



SYMPHOLIGHT 5.5

User Manual

Read the User Manual carefully. Subject to modification without prior notice. The current status of the manual does not correspond in all parts to the current status of SYMPHOLIGHT. Check our website for updates to this manual. Typographical and other errors do not justify any claim for damages. Modification of the product is prohibited. Except for internal use, relinquishment of the instructions to a third party, duplication in any type or form - also extracts - as well as exploitation and / or communication of the contents is not permitted.

This document is designed for all users of the product.

All product names and trademarks mentioned in this manual are trademarks of their respective owners.

Downloads and more information at:
www.ecue.com

Edition:
08.2025

Published by:
Traxon Technologies Europe GmbH
Karl-Schurz-Strasse 38
33100 Paderborn, Germany

©2025 Traxon Technologies Europe GmbH
All rights reserved

Traxon Technologies Europe GmbH
Sales Operations
Karl-Schurz-Str. 38
33100 Paderborn, Germany
+49 5251 54648-0
support@ecue.com

Available for free download from www.ecue.com

or

<https://eu.traxon-ecue.com/products/sympholight/>

Scan QR code for quick access.



Questions and comments to: sympholight@ecue.com

Table of Contents

1	Preface	14
1.1	About This Manual	14
1.2	How to Read This Manual	14
1.3	Special Markup	14
1.4	Special Notes	15
1.5	Online Help Functions	15
1.6	Common Time Format in SYMPHOLIGHT	15
2	Introduction	15
2.1	Digital Lighting Control	15
2.2	About SYMPHOLIGHT	16
3	System Prerequisites	17
4	Installation	17
4.1	Trial and Full Version Mode	18
4.2	Credit System	18
4.2.1	Ordering Credits	19
4.2.2	Upgrading Credits	20
4.3	Automatic Updates	20
4.4	Keyboard Shortcuts	21
5	Principles	22
5.1	Operational Sequence	22



5.2	Migration from Lighting Application Suite to SYMPHOLIGHT	22
-----	--	----

6	Application Concept	23
----------	----------------------------	-----------

6.1	Setup	23
6.2	Content	24
6.3	Automation	25
6.4	Output	25
6.5	Log	26

7	User Interface	28
----------	-----------------------	-----------

7.1	Main Menu	28
7.2	Usability	30
7.3	Status Bar	30
7.4	Hidden Menus	30
7.5	Hide / Show Description of Properties	31
7.6	Saving Project Files	31
7.7	Examples & Templates	32
7.8	Closing SYMPHOLIGHT	32

8	Settings	32
----------	-----------------	-----------

8.1	Configure Backup	32
8.2	Application Settings	35

9	Setup	45
----------	--------------	-----------

9.1	Setup Window	45
9.2	Adding Devices - Device Manager	46



9.2.1	Virtual Devices	49
9.2.2	Butler Classic, XT, XT2 and S2	50
9.2.3	Video Capture Device	50
9.2.4	Non-DMX and non-DALI Devices	51
9.2.5	Attached Devices	51
9.2.6	Glass Touch T6, T6R, T12 and IR	52
9.2.7	Serial Modbus Devices (e.g. Genius)	52
9.3	SYMPHOLIGHT Instances	52
9.4	Icons in the Mode Selection	54
9.5	Placing Fixtures - Build Mode	54
9.5.1	Moving Fixtures	56
9.5.2	Changing View	56
9.5.3	Custom Fixtures	57
9.6	Configuring Properties	57
9.6.1	Network Addresses	58
9.6.2	Additional Functions	59
9.6.3	SYMPL Node and SYMPL Fusion modules Properties	59
9.7	Layout	65
9.7.1	Groups	66
9.7.2	Effect Rectangle	66
9.7.3	Export, Import and Merge of Layouts	67
9.7.4	Configuring Universes	68
9.8	Patch Mode for DMX and e:pix	68
9.9	DALI Devices	70
9.9.1	Placing DALI Devices	72
9.9.2	Addressing DALI Devices	72



9.9.3	Grouping DALI Devices	76
9.9.4	Additional DALI Functions and Features	77
9.10	RDM Devices	80
9.10.1	Addressing RDM Devices	81
9.10.2	Sorting RDM Devices	85
9.11	Art-Net Devices	85
9.12	sACN Devices	87
9.13	Background Mode	88
9.14	Test Mode	88
9.14.1	DMX in Test Mode	89
9.14.2	DALI in Test Mode	89
9.15	Patch Files	90
10	Content Window	91
10.1	Principles	91
10.2	Window Structure	92
10.3	Preview Window	93
10.4	Control Bar	94
10.5	Layers	96
10.6	Media Library	97
10.7	Properties Window	98
10.8	Adjust Mapping to Fixtures	99
10.9	Cues	99
10.10	Sequence and Cue Properties	100
10.11	Fading	100
10.12	Scenes and Scene Editor	101

10.13	Keyframing	104
10.14	Working with Gradients	104
10.14.1	Gradient Editor	105
10.14.2	Single Color	105
10.14.3	Gradient Markers	105
10.15	Effects	105
10.16	Creating New Effects	106
10.16.1	Example of a New Effect: Transformation plus Render	107
10.17	Media Library	107
10.18	DALI Scenes	108
10.19	Properties of Content in Layers	108
10.19.1	Gamma Correction	109
10.19.2	Configuration of the White Channel	110
10.20	Sequence Properties	111
10.21	Cue Properties	113
10.22	Special Function Keys	113

11	Automation	115
11.1	Principles	115
11.2	Workflow Designer	116
11.3	User Interface of the Workflow Designer	117
11.4	Tab Control Bar	118
11.5	Creating a Workflow	119
11.6	Connecting Blocks with Links	120
11.7	Signal Flags	120
11.8	Working with Blocks	122

11.8.1	Target Based Blocks	124
11.8.2	DALI Line Blocks	125
11.9	Configuring the Settings of Blocks	125
11.9.1	Setting Port Types and Modes	126
11.9.2	Data Types for PortIn and PortOut Values	127
11.9.3	Setting Data Types of Ports	128
11.9.4	Setting Input Types for PortIn Values	128
11.9.5	Manually setting PortIn data values	129
11.9.6	Adding an Event Based Trigger	130
11.9.7	Setting Port modes	131
11.9.8	Enabling and Disabling Blocks	132
11.10	Adding Tabs, Deleting, and Renaming	133
11.11	Adjusting the View	133
11.12	Adjust Arrangement	134
11.13	Search for Blocks	134
11.14	Block Library: Workflow Blocks	135
11.15	Workflow Blocks in Detail	145
11.15.1	Adding Templates	145
11.15.2	Project Initialized	147
11.15.3	Time Trigger	148
11.15.4	Periodic Trigger	150
11.15.5	Astronomic Trigger	151
11.15.6	Data Changed Trigger	155
11.15.7	Push Button	156
11.15.8	Toggle Button	160
11.15.9	Image	164



11.15.10	Password Input	166
11.15.11	Text Input	169
11.15.12	Text Label	171
11.15.13	Fader	173
11.15.14	Inline Frame	176
11.15.15	Color Box / Color Wheel	178
11.15.16	Page Navigation	181
11.15.17	Value	183
11.15.18	State To Event	186
11.15.19	Event To Value	187
11.15.20	Combine Text	189
11.15.21	Value to Event / State	191
11.15.22	Converter	194
11.15.23	Permit Event	196
11.15.24	Math Formula	197
11.15.25	Addition	200
11.15.26	Subtraction	202
11.15.27	Multiplication	204
11.15.28	Division	206
11.15.29	Moving Average	208
11.15.30	Time Condition	210
11.15.31	Compare Condition	212
11.15.32	State Condition	214
11.15.33	AND	216
11.15.34	OR	218
11.15.35	NOT	220

11.15.36	Threshold	221
11.15.37	Fade	223
11.15.38	Measure Compare Adjust (PID)	227
11.15.39	Human Centric Lighting	231
11.15.40	Toggle	236
11.15.41	Single Delay	237
11.15.42	Multi Delay	238
11.15.43	Control Sequence	243
11.15.44	Adjust Sequence	245
11.15.45	Control Cue	247
11.15.46	Trigger Point	249
11.15.47	Generic Control Sequence	250
11.15.48	Mutex Group	252
11.15.49	Ballast Group and Daylight Harvesting	255
11.15.50	Button	269
11.15.51	Absolute Input	273
11.15.52	Generic Input	275
11.15.53	Presence Sensor	278
11.15.54	Light Sensor	280
11.15.55	Ballast Line	282
11.15.56	Generic Input Line	291
11.15.57	Button Line	296
11.15.58	Input Automation Line	300
11.15.59	Presence Sensor Line	306
11.15.60	Feedback Channel Line	310
11.15.61	DALI Observer	313



11.15.62	Build RDM Command	317
11.15.63	Send RDM	319
11.15.64	Observe RDM	321
11.15.65	Analog / Digital Contact	326
11.15.66	Dry Contact	330
11.15.67	Relay	333
11.15.68	Modbus	335
11.15.69	Observe Controller	337
11.15.70	Observe System	339
11.15.71	Receive DMX	340
11.15.72	LCE2 Display	342
11.15.73	Glass Touch T6	344
11.15.74	Glass Touch T6R	348
11.15.75	Glass Touch T12	355
11.15.76	Glass Touch IR	356
11.15.77	Sympl Fusion Action Buttons	360
11.15.78	Sympl Fusion Dry Contacts	363
11.15.79	Send E-Mail	366
11.15.80	Send MQTT	369
11.15.81	Receive MQTT	371
11.15.82	Send UDP	373
11.15.83	Receive UDP	376
11.15.84	Send & Receive Serial	379
11.15.85	Write Logbook	384
11.15.86	Send to Network	386
11.15.87	Receive from Network	388

11.15.88	Script	390
11.15.89	Set Property	391
11.15.90	Get Property	392
11.15.91	Check Property	396
11.15.92	Call Method	398
11.16	Scripting	399
11.17	Examples for SYMPHOLIGHT Scripting	402
11.17.1	Adding an Entry in the SYMPHOLIGHT Logs	402
11.17.2	Automation of an Effect	403
11.17.3	Handling UDP Messages	404
11.18	Supported Function Param Types	406
11.19	ActionPad Designer	409
11.19.1	Element Properties	414
11.19.2	Assigning Actions to GUI Elements	415
11.19.3	Accessing the ActionPad on Mobile Devices	415
11.20	SYMPHOLIGHT Remote Console	415
<hr/>		
12	Output Window	417
12.1	Principles	417
12.2	Player Window Icons, Commands and Functions	418
12.3	Z-Position and Priority	418
12.4	DMX Output Window	419
<hr/>		
13	SYMPL Fusion & SYMPL Core S(p)	420
13.1	System Overview	420
13.2	Configuring the SYMPL Fusion or SYMPL Core S(p)	423

13.3	SYMPL Fusion network configuration	426
13.3.1	Default network properties	427
13.3.2	Network configuration with the web interface	427
13.4	Art-Net and sACN on SYMPL Fusion or SYMPL Core S(p)	429
13.5	Date, Time & NTP settings	430
13.6	Using SYMPL Fusion or SYMPL Core S(p) as License Key for SYMPHOLIGHT	432
13.7	Uploading a Project	432
13.8	Selecting the ActionPad	433
13.9	Using Video with the SYMPL Fusion or SYMPL Core S(p)	433
13.10	Maximum Number of SYMPHOLIGHT Instances	433
13.11	Accessing the Logfiles	433
13.12	Using the Scheduler	433
13.13	Automatic Node Replacement Mode	433
13.14	Updating SYMPHOLIGHT in the SYMPL Fusion or SYMPL Core S(p)	434
13.15	Configuring SYMPL Fusion modules	436
13.15.1	Scenario: Modules from SYMPL FUSION Base Device	436
13.15.2	Scenario: New modules for not specified slots	437
13.15.3	Scenario: New modules for designated slots	437
13.15.4	Return of control	438
13.16	SYMPL Fusion Test mode	438
13.18	Updating SYMPL Fusion modules	439
13.19	Resetting to Factory Default	441
14	REST API	442
14.1	REST API Setup	442

14.2	SYMPHOLIGHT Web API	442
14.3	Examples	446
<hr/>		
15	Scheduler	447
<hr/>		
16	RDM in Detail	452
<hr/>		
16.1	RDM Glossary	452
16.2	Remote Device Management (RDM)	452
16.2.1	Topology	452
16.2.2	Addressing	453
16.2.3	Discovery	453
16.2.4	Update Discovery	453
16.3	RDM View in Setup Tab	453
16.3.1	Update Optimization	454
16.3.2	Adjust Attribute Update Priority	454
16.4	RDM in Workflow-Designer	455
16.4.1	Observe RDM	455
16.4.2	Build RDM Command	456
16.4.3	Send RDM	457
16.5	RDM in Scripting	458
16.5.1	Finding RDM Devices	458
16.5.2	Working with RDM Devices	458
<hr/>		
17	DALI Feedback Function	461
17.1	Introduction	461
17.2	Sending Feedback	462

17.2.1	DALI Button Block	462
17.2.2	DALI Feedback Channel Line Block	464
<hr/>		
18	SymphoCity	468
<hr/>		
19	Glossary	469
<hr/>		
20	Notes	473



1 Preface

1.1 About This Manual

With version 5, SYMPHOLIGHT has become a very powerful and full-featured tool. As with every complex tool, every documentation has to decide if it wants to show any microscopic feature and function, or to guide the user through the workflow and give him or her a basic understanding. We decided to take the second approach. To make yourself familiar with SYMPHOLIGHT, one way is to install and discover what SYMPHOLIGHT does and is able to provide. Help text and explanations, as well as mouse-over help will tell you what functions are available. The other way is to read the manual and trying the various features. The best way will be a way in between.

This manual is not intended to explain every drop-down menu and every context menu. Instead, it explains the different parts and windows in SYMPHOLIGHT, and there is always a chance to find even more features. It is a basic guide line for SYMPHOLIGHT to help you to understand the main features and what to do with them. If you have questions or comments, please visit our forums on our website to get additional support or help. On our website you can also find updates on the user manual, release notes (also presented inside the software) and further information about topics around the e:cue world.

1.2 How to Read This Manual

The manual is divided into three main sections. The first section gives a general overview how to install SYMPHOLIGHT and prerequisites as an overview. The second section shows basic principles. The third section describes in detail the single areas and feature sets.

1.3 Special Markup

Preconditions and critical functions are shown like this (example):



Change the Windows password within six weeks after installation. Otherwise the user account will be locked and cannot be unlocked.

Useful hints, short-cuts and functions are shown like this (example):



You can zoom in and out with the scroll wheel of your mouse. Keeping the mouse wheel pressed, will zoom in steps of 10 percent.

In the whole manual, all keywords describing key elements of SYMPHOLIGHT will be written with capitals. Examples: Properties Window, Workflow Designer or Media Library.



1.4 Special Notes



Throughout the manual, the term “multi-select” is used. This means the possibility in Windows to select groups or sequences of elements by keeping the Ctrl or SHIFT key pressed to select elements in a list. See the Windows documentation for a description.



There is also a term “long-click” used. A long-click are two mouse clicks on an object on two different positions. This makes the object editable, e. g. to change the name of an element or a value.

1.5 Online Help Functions

Mutual Exclude Collection ID	0
Priority	1
Start Position	+0.0:0:0.0

Priority
 (Integer)

Render Priority. Sequences with higher priority are overwrite values of sequences with lower priority.

When editing properties in the Properties Window, a help text and a description is displayed in the lower part of the Properties window.

1.6 Common Time Format in SYMPHOLIGHT

SYMPHOLIGHT uses time values in the format dd.hh:mm:ss:ms, where dd = days, hh = hours, mm = minutes, ss = seconds and ms = milliseconds.

2 Introduction

2.1 Digital Lighting Control

Over the last decades digital systems replaced conventional analog lighting control instruments. However, this is not the only change in lighting technology. The story began with single RGB elements like pars and flood lights, developed over moving heads and more complex fixture types. Today, media LED solutions with thousands, if not ten-thousands of RGB pixels became a standard.

e:cue released its first version of the Lighting Application Suite (in short: LAS) in 2002. The LAS was an application suite to setup a configuration with engines, fixtures and everything else to have a physical representation of the installation. Later you can create illuminations, light projects with effects and much more to run on your selection of fixtures. The LAS was the central application for creating light projects and controlling

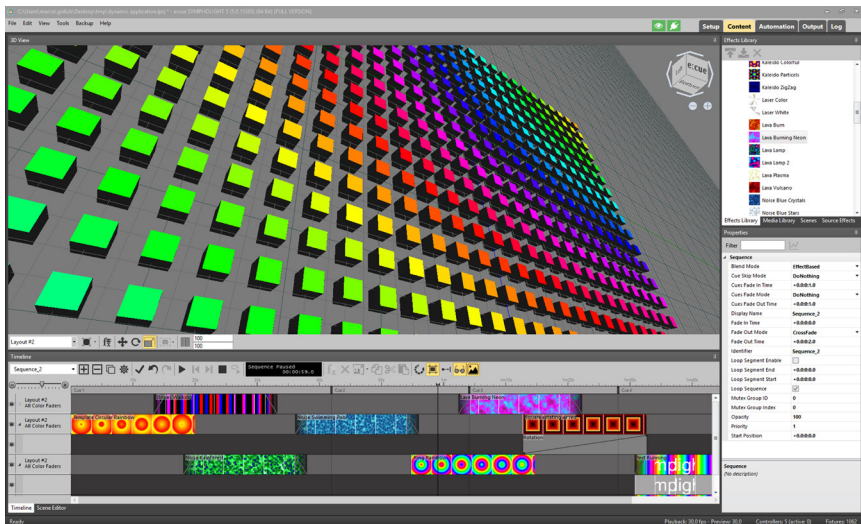


complex lighting projects, and to control your network and all devices in it. In its life cycle the LAS began in the world of some cuelists with triggers and actions and grew into the world of media displays and complex effects.

The overall development of lighting went from fixture-oriented to media-oriented applications, as mentioned before with high resolution media walls and very complex solutions. Today configurations with hundreds of DMX universes are no exception today, they are daily business for Traxon and e:cue. For this reason, the LAS needed a successor that reflects these new world of lighting control. This is e:cue SYMPHOLIGHT.

2.2 About SYMPHOLIGHT

SYMPHOLIGHT is a software application to design, test and program medium to very complex lighting designs and lighting applications.



SYMPHOLIGHT creates a single application for the complete cycle of a project, from setting up fixtures and interfaces (“patching”), design and test the project and execute and control the project in real-time.

Main features include:

- Dynamic lighting with media and effects and static lighting with Scenes.
- Timeline design.
- Layer concept for media like images and videos, and effects.
- Built-in effect library with color and intensity effects including fades, channel effects and more.
- Built-in fixture library with generic and vendor-specific fixtures.
- A flexible and powerful Workflow Designer for automation and control.
- Separation between design and execution state for flexibility and stability.

- Separated preview of projects in design and execution.
- Extendible with C# programming language.

3 System Prerequisites

For running SYMPHOLIGHT the system must meet the following specifications:

Minimum configuration for all sizes:

Microsoft® Windows® 10 64 Bit

Internet Explorer 10 or better

8 GB RAM

A minimum of 20 GB hard disk space

Dual-Core processor or better. Quad-Core processor is recommended.

DirectX 9.0c supported graphics adapter

A minimum of one Ethernet interface for connections to engines and interfaces

Recommend configuration for small to medium projects (~50.000 pixels)

Quad-Core desktop processor (e.g., Intel® Core i5)

nVidia® GeForce® GTX 760 class graphics adapter

Second Ethernet adapter dedicated to e:net

Recommend configuration for large projects (200.000+ pixels)

16 GB RAM

Quad-Core high-end desktop processor (e.g., Intel Core i7)

nVidia GeForce GTX 780 class graphics adapter

Second Ethernet adapter dedicated to e:net



4 Installation

On the e:cue Lighting Control Engines SYMPHOLIGHT is pre-installed.



Please make sure the installed SYMPHOLIGHT Version on your Lighting Control Engine is the latest version. Otherwise proceed as follows below.

On all other systems proceed as follows:

- You must have administrator rights to install e:cue SYMPHOLIGHT on Windows.
- Insert a CD/DVD with the SYMPHOLIGHT installer into the drive or plug in an USB stick with the SYMPHOLIGHT installer.
- Open the Windows Explorer, navigate to the CD/DVD drive or USB drive.
- Double-click on the installer, the installer will ask for permission and acceptance of the copyright conditions.

- Select the installation language.
- Select the components. The Dongle Driver is necessary to enable all Dongle functions, e.g. the full version mode and the use of credits.
- After the installer has finished, click OK.

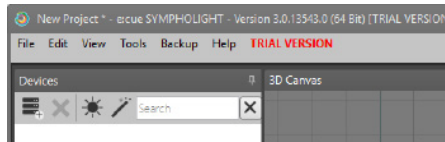
The installer places an optional short-cut to SYMPHOLIGHT on the desktop. Double-click the icon to start SYMPHOLIGHT. If application updates are available, SYMPHOLIGHT will download and install updates.



After starting the LCE3, do not shut down the system until the desktop shows.

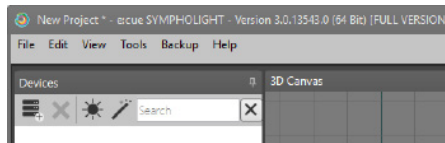
4.1 Trial and Full Version Mode

SYMPHOLIGHT requires a protection dongle to run in full mode. This dongle is part of the delivery when you purchase SYMPHOLIGHT as separate software. In the Lighting Control Engines, the dongle is part of the server system and already built-in on the main board.



Without a dongle, SYMPHOLIGHT runs in trial mode with full functionality, but without generating any output, neither DMX or for relay or serial Nodes. If you forgot to plug in the protection dongle while starting SYMPHOLIGHT, plug it in while SYMPHOLIGHT runs and it will switch to full mode with generated output data.

You can check the current status at [Help | License and Credits](#) overview in the main menu.



All existing LAS Programmer Enterprise users with a LAS Enterprise dongle can use SYMPHOLIGHT's full version without additional costs. The SYMPL Core S(p) and SYMPL Fusion can also be used as a license key for the SYMPHOLIGHT software. For more information see „[13.6 Using SYMPL Fusion or SYMPL Core S\(p\) as License Key for SYMPHOLIGHT](#)“ on [page 432](#).

4.2 Credit System

SYMPHOLIGHT allows the use of external systems and interfaces over open and standard protocols. Certain universe types and features require credits for operation. Without credits, there will be no data output for these elements. All devices, if DMX



engines or user terminals, from e:cue or Traxon do not require credits and the number of connections is practically unlimited.

There are different types of credits:

eDMX credits: Art-Net and sACN devices consume credits and the maximum number so is limited. Universes that require credits are marked with a dollar symbol (\$) in the Setup Window. One credit is consumed for one physical DMX universe. The IP addresses used do not count, only the number of universes. If the credits range is exhausted, an error message is displayed.

Features: Contains the credits for the APIs for the Scheduler („15 Scheduler“ on page 447), Content Exchange (all ContentManagement Rest API methods), Gateway Communication (enables MQTT communication for „11.15.80 Send MQTT“ on page 369 and „11.15.81 Receive MQTT“ on page 371 and makes MQTT settings available), SymphoCity Play Board, Energy Board and Fixture Board.

A credit each is necessary to enable the respective API with data transfer.

You can extend your credits by the online credits extension function. For example, at [Help / License and Credits overview](#) in the main menu, or via web interface of SYMPHOLIGHT or of the SYMPL Fusion or SYMPL Core S(p). Order credits and upgrade them to your dongle here.



Credits are bound to the dongle or license number stated in the ordering. Therefore ordering credits must be performed on the very same SYMPHOLIGHT instance that will employ the new credits. Additional SYMPHOLIGHT instances need to be charged separately.

License and Credits overview	
License status:	Full version without content management.
Dongle type:	ecue Enterprise dongle
Dongle serial number:	<input type="text"/>
Available eDMX credits:	2
Available Features:	<no features active>
<input type="button" value="Order credits"/> <input type="button" value="Upgrade dongle from file"/> <input type="button" value="Close"/>	

4.2.1 Ordering Credits

To order credits, click the “Order credits” button. Enter the required order information and the desired credits: for eDMX credits define an amount of credits; to enable features tick the Order checkbox of the feature.



Order additional dongle credits

Additional credits to order: 0

Features:

- ☒ Order SymphoCity Scheduler
- ☒ Order Content Exchange
- ☒ Order Gateway Communication
- ☒ Order SymphoCity Play Board
- ☒ Order SymphoCity Energy Board
- ☒ Order SymphoCity Fixture Board

API to schedule events.
API to replace media files.
Gateway communication over MQTT
License for the SymphoCity Play Board
License for the SymphoCity Energy Board
License for the SymphoCity Fixtures Board

Proceed Cancel

Proceed and send an email with the order text to your e:cue distributor. Multiple order texts (e.g. for different SYMPHOLIGHT instances) per mail are possible.

Ordering credits for a SYMPL Fusion or SYMPL Core S(p):

- Ordering credits must be performed on the very same SYMPL Fusion or SYMPL Core S(p) device that will employ the new credits.
- It is only possible to create and employ one order for a SYMPHOLIGHT instance. If you need to create another order for the same instance, the previous order must have been employed beforehand.

4.2.2 Upgrading Credits

To employ the ordered credits, wait for a response from your distributor. Once you have received a dongle upgrade file from the distributor, click the “Upgrade dongle from file” button in the window. Select the received file and your credits in SYMPHOLIGHT will be extended.

4.3 Automatic Updates

On start-up, SYMPHOLIGHT is able to find and install updates automatically. This requires an active Internet connection during start-up. The [check for updates](#) can be enabled or disabled in the application settings from the main menu.

e:cue SYMPHOLIGHT Update Check

An update for e:cue SYMPHOLIGHT is available:

Installed Version: 1.0.0

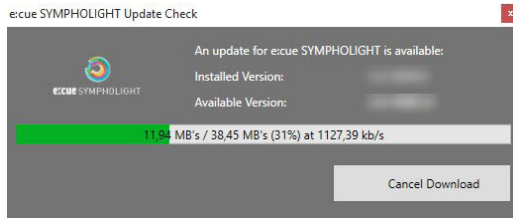
Available Version: 1.0.1

☐ Do not check for updates (re-enable in Tools/Settings...)

Download and Install update now... Skip Update

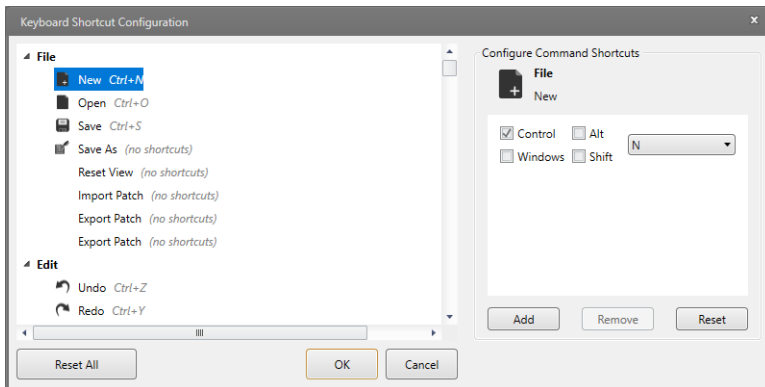


After downloading the update, the update will be installed. This again requires administrator permissions on the system.



4.4 Keyboard Shortcuts

As many users like to work with keyboard shortcuts instead of clicking menu entries, SYMPHOLIGHT supports a predefined set of keyboard commands. To add keyboard shortcuts or edit the existing definitions, use [*Tools / Configure keyboard shortcuts*](#) from the main menu.



5 Principles

5.1 Operational Sequence

The procedures with SYMPHOLIGHT for most projects will follow these steps:

- Setup your DMX fixture configuration and addressing scheme. Distribute DMX universes and start addressing to the target settings of your DMX output engines.
- Place your DALI ballasts and input devices, e.g. sensors.
- This configuration scheme is called a patch. You can also import patch source files created with the Patchelord of the Lighting Application Suite from version 5.x and higher.
- Insert and setup your external engines and interfaces, e. g. SYMPL Core Sp, SYMPL dmx Nodes or SYMPL dali Nodes.
- Setup DMX universe distribution and scan your DALI lines, assign DALI addresses and create DALI groups.
- Create your show under Content. Import videos or images in various formats. All imported media will be converted to a final format and placed in the Media Library. This media can be dragged into the sequence. Or:
- Apply color and intensity effects as main content and drag them into the sequence. Create fades, overlays and other visual effects. The applied content can be previewed in real-time.
- Create automation and control in the Automation area with the Workflow Designer.
- Create an end-user control interface with the graphical user interface Action Pad.
- Publish your show and execute it.



