the values, which can be particularly useful in switching between various preset window layouts ([Create and Recall Canvas Arrangements](#)<sup>(130)</sup>).

Presets capture arrangements and settings for components in a specific 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.

Presets cannot be used in 'foreign' projects.

### Applying presets to a selection

**Example use:** to save repetitious work in configuring many Jugglers, this option allows the core configuration properties of one Juggler of a kind to be saved and recalled into many selected Jugglers of the same type in the Nodes panel, not just to Jugglers of matched UUIDs. Non-transferrable properties such as Name are not copied over.

### Clones: file type .clo

If instead you want to create standardised components, for example a display wall of a number of UHD panels, with or without windows, these can be saved as clones. They are no longer a member of any project, but can be loaded as new elements into a different project, from where they can be added into assembly panels and incorporated and repurposed as necessary.

Unlike presets, clones are saved without their project UUIDs. This is why they are 'loaded' (new), not 'recalled' (remembered).

## Options

### Save As: Selection

This option in both types will save out just those items selected in the Project panel. If the selected items require a parent element (e.g. a Window or Output) a default parent will be inserted into the clone file. Child elements cannot be transferred to a different canvas when reloaded. When using canvas clones, it is best to 'save as canvas'.

The exception would be that if you already have more than one canvas, you may want a clone of just one of them.

### Save As: Canvasses

This option will save out every canvas parent and every child output and window. No selection is needed.

### Export

Whilst the save options will direct you to the preset and clone default locations in the Compere root location (set in Preferences), export will direct you to local locations on the PC using the Compere UI at the time.

## Logs and Diagnostics

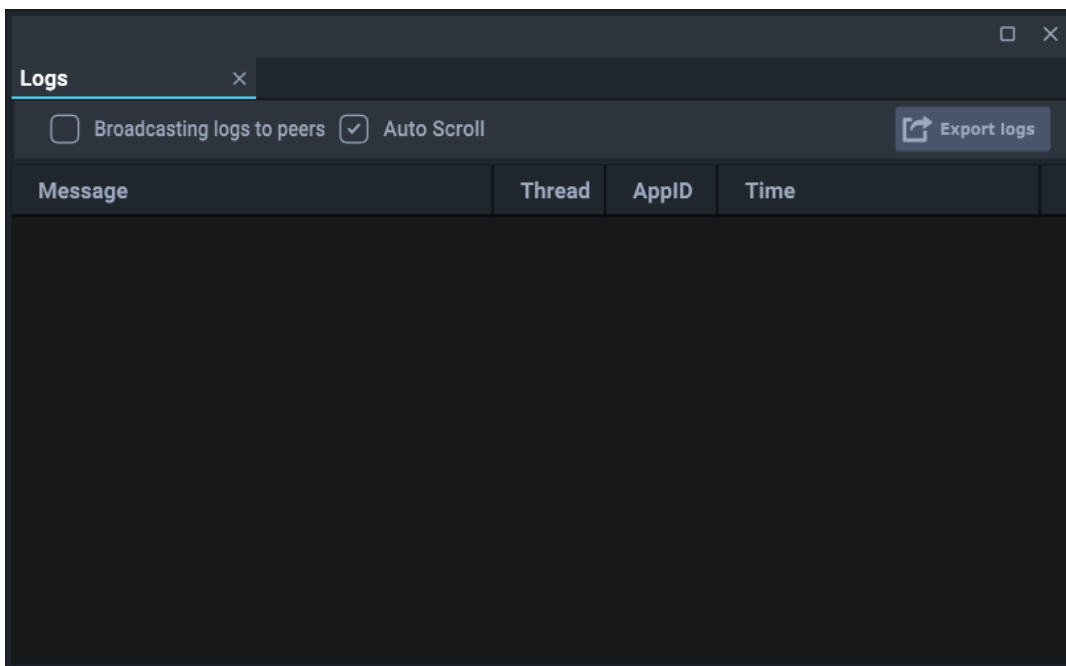
Compere generates a log file (CompereLog.txt) which can be very useful for system debugging and troubleshooting. On a Windows machine this can be found by default in: C:\Users\[USERNAME]\AppData\Local\Compere\logs.