5.2 Migration from Lighting Application Suite to SYMPHOLIGHT

If you are familiar with the e:cue Lighting Application Suite (LAS), SYMPHOLIGHT will represent a completely new interface and way of programming. The LAS was mainly cuelist-oriented.

SYMPHOLIGHT works with a timeline concept similar to the Sequencer and Emotion FX. In general it is possible to work with the concept of cues. This will be explained in a later chapter “Cue creation”.

Setting up the fixture configuration is similar to the Patchelord. You can even import patch files (.pdoc) from the Patchelord into SYMPHOLIGHT and reuse your configuration. There are even more similarities with the LAS, like the Media Pool or the Fixture Library. It is not possible to use or import shows from the Programmer, as the concepts and formats are completely different.

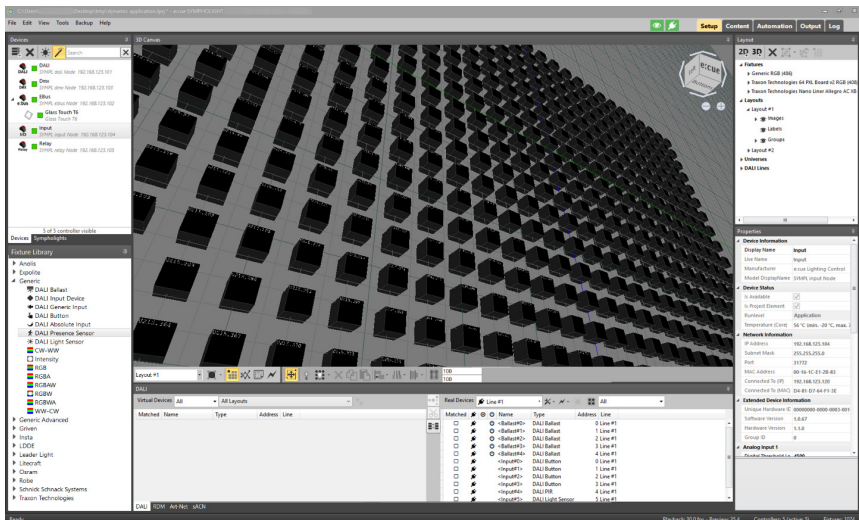
An important change is the layer concept. The Sequencer of the LAS did not support a layer concept, the tracks were played in parallel. Also the Live FX window is now replaced by effects from the effects library directly. Also, the Workflow Designer of SYMPHOLIGHT is much more powerful and flexible compared to the Programmer's Trigger Engine.

Scripting in the Programmer of the LAS was done with a C-like dialect. SYMPHOLIGHT also allows scripting, but now in C#. C# in SYMPHOLIGHT runs in a .NET environment, so SYMPHOLIGHT scripting is far more powerful than the C scripting in the Programmer of the Lighting Application Suite. Use the standard C# and .NET documentation to learn more about C#.

6 Application Concept

6.1 Setup

The first step is setting up your lighting configuration with fixtures and all other components and elements. You can select engines and interfaces like SYMPL Nodes or Butlers and add them to the project. As the next step, you select fixtures from the list on the left side. You can place single fixtures or arrays of fixtures like pixel boards. To represent your later configuration, fixtures can be bound to DMX universes and addresses.



Also, fixtures can be grouped. You can put fixtures in one or more than one group. Grouping has a direct effect on the later content creation, as fixtures in one group have their own Layer in the Timeline. If you do not create any specific groups, all placed fixtures will belong to one Layer. SYMPHOLIGHT supports a 2D and a 3D model for creating the Setup. The Setup Window also contains the RDM section, where RDM is supported.



6.2 Content

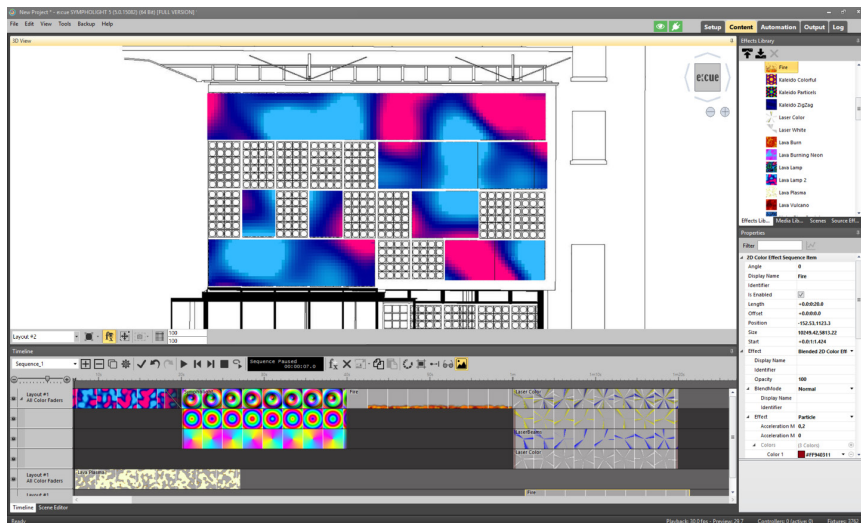
In the Content step, the show with media, effects and Scenes are created.

This happens in the Content Window. Understanding the features and principles is important to use SYMPHOLIGHT effectively and efficient. Content is built from one or more Layers. Each Layer is comparable to layers in graphic or image editors.

SYMPHOLIGHT works with a timeline and layers as the main control element.

Nevertheless, static content is still possible as Scenes and will be shown in detail later.

At the beginning, there are layers generated for all Layouts in the Setup. This is the collection of all color channels of all fixtures that have RGB channels, or a Layer for fixtures with intensity channels only. When placing fixtures in the setup step, fixtures will be assigned to the respective Layer. Building more than one group during setup will create additional Layers. In these Layers you can place media like images or videos, or effects from the Effect Library.

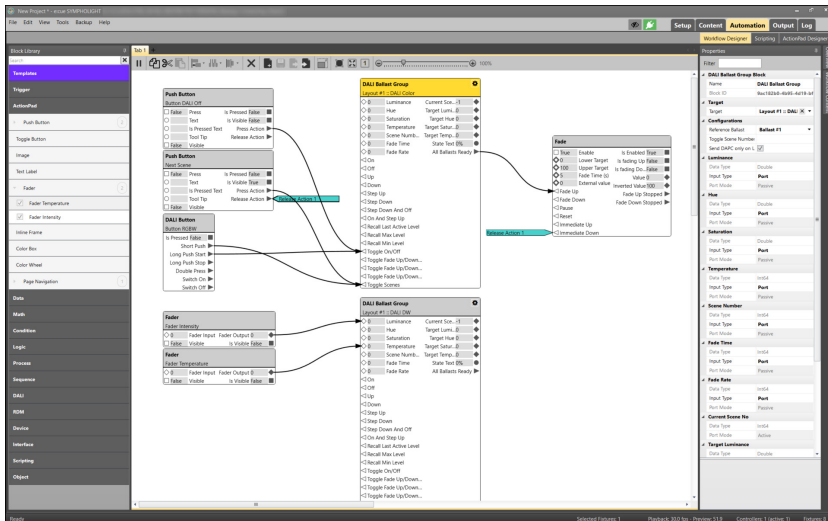


Layers higher up in the layer hierarchy have a higher priority in visibility. Effects can be color or intensity effects. Be aware, that all fixtures are assigned to the two initial layers for color faders and intensity faders automatically, depending on the data in the Fixture Library during setup of the fixtures. To place media or effects from the windows in the upper right, drag-and-drop them from Effects or Media Library into the Layer in the Timeline. After placing elements in the Layer, properties for the element can be adjusted in the lower right Properties Window.

One set of Layers and content builds a Sequence. You can have as many Sequences as you need. Each Sequence realizes a part of the show or application. You can add or delete Sequences as necessary. Later, when adding Automation, Sequences are the main target of operation.

6.3 Automation

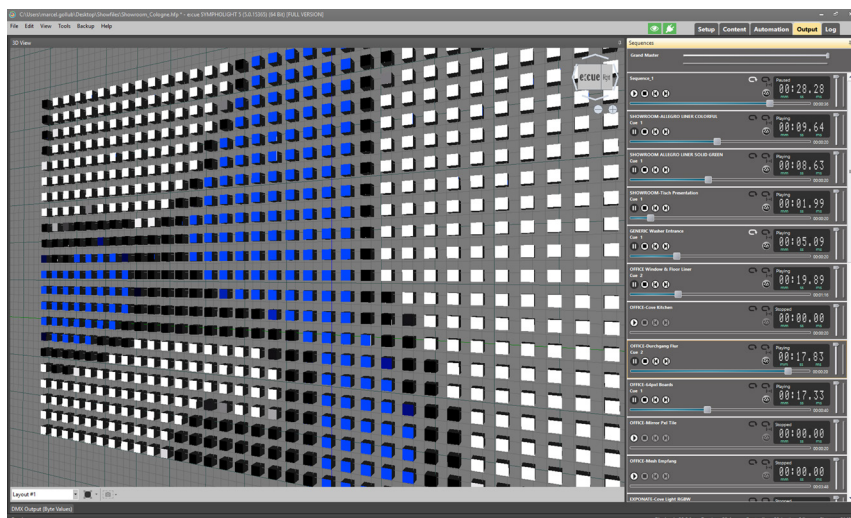
In the Automation Window, you will define a Workflow for control and user interfaces. With Automation you prepare the show how it will be executed live and how to react on external signaling. After the fixtures and content have been created, you can setup automation. The Automation Window consists of the Workflow Designer, the Scripting Window and the GUI Editor.



The Automation Window contains all functions that are necessary to make your show really complete. This window contains a graphical interface for the end user, so the user has only access to functions he needs and not the interface of the original SYMPHOLIGHT during development of a show. Automation also defines the reactions to external signaling, time-controlled execution and necessary calculations.

6.4 Output

In the Output Window you can control the Sequences and monitor the output of SYMPHOLIGHT.

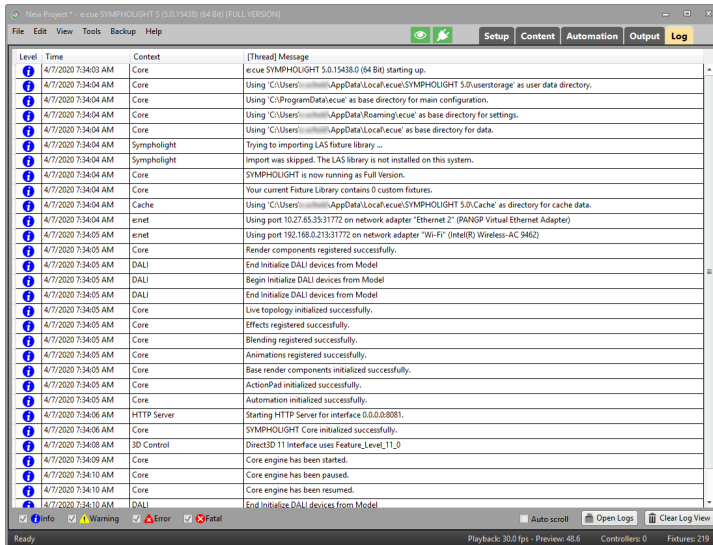


From the Output Window, you can start and stop sequences. The Output Window shows what is really sent to the fixtures. If you edit the content in the Content Window, the changes are not valid until you make the changes valid with the check symbol. Editing happens in the so called blind mode, except the parameter *Auto save changes to content* is set in the Application Settings.

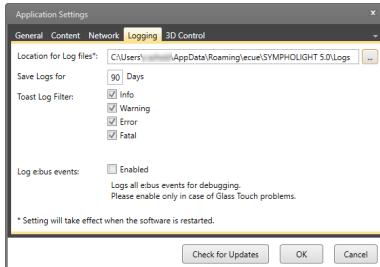
6.5 Log

The Log is the last tab of the main window of SYMPHOLIGHT. Here error messages and warnings are collected from the whole system.





The Log Window has a selection for the levels of entries and an Autoscroll feature now.



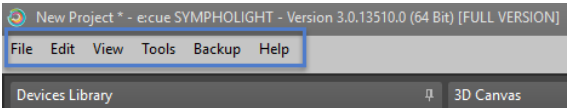
In SYMPHOLIGHT main settings, via the main menu, you can also select which toaster (in Window's taskbar) messages are logged to the global logging file.



7 User Interface

7.1 Main Menu

In the upper left corner you find the main window menu for SYMPHOLIGHT. Here is a list of selectable commands and their functions.



File	
New	Create a new project.
Open	Load an existing project.
Save	Save the current project.
Save As ...	Save the current project under a new name.
Import Patch ...	Import a SYMPHOLIGHT (.slpatch) or Patchelorn (.pdoc) patch files.
Export Patch ...	Export current setup as SYMPHOLIGHT patch file (.slpatch).
Export Project with Media	Export project to file with used media (.hfp).
Recently Used Project Files	A list of the most recent used projects.
Exit	Leave SYMPHOLIGHT.

Edit	
Undo	Reverse the last change.
Redo	Repeat the last undone action.

View

Reset View	Reset the window distribution to the default values.
------------	--

Tools

Application Settings	Define application options like paths, network interface and logging. For details see „8.2 Application Settings“ on page 35.
----------------------	--

Bulk Change	Change properties for all media or effects.
-------------	---

Configure Keyboard Shortcuts	Assign keys to functions.
------------------------------	---------------------------

Backup

Configure Backup	Enable Backup and define role. For details see „8.1 Configure Backup“ on page 32.
------------------	---

Reupload Project	Reupload a Backup project file on the Backup engine.
------------------	--

Give control back	Check whether a Master server is available.
-------------------	---

Reset Backup	Reconnect the Backup system.
--------------	------------------------------

Help

System Manual	See the Manual.
---------------	-----------------

Scripting Help	SYMPHOLIGHT Scripting API.
----------------	----------------------------

First Steps Manual for DMX	Quick introduction to DMX installations.
----------------------------	--

What's New?	Change Log.
-------------	-------------

License and Credits overview	View license and credits status („4.1 Trial and Full Version Mode“ on page 18). Order new credits „4.2 Credit System“ on page 18.
------------------------------	---




About Application information.

7.2 Usability

In the upper right corner, left of the concept Windows, you find the two buttons to handle general behavior of SYMPHOLIGHT.






Define which content is provided to the fixtures. Important when running a live operation.

Green: Actions of every shown Window provide content for the fixtures (Patching test mode and Content preview included). Helpful during commissioning phase.

Red: Blind Mode activated. Actions in the Setup and Content Window are ignored when shown - only actions in the Automation and Output Window provide content to the fixtures (Patching test mode and Content preview excluded). Recommended during live operation.



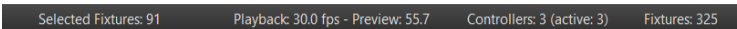
Define e:net (network) communication to hand over the control of devices to other SYMPHOLIGHT instances, e.g. SYMPL Core Sp.

Green: e:net communication active. This very SYMPHOLIGHT has the control over devices.

Red: e:net communication disabled (muted). Another SYMPHOLIGHT instance can take over control over controller.

7.3 Status Bar

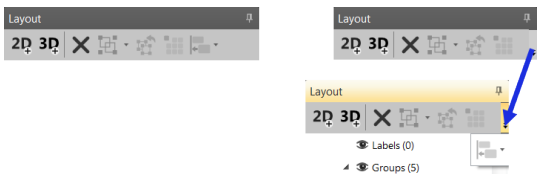
At the bottom, the status bar informs about the current status of the software.



You can see the performance status as well as how many controllers you have in the network and in the project, how many fixtures in the project and selected.

7.4 Hidden Menus

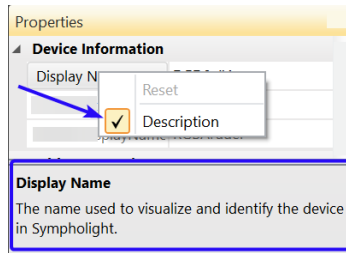
When sub windows are resized, so that not all buttons can be displayed, an additional drop-down mark is displayed on the right border of the window:



Click on the small triangle in this mark to display the hidden drop-down lists.

7.5 Hide / Show Description of Properties

The Properties throughout SYMPHOLIGHT are accompanied by descriptions of each property. The description is located in the lower part of the Properties Window. By default, the descriptions are visible. You can hide / show descriptions by right-clicking any property and disabling / enabling **Description** in the context menu. Toggle the visibility for each main window separately.

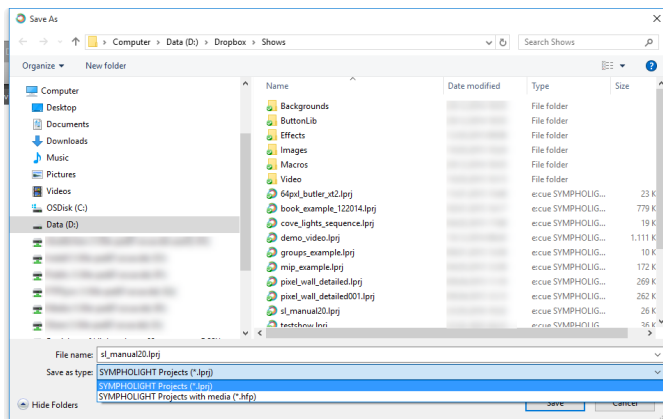


7.6 Saving Project Files



It is not possible to open a project with an older version of SYMPHOLIGHT after saving it with a newer version.

SYMPHOLIGHT project files use the extension *.lprj. You can open or save project files as with any other Windows application. SYMPHOLIGHT remembers the last recently used project files, the list size of the most recently used files can be set in the application settings. Set the default location for projects in the application settings. There are two types of project files. Select the type while saving the project file.



Use the *.lprj version to save only setup, automation and content data. Use the *.hfp version to save the project including media or custom fixtures in the project file. This way, you are independent from your Media Library when using the project on another system. The *.hfp file also includes customized fixtures and user defined effect templates. To save only Setup results, see „9.15 Patch Files“ on page 90.

7.7 Examples & Templates

SYMPHOLIGHT provides a couple of example show files. They are stored at C:\ProgramData\ecue\SYMPHOLIGHT 5.2\Examples.

In addition, you can also find templates for ActionPad themes, Effects and RDM commands at C:\ProgramData\ecue\SYMPHOLIGHT 5.2.

How to open/ load the templates:

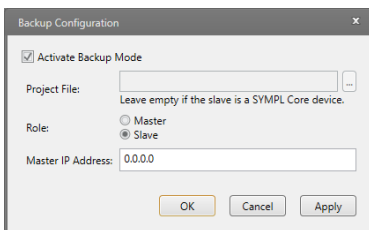
- ActionPad themes (*.appt): Import in Theme Library (see „Theme Library“ on page 413).
- Effects (*.slet): Import in Effects Library (see „10.15 Effects“ on page 105).
- RDM commands (*.slrt): see „Build RDM Command - Advanced“ on page 457.

7.8 Closing SYMPHOLIGHT

To close SYMPHOLIGHT use the Exit command in the Main Menu / File or close the application window. Shut down the system properly before turning it off (exception: SYMPL Core engines) to ensure all processes, e.g. saving, are properly executed, completed and closed.

8 Settings

8.1 Configure Backup



With Backup / Configure Backup you define a master/backup configuration for fail-safe operation. If the Master engine fails, a Backup system takes over control automatically. At first, make sure that the web server is activated on the Backup engine and that it has the same credentials as the Master. Run both engines simultaneously only after the configuration is

completed.

Set the check mark for Activate Backup Mode. Select the project file used as the Backup project file (exception „Core Engine as Backup engine“ on page 28).

Define the role of this engine, if it is the Master or the Backup engine.



Define the IP address for the Master engine and the Backup engine: On the Master engine define the IP address of the Backup engine; on the Backup engine define the IP address of the Master engine. Master and Backup engines must be configured to share the same IPv4 subnet. Both must have e:net enabled for that subnet.

The Backup status is displayed in the title bar of SYMPHOLIGHT and in the Web UI at the Settings page. A restart of the engines might be necessary after the configuration is completed. Alternatively use the [Reset backup system](#) as described below. The backup project file and the currently used SYMPHOLIGHT project do not need to be identical. This can be used for example when a Core Engine is configured as the Backup engine for an LCE Master engine and provides a minimal project file.

States of the Backup system:

Status	Description
Disabled	The Backup is not enabled.
Connecting	The engine is waiting for the other partner to get online.
Connected	The connection between the Master and the Backup engine is successfully established.
Upload Pending	The upload of the project file from the Master to the Backup engine is pending because the Backup engine is started with a delay of 5 seconds. Appears only on a Master engine.
Uploading	The upload of the project file is in process.
Error	An error occurred. Details are available in the Log. The Backup system remains in this mode until SYMPHOLIGHT is restarted.
Timeout	If the Master gets offline for more than 60 seconds, the Backup engine changes to Timeout mode for 30 seconds and waits for the Master to come back online.



Backup

If the Timeout period of 30 seconds has elapsed without the Master coming back online, the Backup engine changes to the Backup mode. The Backup engine loads its project in SYMPHOLIGHT and takes over control. The Backup engine stays in the Backup mode until either its Backup feature gets disabled or the Backup engine is changed to Master. All controllers will be connected to the Backup engine even if the original Master comes back online.

To control the Backup system, two functions are available at [Backup](#) and in the web interface:

Use [Give control back](#) for the Backup engine to check whether a Master engine is available. If a Master is available, the Backup engine hands over control to the Master.

Use [Reset backup system](#) to reconnect the Backup system after an error occurred and got fixed.

- You can reupload a Backup project file on the Backup engine with [Backup / Reupload Project](#).
- If a Backup engine cannot establish a connection to the Master, the Backup engine changes into Timeout mode after 60 seconds. After another 30 seconds, the Backup engine loads its project in SYMPHOLIGHT and takes over control.
- When you encounter upload problems of a project file, try a manual project upload to discover network problems like firewall or configuration reasons.
- If you use any UDP Block in the Workflow and use the same project on both devices, set the IP address of the UDP Block for receiving to 0.0.0.0.
- Note that local serial ports are not supported because you typically cannot connect both the Master and the Backup engine to the same target device. In this case use a SYMPL serial Node.
- To replace the Master engine, use the same IP address of the previous Master and the same software version. Configure the new Master in a discrete network first. To be able to replace the Master, it is recommended to keep the SYMPHOLIGHT setup files of the Master to have the same software version available when replacing it.
- When you use custom fixtures, use the *.hfp file format on both devices.
- A Core Engine that is configured as a Master will not automatically upload projects to the Backup engine. On a Core Engine that is configured as a Backup engine you have to upload the Backup project file manually: Upload the project file for the Backup mode at [Project File / Select a file](#). This stored project file for the Backup mode is not visible in the web interface. The [Project file](#) field of the Backup dialog at [Settings](#) needs to be left empty.



8.2 Application Settings

With [Tools / Application Settings](#), you define the general application settings for SYMPHOLIGHT. The dialog for Application Settings is divided into five tabs: General, Content, Network, Logging and 3D Control. Additionally you can now check for updates for SYMPHOLIGHT with a button in the lower part of the window.

General

Location for Project files	This is the default location where projects and project files are stored.
Location for User Storage	The location where user managed data is stored. For example when using the FileDownload feature of REST API or Scripting, this directory is the download location.
Load last saved Project on startup	If this is checked, SYMPHOLIGHT will load the last project after SYMPHOLIGHT was closed.
Ask the user to save changed projects	If SYMPHOLIGHT gets closed and this mark is checked, and there are changes in the projects, SYMPHOLIGHT will ask to save these changed projects. Default = on.
Automatically save the project after	Select a time in minutes. This is the interval after which the project file is automatically saved by the program after a modification is made. Only saved to an existing project file: no versioning, i.e. no recall of previous states. Default = 0 (off). When using the Scheduler, set a value > 0.
Number of recently used files	In the File menu, the last used projects are shown. This is the maximum number of shown project files. Default = 10.
Check for updates at start	Check the e:cue website if updates for SYMPHOLIGHT are available. Default = on.
Start with Windows	Start SYMPHOLIGHT after Windows has booted. Default = off.



Speed of Test Mode / Running Highlight	The speed of the “Running Highlight” type of the Test Mode in the Setup Window (see „9.14 Test Mode“ on page 88).
Default start up tab	SYMPHOLIGHT will show the selected tab after start of the application. Default = Setup Tab.
Maximum frames per second	This is maximum frame rate used in rendering the content. Default = 60. Devices that do not support this frame rate are fed with 30 fps.
Language	Select a language for SYMPHOLIGHT. Setting will take effect after a software restart.
Enable the local timeseries database	Enable for using the Fixture board - History in SymphoCity.

Content

Fading

Fading of effects enabled on creation	Add a fade-in and fade-out automatically when placing effects. Default = on.
Default fade times for effects	Default automatic fade time in and out.
Fading of media enabled on creation	Add a fade-in and fade-out automatically when placing media. Default = on.
Default fade times for media	Default automatic fade time in and out.
Auto save changes to content.	When changing Content, the changes are saved to live output automatically. Default = on.

Timeline

Default height for Tracks	The height of the Layers in the Content Tab.
---------------------------	--



Gamma preset settings

Red, Green, Blue

The default gamma correction for the red, the green and the blue channel. After configuration, automatically applied to all newly added videos, images and live videos. Does not apply to existing elements. Overwritten by the individual Gamma Correction (see „10.19.1 Gamma Correction“ on page 109).

Value range from 0 to 15. Default = 1.

Network

e:net

Interfaces for e:net

Select the IP address(es) of the network interface that is/are used for connections to engines and interfaces. Default = all network interfaces.

Instance Name

Name of the SYMPHOLIGHT instance in the network. Use %u to insert the current user name. Use %h to insert the local hostname.

Is an instance name invalid (invalid characters in use, eg. empty space, comma, special characters), then SYMPHOLIGHT automatically replaces it with a valid instance name.

The name replacement takes place immediately. The displayed new name is saved in the system and used immediately.

Art-Net

Enable Art-Net

Enables the Art-Net communication.

Enable ArtSync packets

Enables ArtSync packets to synchronize the Art-Net outputs.

For large installations. Only performing on Art-Net devices that support ArtSync. Check whether your Art-Net devices support ArtSync.



Optimize timing	<p>Enables the frame optimization. SYMPHOLIGHT sends the Art-Net outputs evenly distributed within the time frame.</p> <p>For large installations. For improved performance stability. Only for Art-Net devices that do support ArtSync packets. ArtSync must be enabled and all Art-Net devices of the installation must support ArtSync for this feature to work. If you do not know whether your devices support ArtSync, simply try and test the optimized timing on the installation.</p>
Interfaces for Art-Net	<p>Select the IP address(es) of the network interface that is/are used for connections to Art-Net devices. Default = all network interfaces.</p>
Default interface	<p>Select the IP address of the subnet for an IP address specific broadcast. This is only used for universes at which the correct subnet cannot be determined based on the universe configuration.</p>
sACN	
Enable sACN	<p>Enables the Streaming-ACN communication.</p>
Optimize timing	<p>Enables the frame optimization. SYMPHOLIGHT sends the sACN outputs evenly distributed within the time frame.</p> <p>For large installations. Only for sACN devices that do not support Sync frames. All sACN devices of the installation must support Sync frames for this feature to work. If you do not know whether your devices support Sync frames, simply try and test the optimized timing on the installation.</p>
Interfaces for sACN	<p>Select the IP address of the network interface that is used for communication to sACN devices.</p>



Web

Web Server

Enable Web Server	Enables the SYMPHOLIGHT web server for the web interface and ActionPad.
Network Interface	The network interface for the web server. Default = 0.0.0.0 (all network interfaces).
Port	The network port for the web server. Default = 8081 (on Windows), 80 (on SYMPL Core Engine).
User Name	Default: admin.
Password	The access password for web server configuration. Initially, the password is a random value to prevent unauthorized access to the WebUI configuration, which is similar to the Core Engine and is also available on Windows.

In Windows 7 and later a user without administrative rights is not allowed to change the system configuration. To access the web server your system administrator can set an exception rule, so users without administrator rights may start the web server and access it externally. Ask your administrator to enter credentials for an administrator login.

REST API

Used ports	Specifies whether the REST API is available via HTTPS (Port 5001) or via HTTP (Port 5000). Setting will take effect after a software restart.
Use own SSL certificate	Specifies whether the user provides a custom SSL certificate. Using a custom SSL certificate is recommended. Otherwise a self-signed certificate is provided by SYMPHOLIGHT. Self-signed certificates can lead to problems in environments with high security requirements. Setting will take effect after a software restart.



Path to SSL certificate	Specifies the path to the custom SSL certificate. A *.pfx file is required here. Setting will take effect after a software restart.
Certificate Password	The password defined when the certificate was created. Setting will take effect after a software restart.

Static File Hosting

Enabled	Enables the static file hosting feature. When enabled, a "PublicHtml" directory is created within the directory specified as "Location for User Storage". All files and directories in "PublicHtml" are accessible from the network. The access is protected by Basic-Authentication which has to be configured with below <u>Username</u> and <u>Password</u> . The files will be available at https://YourIp:5002/StaticFileAccess/FileName afterwards. Setting will take effect after a software restart.
Port	The port used to access the hosted files. Since a port can only be bound to one application, the ports used for the REST API and web server are not valid here. Setting will take effect after a software restart.
Username	The username for the basic authentication when accessing the statically hosted files. Setting will take effect after a software restart.
Password	The password for the basic authentication when accessing the statically hosted files. Setting will take effect after a software restart.

Another option to share static files via http(s): Place the files in C:\ProgramData\ecue\wwwroot. The files will be available at <http://YourIp:5004> and <https://YourIp:5005>. Do not use the "scheduler" folder - otherwise SYMPHOLIGHT has to be re-installed. Prohibited sub-folder names: "datahub", "logs", "api" and "swagger".



MQTT



The MQTT interface is in the testing phase and should not be used in production environments. It is subject to ongoing changes without backward compatibility. Currently, the interface is intended exclusively for use with SymphoCity. Only available when the Gateway Communication feature is activated.

External MQTT Broker

Connection to External Broker Enable to connect to an external MQTT broker, e.g. when using SymphoCity.

Host The IP address or hostname for the external MQTT broker.

Port The network port for the external MQTT broker. Enable the Encryption accordingly.
 Default ports:
 MQTT encrypted: 8883
 MQTT unencrypted: 1883
 Encrypted MQTT for SymphoCity: 8884

Username The username for the MQTT connection. For SymphoCity, use the same credentials as for the MQTT instance connection in SymphoCity. When using multiple Sympholight instances with SymphoCity, make sure that every Sympholight instance has a unique MQTT username.

Password The password for the MQTT connection. For SymphoCity, use the same credentials as for the MQTT instance connection in SymphoCity.

Encryption (tls) Enable to use encryption (tls). Select a port accordingly.

Use Web Sockets Enable to use web sockets. Disabled by default.



Web Sockets Base Path The network interface for the web sockets. Empty by default.

Validate Certificate Enable to validate the broker's certificate. Disabled by default.

Publish Changes

Publish DALI changes Enable to actively send updates of DALI states to SymphoCity's Fixture board. Disable when using SymphoCity's Traffic Saver.
Default: on.

Publish RDM changes Enable to actively send updates of RDM states to SymphoCity's Fixture board. Disable when using SymphoCity's Traffic Saver.
Default: on.

Logging

Location for Log files Select the path for the SYMPHOLIGHT log files. Default = C:\Users\[username]\AppData\Roaming\ecue\Sympholight\Logs. Setting will take effect after a software restart.

Save Logs for Time frame for which the logs are kept. Older Logs will be deleted. Default = 90.

Toast Log Filter Which information will be logged. Default = Info off.

Log e:bus events Logs all e:bus events for debugging. Please enable only in case of Glass Touch problems.

Debug messages available via REST Enable to activate debug messages via REST.



Canvas

Controls

Control Layout The used Layout mode, SYMPHOLIGHT, SYMPHOLIGHT Classic, or Blender.

Visual

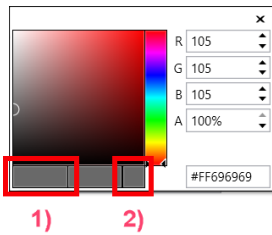
Smooth LEDs Use a constant light value or fading of LEDs in the Preview.

Element Borders Show the borders of elements in Setup, e.g. for luminaires.

Small Grid Sets the color for the single raster cell. Default = FF696969.

Large Grid Sets the color for the raster of $5 \times 5 = 25$ small grids. Default = FF2F4F4F.

Background Sets the background color. Default = FF7B7B7B.



Color Picker for small grid, large grid, and background.

- 1) The initial color. Click to cancel a change.
- 2) The selected color.

Reset colors Use the default color values for Small Grid, Large Grid, and Background.

Grid Defaults

Grid Size The total grid size for X and Y axes in mm. Default = 25.000 mm.



Cell Size	The raster size of the grid for X and Y axes in mm. Default = 100 mm.
-----------	---

Performance

Mute Visualization	Static / disabled display in Workspace (Preview). Saves engine performance. Auto: set the idle time at Mute Visualization After. Off: no muted visualization. On: immediate muted visualization; remains despite of working activities.
--------------------	--

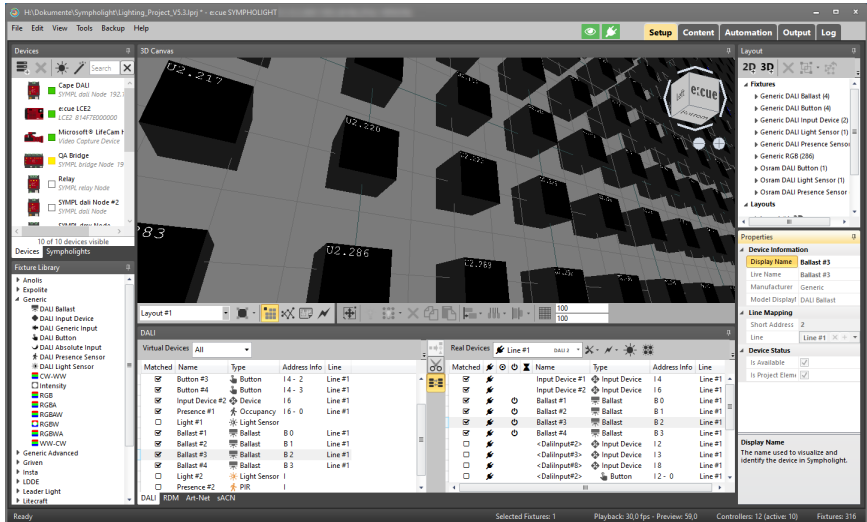
Mute Visualization After	The idle time after which the Workspace gets muted when Mute Visualization is set to Auto.
--------------------------	--



9 Setup

9.1 Setup Window

In the Setup Window, you patch your fixture configuration with all other components like engines, interfaces and fixtures. You also assign DMX universes and addresses as well as DALI lines to fixtures and interfaces.



Devices and Sympholights

Workspace
2D or 3D canvas

Fixture Library

Drag and drop them into the Workspace to add them to the Setup.

Mode Selection

Select Build Mode, Patch Mode, Background Mode and Test Mode.

DALI / RDM / Art-Net / sACN Window

Show DALI configuration, or watch and filter RDM events and messages, or configure and patch Art-Net and sACN devices.

Layout

Project's Fixtures, layouts, universes, and DALI lines

Properties

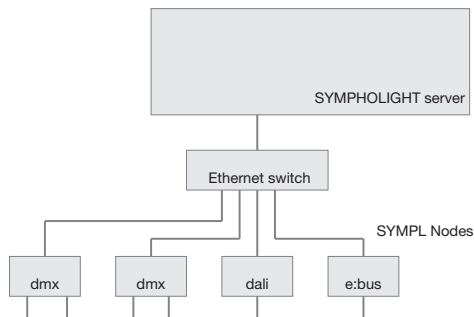
Edit device properties like DMX address, universe, and DALI device type.

9.2 Adding Devices - Device Manager



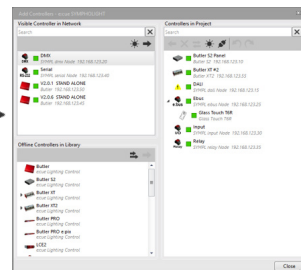
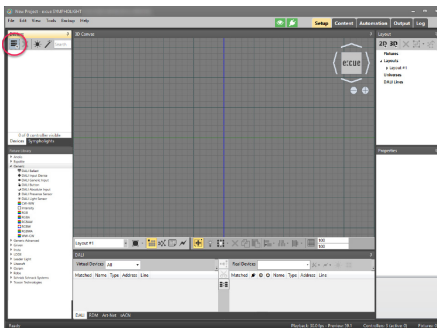
SYMPHOLIGHT supports interfaces and engines of the e:cue SYMPL range and recent e:cue Butler engines. The ETH2DALI and the e:bus Input Modules are only supported by the Lighting Application Suite.

To use SYMPL Nodes as interfaces in SYMPHOLIGHT, connect the Nodes via an Ethernet switch to the SYMPHOLIGHT server. Read the Setup Manual for the device and take care that the given installation requirements are met.

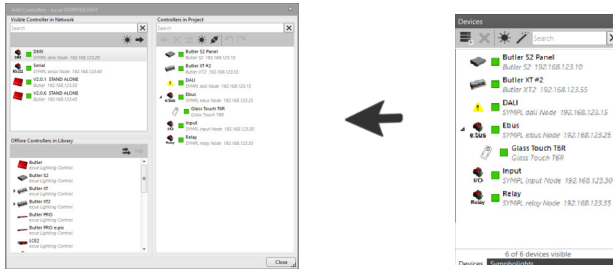


Before adding interfaces, select the proper network interface for e:net in SYMPHOLIGHT's Application Settings. Use the preferred local network range 192.168.123.xxx with a system-internal network card.

Open the Device Manager to add or remove devices in the project:



Connected Nodes in this network become available in the Devices Manager as visible devices in the network. The list of devices in the project is also shown when the Device Manager is closed in the Devices Tab.



To add or remove a device, use Drag & Drop of visible devices or offline devices to the Devices in the project window. You can also use the context menu of the device or these buttons:



Add selected device to the project.



Add multiple devices to the project.



Remove selected devices from the project.



Unassign selected devices. The real device becomes a visible device, while an unassigned device remains in the project.



Replace online device: To assign a real device to an unassigned virtual device, select an unassigned device in the project and an visible device in the network of the same type.




Enable/disable to add interfaces to show automatically: press the hardware button to add when enabled.





Undo the last action.




 Redo the last undo action.


Further functions are available in the Device Manager or the Devices Window using the following buttons:

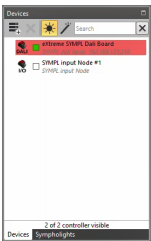
 Enable/disable interfaces identify mode: make Error LED and e:net LED on the device blink to identify the device.

 Enable/disable replace interfaces automatically: after finishing system setup, activate this icon. If a SYMPL Node fails, you can replace it and SYMPHOLIGHT will reconfigure the new device to the current settings. Replace one Node at a time. The project file needs to be saved afterwards.
For Node replacement on the SYMPL Fusion or SYMPL Core S(p) see „13.13 Automatic Node Replacement Mode“ on page 433.

The Identify icon () determines how the Identify button on the SYMPLs is handled.

A press of the Identify button makes the device blink in red in the Device window.

 The e:cue SYMPL Fusion or SYMPL Core S(p) has a separate chapter in this manual. See „13 SYMPL Fusion & SYMPL Core S(p)“ on page 420.



After the device was added to the project, you can change the SYMPHOLIGHT internal name of the device, or the network data for the device, if a manual configuration is needed. Click into the name field of the device to change the name, or change properties in the Properties Window. The following colors are assigned to the devices:

Indicator Color	Local Sympholight instance perspective
White	An e:net device which is unassigned. A module which is unassigned.

Grey

An e:net device that is required but not found in the network.
 An e:net device disconnected but desired in the show, and the Sympholight tries to establish the connection.
 A module is required but not properly inserted into the slot.
 A module is disconnected but desired in the show, and Sympholight tries to establish the connection.

Green

An e:net device that is available but not required.
 A module that is available, but not required.

Lime

An e:net device that is available and required.
 A module that is available and required.

Yellow

An e:net device that is in test mode.
 A module that is in test mode.

Red

An e:net device that is in the wrong runlevel or unavailable.
 A module that is in the wrong runlevel or otherwise occupied or unavailable.
 A module that is mismatched and does not fit.

Black

An e:net device which is connected to another instance and the user tries to add it to his local Sympholight.

Blue

An e:net device which is connected to another instance.



When running the Programmer of the LAS and SYMPHOLIGHT on the same e:cue LCE3 server, the IO ports of the LCE3 will be disabled in SYMPHOLIGHT. The LCE display is still available.



9.2.1 Virtual Devices

You can also drag-and-drop offline devices into the project. These virtual interfaces can be used like real physical interfaces. This is helpful when setting up configurations without the real devices present. Later, you can replace the virtual devices with real devices. The same is valid when assigning DMX universes. You can place fixtures in the Workspace and delete/add/assign DMX addresses and universes, when still no interfaces are present. Later, you can add interfaces and map the outputs accordingly.

9.2.2 Butler Classic, XT, XT2 and S2

In general, these e:cue Butlers are supported by SYMPHOLIGHT as DMX interfaces and with dry contacts for Butler XT and XT2. The e:bus interfaces of Butler XT and XT2 are supported. Also the RDM functionality of the Butler S2 and XT2 is available.



e:cue Butler devices must be configured with the Programmer/Patchelor from the Lighting Application Suite. IP addresses must be set manually. Take care for possible IP address conflicts.

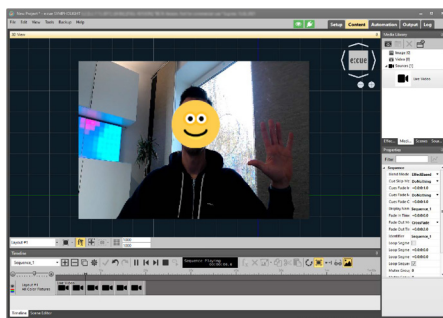


Please be aware that SYMPHOLIGHT doesn't support the standalone mode of this devices.

9.2.3 Video Capture Device

Every video device you connect to your PC should be available in the Device Manager, e.g. webcams, USB capture devices, or capture cards. You can integrate other video systems such as the lighting designer's PC/Mac, a TV picture or just a camera. Configure the resolution, FPS, and color pixel format in the device properties. Via video capture devices live video-in is possible.

For live video, add the video capture device to your project and then you can drag and drop the "Live video" source from the Media Library in the Content tab single or multiple times onto the timeline and mix it with other SYMPHOLIGHT effects.



The typical effect options, e.g. fading or keyframing, can be used on live videos as well. See [„9.14 Test Mode“ on page 88](#) for more. You can also use multiple Video-In sources.

Note: Required minimum version for live video is Win10 1803. Native Video-In for the Core Engine is not possible.



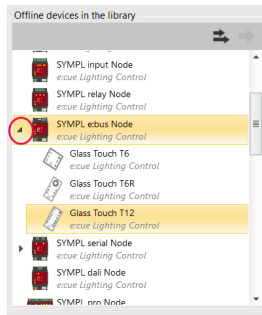
9.2.4 Non-DMX and non-DALI Devices

Other devices and interfaces like serial interfaces, UDP interfaces, SYMPL input Nodes or SYMPL relay Nodes are not related to the content as media. They are added and configured in the Setup Window, but they require basic configurations like system interfaces, IP addresses and names only. Add them like the SYMPL dmx and dali Nodes via the Device Manager to the project. You can configure them in the Properties Window.

9.2.5 Attached Devices

In SYMPHOLIGHT, an “attached device” is a second level controller. It is physically connected to a controller (“parent device”) such as the SYMPL nodes. Examples are Glass Touches which are attached devices to the SYMPL e:bus Node, or Serial Modbus Devices (e.g. Genius) which are attached devices to the SYMPL serial Node. The physical connection and virtual assignment in SYMPHOLIGHT to a “parent device” are prerequisites to use the attached device. When in SYMPHOLIGHT a parent device is not defined for the attached device, it is “unassigned” and not controllable.

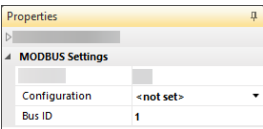
An attached device is added to the project as any controller via the Device Manager. The attached devices are available at the small triangle ► in front of the controllers in the offline devices window. Click the triangle and the list of possible attached devices expands.



- Adding an assigned attached device: To add an attached device including its parent device drag&drop it from the “Offline devices in the library” into “Device in project”. Or use the single “Add device to project” button. The attached device will be assigned to the parent device.
- Adding an unassigned attached device (without a parent device): Use the multiple add button.
- Assigning of an attached device: Drag&drop an attached device within the “Device in the project” list to the target parent device. This way you can assign unassigned devices or change the assigned parent device.



Configure the attached device in the Properties window once they are in the project. For the Serial Modbus Device, it is important to set a Configuration (= Modbus device type, e.g. Genius) and unique Bus IDs (= Modbus device address on the serial bus). For the Glass Touches, it is important to configure the UID.



9.2.6 Glass Touch T6, T6R, T12 and IR

SYMPHOLIGHT supports the e:cue Glass Touch T6, T6R, T12, and IR user terminals via e:bus as “attached devices”. They can be used and configured in the Workflow Designer in the Automation. For details see [„11.8.1 Target Based Blocks“ on page 124](#), [„11.15.74 Glass Touch T6R“ on page 348](#), [„11.15.75 Glass Touch T12“ on page 355](#), or [„11.15.76 Glass Touch IR“ on page 356](#).

9.2.7 Serial Modbus Devices (e.g. Genius)

SYMPHOLIGHT supports Modbus devices when a serial controller (SYMPL serial Node) is installed as an intermediate protocol transformer. The Modbus device is called “Serial Modbus Device”, is of the type “attached device”, and owns a target based Workflow Block, the “Modbus Device” block.

In the Properties Window at Configuration under the Modbus Settings, define which Genius product the Serial Modbus Device represents. And set unique Bus IDs.

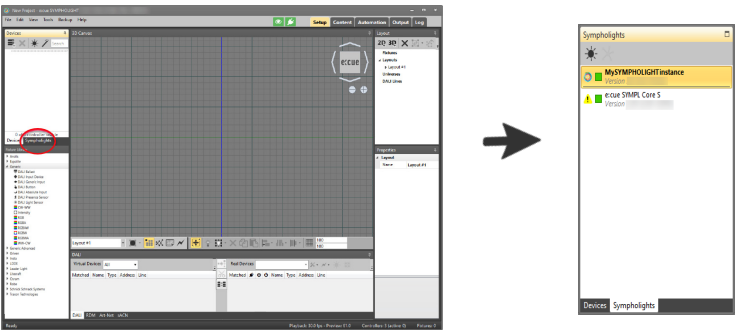
At two points in SYMPHOLIGHT you can configure parameters of the Modbus data: In the Devices window, in the context menu of the Serial Modbus Device at “Modify device” set the refresh rate for each information point. This is also available in the workflow block.

Here in addition it is configurable which information is enabled and its unit. For block details see [„11.15.68 Modbus“ on page 335](#).

To disable internal value reading from the Modbus device, set the Refresh Rate to 0.

9.3 SYMPHOLIGHT Instances

Select the Sympholights Tab to see all SYMPHOLIGHT instances in the network.



Click into the name field of the instance to see its properties in the Properties Window. The following colors are assigned to the devices:

Indicator Color	Other instance perspective
Grey	A SYMPL Fusion / Sympholight / SYMPL Core Sp which is muted.
Blue	A SYMPL Fusion that redirects its modules and input to some instance (higher priority).
Sky blue	A SYMPL Fusion that is occupied by another system but the local system wants to redirect to itself (higher priority).
Black	A SYMPL Fusion / Sympholight that has none of its modules/nodes available. A SYMPL Fusion / Sympholight that has not all but some required modules or nodes.
Yellow	A SYMPL Fusion / Sympholight that has at least one unassigned module (or node), i.e. the system device's address equals the empty address of the specified system device type.
Lime	A SYMPL Fusion / Sympholight which has all required modules or nodes and is available.



The context menu of an instance allows you to manage your own and other instances. You can for example mute your own detection or other instances, open the web interface, share the license, upload a project, trigger a system update, open the REST API web interface or copy the instance ID. You can see the SYMPHOLIGHT instance ID also in the Properties window and web interface's dashboard.

Further functions are available in the Sympholights Tab using the following buttons:



Enable/disable interfaces identify mode: on a SYMPL Fusion or SYMPL Core Sp, depending on the device type, make the respective LED on the device blink in Identify mode to identify the device.



Enable/disable share license: Use the SYMPL Fusion or SYMPL Core S(p) as License Key. For details see „[13.6 Using SYMPL Fusion or SYMPL Core S\(p\) as License Key for SYMPHOLIGHT](#)“ on page 432.

9.4 Icons in the Mode Selection

These are the first five functions on the left of the Setup Mode, the Mode Selection:



Optimize view, reset to default view or define as new default view, select with the small triangle. Zooms to an optimum view of the fixture set.



Build Mode: place fixtures. See „9.5 Placing Fixtures - Build Mode“ on page 54.



Patch Mode: set wiring and distribution. See „9.8 Patch Mode for DMX and e:pix“ on page 68.



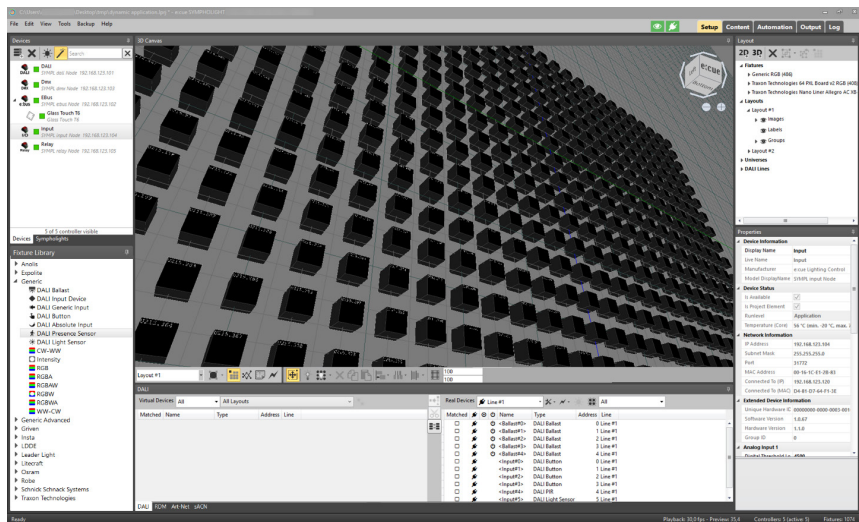
Background Mode to set text and backgrounds. See „9.13 Background Mode“ on page 88.



Test Mode: test fixtures and patching. See „9.14 Test Mode“ on page 88.

9.5 Placing Fixtures - Build Mode



In Build Mode, you place, address and arrange all fixtures.



The following functions are available in Build Mode:



Combined move, scale, and rotate tool for single or multiple selected fixtures, labels, and images.

- Move: hover over the selection and use the .
 - Scale: use the cubes at the border of the selection.
 - Rotate: hover over the circle at the top of the selection and use the .
- You can also change the position, the size, and the rotation angle in the Properties.



Create fixture: after selecting a fixture from the library, places a single instance of this fixture in the Layout.



Create fixture matrix: create a fixture matrix with left mouse button pressed. You can use an offset for X and Y with the drop-down menu under the small triangle to keep space between the fixtures.



Delete the selected fixtures.



Copy selected elements to clipboard.



Paste the clipboard contents.



Align selected fixtures left.



Space selected fixtures evenly.



Distribute selected fixtures left edge.



Enable or disable snap to grid.

Change cell size in the right fields. Overrides the grid settings of the Application Settings for the selected layout.

There are three ways to add and place fixtures:

- In any case switch to Build Mode.
 - Select a fixture from the Fixture Library and drag-and-drop it into the workspace.
- Or
- Select a fixture from the Fixture Library.



- In Build Mode, select Create Fixture Matrix and draw a matrix in the Workspace keeping the left mouse button pressed.

Or

- Select a fixture from the Fixture Library.
- In Build Mode, select the Add Single Fixture and click into the workspace. A new fixture will be added to the Workspace with every new click.

9.5.1 Moving Fixtures

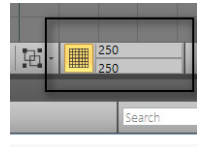
Before you can move elements, you have to select them. This is done in a similar way as in other Windows applications.

- Click a single fixture to select it.
- With the left mouse button pressed, draw a rectangle around fixtures. These fixtures get selected.
- Keep the **Ctrl** key pressed when selecting fixtures. These fixtures are added or removed from the selection.
- Keep the Shift key pressed when selecting fixtures. These fixtures are added to selection, not removed.


The selected fixtures get a blue frame. You can now click and move the fixture with the blue frame or drag the arrows. Placement can be aligned to a grid.

If the grid is enabled (click [Snap to grid icon](#) in the Mode Selection), every move snaps to the underlying grid. Click on the grid size values to edit. If disabled, the fixtures can be moved freely. Use the pressed left mouse button or the cursor keys to move.

Pressing the **Ctrl** key and using the cursor keys shifts the view port of the workspace.



9.5.2 Changing View

Zoom: To zoom in and out for the workspace, there are two circular buttons  in the upper right corner of the Workspace. You can also zoom in and out with the mouse wheel.

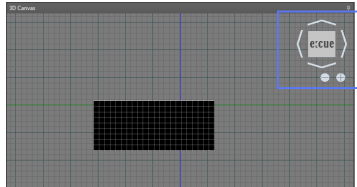
Zoom to specific devices in the workspace via the Layout Window:

- for a single device, go to the fixture (Layout Window - Fixtures - fixture type) and click its “type” icon (finger cursor). The view zooms to the device.
- for a group of devices, go to the group (Layout Window - Layouts - Groups) and click its “type” icon (finger cursor). The view zooms to the group.

If the device or group is on another layout than the currently visible one, you need to click twice: the first click switches to the concerned layout, the second click zooms to the selected device / group.

Viewpoint: The four arrow-like elements around the e:cue box in the upper right corner of the workspace move the view of the workspace.

To rotate the workspace use **alt + pressed mouse**. The e:cue box in the top right corner indicates the current viewpoint.



9.5.3 Custom Fixtures

It is possible to integrate custom fixtures. In order to create a custom fixture, the Library Editor of the Lighting Application Suite can be used. Once the fixture is defined, SYMPHOLIGHT will update the fixture Library after a restart of SYMPHOLIGHT.



Custom fixtures may only contain **either** color **or** intensity channels. Supported color channels are: red, blue and green. All fixtures containing different channel types will not be loaded on start-up.



Please make sure to export a *.hfp file onto a SYMPL Fusion or SYMPL Core S(p) if your show contains custom fixtures.

9.6 Configuring Properties

The protocol-specific properties of the devices are defined in the Properties Window. For example you can assign DMX universes to the SYMPL dmx Node outputs. After adding the Node to the project, new DMX universes are assigned. To change properties of a fixture, click the fixture in the Workspace. In the Properties Window, you now can change values like assigned DMX universe and DMX address (for further information on addressing see [„9.8 Patch Mode for DMX and e:pix“ on page 68](#)). You can assign the same universe to different output ports of devices or for streaming. However, do not assign the same universe and address to more than one fixture. In case of an address conflict where the same address is assigned to another fixture in the same universe, SYMPHOLIGHT checks the address range. A confirmation window pops up. When you click “No”, the address change will be discarded and both

Properties	
Device Information	
Display Name	Dmx
Live Name	Dmx
Manufacturer	e:cue Lighting Control
Model DisplayName	SYMPL dmx Node
Address Mapping	
DMX 1	U1 X + -
DMX 2	U2 X + -
Device Status	
Is Online	<input checked="" type="checkbox"/>
Is Available	<input checked="" type="checkbox"/>
Is Show Element	<input checked="" type="checkbox"/>
Runlevel	Application
Temperature	43 °C (min. -20 °C, max. 70 °C)
Network Information	
IP Address	192.168.123.3
Subnet Mask	255.255.255.0
Port	31772
MAC Address	00-16-1C-E1-01-20
Connected To (IP)	192.168.123.154
Device Information (No description)	

fixtures will retain their individual addresses. When you click “Yes”, you will proceed with the conflict and the address will be given twice, i.e. the old fixture will keep the address, and the new fixture will get the same address.

Click on the plus-symbol in the Address Mapping to assign a new DMX universe, or click the down-arrow to assign an already used DMX universe, which was created while placing fixtures. Click the x-symbol to remove the assigned universe.

In the same way, you can select DALI lines for the SYMPL dali Nodes, or any other functional property for the rest of the SYMPL Node family.

Properties shown in grey are not editable in SYMPHOLIGHT, they are read-only values like the MAC address, model numbers or unique hardware IDs given during production of the device. The Device Status in the Properties Window shows the status of the device, if it is online or added to the project. Additionally, the internal temperature of the device is displayed.

See „9.6.3 SYMPL Node and SYMPL Fusion modules Properties“ on page 59 for a description of SYMPL device properties and their meaning.



Do not change properties if you are not fully aware of the results, for example parameters for the DMX protocol. Changing parameters in an uncontrolled way can make the device fail or cause spooky results.

9.6.1 Network Addresses

There are two ways to assign a network address to a device. As a default, the device gets an IP address assigned by the DHCP (Dynamic Host Configuration Protocol) server. This assures that only unique IP addresses are assigned. If you want the device to have a fixed IP address, add the device to the project, right-click on the newly added device and select Configure Network. In the configuration window you can change the IP Address, the Subnet Mask, the Gateway and in case you’ve selected a Core Engine, the Network Mode and the password to write the new settings to the device.