➤ Default file locations can be edited. See File Locations in [Preferences Panel](#)<sup>42</sup>.

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

- On Mac these should be found in ~/Library/Compere/logs.
- On Juggler 1, the logs can be found in /share/compere/logs.
- On Juggler 2 the location is /7thApps/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: C:\Users\[UserName]\AppData\Local\compere\logs).

**Note:** The times in the log files are UTC. The start of each run includes the local time.



## Collated System Reports

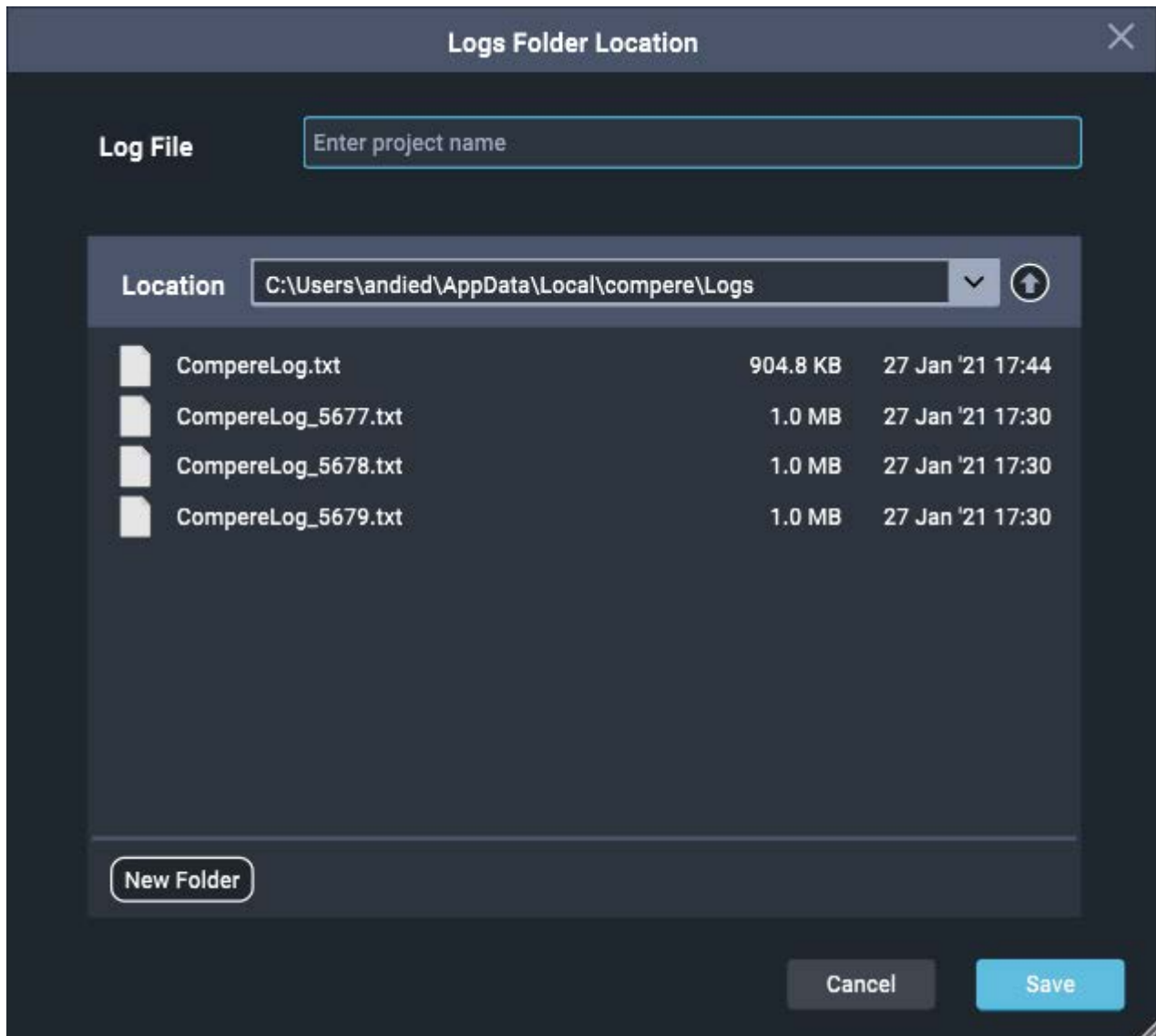
Sometimes things can go wrong, and in a complex system, gathering various log files from devices is burdensome. Collated system reports can be quickly and consistently gathered together, with all information that might be required to allow any observed behaviour to be investigated.