The left picture shows the Network Configuration Dialog for the SYMPL Fusion or SYMPL Core S(p). Please adjust the settings and enter the correct Password in order to transfer the configuration onto the



SYMPL Fusion or SYMPL Core S(p). Please make sure the device is configured properly, otherwise the new settings won't be applied. See „13.2 Configuring the SYMPL Fusion or SYMPL Core S(p)" on page 423 for more information. The right picture shows the dialog if some interfaces and Core Engine have been selected in the Device Window at once. The Sympholight Network Configuration will set the address for the SYMPL Fusion or SYMPL Core S(p), while the Controller Network Configuration will readress, e.g. the SYMPL Node devices.



Make sure that all devices including the SYMPHOLIGHT server have a unique IP address. Duplicated IP addresses will cause malfunctions or a complete fail of the network.

Compared to the Programmer in the former Lighting Application Suite, SYMPHOLIGHT does not identify interfaces by their IP address, but by their unique hardware ID. This means that a change of the IP address does not change the DMX universe mapping. For this reason, SYMPHOLIGHT can use DHCP with changing IP addresses without problems.

9.6.2 Additional Functions

A right-click on the device in the Device Window opens a context menu, where you can remove the device from the project, reset the device and enter or leave the bootloader for uploading a new firmware to SYMPL Node interfaces. SYMPHOLIGHT comes with the necessary firmware versions for SYMPLs built-in, so you can upload a firmware without an additional file.

The command Unassign in the context menu removes the device from the list and replaces it with a virtual device of the same type. This is useful when replacing a device by another device of the same type. The port assignments are kept.

9.6.3 SYMPL Node and SYMPL Fusion modules Properties

Properties that are read-only and cannot be modified, are shown in grey.

Properties of all SYMPL Nodes

Device Name	The name used to visualize and identify the device in SYMPHOLIGHT. Becomes the Live Name when the device is available.
-------------	--

Live name	The name stored on the device, read-only.
-----------	---

Manufacturer	e:cue Lighting Control, read-only.
--------------	------------------------------------

Model	The type of the device, read-only.
-------	------------------------------------



Is Available	Shows if the device is available, read-only.
Is Project Element	Shows if this device is used in the project, read-only.
Runlevel	Current state of the device, e.g. while updating or resetting, default = Application, read-only.
Temperature	The internal temperature of the device, read-only.
IP Address	The assigned IP address, e.g. 192.168.123.11, read-only.
Subnet Mask	The subnet mask, default = 255.255.255.0, read-only.
Port	The UDP port number, default = 31772, read-only.
MAC Address	The hardware MAC address, read-only.
Connected To (IP)	The IP address of the SYMPHOLIGHT server the Node is connected to, read-only.
Connected To (MAC)	The MAC address of the SYMPHOLIGHT server the Node is connected to, read-only.
Unique Hardware ID	The internal hardware ID of the device to make the device addressable without relation to the IP address, read-only.
Software Version	The version of the internal firmware, read-only.
Hardware Version	The production release of the hardware, read-only.

Group ID

Properties of SYMPL pixel Node

Out 1	The universe number port 1 of the Node sends to. Assign a used universe or create a new one.
-------	--



Out 2 The universe number port 2 of the Node sends to. Assign a used universe or create a new one.

DMX Break Length The Break resets the line, signaling a new DMX Packet, default = 100 us.

DMX MAB Length Length of Mark After Break, MAB signals the receiver to begin reading data default = 17 us.

Properties of SYMPL dmX Node

DMX 1 The DMX universe number port 1 of the Node sends to. Assign a used universe or create a new one.

DMX 2 The DMX universe number port 2 of the Node sends to. Assign a used universe or create a new one.

Direction Define this port as DMX input or output port. In case the direction is set to input, a DMX port will be added in the Workflow Designer for further processing and automation. Default = Output.

DMX Receive Length The block size after which a received DMX block is seen as complete, default = 512.

DMX Break Length The Break resets the line, signaling a new DMX Packet, default = 100 us.

DMX MAB Length Length of Mark After Break, MAB signals the receiver to begin reading data default = 17 us.

RDM Break Length The Break resets the line, signaling a new RDM Packet, default = 180 us.

RDM MAB Length Length of Mark After Break, MAB signals the receiver to begin reading data default = 17 us.

RDM switch delay The delay between sending a DMX and an RDM block, default = 176 us.



Startup Behavior State of the outputs after startup, default = HighImpedance.

Offline Behavior State of the outputs when going offline, default = HoldLastFrame. You can define offline content in the Content Window and use this content as offline content. See „10.3 Preview Window“ on page 93.

Properties of SYMPL e:pix Node

e:pix 1 The e:pix universe number port 1 of the Node sends to. Assign a used universe or create a new one.

e:pix 2 The e:pix universe number port 2 of the Node sends to. Assign a used universe or create a new one.

The other properties are the same as for the SYMPL dmx Node.

Properties of SYMPL input Node

Digital Threshold Low Input voltage for digital signaling off, this is also the level for switching the LED, default = 4000 mV.

Digital Threshold High Input voltage for digital signaling on, this is also the level for switching the LED, default = 6000 mV.

Event Delta Voltage range that is ignored between the two above thresholds, default = 100 mV.

Properties of SYMPL relay Node

Initialize relays on project load After loading the project, the relays are initialized to a state, NC or No, default = not initialized.

Initial Relay 1 State to be initialized to, default = not initialized.

Initial Relay 1 State to be initialized to, default = not initialized.

Initial Relay 1 State to be initialized to, default = not initialized.



Properties of SYMPL e:bus Node

The SYMPL e:bus Node has no properties to be set or modified.

Properties of SYMPL serial Node

Baudrate	The data rate of the interface, default = 115200.
----------	---

Data Bits	Length of one data block, default = 8.
-----------	--

Parity	Parity check bit, default = None.
--------	-----------------------------------

Stop Bits	Number of stop bits, default = One.
-----------	-------------------------------------

Properties of SYMPL dali Node

Line A	The DALI line in the setup this Node is used for. Select an already used one or create a new line.
--------	--

Bus Power On Start	Sets the DALI bus power on startup to On or Off.
--------------------	--

Bus Power Off Timeout	The time in seconds after which the DALI bus power will be switched off in offline mode.
-----------------------	--

Properties of SYMPL pro Node dmx

Min. frame distance	The minimum frame distance, default = 0 us..
---------------------	--

RDM packet space	The delay between sending a DMX and an RDM block, default = 176 us.
------------------	---

RDM DISC timeout	The RDM discovery timeout, default = 2800 us.
------------------	---

The other properties are the same as for the SYMPL dmx Node.



Properties of SYMPL pro Node e:pix

The properties are the same as for the SYMPL e:pix Node and the SYMPL pro Node dmX.

Properties of SYMPL pro Node m:pix

DMX Baudrate Set the baudrate to 750,000 (=1536 channels). Configure the universe as "e:pix" ([„9.7.4 Configuring Universes“ on page 68](#)).

The other properties are the same as for the SYMPL pro Node dmX.

Properties of SYMPL essential Node

See the SYMPL dmX Node and SYMPL pro Node dmX for the property descriptions.

Properties of SYMPL essential2 Node dmX

See the SYMPL pro Node dmX for the property descriptions.

Properties of SYMPL essential2 Node m:pix

DMX Baudrate Set the baudrate to 750,000 (=1536 channels). Configure the universe as "e:pix" ([„9.7.4 Configuring Universes“ on page 68](#)).

See the SYMPL pro Node m:pix for the property descriptions.

Properties of SYMPL bridge Node

See the SYMPL e:pix Node and SYMPL pro Node dmX for the property descriptions.

Properties of SYMPL pure Node

See the SYMPL dmX Node and SYMPL pro Node dmX for the property descriptions.



Properties of SYMPL Fusion DMX modules

Slot Slot number where the module is expected or plugged in.

See the SYMPL dmx Node for the other property descriptions.

Properties of SYMPL Fusion DALI module

Slot Slot number where the module is expected or plugged in.

See the SYMPL dali Node for the other property descriptions.

9.7 Layout

A Layout contains all fixtures, devices, and elements (images and labels) for a project. You can have one or many Layouts, representing parts or subdivisions of the whole project. SYMPHOLIGHT supports two Layout models, 2D Layouts and 3D Layouts.

For every Layout with DMX fixtures, SYMPHOLIGHT creates an associated Layer in the Timeline in the Content Window. Depending on the fixture types, these Layers are used for color or for intensity only, if the used fixtures only support intensity control. For every Layout, you can define as many Sequences and Layers in the Content Window as necessary. There will be at least one Layout and one Layer, if you place fixtures in one Layout.



Create new 2D layout.



Create new 3D layout.



Delete the selected items: fixtures, layouts, images, labels, groups, universes, or DALI lines.



Group the selected fixtures. Choose whether to add to another existing group.



Merge groups.



Edit the selected group. Select fixtures to add (green border) and fixtures to delete (red border) from a group. Save ✓ or cancel ✗ the changes.





Align selected fixtures.

Within the Layout window the following functions are available:



Item is visible in canvas and its properties available. Click to hide the item.



Item is unlocked. You can edit its properties, group and delete it. Click to lock the item.



Click to display the effect rectangle of the item in the canvas. See „9.7.2 Effect Rectangle“ on page 66.

Icon of
Fixture
type

Click to zoom to the item in the workspace. See „9.5.2 Changing View“ on page 56.

9.7.1 Groups

If you want to address only certain fixtures in a Layout, you can group fixtures. In the Layout, you can select or multi-select fixtures by keeping the **Ctrl** key pressed and clicking fixtures, or keeping the left mouse button pressed and drawing a rectangle around fixtures in the canvas. Click the Create Group icon  in the upper Layout icon menu to create a new group. A group can only exist of fixtures of the same type.

You can add new fixtures to already created Groups or merge different Groups by using the context menu of each Group or the tool bar in the Layout. If you select a fixture or a selection of fixtures, SYMPHOLIGHT will show the Groups they belong to, in the Properties Window.

In the Content Window, each Group gets a separate Layer. You can place effects or media in the Layer and they will only be mapped to the fixtures in this group.

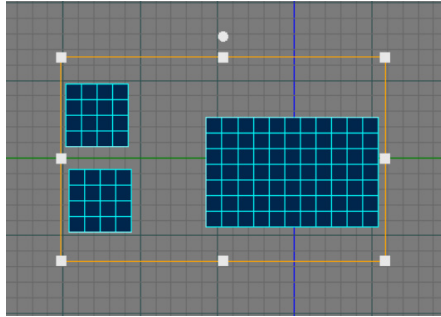
Apart from the fixtures, a group also consists of an Effect Rectangle.



For each fixture type, on first addition, a default group is automatically created. In this group, indicated with a star icon, all fixtures of the same type will be collected. A default group cannot be deleted.

9.7.2 Effect Rectangle

The Effect Rectangle defines the area on which content like Effects or media is mapped.






Only user defined groups have adjustable effect Rectangles. You can see the Effect Rectangle in the Workspace of the Setup Window by selecting a self-made Group in the Layout and clicking the  icon left of it. In the Content Window by selecting an Effect in the Timeline and enabling its visibility via the  icon.

When you create a group, it automatically gets an Effect Rectangle covering the group's fixtures.

The Effect Rectangle has properties which you can manipulate either directly in the Workspace or in the Properties of the Group or Effect. Change its position and form - if you want unrelated to the fixtures in the Group.

Each Effect or media content that you place into this group's Layer in the Content Window, will at first cover the same area as the group's Rectangle. However, you can change the Effect Rectangle for each and every content in the Timeline.

To harmonize the Rectangles of Timeline content to the group's size, either individually or by a chosen range, use the resize icon  in the Content Window.

9.7.3 Export, Import and Merge of Layouts

Exchange layout files to enable joined working on a project through data exchange and a separation of content editing from setup editing.

Export of a layout enables you to exchange layout data. When you export a layout, only properties of the layout are saved in one single file. The file format is *.slpl.

Saved properties are:

- connected and matched DMX controllers and their DMX universes
- layout groups

Properties that are not saved:

- wirings
- DALI controllers and their properties

To export a layout, select a specific layout (#) in the Layout window of the Setup tab, open its context menu and click "Export Layout".



Import of a layout opens a layout file (*.slpl) and adds this layout to the project as an additional layout. To import a layout, click "Import Layout" in the context menu of Layouts (upper level grouping) in the Layout window.

Import and Merge of a layout opens a layout file (*.slpl) and integrates this layout into the selected existing layout (target layout). The merging renames all already existing names of the integrated layout. The following consequences are to be considered when merging:

- Controllers get duplicated but receive new universes so that there are no duplicated universes.
- If layout groups with identical names already exist in the target layout, the fixtures from the imported layout are both created there and duplicated again (the duplicated layout group gets a new name with #). This allows the selection of the imported fixtures in the merged group with the help of the duplicated group (select the duplicate - switch to the merged group - selection of fixtures remains) to be able to delete them if necessary.
- If the imported layout has a group effect frame AND if layout groups with identical names already exist in the target layout, the group effect frame will be handled as follows:

The group effect frame of the destination layout is not defined (0, 0, 0): the group effect frame values from the imported layout is taken.

The group effect frame of the target layout is defined (unequal to 0, 0, 0): the group effect frame from the imported layout is not applied; values from the target layout remain.

To import and merge, select a specific layout (#), open its context menu, and click "Import and Merge Layout".

9.7.4 Configuring Universes

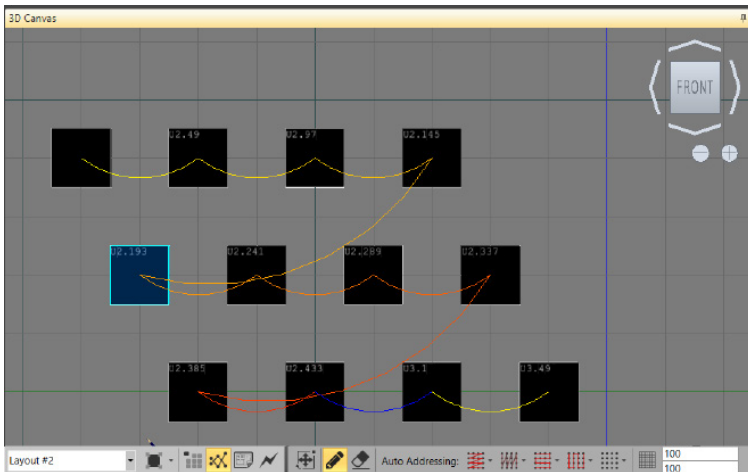
For all universes, you can set the name and length in Layout at Universes. Select a universe and its properties can be accessed in the Properties window. Here you can see which devices are connected to a universe, which protocol is used (e.g. e:pix) and whether you can change it which depends on the connected devices. Set the universe to "e:pix" for using the m:pix protocol.

Please note: Changing a connected devices here in the Properties only renames the device. Enter the Properties of the device to connect it at Address Mapping to a universe.

9.8 Patch Mode for DMX and e:pix

In Patch Mode for DMX devices, you select the wiring of fixtures in the DMX universes. Usually, when placing fixtures in the Workspace, SYMPHOLIGHT assigns DMX universes and addresses according to the number of addresses the fixtures cover in linear mode.





Mode selection in Patch Mode has the following additional icons with drop-down menus behind:



Create a single patch line.



Remove a single patch line.



Address selected fixtures row-wise.



Address selected fixtures column-wise.



Address selected fixtures row-wise in snake mode.



Address selected fixtures column-wise in snake mode.



Remove all connections/from selected fixtures/in Layout.

Check for patch conflicts. In case of patch conflicts, you will get a list of the conflicting fixtures containing their universe ID, address, footprint, and fixture name.



You can assign the same universe to different output ports of devices or for streaming. However, do not assign the same universe and address to more than one fixture.



↑
content

You can change the wiring by clicking on a connection, keeping the mouse button pressed and redirecting the connection to another fixture. Use the dots in the middle of a fixture as start or end point of a connection. SYMPHOLIGHT can also do an automatic addressing with the four icons in the center of the Mode Selection. Select fixtures with dragging a rectangle over all fixtures to select. Click on an auto-addressing icon to execute automatic address assignment.

9.9 DALI Devices

DALI devices are handled separately to DMX devices throughout SYMPHOLIGHT. In addition to the typical operational sequence („5.1 Operational Sequence“ on page 22 or „9.9.2 Addressing DALI Devices“ on page 72) where you first create virtual devices that you match later to real devices, you can also use real DALI devices directly for simple (no DALI device configuration, only “DALI line” Workflow Blocks) and on-site project creation:

- Working with virtual devices allows you to configure DALI device properties. This enables a flawless device replacement. That is not possible for real devices.
- For matched devices all related DALI Workflow Blocks are available: device specific and line bound Blocks. Unmatched real devices can be integrated by a reduced number of Blocks but in a direct way: with the DALI “line” Blocks.

A DALI device can be a single unit or can consist of several so called “instances”, e.g. a light sensor and a presence sensor. Each instance and single unit is represented in SYMPHOLIGHT by a DALI device. For example, for a device with a combined light and presence sensor, add from the Fixture Library one Light Sensor as well as one Presence Sensor to the Workspace.

These instances - which belong to one “parent device” - share the same address (short address) with individual “sub addresses” (instance numbers). You can interact with instances without having the “parent device” (input device) as a virtual device in the project.

In SYMPHOLIGHT available DALI devices:

DALI device	Fixture group	DALI protocol	Device type*	Description
Ballast	Generic	DALI v2	B	A DALI ballast.



Input Device	Generic	DALI v2	I	The basic DALI hardware device, usually consisting of multiple (max. 32) instances such as a Generic Input and sensors. Only necessary for memory bank settings.
Generic Input	Generic	DALI v2	I	Instance for basic device values like temperature or humidity. Use the Event Filter if only certain values are of interest and not every value change.
Button	Generic	DALI v2	I	A DALI button.
Absolute Input	Generic	DALI v2	I	A slider, rotary knob, or a switch. Instance of the Input Device. Minimum, maximum, and resolution values are defined within the device.
Presence Sensor	Generic	DALI v2	I	An occupancy sensor.
Light Sensor	Generic	DALI v2	I	A light sensor.
Light Sensor	Osram	Osram	I	A light sensor.
Presence Sensor	Osram	Osram	I	A presence detection sensor (PIR).
Button	Osram	Osram	I	A DALI button.

* Device type “B” for “ballast”, “I” for “input”. Appears in the Address Info of a device.

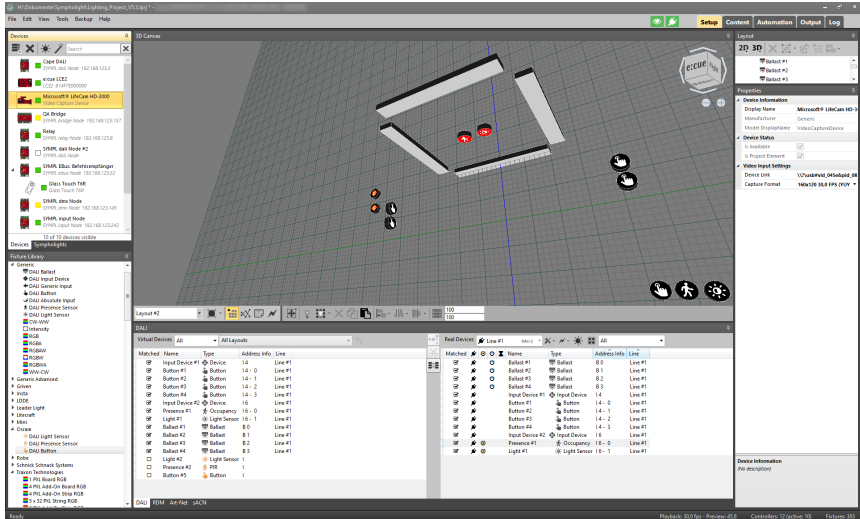
Each device type has its own Workflow Block in addition to the more generic and DALI line related “line” Blocks.

The devices of the type “I” are handled individually by its own Workflow Block: the Block is assigned to a single device only. The devices of type “B” are handled in groups.



9.9.1 Placing DALI Devices

DALI devices are also part of the Fixture Library and are placed and arranged the same way as DMX devices.



There are two groups in the Fixture Library containing DALI devices. In the group Generic, you will find DALI devices that are supported vendor-independently, e.g. ballasts and sensors, and comply with DALI version 2. SYMPHOLIGHT can control all DALI ballasts that conform to the DALI Standard and the version 2. In the group OSRAM, DALI buttons and sensors from the OSRAM DALI professional range are available. Dragging the devices from the Library into the Workspace will create virtual devices. You can configure their properties in the Properties Window.

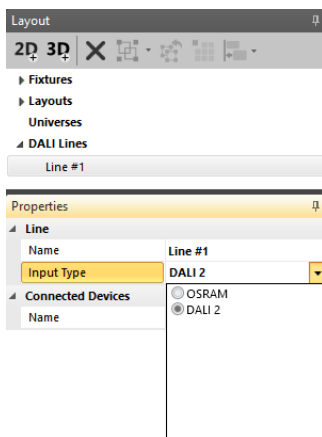
When you have real DALI devices connected to the SYMPHOLIGHT network, they are automatically displayed in the “Real Devices” list on the right side of the DALI Window. For unmatched real devices you cannot change properties or use the whole range of Workflow Blocks. To create matching virtual devices, simply drag and drop a real device from the list into the Workspace. This works for all DALI device types. When a real input device is dragged and dropped onto the canvas, you can add all available instances as well: press the **Ctrl** key while adding the device. The matched virtual input device and the matched virtual input instances of the real instances are created automatically.

9.9.2 Addressing DALI Devices

Using DALI requires at least one SYMPL dali Node.

- Add it to the project as usual. A DALI line will be assigned automatically to the Node.

- Define the DALI protocol for the line according to the DALI devices you plan to have on the line: Select the line in the Layout Window and choose its protocol in the Properties Window at "Input Type":



- Connect the SYMPL dali Node to the actual DALI equipment.
- Place the expected DALI devices in the Layout, e.g. based on a picture of a floor plan.
- In the DALI / RDM Window, select the DALI tab.
- The DALI Window is split into two different areas: the left area is called Virtual Devices and is showing all devices that have been added to the 3D view before. The area on the right hand side is called Real Devices and shows all DALI devices that have been automatically discovered on the selected DALI Line.
- In the DALI subwindow, select a DALI Line on the right part (Real Devices). The SYMPL dali Node will scan all devices.
- All detected and configured DALI devices are visible on the right side of the DALI Window, all DALI elements in the Layout are available on the left side.

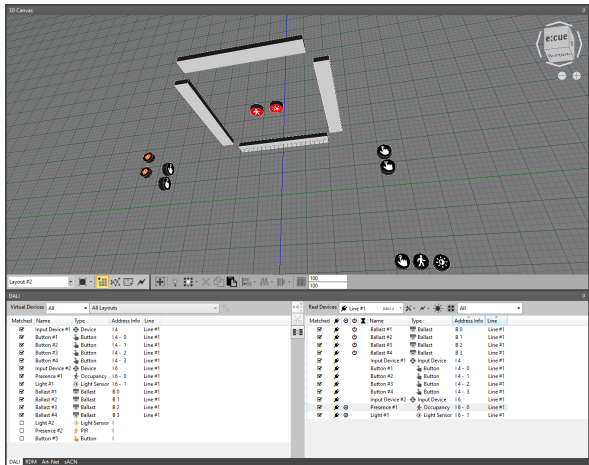


The current SYMPHOLIGHT release will not support reading DALI device parameters back into the software but will only push the device configuration from virtual devices onto the real DALI devices.

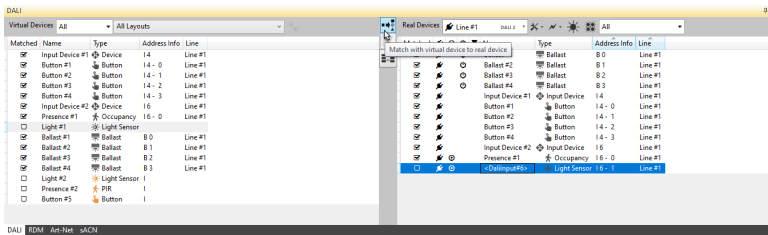


Do not mix DALI devices from the group Generic and the group OSRAM on one DALI line. They have different protocols. Only put devices from one group on a line.





Now map the virtual to the connected devices, i.e. to the real devices, if unmatched. Click on a device in the left list for a virtual DALI element, or select the device in the workspace. Then select in the right list a detected DALI device. The left DALI device will stay selected in light grey. Click on the [Match virtual device to real device](#) icon located between the two lists to match the two devices. Or drag-and-drop a virtual device from the left list to the real device on the right list.



To unmatch a device pair, click [Break match virtual device with real device](#) in the middle pillar. To highlight the mapped devices, click [Associated selection on/off](#), also located in the center: First activate the button, then select the device (virtual or real) to which you want to know the mapped counterpart.

Here are the columns displayed in the DALI Manager:

Virtual DALI Devices

Matched If marked, the virtual device is mapped to a real device.



Name	The name of the virtual device.
Type	The DALI type of the device.
Address Info	Combined information of the general device type ("B" for "ballast", "I" for "input"), short address (first figure), and instance number (second figure).
Line	The matched DALI line.

Real DALI Devices - in addition to the above

Matched	If marked, the real device is mapped to a virtual device.
---------	---



The device is connected.



An exclamation mark is shown, if the device is virtual but not connected.



A flash is shown when two or more devices have the same short address.



Signals an event for a device, e.g. a button being pressed or a sensor signaling a change. To react on these events use the device fitting Workflow Blocks.



Switches the DALI device on / off.




Click to activate/stop the feedback channel of the instance. Only available for instances that support DALI feedback functionality.

For a comprehensive overview over DALI feedback functions in SYMPHOLIGHT see „[17 DALI Feedback Function](#)“ on page 461.



The Update State indicates that property settings are pending to be transmitted to the DALI device.

If there are unaddressed devices at the line available, this icon  indicates this situation.



In the list of virtual and real DALI devices, there are filter functions in the top menu bar, to select which devices are displayed or hidden. If the subwindow is too small to show all of these drop-down lists, click the small triangle on the right border of the window.

DALI

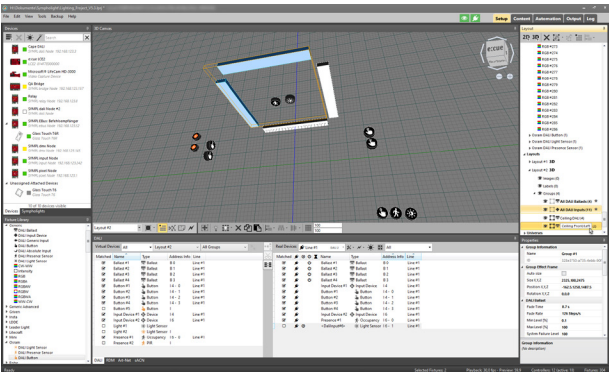
Virtual Devices		All	Layout #2	Ceiling DALI	
Matched	Name	Type	Address Info	Line	
<input checked="" type="checkbox"/>	Ballast #1	Ballast	B 0	Line #1	
<input checked="" type="checkbox"/>	Ballast #2	Ballast	B 1	Line #1	
<input checked="" type="checkbox"/>	Ballast #3	Ballast	B 2	Line #1	
<input checked="" type="checkbox"/>	Ballast #4	Ballast	B 3	Line #1	

To switch the sorting order of the DALI devices, click on the column heads. You can sort by multiple criteria: press the <Shift> key and select several headers. Example: You want to sort by Line and by Address Info. Then keep <Shift> pressed while clicking “Line” and “Address Info”. The List entries will be sorted first by Line, secondly by Address Info.

There is no explicit Patch Mode for DALI devices like for DMX devices. Set the properties of a DALI device either in Patch Mode or in Build Mode. Select a DALI device from the left list and change the properties of the device in the Properties Window, e.g. Fade Rate or Fade Time.

9.9.3 Grouping DALI Devices

SYMPHOLIGHT handles DALI ballasts and DALI inputs always in groups. To create a group of DALI ballasts and inputs, select the ballasts and inputs in the Workspace, either by drawing a frame with pressed left mouse button, or by clicking ballasts keeping the **Ctrl** key pressed. Click the Group icon in the Layout window and a new DALI group will be created in the Layout list. Rename the group as required. If you want to control a single DALI ballast or input, create a DALI group with this ballast or input as single member.



Selecting the DALI groups in the Layout list will highlight the included ballasts and inputs. Also selecting the All DALI devices will highlight the included DALI devices.



When selecting a DALI device, you can edit the device properties in the Properties Window. This includes the usual DALI parameters like Min and Max Value, Fade Rate and Fade Time, Power On Level etc. When more than one ballast is in a group, you have to define a Reference Ballast in the DALI Group Block in the Workflow Designer. See [„11.15.49 Ballast Group and Daylight Harvesting“ on page 255](#) for more details.



Only put DALI devices with identical Dimming Curve AND Color Type settings into one DALI group.

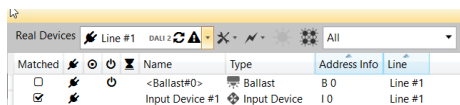
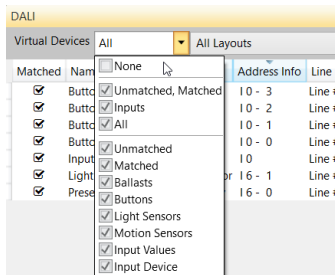


It is recommended to use state of the art DALI devices.

9.9.4 Additional DALI Functions and Features

There are further DALI functions in SYMPHOLIGHT.

Filter: The drop-down lists at the top provides a filter function for which devices are displayed.



Status information: Next to the line filter of real devices, status information about the selected line(s) are displayed:

DALI 2 Line is configured to DALI v2 protocol.

OSRAM Line is configured to OSRAM protocol.




The line is busy, i.e. scanning, addressing or initializing.




There is a problem. To see details about it, hover over the icon in the drop-down list. The tool tip displays the details.



Zoom: Zoom to specific DALI devices in the Workspace like to every other fixture via the Layout Window. Additionally, you can zoom from within the DALI Window: for a single device, go to the device and click its “type” icon (finger cursor). The canvas zooms to DALI device.

Testing: To test or verify your DALI connections, there are two test functions available in the top menu bar of the list with real devices. To use them, select the DALI line to test, here Line #1. Do not use All Lines! Click the single light  and select a ballast. The selected ballast will blink. Click the light matrix and the blink commands will be broadcast to all connected DALI ballasts, even though they are not addressed yet. For sensors like PIRs or for buttons, the Event icon in the list of real devices will blink. This way, you can identify buttons or sensors by watching the list of real devices. If identify functionality is supported by the DALI devices, the devices themselves give an identifying signal as well.

Test a whole DALI Line using Broadcast by clicking the lightning icon  :

Fader	Use the fader to set the current value (0 - 254) of the selected DALI Line. Consider fade time.
	The power button toggles to the last ON value (1 - 254) or to 0. Consider fade time.
Min	The Min button sets the DALI Line to its physical minimum value (never 0). Without fade time.
Max	The Max button sets the DALI Line to its physical maximum value. Without fade time.

Another test function setting the luminance is available in the Test mode (see „9.14.2 DALI in Test Mode“ on page 89).

Tools: The following DALI Line Tools are available at  :

Scan All	Scans the complete line for ballasts and inputs.
Delete Devices with Address Collisions and Rescan	First, all collision ballast and input devices will be deleted. Then a new scan on all devices starts.
Address only unaddressed	Only devices without short address will be addressed.



Readdress all Devices	First, all devices will be deleted. Then the complete line will be addressed. This eliminates the complete previous address settings.
Stop current Scan or Addressing	When one of the actions has been triggered erroneously, you can stop this action here. Changes, that have already been implemented, cannot be reverted.


The context menu of a real device offers the following functions:

Match with Virtual Device	Matches the real device with the selected virtual device. For devices and instances.
Break Match with Virtual Device	Breaks the match with the connected virtual device. For devices and instances.
Check Device Address for Collisions	Checks a single device for addressing conflicts. For devices with short address only.
Reset Device	Resets the device to its default settings. Use only with expertise and in case the device does not longer work properly or is misconfigured. For devices with short address only.
Remove Device	Removes the device from the Real Devices list (= hide). If the device is no longer available, its short address is available again for assigning.
Delete Short Address and Remove Device	First, the device gets unmatched. Then its short address is deleted and the device is removed from the line. The short address is available again for assigning. For devices with short address only.
Modify Short Address to	The device gets the selected new short address and its data are scanned again. For devices with short address only.



Bus Behavior: For an interrupted connection (Network problem) from SYMPL to SYMPHOLIGHT (PC/Core Engine), you can define the DALI bus behavior. Set the settings in the Properties of the controller (SYMPL dali Node) in the Setup tab:

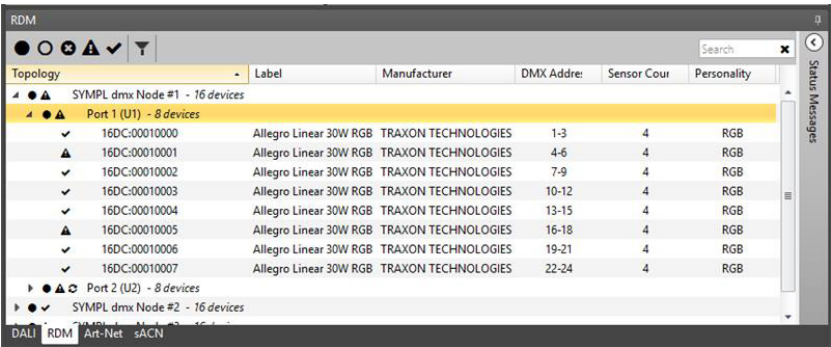
Bus Power On Start	Set the DALI bus power on startup [on] or [off].
Bus Power Off Timeout (s)	The time in seconds after the DALI bus power will be switched off in offline mode (0 = never).



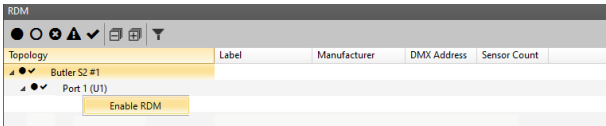
SYMPHOLIGHT will automatically keep scanning the DALI line and addressing new devices if the DALI window is visible in the Setup tab. Please be aware that a constant scanning of the DALI bus might slow down the execution of other DALI relevant commands.


9.10 RDM Devices

In the RDM Window, SYMPHOLIGHT collects for all DMX devices and their subdevices data like sensor values or functionality indicators. The RDM browser is able to handle up to 200 DMX universes.



By default, RDM is not enabled for interfaces. Right-click on a port of a DMX node in the RDM Window and select Enable RDM:





Only enable RDM where and when needed.

In the top icon row of the RDM Window, you can filter RDM events by their level or you can search for specific RDM entries. Additionally, hide or show devices depending on their status.



Show / hide active devices.



Show / hide inactive devices.



Show / hide faulty devices.



Show / hide devices with warnings. Warning = sensor value is out of range.



Show / hide valid devices.



Enable / disable filter.

Filter options: Manufacturer Label, Device Label, Software Version Label, Device Model Description, Power State, DMX512 Start Address.



Collapse all. Only available when the filter is enabled.



Expand all. Only available when the filter is enabled. Only use temporarily and when necessary because the expanded view consumes many resources.

The Topology column shows the node structure, expand/ collapse buttons, status icon and the device name.

Clicking a single device or a subdevice, you can edit values not greyed-out in the Properties Window, if the device supports this.

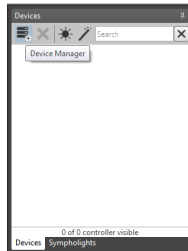
9.10.1 Addressing RDM Devices

You can manually set the DMX start address for RDM devices in SYMPHOLIGHT. The following steps describe how to set the DMX start address in SYMPHOLIGHT. It is device-dependent which RDM properties are available to set the DMX start address at [step 8](#). To find out the device-dependent properties and the instructions to change the DMX start address, read the Information for Use of the individual RDM devices.

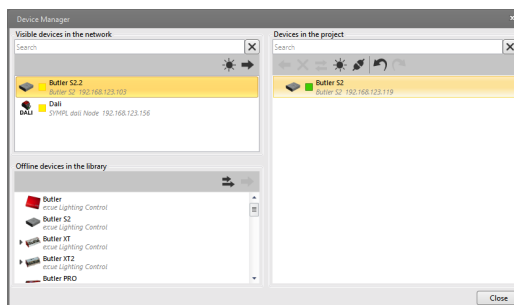
You need to have the RDM devices connected to the network and running.



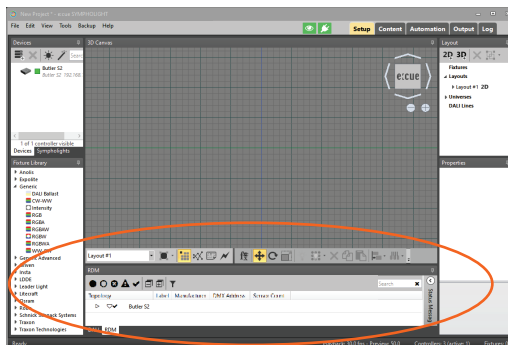
1. Open SYMPHOLIGHT and add your DMX node or Butler etc. by clicking on Devices tab (in Setup tab) and then select Device Manager.



2. Add the output device to the project by double clicking the device in the left panel, it will then appear in the right panel. Once the output device is in the right panel you can close the device manager.

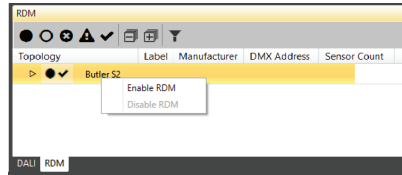


3. Select the RDM tab at the bottom of the screen.

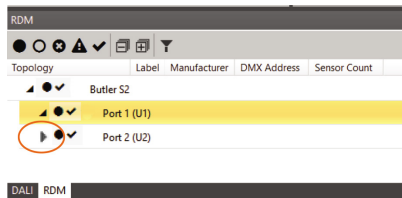


content

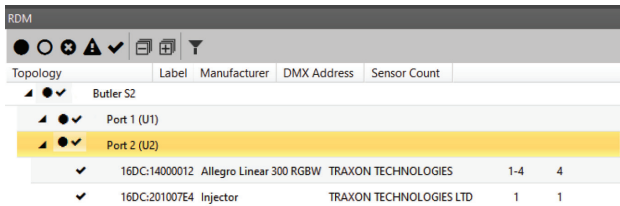
4. Enable RDM mode for the output device by right clicking on the output device and select **Enable RDM**.



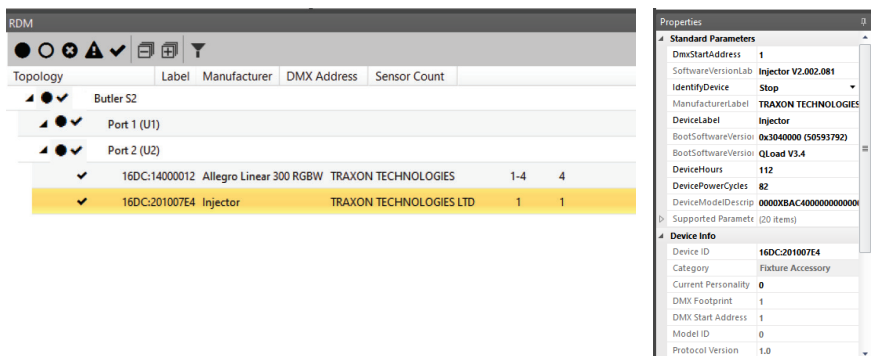
5. If there are any RDM devices on the output device line then a triangle arrow will appear to the left, in this case port 2.



6. Click the triangle arrow and the list will expand revealing all of the detected devices on the output port of the output device.



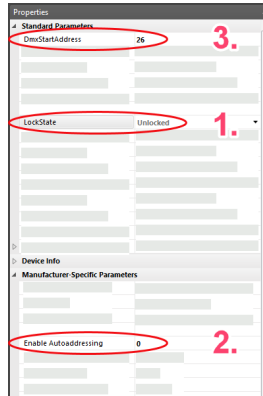
7. Click on an RDM device, here the Injector. Now its properties are displayed in the Properties tab on the right hand side of the screen.



8. Configure the properties that effect the addressing. It is device-dependent which RDM properties these are and how to. The settings are immediately effective.

Two examples for different devices to edit the DMX start address.

Traxon Allegro Dot: First, enter a LockState in which Enable Autoaddressing and DMXStartAddress are editable. Second, disable Enable Autoaddressing by setting it to 0. Third, set the DMXStartAddress to the desired value.



Traxon Allegro Liner: DMXStartAddress: To affect this property, you must change the value of Offset Segment Data.

Offset Segment Data: This will offset the DMXStartAddress of the device by the value you specify plus 1, that is: Offset Segment Data value = DMX Start Address value - 1.

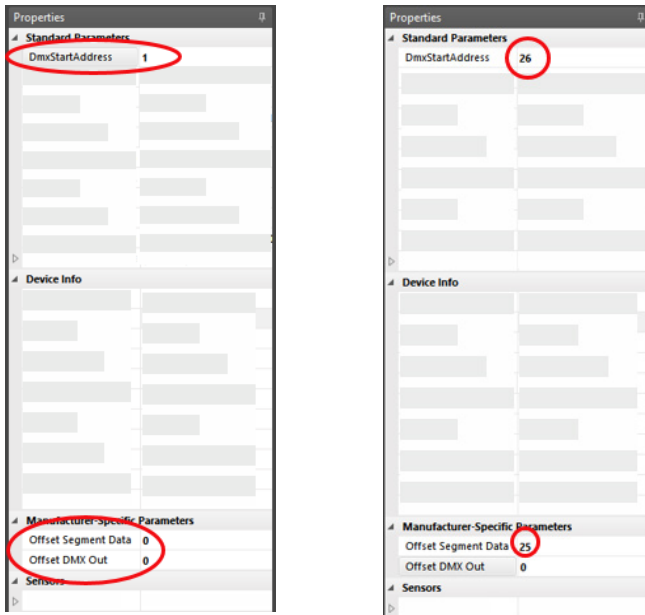
Example: required DMX Start Address value = 26.

Offset Segment Data value = $26 - 1 = 25$.

Offset DMX Out: This value will offset the DMX data output of the current device to the next device on the circuit.



Left: initial state. Right: changed DMXStartAddress.



The addressing of the device is completed.

9.10.2 Sorting RDM Devices

You can sort the displayed RDM devices by using the table head.

RDM nodes are sorted by name only.

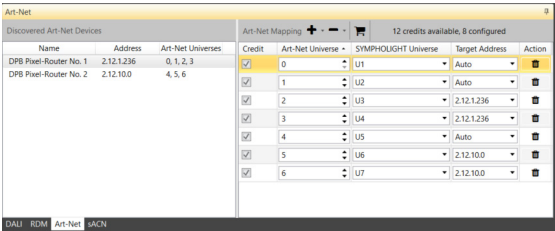
To sort RDM devices and subdevices click on the column head you want to sort by: Label, Manufacturer, DMX Address, Sensor Count or Personality. The Personality column is only for rootdevices/subdevices that support the RDM parameter "DMXPersonalityDescription".

It is only possible to select one column for sorting; multi columns sorting is not possible. And it is only possible to use ascending sort direction.

9.11 Art-Net Devices

In the Art-Net Window, the discovered Art-Net output devices are listed.





In the Art-Net Mapping list, add Art-Net Mapping entries, map Art-Net universes to SYMPHOLIGHT universes and assign credits to enable content output.

Define with the Target Address to which IP address the data is to be transmitted. Auto (automatic mode) transmits to the IP addresses of all Art-Net devices that are subscribed to the specified Art-Net universe. Or it falls back to broadcast if there is no unique subscription to an Art-Net universe:

Number of Art-Net universe occurrences	Subnet mask of Art-Net devices	Function of “Auto”
1	irrelevant	Sends data to the Art-Net device (1 receiver, unicast)
multiple	same	Sends data to the subnet mask (multiple receivers, subnet mask specific broadcast)
multiple	different	Sends data to Default Interface (IP address specific broadcast)
none	-	Sends data to Default Interface (IP address specific broadcast)

Auto as unicast or subnet mask specific broadcast only works when all Art-Net devices are listed as discovered devices. In the bottom two scenarios of the table above, Auto broadcasts the data to the subnet as specified by the IP address that is defined at Application Settings | Network | Default Interface (see „Default interface“ on page 38). Use the Bulk Add option at the Add drop-down menu to create multiple new Art-Net universes including mapping and Target addresses.

Note that SYMPHOLIGHT cannot configure the Art-Net devices and that changes in the device configuration are only inherited in the mapping when using Auto.



To enable an fast and easy configuration of large Art-Net installations in SYMPHOLIGHT, for example with the **Bulk Add** option, it is recommended to

- prior to the mapping in SYMPHOLIGHT and
- in accordance with the Art-Net standard
- carefully plan and configure the Art-Net devices with their addresses and universes, and
- create and patch all needed SYMPHOLIGHT universes and fixtures.

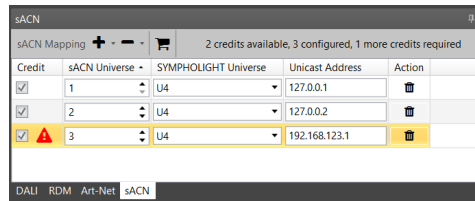
For example, use consecutive Art-Net and SYMPHOLIGHT universe numbering. Also stay within the tolerated universe range, usually 16, when configuring the ports of an Art-Net node. I.e. the assigned universes of the ports remain within a block of 16, e.g. universes between 0 and 15.



When sACN or e:net are used in addition to Art-Net in the installation, ArtSync (see „[Enable ArtSync packets](#)“ on page 37) has to be enabled on to ensure stable DMX output.

9.12 sACN Devices

In the sACN Window, the addressing of the sACN devices is located. First add sACN Mapping entries, then map sACN universes to SYMPHOLIGHT universes, and assign credits to enable content output. Enter the **Unicast Address** - with an empty **Unicast Address** field, the sACN Universe data is sent multicast.



Use the **Bulk Add** option at the **Add** drop-down menu to create multiple new sACN universes including mapping.

For sACN multicast make sure your network hardware is sACN compatible. Please note that for sACN multicast you need:

- one switch that is configured as an IGMP Querier;
- all switches to have IGMP snooping enabled.

To enable an fast and easy configuration of large sACN installations in SYMPHOLIGHT, e.g. with the **Bulk Add** option, it is recommended to

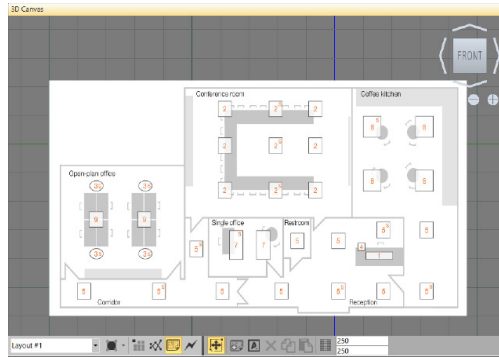
- prior to the mapping in SYMPHOLIGHT
- carefully plan and configure the sACN devices with their addresses and universes, and

- create and patch all needed SYMPHOLIGHT universes and fixtures.

For example, use consecutive sACN and SYMPHOLIGHT universe numbering.

9.13 Background Mode

In Background Mode you can place background images or labels with text in the Workspace.



The Background Mode has two additions to the know functions and icons.



Image Mode: Add an image to the workspace.

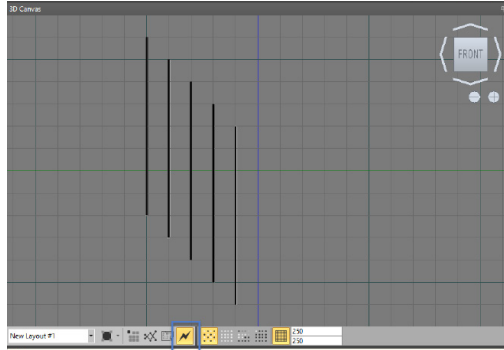


Label Mode: Add a label (text box) to the workspace.

Click the picture or the text block icon in the Mode Selection. Select an element to add or remove the current selected element. Use the Properties Window to change scaling or size, color, font or other parameters.

9.14 Test Mode

In Test Mode, you can test and check your DMX setup and patching. And you can test your DALI ballasts using a brightness slider.



9.14.1 DMX in Test Mode

The Test Mode supports four test scenarios for the connected DMX fixtures.



Blink: all channels in the selected fixture blink.



Highlight: all channels in the selected fixture are set to full intensity,




Running Highlight: A pattern runs from lowest to highest channel though the selected fixture. You can define the speed in the Application Settings.



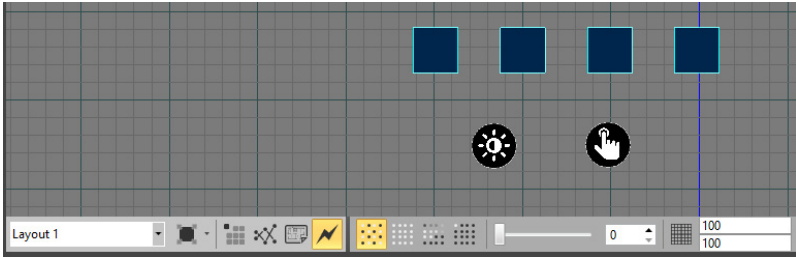
Tile Orientation: The lowest channel in the selected fixture is at full intensity. The other channels are dimmed. The other unselected fixtures are dimmed in blue.

Click a fixture, select the test pattern. For the Running Highlight, make sure that the direction in the preview in the workspace and on the fixture is the same. Otherwise rotate the fixture.

9.14.2 DALI in Test Mode

Select the DALI ballasts you want to test. The ballasts have to be available as virtual devices and linked to real devices. Select them in the canvas or the Layout window at Fixtures or Layouts - Groups. In Test Mode, use the slider or the entry field next to it  to set the luminance of the DALI ballasts. The tools are only available when compatible DALI devices (e.g. ballasts that are matched) are selected.





To switch between 0 and 100%, press the H key.

The test value will not be send while moving the slider, it will be send on mouse up.

Note: Brightness changes that come in (e.g. workflow) immediately overwrite the set test value.

9.15 Patch Files

While all parts of a project, Content, Setup, and Automation, are stored in the project file, you can also save and load the Setup results in a separate file. These files are called Patch files and contain only the fixture and interface configuration. To save or load a Patch file, use [File / Import Patch](#) or [Export Patch](#). With [File / Export Patch with Media](#) also images and Labels are included.



10 Content Window

10.1 Principles

The Content Window of SYMPHOLIGHT is the central area where you create the project, its content and how this content is handled, including fades and control structures. SYMPHOLIGHT works with a timeline like a sequencer. All functions and control is based on this timeline.

SYMPHOLIGHT is a media-oriented system with focus on media and effects, at first sight. These videos, images and effects are placed in the Layers of a Sequence. But, it is also possible to control even single LEDs in a fixture. Even if a timeline is often seen as not so exact and controllable as, for example, cuelists, this is not true. In SYMPHOLIGHT, all components of a Sequence can be adjusted down to milliseconds. The snap mechanism in SYMPHOLIGHT allows very precise adjustment even for fade times and cues for run control. The advantage of a timeline is the clear graphical user interface, which leads to a more intuitive work flow. You always see what you will get, compared to other concepts, like cuelists.

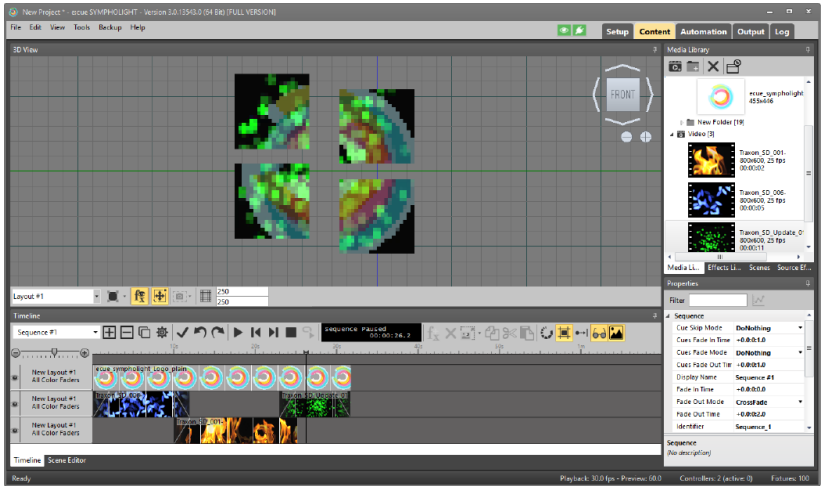
The content of a project is organized in Sequences, these Sequences contain Layers. The basic Layers are created automatically during the Setup process. Depending on the types of fixtures placed in Layouts, a Sequence contains different Layers. If the fixtures in a Layout are RGB-capable, the corresponding Layers will be set for color channels. If the fixtures only support intensities, the Layers will be intensity channels. DALI ballasts are handled in a separate way and do not have Layers. While the basic Layers are created while defining the Setup, you can add additional Layers by placing media content or effects in the timeline.



This chapter deals in general with DMX-oriented functions in SYMPHOLIGHT. DALI functionality is explained in more detail in the chapter Automation, even though some few DALI functions are part of the Content Window, e. g. DALI Scenes (see „[10.18 DALI Scenes](#)“ on page 108).



10.2 Window Structure



Preview Window

Media, Effects,
Scenes

Control Bar
Play/Pause/Stop/Skip, select Sequence handling.

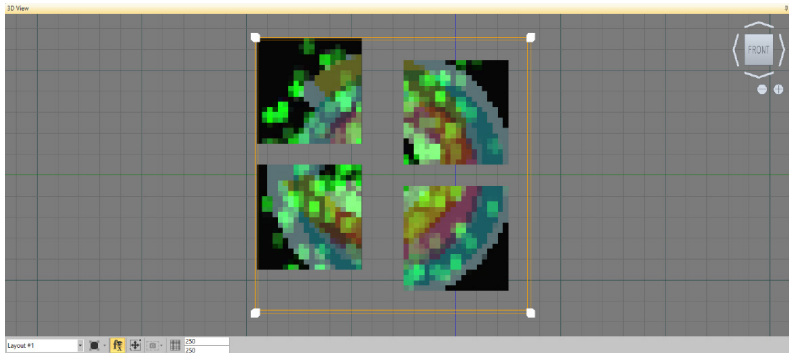
Properties
Window
Edit properties
of timeline
components,
Effects or
videos.

Timeline and Layers in Sequence
Sequences with Effects and Media



10.3 Preview Window

In the Preview Window, you see the simulation of the output that is sent to the fixtures.



The Layout, that is displayed, can be selected by the drop-down box at the bottom of the Preview Window. Zooming in and out is done with the control elements in the Preview Window or with the mouse wheel. The yellow frame is the area on which an Effect or a media is mapped: the Effect Rectangle. Additional functions are available with the icons below. The Effect Rectangle frame is the view port and defines, how the content of the Sequence is mapped on the fixtures. So the view port can be smaller than the fixture area, where only a part of the content is mapped.



Optimize view for current window size.



Hide or show the Effect Rectangle of a selected effect in a Layer.



Combined move, scale, and rotate tool for the selected effect rectangle.

Move: hover over the selection and use the .

Scale: use the cubes at the border of the selection.

Rotate: hover over the circle at the top of the selection and use the .

You can also change the position, the size, and the rotation angle in the Properties.



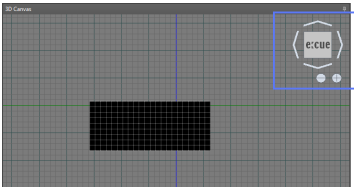
Current content as offline image: run the Project, pause it at a position and use this function to set an output image for SYMPL dmx and e:pix Node for offline situation. They will only receive that part of the output they are responsible for.



Set the grid size for the Preview Window and the View Port.

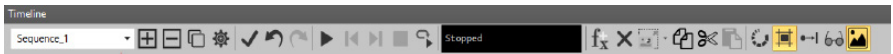


To rotate the Preview Window use *alt + pressed mouse*. The e:cue box in the top right corner indicates the current viewpoint. The four arrow-like elements in the upper right corner of the workspace move the view of the workspace.












10.4 Control Bar

The Control Bar provides a selector for the currently working Sequence, an icon to create a new Sequence or to delete a Sequence as well as control functions.



Changes in the Sequence can be saved or undone. Additionally, information about the running Sequence is displayed and you can add or delete elements in a Layer.

-  Create a new Sequence.
-  Delete the current Sequence.
-  Duplicate the current Sequence.
-  Change the Properties of the Sequence in the Properties Window. You can also click into an empty area of the Sequence to edit the Sequence Properties.
-  Save the Sequence to the Live View. If the option *Auto save changes to content* in the main settings is set, this is done automatically if the Tab is changed or another Sequence is selected.
-  Undo last change.
-  Redo last Undo command.
-  Revert to original Sequence, undo the last changes that have been saved with a Tab switch or before the Sequence has been changed.
-  Play the Sequence from current cursor position.





Pause the Sequence, click again to return to playing.



Skip to previous Cue.



Skip to next Cue.



Stop Sequence.



Play selected Effect and loop.



Create Effect Template: creates a new Effect template from the currently selected Effect and its properties, adds a new template to the Effects library.



Delete Effect/Track: deletes the selected Effect or Track in the Sequence.