Compere System Reports (collated logs) can be generated from the *Help > Generate System Report* menu item. They can also be generated via JSON External Control using the [generate-system-report](#) command.

A named location for these can be set in *Configuration > Settings > Preferences*.

## Gather Project Data

Lists logs per device 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](mailto:support@7thsense.one).

## 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)
Broadcast Logging UDP	Default, editable via comms.ini or in Preferences	5596
Watchdog control and comms	Fixed	5599
Juggler to Compere UDP hello packets	Fixed	9001
Juggler thumbnail data to Compere, UDP	Fixed	9009
Juggler 1 internal comms	Fixed	9012
Juggler 1 Console TCP	Fixed	9013
Juggler 2 TCP xml transmit	Fixed	9014
Juggler 2 TCP xml receive	Fixed	9015
Juggler to Compere TCP Console transmit	Fixed	9016
Compere to Juggler TCP Console receive	Fixed	9017
Multicast discovery UDP	Fixed	5353
External control comms, default UDP receive	Yes, in Preferences	8001
External control comms, default UDP transmit	Yes, in Preferences	8002
TCP (ASCII Ext Ctrl)	Yes, in Preferences	2300
External control adapter TCP	Yes, in Preferences	5584
Time Server	Default, editable in Preferences	18888
Front Panel touch display	Fixed	5000
Failover Heartbeat	Yes, in Preferences	5678
Project Group Data	Yes, in Preferences	5679

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.

## Activate a Codemeter USB Dongle

---

7thSense software – Compere, Actor and Conjurer performer roles – and some third party inclusions, are licensed options.

Wibu CodeMeter provides this software protection, licensing and security. A licence is required for Compere and third party software such as Notch, and can be provided as a software licence or USB dongle.

### Register a Licence

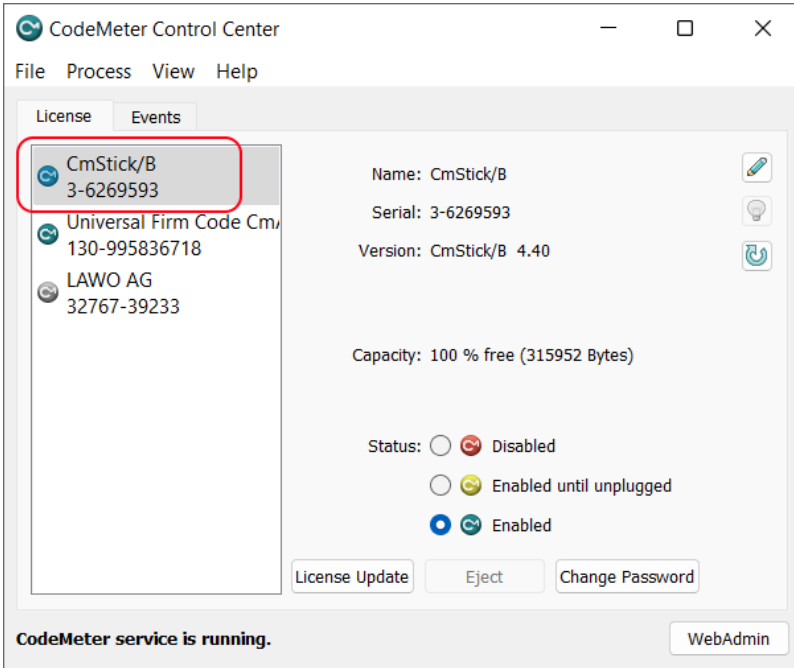
Insert USB CodeMeter licence dongle into the PC