Resizes the Effect Rectangle to cover all fixtures in the current Layout, if you resized or moved the effect mapping to fixtures.



Copy the currently selected Effect or media to the clipboard.



Cut the currently selected Effect from the Track to the clipboard.



Paste the clipboard content into the currently selected Layer. The item will be pasted at the position of the time cursor on the selected track.



Auto Preview Selected Effects: toggles preview of the selected Effect only.



Enable/Disable Snapping on Timeline: toggles snapping of media, Effects and Cues in the Layer in the Timeline.



Follow Cursor on Timeline: if enabled, the view on the Timeline always follows the cursor while playing. If disabled, the view on the Timeline stays fixed.





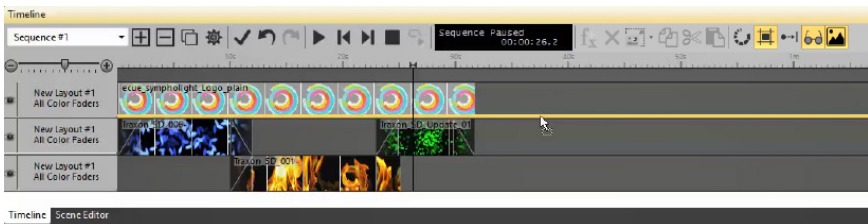
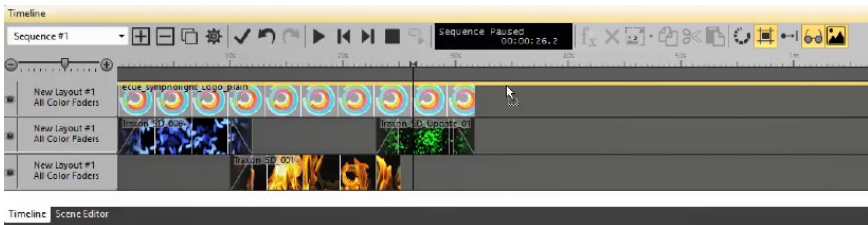
Show/Hide tracks from other Layouts: if enabled, all Layers from all Layouts are visible. If disabled, only Layers for the current Layout are visible.



Enable and disable thumbnails. If activated, the previewed content in the Layer is rendered as the real content will be played. Otherwise only thumbnails are displayed.

10.5 Layers

The Layers contain the real content of the project. In the Layers, media content or effects are placed. For every Layout with DMX fixtures, SYMPHOLIGHT creates an associated Layer in the Timeline in the Content Window. To place media or effects, select Media Library or Effects in the upper right window and drag it keeping the left mouse button pressed into the Layer of the Sequence. For the selected media or effect in the Layer, you can change properties in the Properties Window. Looping can be assigned to a media element or an effect by selecting the element and dragging the end of the element. To create a new Layer for a Layout in a Sequence, drag-and-drop an element of the Media Library, the Effects Library, or the Source Effects on top of or below an existing Layer.

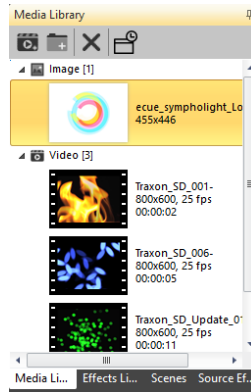


You can create new Layers by dragging media or Effects from the libraries into the Timeline, but you can also create new Layers by dragging content elements from one Layer to the upper or lower border of a Layer.

Next, you can rearrange the Layers by keeping the header of a Layer clicked and moving the Layer above or below another Layer. Every Layer has two small buttons in the Layer header. With the M icon you can mute (switch off) a Layer. With the S icon you can solo the Layer, so it becomes the only active Layer.

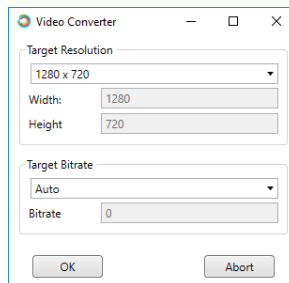
10.6 Media Library

To add media to the Media Library, select the chapter - Image, Video, or Sources- and use the clap board icon to add a file.



In the opened dialog, select media to be added to the Media Library. You can create subfolders in the Media Library with the folder icon in the top menu of the Media Library.

Video files will be converted to an internal format. Several video formats are supported and will be converted to a certain format in the Media Library. Define and customize the resolution of the imported videos and the bitrate.

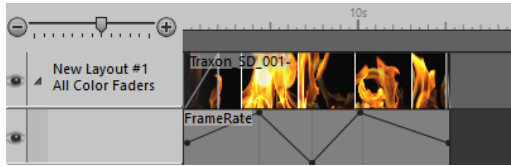


The SYMPL Fusion or SYMPL Core S(p) supports resolutions with 720p and lower. It does not support HD resolution.

For video files, SYMPHOLIGHT supports a tool for frame rate interpolation for videos. The tool is available via the link in the Media Library. This tool re-samples videos for slow motion or high-speed playback to avoid hard frame rate delays. Download the Free Video



FPS Converter from the site and install it in your system. When the tool is installed, the message will be replaced by an icon for the tool.



SYMPHOLIGHT will detect the tool automatically. Place a video in a Layer in the Sequence, select the video, right-click in the Properties window of the video on Frame Rate and select **Add Keyframing**. Double-click in the Keyframes below the video to place key frames and change the frame rates to create slow-mo or hi-speed effects.



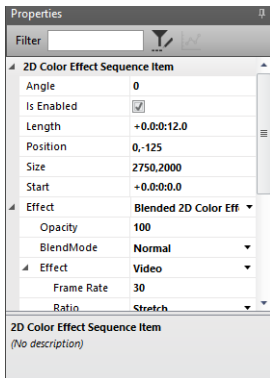
Only videos imported (or re-imported) after installation of this tool can be used with frame rate animation.

The “Live Video” is a place holder for content from a video capture device. Drag it to the Timeline, then select at “Source” in the Properties window the video capture device from which the content is sourced. The place holder has now changed to the media content and is matched to a device. You can adjust the media content with the same options as other content.

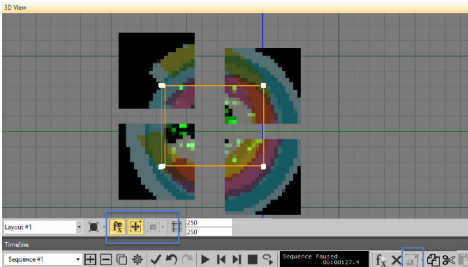
10.7 Properties Window

When an Effect or a media element is selected in a Layer, you can edit the properties of this element in the Properties Window.

Click the item in the Layer to select it. There are many properties for Effects and Media to edit. Most properties and the names are obvious. Videos and images are always used as a 2D Color Effect Sequence. When you adjust an Effect with properties different from the basic settings, you can save this Effect in the Effects Library above the Properties Window. Give the Effect a new, unique name and save it with the Create New Effect Template icon. There are some more control elements in the main menu of the Timeline Window: You can set up an Effect Template, delete Effects or Layers, enable or disable timeline snapping and much more. Use the filter option to select only certain content.



10.8 Adjust Mapping to Fixtures



By default, an image, video or Effect will be mapped to the full Layout size of the fixtures. Adjust the mapping by selecting the object in the timeline and change the mapping. Use the icons below the preview to move, rotate or resize the Effect Rectangle.

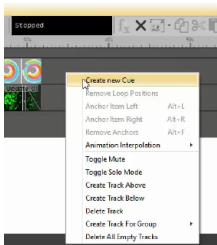
Click the icon in the lower right to reset the mapping to the complete Layout

10.9 Cues

You can add Cues to the Timeline. Cues are control elements for fades or to control playing of content. Cues control fade ins and outs as well as playing states.

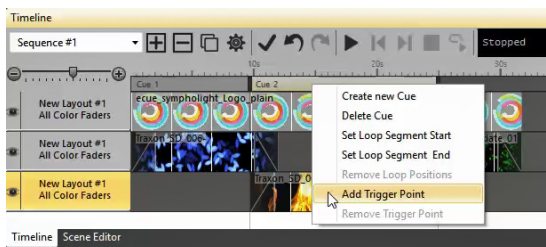


If you know Cues from the Programmer of the LAS, do not be confused. Cues in SYMPHOLIGHT are control elements, not lighting scenes!



To add a Cue, right-click in a free area of a Layer, or right-click in the Cue track (the grey band between time markers and Layouts) and select **Create new Cue**, or double-click into the Cue track. A new Cue will be created. Click on the Cue and adjust its properties in the Properties window. By default Cues refer to the Cue settings of the Sequence.

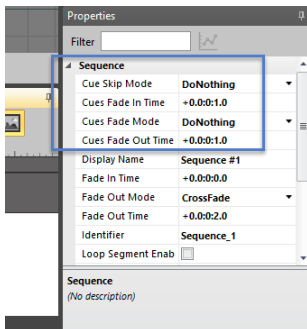
When executing the project, you can use functions to jump to Cues forward or backward, or to continue from a definite point in the Timeline. An additional control element in Cues are Trigger Points. To add Trigger Points, right-click into a Cue and select **Add Trigger Point**. Later, in the Workflow Designer, you can react on such a Trigger Point. Please note that Trigger Points only perform when triggered from the Output tab.



You can change the length of a Cue by dragging its end or start. There are three modes to do this. Without additional keys pressed, the length of a Cue changes and other Cues move. Keeping the Ctrl key pressed, the length of the affected Cues are changed, whereas the other Cues do not move. Keeping the Alt key pressed, also the Effects of the affected Layers in this Cue are moved. You may combine the Alt and Ctrl key.

10.10 Sequence and Cue Properties

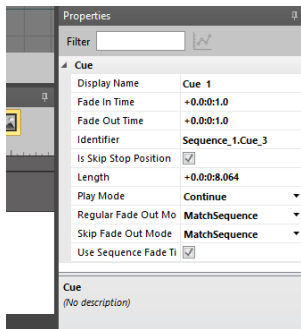
To set the Sequence properties, click in an empty area of the Timeline or click the cogwheel icon in the Control Bar: .



In the Sequence properties, the default behavior for all Cues is defined. This is the Skip Mode (DoNothing, CrossFade or FadeFromLastFrame), the fade in and fade out times and more. These default Cue properties are used if not defined otherwise. To change the Cue properties for a single Cue, click the Cue to select and change the properties in the Properties Window.

Important parameters for a Cue are Play Mode (Continue: continue playing, Wait: pause on end of Cue, Repeat: loop this Cue) and Regular Fade Out Mode, e. g. to realize a cross fade between two content elements in a Layer. Uncheck the box Use Sequence Fade Times to set fade times different from the times defined in the Sequence properties.

On the top of this window a small filter icon is shown. If selected, the properties window will show all properties relevant for this



item. If deactivated, only the most important properties are shown.

10.11 Fading

Fading can be used in SYMPHOLIGHT for all elements in a Layer. There are various sources for fading in SYMPHOLIGHT. Here are the control possibilities for fades.

- Element-specific: Fade-In and Fade-Out are directly defined in the properties of the element.
- Fade-configuration: this is defined in the applications main settings: Tools / Settings / Content. This is the default fade time for all Effect and media Layer elements that get inserted. It is also the default setting after installation and enabled.

- Manual edits: at the beginning and the end of an element, there is a ramp. Keep the left mouse button pressed over the border and move the border. This will change the fade time, but will not change the applications settings.

10.12 Scenes and Scene Editor

Scenes are the tool for use when you want to send DMX output to non-fixtures like DMX switches or if you want to define static DMX output to few fixtures. To create static content as Scene the Scene Editor is used. Scenes can be used in the All Color Faders/Intensity Faders Layer like any other content. We show Scenes for the All Color Faders Layers here. Intensity Scenes work the same way, but with only one or few channels. To create a Scene in the Content window, click Scenes in the bottom of the Media Library. As the first step you have to select the fixtures the Scene will be created for. Equivalent to the Setup procedure, click on a fixture to select it, use the Ctrl key while selecting to add or remove fixtures in an existing selection or draw a rectangle with pressed left mouse button.

Select the Scene Editor at the bottom of the Timeline Window. While working with this Scene, the following commands are available.



Create a new Scene for the selected fixtures. Rename the Scene in the name field. You can still modify the selection of fixtures in the Preview Window to add or remove fixtures to the selection.



Save the Sequence. If the option Auto save changes to content in the main settings is set, this is done automatically if the Tab is changed or another Scene is selected.



If enabled, values are percent values, if disabled, values are Byte values.



Enable or disable the color picker to select colors.



Show all fixtures in Scene. This overwrites to previous selection and includes all touched fixtures in the selection.



Copy the current selected values to the clipboard.



Paste the values from the clipboard to the selected channels.

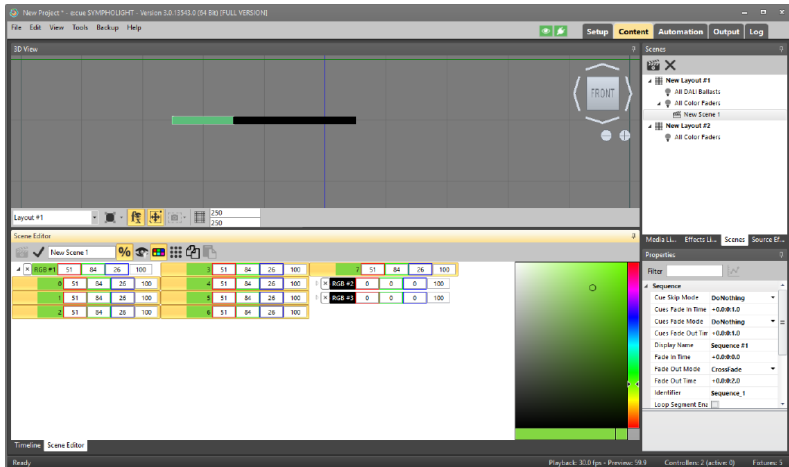
You now have the Scene Editor with the selected fixtures for the Scene and you can assign color or intensity values to the complete fixture or to single pixels in the fixture. To



open or close a fixture with more than one channel, click on the arrow left of the top fixture line.

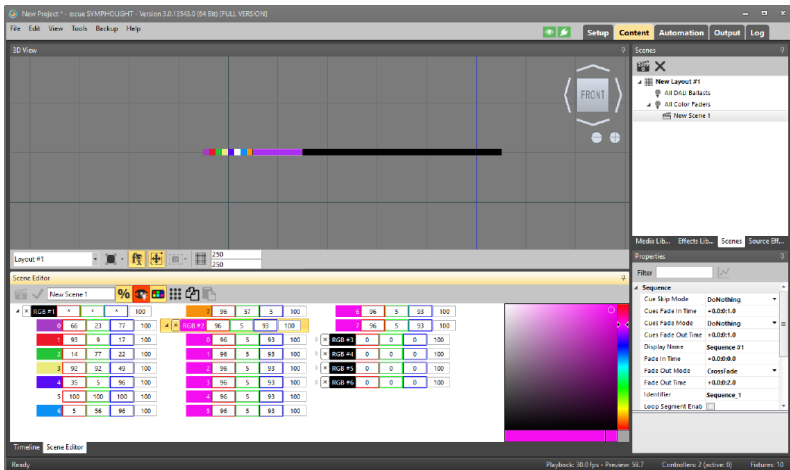
You can remove (untouch) fixtures from the Scene with the cross icon on the left side of a fixture in the list. To place it again in the selection, click the fixture again in the Preview Window. Or click the **Show All** icon to show all touched fixtures of that Scene.

You can copy and paste value settings from one element to another, or for several elements. When copying multiple elements, there may be no gap between them.

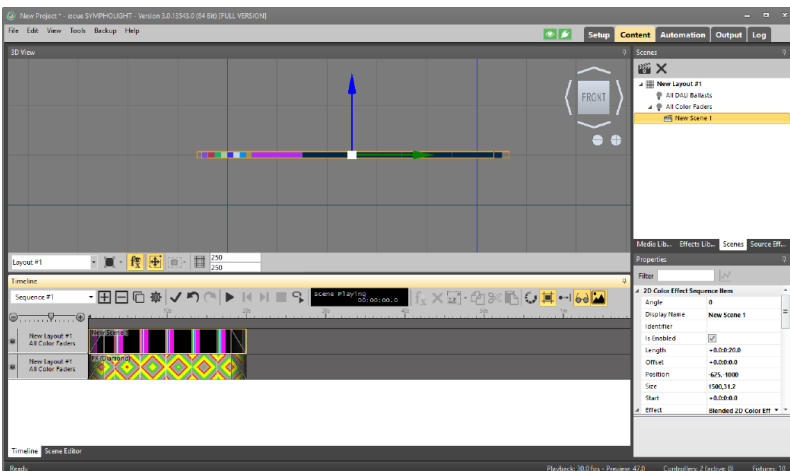


Select a complete fixture or a single channel, or use multi-select with a pressed Ctrl key. Now you can either assign colors to the complete fixture or to the channel with the color picker, or enter values for fixture or channel directly. After assigning value, click the check mark to save the Scene. Fixtures, which are not included in the Scene, receive their channel values from the content defined in the Sequence. Only the fixtures and channels used in the Scene receive their content from the Scene. The assigned colors are displayed in the Preview window and, if enabled by the Eye icon in the upper right corner, on the fixtures.





After saving the Scene, you can drag-and-drop the Scene from the Scenes tab onto a Layer in the Timeline window like any other Layer content. Playing the content now shows the color settings of this Scene as content if the Scene is placed in an All Color Faders Layer.



Use Cues for play control and fades as required. For Scenes in RGB Layouts, you can set the color channels and opacity. For Scenes in Intensity Layouts, you can set intensity and Opacity. The Opacity defines how much of the Layers below is visible. A zero means full transparency, 255 or 100% means no transparency.



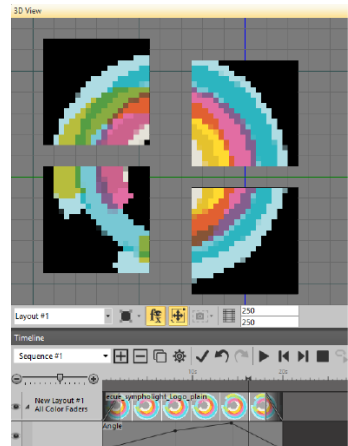
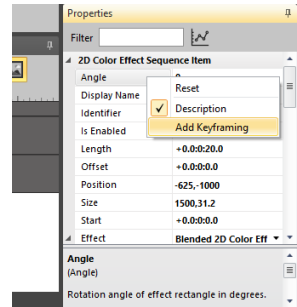
10.13 Keyframing

For certain properties of an effect or a media element you make properties dynamic.

This way, a property of an element can change over time. Not for all properties Key Framing is possible. Right-click on the property name, in the context menu either Add keyframing is disabled (greyed out) or enabled. Select Add keyframing.

A sublayer below the selected Layer will be added, where the animation is defined. Two keyframes were already created: one for frame 0 and one keyframe for the last frame. Double-click into the keyframe sublayer to create a new keyframe. Click the new keyframe to edit its properties, time position and value, in the Properties Window. Right-click in the keyframe sublayer Right-click into the Keyframe Layer to select the animation mode. The context menu contains the following options:

- Animation Interpolation | Step Interpolation will set the selected property directly in one step.
- Animation Interpolation | Linear Interpolation will calculate intermediate steps for a continuous property change.



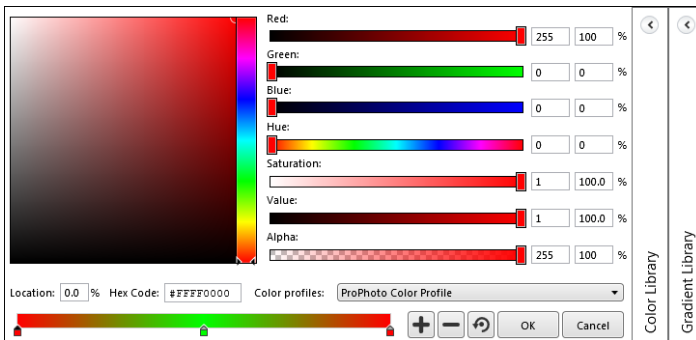
For angle animations, positive values are counter-clockwise rotation, negative values are clockwise rotation.

The keyframe layer also has a mute button in the Layer header, to disable or enable the animation.

10.14 Working with Gradients

In many places in SYMPHOLIGHT, you can select Gradients, for example for Effects or for backgrounds or colors in the GUI Editor. Gradients are not a single color, but a span of one or more colors over an area. A single pure color is handled as a special case of a Gradient.

10.14.1 Gradient Editor



A single color can be defined either by picking a color in the left color picker, or by editing the color and intensity values with faders and value entry fields on the right. The color values depend on the selected Color Profile. You can also define a color with its hex code. The line at the bottom of the editor is the color span, indicating the Gradient Stops and is called the Gradient Preview. The Gradient Stops define the points where color changes are placed.

The faders and value entry fields have the following functions:

- Red/Green/Blue (Amber/White): the color channels of a color.
- Hue: the color type.
- Saturation: deepness of the color.
- Value: intensity of the color.
- Alpha: amount of transparency.

10.14.2 Single Color

Use a single Gradient Stop, use the minus icon right of the Gradient Preview to remove additional Gradient Stops. The position of the single Gradient Stop has no meaning now. Define the color now. Open the Color Library on the right to save this color as color template.

10.14.3 Gradient Markers

Use at least two Gradient Stops. If Gradient Stops are not placed at the beginning or the end, the area in front or after will wear the same color as the Gradient Stop. Add Gradient Stops with the plus icon right of the Gradient Preview. Click a Gradient Stop to select it, move it and set the color properties. Click another Gradient Stop to edit this one. Save this Gradient as a template in the Gradient Library.



10.15 Effects

Effects are an important part of SYMPHOLIGHT as a media application.



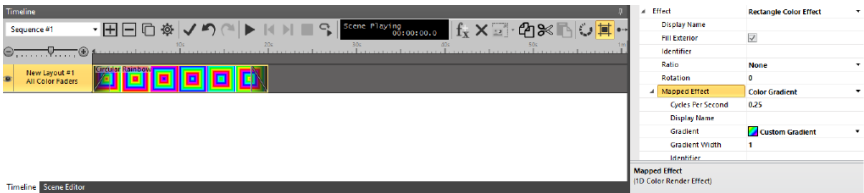
Effects are available as color or intensity versions. SYMPHOLIGHT comes with a rich Effects Library of predefined, ready to use Effects. These Effects can also be adapted by changing the properties of the Effect. You can create your own new Effects and save them for later use in your project. In the Effects Library, user-defined and factory Effects are stored in separate trees. Preview icons give an idea what the Effects look like. Under Source Effects there are the building blocks for creating new effects. The Effects can be used in a Layer directly.

When new Effects are created or properties of a factory Effect are changed, save them as Effect Templates and they will become user-defined Effects. SYMPHOLIGHT will create a new preview icon when the Effect is saved. Place factory Effects in a Layer like any other Effect. Use-defined Effects can be deleted, factory Effects cannot. To delete an Effect, right-click on the Effect and select Delete Effect Template.

Effects from the Effects Library are saved in your system and can be exported and imported. Select the Effect that you want to export/ import, and use the   icons at the top of the Effect Library or its context menu. The file format for SYMPHOLIGHT Effect Templates is *.slet.

10.16 Creating New Effects

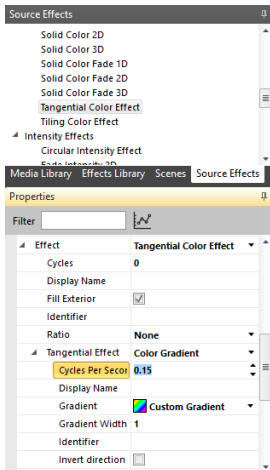
You can create new Effects from the tab Source Effects at the bottom of the Effects Library. To do this, you have to be aware of the way Effects are structured and composed in SYMPHOLIGHT. The two basic building blocks of Effects are Render Effects and Transformation Effects. Every Effect in Source Effects belongs either to Render Effects or Transformation Effects. Render Effects provide colors or intensities. Transformation Effects modify or structure colors or intensities.



Most Effects consist of a Render and a Transformation Effect. In a few cases only a Render Effect can be enough, for example if you only want to use a constant color or a color chase. Both Effect types have properties, which can be changed. This has an influence on how the color rendering happens. The Transformation Effect Particle has many properties like Fade Factor, Lifetime and Movement values. This modifies the transforming Effect. After creating or changing an Effect, you can save this Effect as a user defined Effect. Use the Create Effect Template icon in the Timeline menu or the context menu.



10.16.1 Example of a New Effect: Transformation plus Render

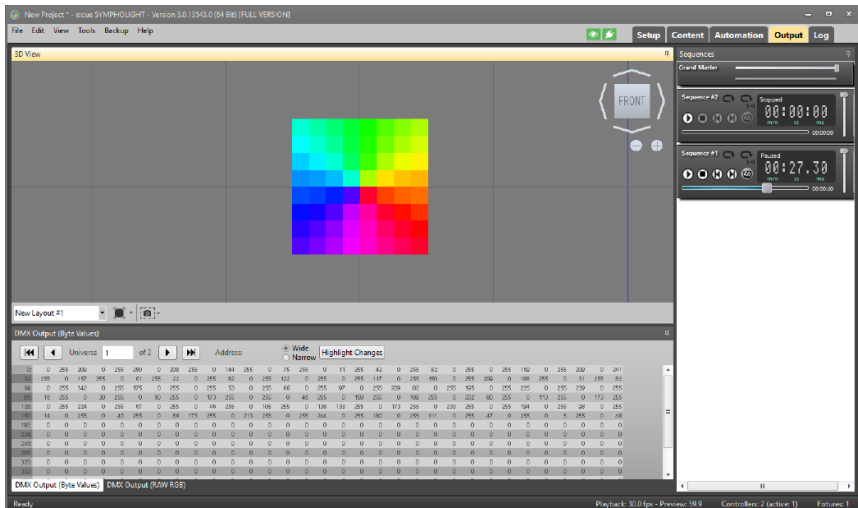


We will now create a new Effect.

Place the Multi-Channel-Spiral Effect from the Effects Library in a Layer. Use the Tangential Color Effect as Effect, the Tangential Effect in the Properties is shown as Color Gradient. Opening the Tangential Effect drop down menu provides the other possible Render Effects.

We use Color Gradient as Render Effect. Now the new Effect is complete, the preview of the Effect is shown in the Layer of the Timeline. You can edit now the Gradient properties by clicking the Gradient property for the Color Gradient Editor. While you change the Gradient, the Preview Window already shows how the final Effect will look like. Save the new Effect via the Timeline menu. Here we use Purple Spiral.

Additionally, the Effect property Cycles is set to 2.5, the Cycles Per Second to 0.15, giving more lines of the spiral.



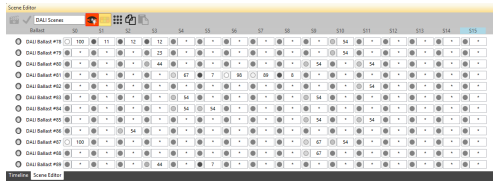
10.17 Media Library

The Media Library contains images and videos to use as content. In the Media Library at "Sources", you find Live Videos from video capture devices. Once you placed a Live Video on a Layer, it becomes a live video effect and is handled like a video effect. Additionally, you can change the source of the live video effect in the Properties at "Source".




10.18 DALI Scenes

Also part of the Content Window is the Scene Editor for DALI Scenes. Select Scenes in the Media/Effects/Scenes Window on the top right and choose a device group you want to work on. Then switch to the Scene Editor in the lower part of the main window. Here you now can define DALI Scenes for the selected DALI group.



With the percent icon in the top left part of the Scene Editor, you can enter percent values or absolute values. When entering percent values, SYMPHOLIGHT recalculates these values to absolute values and the entered percent value may change to binary representation. Calling DALI Scenes for a DALI group is part of the Workflow Designer in the Automation Window.



The values for each DALI scene are stored inside the DALI ballast itself. Due to the DALI standard the number of scenes is limited to 16 scenes only. Therefore it is not possible to create more than 16 scenes per ballast.

10.19 Properties of Content in Layers

The first six properties are common for all objects.

Angle	Rotate the effect(rectangle) in degrees.
Display Name	The name displayed as object name.
Is Enabled	Show the object in Preview and Live view. If disabled, the object is hidden, but still present.
Length	Run length in the Layer.
Offset	An offset used at the beginning of media. Either enter a value or move the beginning of the media with a pressed mouse and Shift key.
Position	Position of center.
Size	Size of the object in mm.



Start Start time in the Layer.

All other properties are specific to the content types like images, videos, Effects, or Cues. For videos, for example, you can choose between three types of Ratio:

Stretch Stretches the effect content to cover the effect rectangle keeping the aspect ratio.

Crop Crops the effect content to cover the effect rectangle keeping the aspect ratio.

Fit Fits the effect content to cover the effect rectangle (without aspect ratio).

10.19.1 Gamma Correction

In addition to the configurable default gamma presets (see „Gamma preset settings“ on [page 37](#)), you can configure the gamma correction individually for videos, images and live videos.

Gamma correction is a conversion between what the source signal says, and what the luminaire output will be. Modulating the gamma values can improve the perception of luminance, shades and colors.

Go to the Properties of the corresponding Layer content. It is possible to configure the red (Gamma Red), the green (Gamma Green), and the blue channel (Gamma Blue). Where applicable, the white channel is manipulated according to the white channel properties settings.

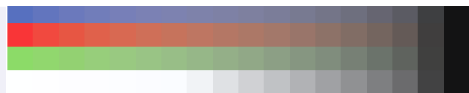
Gamma values range from 0.0 to 15.0. A value of 1.0 causes no gamma correction.

Examples:

Settings

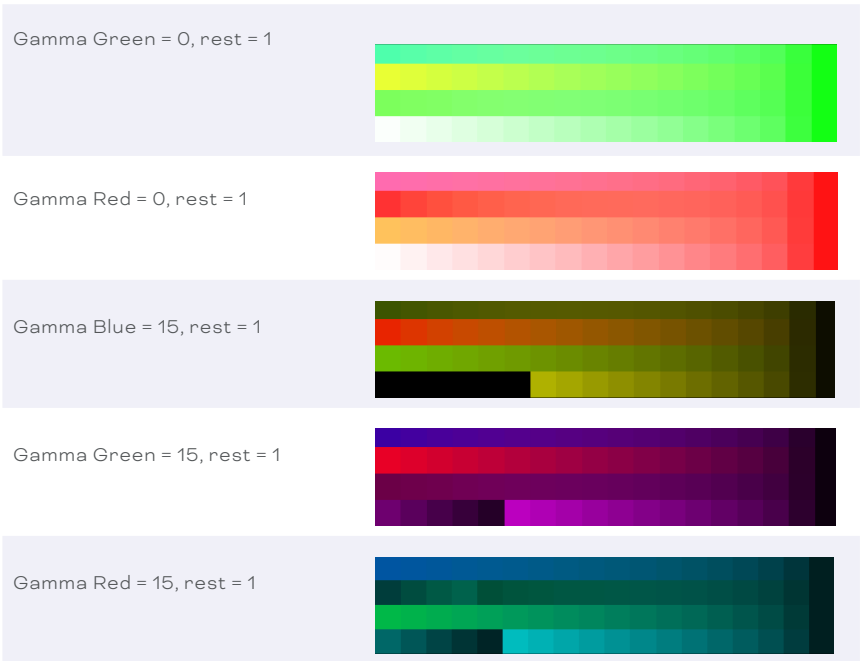
Original showing full blue, red, green and white without gamma correction (Gamma Blue, Gamma Green, and Gamma Red = 1)

Color Result



Gamma Blue = 0, rest = 1





10.19.2 Configuration of the White Channel

When using RGB Effects with RGBW fixtures, the White channel has to be calculated by SYMPHOLIGHT. This will be done automatically by adding the Effect to an “All Color X” Layer. The properties “White Channel”, “White Formula” and “White Offset” allow you to define the calculation.

Property “White Channel”: You have different calculation options for the White channel.

Mapping mode	Description
Average	The value of the White channel is the average between R, G and B. $W = (R + G + B) / 3$
Minimum	The value of the White channel is equivalent to the smallest value between R, G and B. $W = \min (R, G, B)$

Maximum	The value of the White channel is equivalent to the largest value between R, G and B. $W = \max(R, G, B)$
---------	--

Formula	The value of the White channel is calculated with the formula defined in the property "White Formula".
---------	--

Offset	The value of the White channel is set to the value of the "White Offset" property. $W = (\text{White Offset})$
--------	---

Off	The White channel is turned off. $W = 0$
-----	---

Color Conversion	Default. The value of the White channel is the smallest value between R, G and B, reduced to 1/3. And the values for R, G and B are each reduced to 1/3. $W = (\min(R, G, B)) / 3$ $R, G, B = (R, G, B) / 3$
------------------	---

Decolorize Conversion	The value of the White channel is the smallest value between R, G and B. And the values for R, G and B are each reduced by W. $W = \min(R, G, B)$ $R, G, B = (R, G, B) - W$
--------------------------	---

Property "White Formula": Enter the formula for calculating the White channel. Allowed variables are R, G and B. This property is only considered by SYMPHOLIGHT when the property "White Channel" is set to **Formula**.

Property "White Offset": Define an offset which will be added to the calculation result of "White Channel". The Offset value ranges from 0 to 100.

10.20 Sequence Properties

The behavior of a Sequence is mainly controlled with the Sequence properties. Also properties of Cues can be taken from the Sequence properties, or can be defined separately. Click into an empty area of the Sequence to set the properties of the Sequence or simply click the Sequence Properties icon (the cog wheel next to Delete Sequence) in the control bar.



Cue Skip Mode	Fade mode when using the skip functions. CrossFade: perform cross fade when entering next Cue. FadeFromLastFrame: use the last frame for fade.
Cues Fade In Time	The fade-in time for Cues, default = 1 second.
Cues Fade Mode	Fade mode in standard play. CrossFade: perform cross fade when entering next Cue. FadeFromLastFrame: use the last frame for fade.
Cues Fade Out Time	The fade-out time for Cues, default = 1 second.
Display Name	The name displayed as Sequence name.
Identifier	Unique identifier for this Sequence.
Loop Segment Begin	Define a loop segment begin position. Marked in the Cue track in the Timeline with a green marker.
Loop Segment Enable	Enable looping the defined loop segment.
Loop Segment End	Define the loop segment end. Marked in the Cue track in the Timeline with a red marker.
Loop Sequence	Loop the whole Sequence.
Mutual Exclude Collection ID	If not zero, a mutex group ID where only one Sequence may play at one time. If zero, no mutex grouping happens. Mutex is mutual exclusion, only one Sequence in a group may run at a time.
Priority	Render Priority: Sequences with higher priority overwrite values from Sequences with lower priority.
Start Position	Define the start position of the Sequence when triggered from the Automation or the Output Window.



10.21 Cue Properties

Display Name	The name displayed as Cue name.
Fade In Time	The fade-in time.
Fade Out Time	The fade-out time.
Identifier	Unique identifier for this Cue.
Is Skip Stop Position	This is a stop position when using the skip functions.
Length	Length of the Cue.
Play Mode	Behavior during play mode. Continue: continue playing. Wait: wait at the end of the cue. Repeat: repeat this Cue.
Regular Fade Out Mode	Fade mode during regular play mode. CrossFade: perform cross fade when entering next Cue. FadeFromLastFrame: use the last frame for fade.
Skip Fade Out Mode	Fade more when using skip functions. CrossFade: perform cross fade when entering next Cue. FadeFromLastFrame: use the last frame for fade.
Use Sequence Fade Times	Ignore fade times from Cue, use fade times from Sequence properties.

10.22 Special Function Keys

[SPACE]	Using the space tab, you can toggle between play mode and paused.
Ctrl + C	Copy elements in the Timeline to the clipboard.
Ctrl + V	Paste elements from the clipboard to the Timeline.



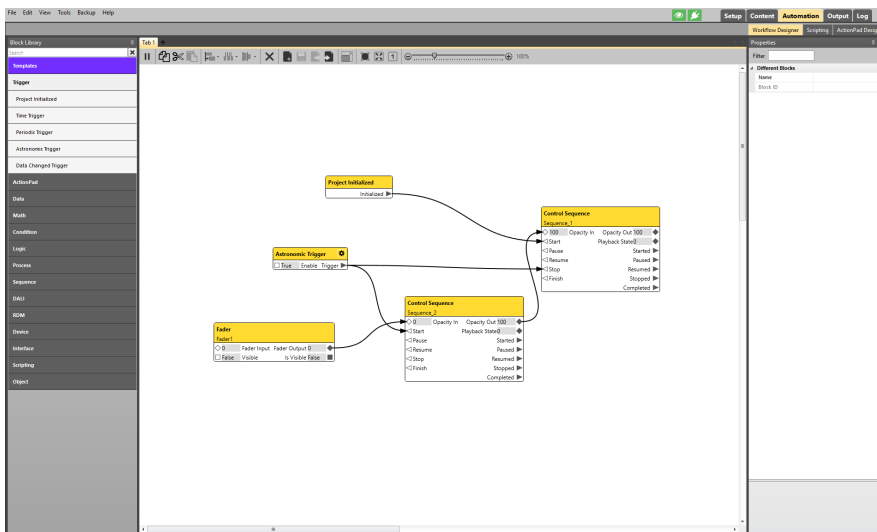


11 Automation

11.1 Principles

In the Automation Window, you will define a Workflow for control and user interfaces. With Automation you prepare the project how it will be executed live and define how to react to external signaling, time-controlled execution, and necessary calculations. The Automation tab contains the Workflow Designer, the Scripting window, and the ActionPad Designer. With the Workflow Designer you can automate SYMPHOLIGHT regarding control, events, external signaling and status changes. With Scripting you can add functions and features of SYMPHOLIGHT by programmed code in C#. The ActionPad Designer is used to create graphical user interfaces for mobile devices or external web browsers.

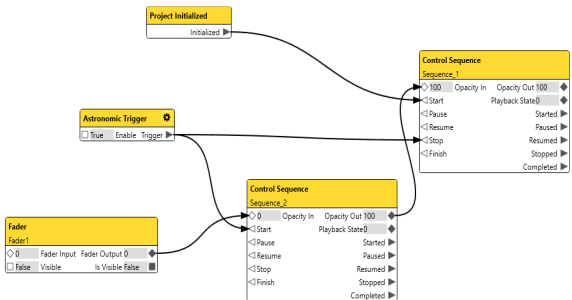
As one of the main tabs, the Automaton tab is located at the top right corner with the Workflow Designer, the Scripting window, and the ActionPad Designer beneath.



11.2 Workflow Designer

In the Automation window, you will define a Workflow for control and user interfaces. With Automation you prepare the project for how it will be executed live. Automation defines the reactions to external signalling, time-controlled execution, and necessary calculations.

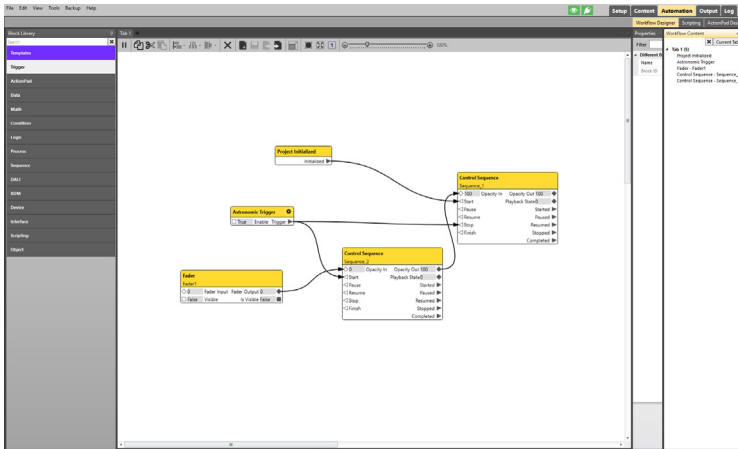
Here is a simple example of a Workflow for controlling two Sequences in SYMPHOLIGHT with an Astronomic Trigger, a Project Initialized trigger and one fader:



Sequence #1 gets started by a trigger that fires on project initialization. Another trigger is configured as "Astronomic Trigger" and stops Sequence #1 and starts Sequence #2". Both Sequences are controlled with a Fader on the input Opacity, which is Brightness here.



11.3 User Interface of the Workflow Designer



Block Library	Tab Control to structure the Workspace	Workflow Content	Overview of the Workspace
	Workspace with SYMPHOLIGHT Blocks		Properties



- The **Workspace** contains the Workflow. You arrange Blocks in the Workspace to create and edit the Workflow. Connect Blocks with a Link to create causalities. All causalities together form the Workflow. In the Workspace, you can move a selected Block. Multi-select is possible to move and edit more than one Block in the Workspace. Either draw a frame with the left mouse button pressed. Or if you keep the **Ctrl** key pressed while drawing a rectangle with the left mouse button in the Workspace, you select all included Blocks. In the Workspace, delete Blocks and Links.
- The **Block Library** contains all elements that can be placed in the Workspace, for example triggers, logical functions, ActionPad contents, or functions of devices like SYMPL dmx or dali Nodes. Blocks, that are already in use and can only have a single instance, like ActionPad buttons, Sequences or devices, have a check mark in front.
- In the **Tab Control** bar, you can create new tabs, create Templates, adapt the viewport with Fit or Fill, or you can zoom in and out manually. You can create as many Tabs and Templates as needed to have different areas for DALI, Sequences or functions. If not one Block is selected, Fit zooms to the best possible view port. If one or several Blocks

are selected, Fit zooms to the selected part of the Workspace. Fill always shows the complete Workspace.

- The [Overview of the Workflow](#) shows the current view port of the Workspace related to the complete tab. You can move the view port by dragging the yellow area. A right-click of the mouse opens a context menu where you can zoom in and out (Fit, Fill, 100%).
- The [Workflow Content](#) has all Blocks that are currently contained in the Workflow. The Blocks are sorted by the Tab (bold entry) to which they belong. Clicking an element here, or in the Workspace, makes the Block editable in its properties in the Properties Window. You can search in the Workflow Content to select only special Blocks in the list.
- The [Properties Window](#) provides the settings of selected Blocks for configuration.

11.4 Tab Control Bar



Suspend the currently running Workflow execution.



Continue the currently running Workflow execution.



Copy selected Blocks, including connected Links. (Ctrl + C)



Cut selected Blocks, including connected Links. (Ctrl + X)



Paste the copied or cut Blocks into the Workflow, including connected Links. (Ctrl + V)



Align selected Blocks. Choose from the drop-down menu the direction of alignment: left, right, top, bottom, horizontal center, vertical center.



Space selected Blocks evenly. Choose from the drop-down menu the axis of spacing: horizontal, vertical.





Distribute selected Blocks. Choose from the drop-down menu the reference point of distribution: left edge, right edge, top edge, bottom edge, horizontal center, vertical center.



Delete selected Blocks or selected Links.



Create a new Template.



Save a Template.



Export a Template.



Import a Template.



Fit the canvas to the content. Any excess area is clipped.



Optimize the viewport. The content of the Workspace is fit to the viewport.



Fill the entire Workspace to the viewport.



Scale the content of the Workspace to 100%.



Zoom in and out of the Workspace (min. 10%, max. 400%).



11.5 Creating a Workflow

1. To create a Workflow, drag and drop a Block from the Block Library into the Workspace. Blocks are the main objects of which a workflow is composed.
2. Configure the settings of the Block in the Properties Window.
3. Add Blocks to the Workspace as required.

4. Connect the Blocks with Links defining their causality, conditional statements and interdependencies. To create a connection, drag a Link: left-click on a PortOut interface, keep the mouse button pressed and move the mouse cursor to the target Port or Event. While doing this, a rectangle indicates if this connection is allowed or prohibited. You can connect Ports of the same Port types indicated by icons of the same shape. When connecting Ports of Blocks, the data type and the value condition must be compatible. In some cases links between different port types are possible. Then an automatic conversion will be done.

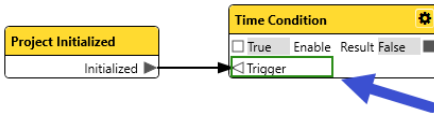
5. Add Links as required.

The Workflow is executable at every state without an explicit command.

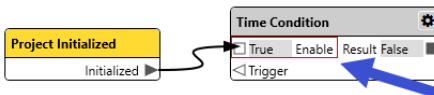
11.6 Connecting Blocks with Links

You connect the Blocks with Links defining their causality, conditional statements and interdependencies.

- To create a connection, left-click on a Port or Event interface, keep the mouse button pressed and move the mouse cursor to the target Port or Event. While doing this, a rectangle indicates if this connection is allowed or prohibited. You can connect Ports of the same Port types indicated by icons of the same shape. When connecting Ports of Blocks, the data type and the value condition must be compatible. In some cases links between different port types are possible. Then an automatic conversion will be done.



The green rectangle shows that the connection is allowed and valid.



The connection is prohibited, because the Port types are incompatible.

- You can change the color of a Link between ports. Click on the Link to see the properties of the Link in the Properties Window. Change the color of this Link in the Properties Window.
- To delete a Link, click on the Link to select it and press the Delete key.

11.7 Signal Flags

Signal Flags are Links that link different Block Ports in different Tabs. They can also be used instead of Links for a cleaner signal routing (better visual clarity) inside the same Tab. Signal Flags can connect one PortOut with more than one PortIn. You can assign Signal Flags and Links to the very same Port.

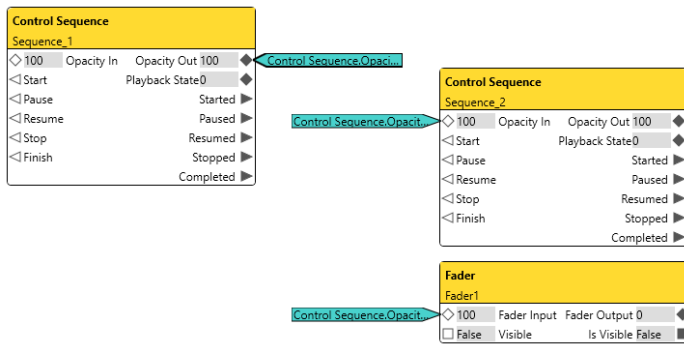
1. To create a Signal Flag, click on a PortOut. The PortOut is marked red.
2. Now press the Space bar on your keyboard. An text entry field opens with an auto-generated name. You can either change the name or press the Enter button and a Signal Flag gets created.

Press the Enter key instead of the Space bar and the text entry field is empty. Enter a name and press the Enter button. A Signal Flag gets created.

Signal Flag names are unique. You can only create Signal Flag with a new, unused name.

3. Assign the Signal Flag to a PortIn: click on the PortIn. The Block can be in a different Tab.
4. Press the Enter key or the Space bar and a list of valid, already existing Signal Flags will be shown.
5. Select a corresponding Signal Flag and press Enter.
6. Assign the Signal Flag to more than one PortIns if needed.


The connection between PortOut and PortIn is established.



- You can edit the color and the name of a Signal Flag. To set the properties of a Signal Flag, select the Signal Flag with a mouse click. The properties of the Signal Flag are shown in the Properties Window and the selected Signal Flag flashes.
- Jump to the location of the counterpart Block of a Signal Flag. Click on a Signal Flag and a list with the connected counterpart Blocks appears. Double-click onto an entry of the list to jump to the connected Block.
- To show the full name of an abbreviated Signal Flag name, move the mouse cursor over the Flag.
- Signal Flags with a red border have no counterpart Blocks, means a Signal Flag has not been assigned to a PortIn.



Signal Flags disappear after changing the data type of one connected Block (e.g. from string to bool).



You can assign Signal Flags and Links to an individual PortIn / PortOut. When deleting one connecting element, the other persists.

Control Sequence
Sequence_1

◇ 0 Opacity In Opacity Out 100

◁ Start Playback State 0

◁ Pause Started ▶

◁ Resume Paused ▶

◁ Stop Resumed ▶

◁ Finish Stopped ▶

Completed ▶

Fader
Fader1

◇ 100 Fader Input Fader Output 0

☐ False Visible ☐ Is Visible True

Control Sequence
Sequence_2

◇ 0 Opacity In Opacity Out 0

◁ Start Playback State 0

◁ Pause Started ▶

◁ Resume Paused ▶

◁ Stop Resumed ▶

◁ Finish Stopped ▶

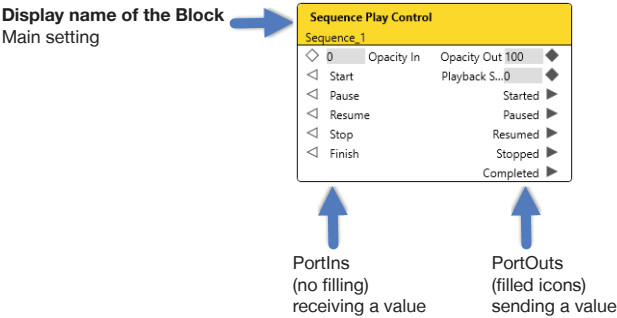
Completed ▶

Control Sequence Opacity

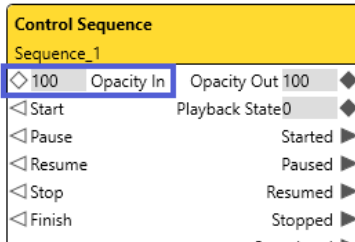
Control Sequence Opacity

11.8 Working with Blocks

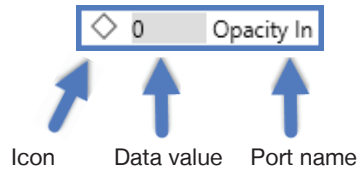
Typically, one Block offers one functionality as indicated by its name. The general structure of Blocks is the following:



For example, the following Sequence Play Control Block has six PortIns and seven PortOuts:



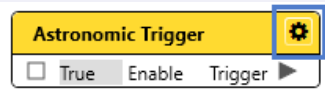
A Port interface:



Additional buttons in Block headers:



Opens a dialogue for setting parameters of a Block.

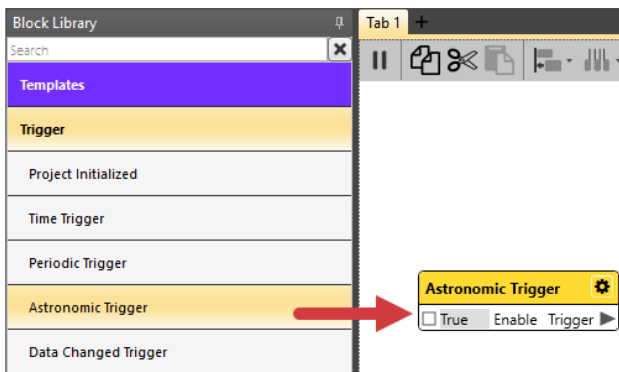


Opens a tab for editing the Template Block.




- To create a Block for the Workflow, drag and drop a Block from the Block Library into the Workspace.

Example of adding an Astronomic Trigger Block to the Workspace:



- Configure and adjust the settings of a Block for the Automation in your Workflow. See [„11.9 Configuring the Settings of Blocks“ on page 125](#).

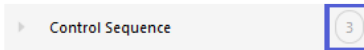
- Rename a Block by changing its Display Name in the Properties window or double-click on the Block header.
- To delete a Block, click on the Block to select it and press the Delete key. Or click the Cut button in the Tab Control bar: 

11.8.1 Target Based Blocks

Blocks that are target based can only have one bound instance per target - with the exception of DALI line Blocks (see „11.8.2 DALI Line Blocks“ on page 125). Targets are hardware devices and control elements in SYMPHOLIGHT like groups, Sequences, or ActionPad elements.

In the Block Library, a circled number at the right side indicates the amount of available instances for target based Blocks when the related target is added to the project (Setup Tab):

Example for three existing Sequences:



The Block Library also contains additional Blocks for each of the targets when the related target is added to the project (Setup Tab).

Example: A Glass Touch T12 is added to the project:



Add either the target based Block to the project or an unbound Block. Unbound, orphan Blocks can be added without restriction and pre-configured to some degree, but are not executed. Orphan Blocks are indicated by the grayed out header.

Example of a DALI Group block when not one DALI groups exist in the project:



Bind an unbound Block to a target by defining the Target value in the Properties Window of the Block. A device has to be in the project (Devices Window in the Setup Tab - "Attached devices in the project") to be available as Target value. When in Trial Mode, attached devices cannot be added to the project.

To undo the binding, delete the Target value in the Properties Window by clicking the cross button or selecting "<not set>" in the drop-down menu.

The name of Target values are always the Identifier of the targets.

Please note that the PortIn values of target based Blocks are set by default: actual settings within the objects/ targets are not retrieved, not displayed in the Block and might differ from the values set by default.





When you change the target device, all connections and flags to the block will be lost.

11.8.2 DALI Line Blocks

A special case of target based Blocks are DALI line Blocks (= all DALI Blocks that contain the word “line” in their name: [„11.15.55 Ballast Line“ on page 282](#), [„11.15.56 Generic Input Line“ on page 291](#), [„11.15.57 Button Line“ on page 296](#), [„11.15.59 Presence Sensor Line“ on page 306](#)). Instead of only one instance, DALI line Blocks can be bound to the same target multiple times. Also in contrast to all other target based Blocks, DALI line Blocks fully work without having virtual devices and therefore without configuration of the DALI devices. No matching required, but possible. You can directly address all devices or selected devices on a DALI line.

Use the DALI line Blocks instead of the device based blocks when you want to work with the events that are available in the Properties of the DALI device (Setup Window).



All DALI line Blocks for input devices do not work on lines with devices from the Osram fixture group.



It is recommended to avoid using DALI line Blocks together with other DALI blocks on the same line.

11.9 Configuring the Settings of Blocks

Configure the settings of a Block to adjust the Block for the Automation in your Workflow:

- Select the Block to access its settings in the Properties Window. To select a Block, left-click on the Block in the Workspace or in the Workflow Content.
- Adjust the settings of the Block in the Properties Window.

Multi-selection of Blocks is possible to set identical properties at one blow:

- To multi-select, press the **Ctrl** key and left-click on the Blocks. Or draw a frame with the left mouse button pressed. The Blocks included in the frame get selected. Both ways are available in the Workspace as well as in the Workflow Content.
- Adjust the identical settings of the selected Blocks in the Properties Window.

There are generic properties and Block specific properties. The generic properties are listed below. For a description of each Block and its properties see [„11.15 Workflow Blocks in Detail“ on page 145](#).



List of Generic Block Properties

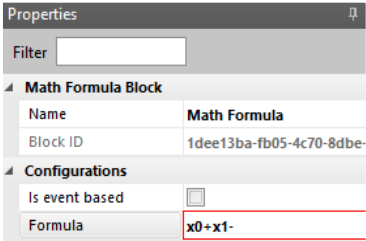
Display Name	Name of the Block. Displayed in the top row of the Block's header.
Block Position X	X-Position of the Block in the Workspace.
Block Position Y	Y-Position of the Block in the Workspace.

Accessibility of Properties

Properties can be either fixed or selectable. Fixed properties are grayed out, selectable properties are bold:

Fixed property	Grayed-out	Not changeable. This is a Port that has been given a constant value.	<div><div>Data Type</div><div>Double</div></div>
Selectable property	Bold	Use drop-down menu, counting arrows or type to change the value.	<div><div>Data Type</div><div>Double</div></div>

A red framed Block or Property entry indicates an invalid setting. The block cannot be executed. Please adjust the property settings.
Example: The formula of the Math Formula block (Advanced Math) is incomplete:



Please note: To apply defined values press the Enter key on the keyboard or select another property in the Properties Window.






11.9.1 Setting Port Types and Modes

Each Port has types and modes assigned. These are:

- Data types (see „11.9.2 Data Types for PortIn and PortOut Values“ on page 127)
- Input types (see „11.9.4 Setting Input Types for PortIn Values“ on page 128)
- Trigger (see „11.9.6 Adding an Event Based Trigger“ on page 130)











— Port Modes (see „11.9.7 Setting Port modes“ on page 131).

The assigned types and the Trigger are graphically indicated by the Port icons:





Data type	Shape of the icon.	E.g.  
Input type: Property	Bold framed icon.	E.g. 
Trigger	Triangle icon.	 

11.9.2 Data Types for PortIn and PortOut Values

The following data types are supported in SYMPHOLIGHT's Workflow Designer:

Data type	Port Icon (PortIn / PortOut)	Explanation: "example".
Bool	 	A bool value, equals True or False: "True".
Double	 	A 64 bit float value.
DateTime	 	A value representing date and time with high granularity.
Dynamic	 	Any data type is compatible.
HdrIntensity	 	(High-dynamic-range Intensity) Tuple of Intensity and Alpha values with 32 bit float precision. Used for Intensity Effects.



Int64		A 64 bit signed integer value.
LogLevel		Values: Warning, Debug, Message, Error. Used in the Write Logbook Message Block.
ProPhotoColor		Tuple of RGB and Alpha values with 32 bit float precision. Used for Color Effects.
String		A string of UTF-16 characters: "Text".



When connecting Ports of Blocks, the data types must be compatible. A rectangle indicates if a connection is allowed or prohibited.

11.9.3 Setting Data Types of Ports

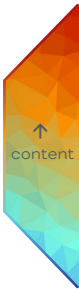
To configure the data types of Ports of a Block, select the Block. In the Properties Window, adjust the data types of the Ports. If data types of Ports are fixed, this is visible when the Properties in the Property Window are grayed-out.

11.9.4 Setting Input Types for PortIn Values

A special type of PortIn is the Property Port. A Property Port does not receive a value, but has a fixed value. As such, a PortOut cannot be connected to a Property Port with a Link. A Property Port is indicated with a bold frame.