If not already available, install Wibu CodeMeter control centre application

<https://www.wibu.com/uk/support/user/downloads-user-software/file/download/11352.html>

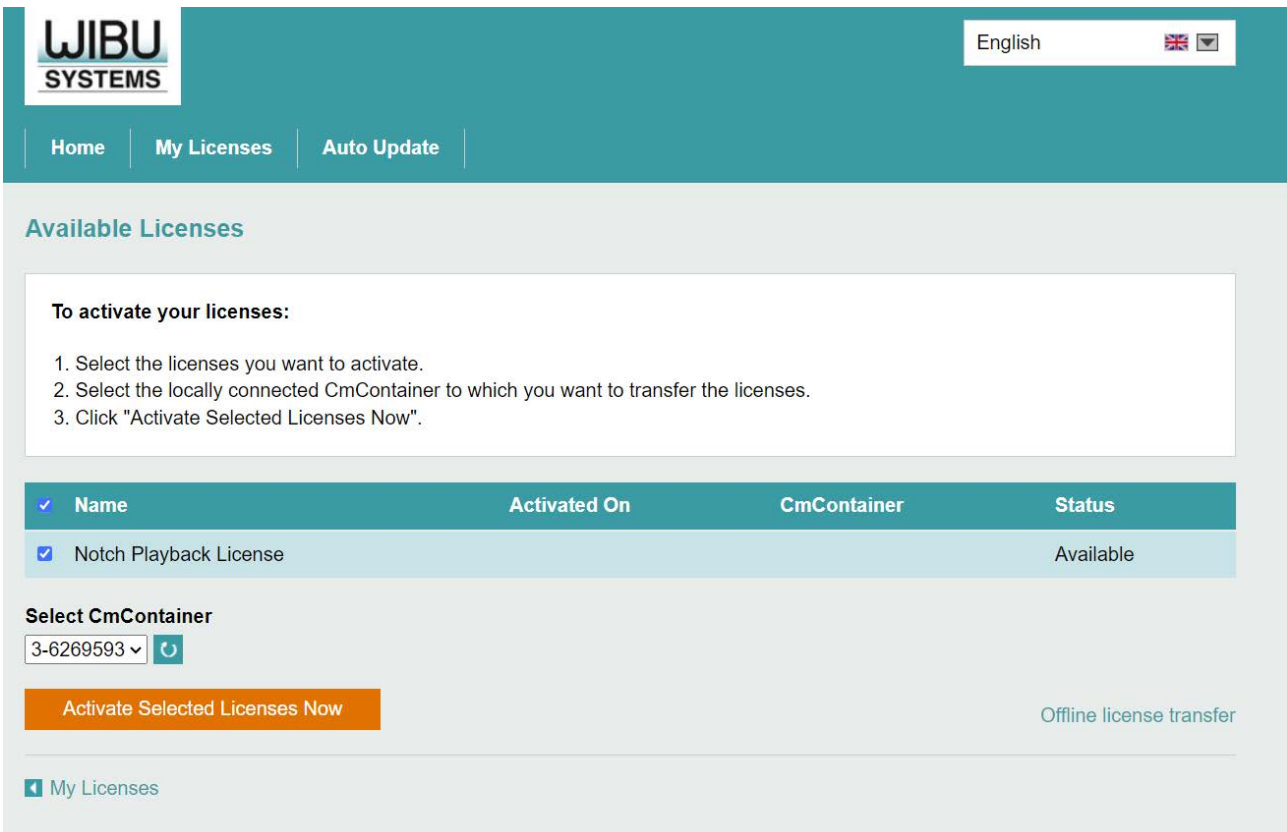
Open the CodeMeter control center app and verify that the USB dongle is present. This will be described as below, CmStick/B. Take note of the dongle number, in this case 3-6269593.



Follow the hyperlink to the licence code as provided by the company supplying the licence. For example, 7thSense, or Notch.

*Example:* <http://lc.codemeter.com/76932/depot/get.php?id=VYM5B-W3M3L-FPQ5Q-EDKSM-3UXTS>

Assuming the licence hasn't already been registered, following the link should look like this:



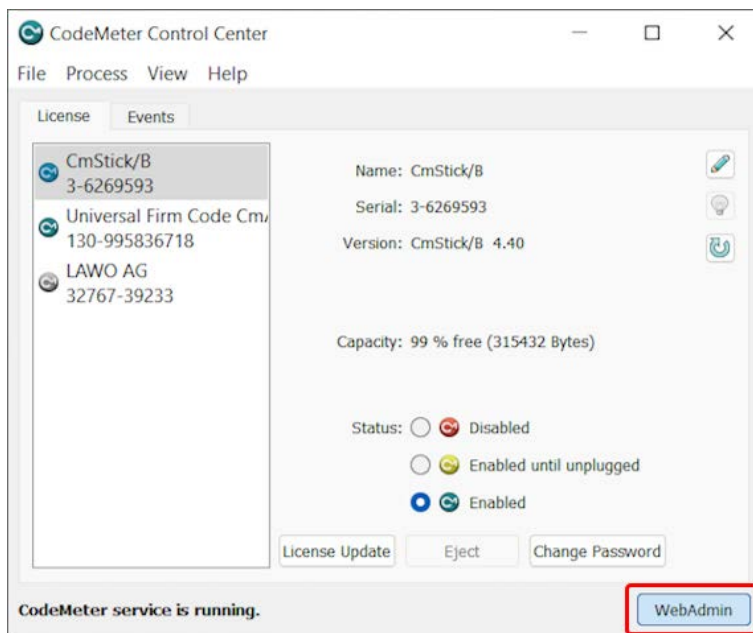
The licence dongle is recognised where the licence status is shown as 'Available' and 'Select CmContainer' shows the dongle number listed.

Assuming its licence number does match what is expected, click the orange 'Activate Select Licenses Now' button. This will automatically download and transfer the licence to the dongle.

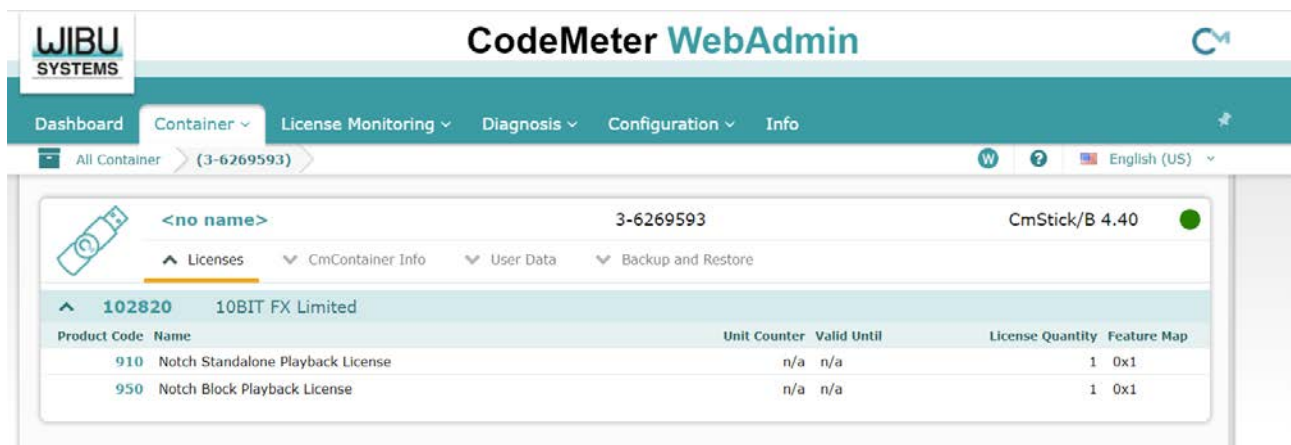
If the licence number doesn't match what you expect, refresh the page to reread the system for available dongles. You can have multiple dongles connected simultaneously, but this will increase the risk of transferring incorrectly.

## Read Dongle Licence Information

If we want to read the licence information back from a dongle, from the Codemeter control panel application we can select the dongle in the list, then click Web Admin. This will open a webpage.



The Wibu WebAdmin page provides information about each licence, its features and expiry:



## Document Information

---

Date	Document edition	Software version	Revision Details	Author/Editor
December 2023	1	Compere 1.0 build 7	New release for Compere 1.0 build 7	Andie Davidson
November 2024	2	Compere 1.0 build 7	Revised Vulkan version requirement from 1.1 to 1.2	Kenny Showler

## A

- activate CodeMeter licence 152
- add, select, edit, delete, remove objects 20

## B

- black level 137
- bus bridging 99
- bus capability 93
- bus connections 93

## C

- canvas 6
- canvas arrangement 130
- canvas outputs 121
- canvas preset 130
- canvas windows 127
- clones 29, 146
- colour meanings 75
- colours in nodes 66
- Compere network ports (list) 151
- connecting Juggler inputs, outputs 113
- context menus 62
- control PC 6
- control PC specification 9
- custom web panels 33

## D

- diagnostics 148
- docking 27
- document information 155

## E

- EDID spoofing 116
- election of project group server 40
- external control 6

## F

- failover 6
- failover systems 85

## G

- gather project data 148
- Glossary, Compere and Juggler 12

## I

- install a control PC 9
- IntelligentSource 85

## J

- Juggler connector properties 89
- Juggler display configuration
  - canvas assembly panel 117
  - canvas object 117
  - canvas panel shortcuts 117
  - interaction mode (canvas panel) 117
  - toolbar, canvas panel 117
- Juggler network ports (list) 151
- Juggler node 66, 75
- Juggler pins 75
- Juggler properties
  - Bus 96
  - defaults 73
  - GPIO 73
  - internal processor 72
  - System Monitor 74
- Juggler Properties: video channels 82
- Juggler unit properties 70
- Juggler Video interface timing 92

## K

- keyboard shortcuts (Compere) 16

## L

- licence 29
- log files 148

## M

- main menu 29
- minGUI 65
- MPCDI 137

## N

- NDI monitoring 6, 113
- NDI source name per channel 86
- network configuration 37
- network discovery 37
- nodes (Juggler) 6
- Nodes assembly panel 75
- nodes, add dynamic pins 139



## O

Output Canvas Bus (OCB)  
  canvas wizard 109  
  wizard 109  
  workflow 103  
overview 6

## P

panels 27, 29  
panels menu 33  
panels, arrangement 20  
PC as server only 65  
positioning elements 27  
presets 29, 130, 146  
profiles 11  
programming (and logic) nodes 139  
project group  
  server failover 40  
project groups 6, 37  
project panel 20, 60  
projects, project files 58  
properties panel 20  
property path strings 56

## R

read CodeMeter Licence info 152  
recall preset 130  
resources 20

## S

server failover 37  
server-client roles 37  
settings  
  Canvas panel 49  
  failover, election 42  
  file locations 15, 42  
  input configuration 51  
  Juggler 46  
  mini map 50  
  Nodes panel 50  
  output configuration 51  
  preferences panel 42  
  Visualise panel 46  
settings and preferences 36  
settings, configuration 41  
shortcuts 27  
software licensing 152

status bar 56  
system admin 64  
system config table 29

## T

task progress 54  
task tracker 54  
technical support 148  
terms and definitions 12  
toolbar: Nodes panel 75

## U

update Compere on Jugglers 9

## V

version 9, 29  
video channel  
  video measure 87  
video channels 75

## W

warp and blend 137  
wizards (menu) 53  
workspaces 27  
  save workspace 20

## Z

z-order layers, windows 127





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  
4207 Vineland Rd  
Suite M1  
Orlando, FL 32811  
USA

T: +1 407 505 5200