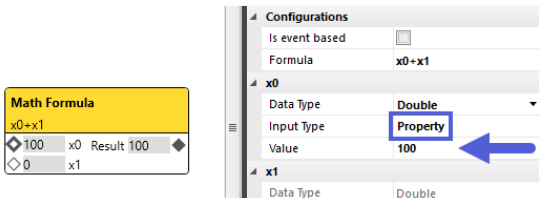
Input type: Port	Regular icon.	E.g. 
Input type: Property	Bold framed icon.	E.g. 

- For some PortIns in some Blocks, you can switch from Port to Property:
- Look for the property “Input Type” in the Properties Window.
 - At the right column, select from the drop-down-list or type “Property”.
Now, the Port is marked with a bold circle, the data type is adjustable, and the property “Port Mode” is replaced by “Value” for setting a fixed value.
 - Configure the data type and set the value as desired.



The PortIn has now a fixed value that is used for the Block's calculation.

Example:



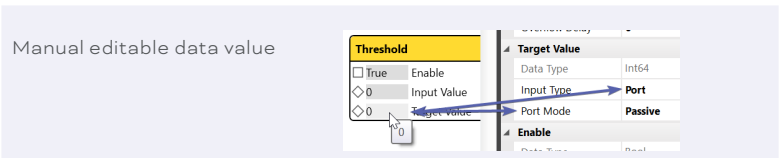
Here for the Math Formula Block, the value x0 is set to a constant of 100. The result will be $100 + x1$, like the Formula says.

A Port with a “Property” input type is accompanied by a Value property instead of a Port mode property.

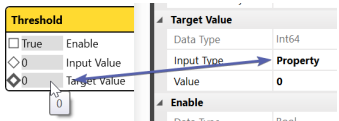
11.9.5 Manually setting PortIn data values

Data values of PortIns can be manually edited from within the Workflow Designer. This function is available when the Input Type of a PortIn is of type “Port”, respectively when the PortIn lists the “Port Mode” property.

To see whether a data value is manually accessible, hover over the data value of a PortIn: A change of the grey background color to a lighter grey indicates the availability.



Not editable data value

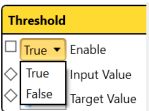


To edit the value, double-click on the data value. The value gets selected: Target Value

To submit the change or to re-send the value, click elsewhere or press “Enter”.

To cancel the edit, press “Esc”.

For a bool value, a drop-down list expands. Select an entry - the chosen value is submitted directly.





You can use this function to test a part of the workflow independently of preconditions.

11.9.6 Adding an Event Based Trigger

A special type of Port is the event based Port. Instead of a specific value, a Port can send or receive a trigger. An EventIn receives a trigger and not a value. An EventOut sends a trigger and not a value. Values are generally set before an event is fired.

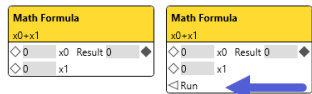


An EventIn input. Receives a trigger or an event from another operation.



An EventOut output. Sends a trigger or the event from an operation.

If Is event based is checked, the Block gets an additional EventIn and creates the result for the output only when the block gets a trigger on the “Run” input. Example: Math Block without (left) and with (right) checked Is event based.

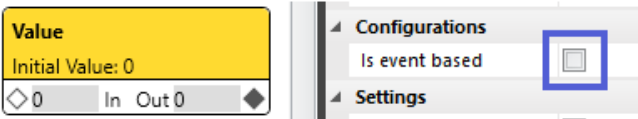


For some Blocks, you can add an event based Port to

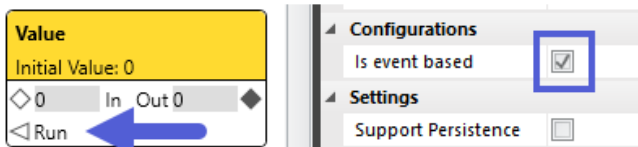
trigger the execution of the Block:

- At the property “Is event based”, check the box for “Value” to enable triggering. An event based PortIn (EventIn) is added to the Block.

Example of Value Block without the Event based Trigger:



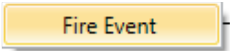
Value Block with the enabled “Is event based” property and the additional EventIn:



When connecting Ports of Blocks, the value condition must be compatible. Setting a PortIn to event based switches the Port mode from passive to active. Is a Block event based, the Block only executes on an external trigger event.



- You can manually trigger EventOut Ports. Right-click on an EventOut and click the “Fire Event” context menu entry.



The defined Trigger is sent instantly to the connected EventIn.

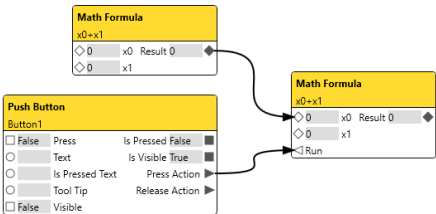


You can use this method to test a part of the workflow independently of preconditions.

11.9.7 Setting Port modes

A Port can be either active or passive. In general, “active” means that the Port refreshes its data values constantly, or when input values change, regardless of what the receiving Port expects whereas “passive” means that the data only gets updated if explicitly ordered to do so. A Port is passive, when it requires a request from its counterpart and then creates the result.

Example:



The upper Math Block has a Passive PortOut, the lower Math Block has an Active PortIn. The lower Block is also set to Is Event Based. If the button on the ActionPad is pressed, the lower Math Block will now request the result from the upper Math Block, calculate its own result and send it to its PortOut. Set the sending Port as active and the receiving Port as passive. This mechanism is helpful when you want to avoid permanent load from the Blocks.

Important is the combination of linked Port modes and to consider the executing block:

PortOut of block A - block A is executing	PortIn of block B	Description
active	active	Data value is forwarded, block B is not triggered.
active	passive	Data value is forwarded, block B gets triggered to execute.

passive	active	Data value stays in the PortOut of block A until PortIn of block B fetches the value.
passive	passive	Do not use: No data flow.
PortOut of block A	PortIn of block B - block B is executing	Description
active	active	No action since data value is up-to-date.
active	passive	Block B executes with existing data value.
passive	active	Block B triggers block A to run again and forward the new data value.
passive	passive	Do not use: No data flow.

- To change the Port mode, go to the property “Port Mode”, select from the drop-down-list or type “Active” respectively “Passive”.
- Set a Port mode to Passive to avoid permanent evaluation of blocks.

By default, PortIns are passive and PortOuts are active. Setting a PortIn to event based, switches the Port mode from passive to active.

Take care to only connect a single Link to an active PortIn. Otherwise you cannot distinguish with which incoming value the block calculates.

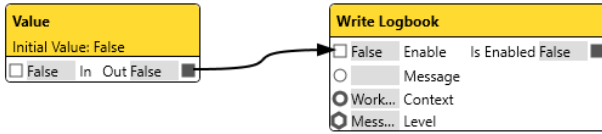
If Port modes are fixed, this is visible when the Properties in the Property Window are grayed-out.

11.9.8 Enabling and Disabling Blocks

Many Blocks have a PortIn named Enable and a PortOut named Is Enabled. By default, these Ports are set to TRUE: Any Block is enabled and executed when placed in the Workflow. Sending the bool value “FALSE” from a linked counterpart Block to the Enable PortIn of a Block disables the Block. The Is Enabled PortOut will forward the value “FALSE” to a connected block.

Example: A Write Logbook block creates entries in the Log as its initial Enable value is “TRUE”. A Value Block can prohibit the Write Logbook block to create entries in the Log. Set the data type of the Value Block to Bool. The default initial value of the Block becomes FALSE. Connecting the Out PortOut of the Value Block to the Enable PortIn of the Write Logbook block disables the Write Logbook Message. No Log entry is created any longer.

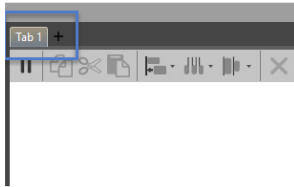




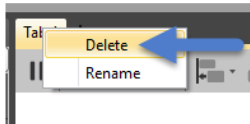
11.10 Adding Tabs, Deleting, and Renaming

A Tab contains one Workspace. You can create as many Tabs as needed to have different areas for DALI, Sequences or functions. The least number of Tabs is one. All Tabs together build the Workflow. You can connect Blocks of different Tabs using Signal Flags (see „11.7 Signal Flags“ on page 120).

- To create a new Tab, click the Plus button above the Tab Control bar: . By default, on opening the Workflow Designer one Tab is already present:



- To delete a Tab, right-click on the Tab. In the context menu click “Delete”:



To rename a Tab, right-click on the Tab. In the context menu click “Rename”. Enter the new Tab name into the entry field and click the “Ok” button.

11.11 Adjusting the View

You can adjust the view of the Workspace to your needs.

- Zoom in and out using the Tab Control bar:



Click to fit the canvas to the content. Any excess area is clipped.



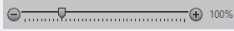
Click to optimize the viewport. The content of the Workspace is fit to the viewport.



Click to fill the entire Workspace to the viewport.



Click to scale the content of the Workspace to 100%.



Zoom in and out of the Workspace using the fader (min. 10%, max. 400%).

- Move the viewport of the Workspace using the Overview of the Workflow: drag the yellow rectangle of the viewport pressing the left mouse key to the desired area and release the left mouse key.

11.12 Adjust Arrangement

You can rearrange the Blocks of the Workspace at any time for a better visual clarity.

- Align and space evenly and distribute selected Blocks using the Tab Control bar:



Click to align selected Blocks. Choose from the drop-down menu the direction of alignment: left, right, top, bottom, horizontal center, vertical center.



Click to space selected Blocks evenly. Choose from the drop-down menu the axis of spacing: horizontal, vertical.



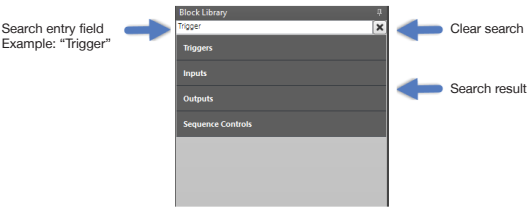
Click to distribute selected Blocks. Choose from the drop-down menu the reference point of distribution: left edge, right edge, top edge, bottom edge, horizontal center, vertical center.



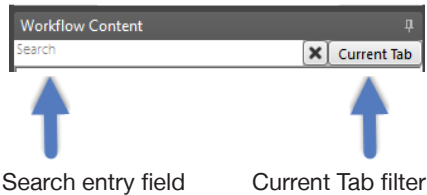
11.13 Search for Blocks

You can search the Workflow Designer for Blocks. Use the Block Library to search the whole range of provided Block items. Use the Workflow Content to search for Blocks utilized in the Workflow, independently of the Tab.

- Using the Block Library for Block items, enter a search text into the entry field at the top of the Block Library window. You can search for Blocks and for Block categories. The Block Library displays the Block categories of the matches. Clear the search text by clicking the button at the right of the entry field:



- Using the Workflow Content for utilized Blocks, enter a search text into the entry field at the top of the Workflow Content. The Workflow Content displays the matching Blocks. Clear the search text by clicking the button at the right of the entry field: ✕
- Filter the displayed Blocks of the Workflow Content to display only the Blocks of the currently shown Tab in the Workspace. Click the "Current Tab" button at the right of the Search entry field.



11.14 Block Library: Workflow Blocks

The Block Library is separated into various function areas.

A quick overview is presented in this chapter. For more information on each Block see the following chapter „11.15 Workflow Blocks in Detail“.



When editing properties of Blocks, details and more information for each property are shown in the help window below the Property Window..

Templates

User Defined Templates	See „11.15.1 Adding Templates“ on page 145 for creating and using Template blocks.
------------------------	--

Input	Use this Block if you would like add input ports to your template. Available in Template Tab only.
-------	---



Output	Use this Block if you would like add output ports to your template. Available in Template Tab only.
--------	---

Trigger

Project Initialized (p.147)	Use this Block to execute actions as soon as the project file has been loaded, e.g. Use this trigger to start a sequence if the system is started.
--	--

Time Trigger (p.148)	Use this Block to react on time events, like every working day at 7am.
---	--

Periodic Trigger (p.150)	Use this Block to trigger actions periodically, e.g. to execute a script periodically.
---	--

Astronomic Trigger (p.151)	Use this Block to react on astronomic events, like Dawn, Dusk, Sunrise and Sunset.
---	--

Data Changed Trigger (p.155)	Use this Block to react on any changes that are related to a dedicated port, e.g. if a Sequence Opacity changes.
---	--

ActionPad

Push Button (p.156)	Use this Block to use button press and release events to trigger elements in the Workflow Designer, e.g. if you want to start a Sequence from an ActionPad Button.
--	--

Toggle Button (p.160)	Use this Block to use toggle on and off events to trigger elements in the Workflow Designer, e.g. if you want to start and stop a Sequence from Toggle Button.
--	--

Image (p.164)	Use this block to set the image item.
---------------------------------	---------------------------------------

Password Input (p.166)	Use this block to define the password for an ActionPad's Password Input element in order to create a password protection. React on password entries at the ActionPad in the Workflow Designer, e.g. if you want to hide ActionPad content when the ActionPad user is not verified.
---	--



Text Input (p.169) Use this Block if you want to use the text of an ActionPad's Text Input in the Workflow Designer, e.g. if you want to write a log entry from within the ActionPad or set a Fader value by entering a value.

Text Label (p.171) Use this Block if you want to change the text of ActionPad labels from the Workflow Designer, e.g. if you want to display different messages on the ActionPad depending on the daytime.

Fader (p.173) Use this Block if you want to set or receive values of ActionPad Faders in the Workflow Designer, e.g. if you want to control the Opacity of a Sequence.

Inline Frame (p.176) Use this block if you want to change the URL for the Frame content.

Color Box (p.178) Use this block to set color.

Color Wheel (p.178) Use this block to set color.

Page Navigation (p.181) Use this Block if you would like to switch to a certain ActionPad Page on all devices that are actually showing the corresponding ActionPad, e.g. if you want to show a specific page in case an emergency comes up.

Data

Value (p.183) Use this Block to add a constant value to your workflow, e.g. you want to trigger a scene of a DALI Group.

State to Event (p.186) Use this Block to react on bool state changes, e.g. if a value changes from TRUE to FALSE.

Event to Value (p.187) Use this Block if you want to output statically defined values on an event.

Combine Text (p.189) Use this Block to combine two or more texts.



Value to Event / State (p.191)	Use this Block to compare Strings and Values that have been received from e.g. external sources and trigger actions based on the input received.
--------------------------------	--

Converter (p.194)	Use this Block to convert data types. Since v3.0 this block is obsolete as the Workflow Designer Blocks are converting data types automatically.
-------------------	---

Permit Event (p.196)	Use this Block to decide whether to pass an incoming event or to block it.
----------------------	--

Math

Math Formula (p.197)	Use this Block for advanced mathematical functionalities, e.g. combine arithmetic calculations.
----------------------	---

Addition (p.200)	Use this Block to add two or more values.
------------------	---

Subtraction (p.202)	Use this Block to subtract two or more values.
---------------------	--

Multiplication (p.204)	Use this Block to multiply two or more values.
------------------------	--

Division (p.206)	Use this Block to divide two values.
------------------	--------------------------------------

Moving Average (p.208)	Use this Block if you want to create a moving average value based on a number of received values or based on values that have been received within a certain time.
------------------------	--

Condition

Time Condition (p.210)	Use this Block to react on time events. Use this Block if you want to limit the execution of other triggers to a specific time range, e.g. only allow to switch on the lights if it is after 7am.
------------------------	---

Compare Condition (p.212)	Use this Block to compare to values with each other.
---------------------------	--



State Condition (p.214)	Use this Block if you would like to switch between two different scenarios depending on one single value, e.g. change the text of a button if another button has been pressed.
----------------------------	--

Logic

AND (p.216)	Use this Block if you want to combine two or more conditions: only if all inputs are TRUE, the output will be TRUE.
-------------	---

OR (p.218)	Use this Block if you want to combine two or more conditions: if at least one input is TRUE, the output will be TRUE.
------------	---

NOT (p.220)	Use this Block to negate a boolean value: if the input is TRUE, the output will be FALSE and vice versa.
-------------	--

Process

Threshold (p.221)	Use this Block if you want to turn off the light if the light sensor value is above a threshold level and to turn on the light if the sensor value is below.
-------------------	--

Fade (p.223)	Use this block to fade between two values.
--------------	--

Measure Compare Adjust (PID) (p.227)	Use this Block if you want to adjust the fixture light level depending on a certain sensor target value, e.g. in Daylight Harvesting scenarios.
--	---

Human Centric Lighting (p.231)	Use this Block to create human centric lighting features like defining a color temperature curve depending on the time of day.
-----------------------------------	--

Toggle (p.236)	Use this Block to achieve a toggle functionality, e.g. first button press: light on, second button press: light off.
----------------	--

Single Delay (p.237)	Use this Block to react on an event a certain time after it has been triggered.
-------------------------	---

Multi Delay (p.238)	Use this Block to react on an event a certain time after it has been triggered. Multiple delays can be specified.
---------------------	---



Sequence

Control Sequence
(p.243)

Use this Block to control the opacity and play state of a sequence or to react on play state changes, e.g. to start, stop or pause a Sequence.

Adjust Sequence
(p.245)

Use this Block to control some advanced parameters of a Sequence, e.g. the priority value of a Sequence.

Control Cue
(p.247)

Use this Block to control Sequences based on cues, e.g. skip to the next or previous cue or react to completed cues of a Sequence.

Trigger Point
(p.249)

Use this Block to react on Trigger Points that have been assigned to a specific Sequence. Whenever the Sequence play time is passing a trigger point an event will be fired and the number of crossings is available.

Generic Control Sequence (p.250)

Use this Block to control multiple Sequences at once and get notifications of them at one place.

Mutex Group
(p.252)

Use this Block to control Mutex Groups, e.g. to jump to the next Sequence within the same Mutex Group.

DALI

Ballast Group
(p.255)

Use this Block to control your DALI Groups. You can also configure Daylight Harvesting.

Button (p.269)

Use this Block to include DALI buttons in the Workflow Designer, e.g. to switch on/off lights based on button presses.

Absolute Input
(p.273)

Use this Block to include DALI Absolute Input in the Workflow Designer, e.g. to get values from absolute instances.

Generic Input
(p.275)

Use this Block to include DALI Generic Input in the Workflow Designer, e.g. to get values from generic instances.



Presence Sensor (p.278)	Use this Block to include DALI presence detection sensors in the Workflow Designer, e.g. to switch on/off lights based on presence information.
----------------------------	---

Light Sensor (p.280)	Use this Block to process DALI light sensor values in the Workflow Designer, e.g. for Daylight Harvesting.
-------------------------	--

Ballast Line (p.282)	Use this block to control DALI ballasts of a line in the direct way by ballast address, group address, and Broadcast. Requires no ballast matching in the Setup Tab.
----------------------	--

Generic Input Line (p.291)	Use this Block to control DALI inputs of a line in the direct way by input address, instance number, and instance type.
-------------------------------	---

Button Line (p.296)	Use this Block to control DALI push button inputs of a line in the direct way by input address and instance number.
---------------------	---

Input Automation Line (p.300)	Use this Block to control a DALI Group Block by inputs like buttons or sensors of a line in the direct way by input address / instance number and execute DALI group actions on incoming events.
----------------------------------	--

Presence Sensor Line (p.306)	Use this Block to control DALI occupancy inputs of a line in the direct way by input address and instance number.
---------------------------------	---

Feedback Channel Line (p.310)	Use this Block to control DALI feedback channels of a line in the direct way by input address and instance number. The Block sends DALI commands to the associated line, device and instance number(s).
----------------------------------	---

DALI Observer (p.313)	Use this Block to process DALI information that can be collected from DALI fixtures.
--------------------------	--

RDM

Build RDM Command (p.317)	Use this Block to build simple RDM commands that can be send to RDM devices using the Send RDM Block or scripting.
------------------------------	--

Send RDM (p.319)	Use this Block to send RDM commands to specified destination remote devices.
------------------	--



Observe RDM (p.321)	Use this Block to process RDM information that can be collected from RDM fixtures, e.g. temperatures.
--	---

Device

Analog / Digital Contact (p.326)	Use this Block to react on the values of analog inputs of e.g. the SYMPL input Node and process the values in the Workflow Designer.
--	--

Dry Contact (p.330)	Use this Block to receive dry contact triggers from, e.g. the LCE3 or the Butler XT2.
--	---

Relay (p.333)	Use this Block to control the relay outputs of e:cue devices, e.g. the SYMPL relay Node.
---------------------------------	--

Modbus (p.335)	Use this block to communicate with external mapped Modbus devices.
----------------------------------	--

Observe Controller (p.337)	Use this Block to control and fetch information from e:net devices.
---	---

Observe System (p.339)	Use this Block to retrieve various information about the running instance and the project.
---	--

Receive DMX (p.340)	Use this Block to receive DMX values from an external system, like a SYMPL dmx Node or a SYMPL e:pix Node, and process them in the Workflow Designer, e.g. to map a DMX channel value to a Sequence Opacity.
--	--

LCE2 Display (p.342)	Use this Block to show a specific message on the display of the e:cue Lighting Control Engine 2 device or the LCE3.
---	---

Glass Touch T6 (p.344)	Use this Block to control your installation with a Glass Touch T6 User Terminal.
---	--

Glass Touch T6R (p.348)	Use this Block to control your installation with a Glass Touch T6R User Terminal.
--	---

Glass Touch T12 (p.355)	Use this Block to control your installation with a Glass Touch T12 User Terminal.
--	---



Glass Touch IR (p.356)	Use this Block to process IR commands from a remote control that are received by any Glass Touch User Terminal. Make sure the IR Remote supports RC5.
---	---

Sympl Fusion Action Buttons (p.360)	Use this Block to trigger automation from the Action buttons of a SYMPL Fusion device.
---	--

Sympl Fusion Dry Contacts (p.363)	Use this Block to trigger automation from the digital inputs of a SYMPL Fusion device.
---	--

Interface

Send E-mail (p.366)	Use this Block if you want to send emails with customizable messages.
--	---

Send MQTT (p.369)	Use this Block to send MQTT data, e.g. when using a SYMPHOLIGHT Gateway BACnet.
--	---

Receive MQTT (p.371)	Use this Block to receive MQTT data, e.g. when using a SYMPHOLIGHT Gateway BACnet.
---	--

Send UDP (p.373)	Use this Block to send UDP network triggers to a given IP address of external systems, e.g. with a Constant Value Block.
------------------------------------	--

Receive UDP (p.376)	Use this Block to receive external UDP network triggers that can be processed in other blocks, e.g. the Value to Event / State Block.
--	---

Send & Receive Serial (p.379)	Use this Block to communicate with external systems using serial ports, like the SYMPL serial Node or a Local Serial Port.
---	--

Write Logbook (p.384)	Use this Block to write a Logbook Message to the Log Tab.
--	---

Send to Network (p.386)	Use this Block to send data from configured ports to the network.
--	---

Receive from Network (p.388)	Use this Block to receive data from the network and dispatch it to configured ports.
--	--



Scripting

Script (p.390)

Use this Block to add Scripts to the Workflow Designer, e.g. if you want to start Scripts based on ActionPad button presses.

Object

Set Property (p.391)

Use this Block if you want to change the value of an object property. An object can for example be an opacity value of an effect or the master fader.

Example: assign a fader on an ActionPad page to the Grand Master Fader (Target Object: ProjectInformation, Target Property: Opacity).

Get Property (p.392)

Use this Block if you want to get the value of an object. The Block gets a property from an object and provides it as value. An object can for example be an opacity value of an effect or the master fader.

Example: fetch the color selection from a Color Picker on an ActionPad page and provide the color value as ProPhotoColor value.

Check Property (p.396)

Use this Block if you want to compare the value of an object property. The Block compares a property of an object against a value. It provides a bool value as result from the comparison. An object can for example be an opacity value of an effect or the master fader.

Call Method (p.398)

Use this Block to call a function of an object.

Example: Play/Pause/Stop of a Sequence, or Cue Control for the Sequence.




11.15 Workflow Blocks in Detail

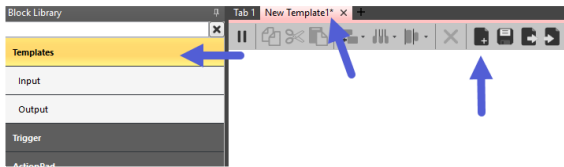
11.15.1 Adding Templates

A special type of Blocks are Templates. With Templates you can design your own customized Block. You can also use Templates to create a complex Workflow logic once (Master Template) and then multiply it effortlessly (Template instances). A template Block consists, as a usual Block, of PortIns, EventIns, PortOuts, and EventOuts. Additionally, a Templates has its own internal logic that is created by an own Workflow, consisting of interconnected Blocks from the Block Library.

Create a Template (Master Template)


Create a Template in the Workflow Designer:

1. Click the  button in the toolbar of the Workflow Designer to create a Template. A new tab with an empty Workspace opens. Newly created Templates are always Master Templates. You can rename the Template over the context menu.



2. Go to the Templates category in the Block Library. You can add Inputs and Outputs in the shape of Blocks to the Template Workflow as desired which will act as Ports in the Template instances. Configure the Input and Output blocks as you need them as Ports.
3. Add Blocks from the Block Library as desired and link them to design the internal logic of the Template.

Please note that target based Blocks are always orphan in Master Templates. Target values can only be assigned in Template instances, using either the Properties window of the selected Template instance or the editing tab.

4. Click the  button in the toolbar of the Workflow Designer to save the Template. When switching from the Template tab to an ordinary tab, the Block Library now contains the newly created Template (category "Templates").

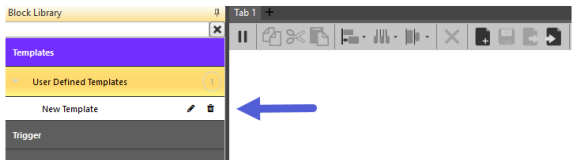
You can also import () and export () Templates. The file format for Templates is *.slwt (SYMPHOLIGHT Workflow Template).



Use a Template (Template instance)

To use a Template in the Workflow after creating the Master Template, an Template instance is automatically created:

1. Go to the Templates category in the Block Library of the Workflow Designer.

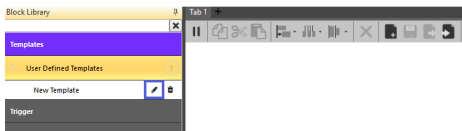



2. Drag and drop a Template into the Workspace like any other Block. The Block created in the Workflow is an instance of the Template. Assign target values to target based Blocks where present.
3. Integrate the Template into your Workflow by defining its properties and linking it with counterpart Blocks.

Edit Templates (Master or instances)

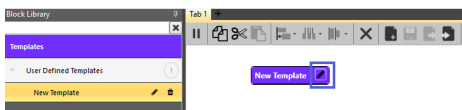
You can edit either a Master Template or a Template instance itself. Changes on an instance only apply to this very Template block as it is deployed in the Workflow. Changes on a Master Template apply to all henceforth deployed Template instances of this Master Template - but not retroactive.

- Master Template editing: access over the  icon in the Block Library.



The Template content of the Master is opened in another tab of reddish color, with the Input and Output blocks again available in the Block Library. You need to save the changes by clicking the  button in the toolbar.

- Template instance editing: access over the  icon in the Template block in the Workspace.



The Template content of the instance is opened in another tab of purple color, with the Input and Output blocks again available in the Block Library. The changes are applied instantly without saving them.

11.15.2 Project Initialized



Properties	
Filter	<input type="text"/>
Project Initialized Block	
Name	Project Initialized
Block ID	d4b4533d-f860-44c0-a75c-!

Use this Block to execute actions as soon as the project file has been loaded, e.g. use this trigger to start a sequence when the system is started.

The Block has one EventOut for the trigger when the project is initialized.

- No configuration needed.



11.15.3 Time Trigger

Time Trigger

No Next Trigger

☐ True Enable Next Event

Trigger

Time Trigger Block Configuration

Specific date

16/03/2021

OK

Daily

Repeat every 1 day

Weekly

Repeat every 1 week

☐ Monday ☐ Tuesday ☐ Wednesday ☐ Thursday ☐ Friday ☐ Saturday ☐ Sunday

Monthly

Days

Month

Every

1

Month

Time

Start at: 0000

Latest start time: 235959

Start at defines the time the trigger has to be executed. If Sympholight is started or the project is loaded after 'Start at' time, latest start time will be evaluated. The trigger will happen when system resumed before the 'Latest start time'. If 'Latest start time' is less than 'Start at' the 'Latest start time' is considered to be next date.

OK

Cancel

Properties

Filter

Time Trigger Block

Name

Time Trigger

Block ID

04de6430-8523-4c71-a97d

Enable

Data Type

Bool

Port Mode

Passive

Next Event

Data Type

Dynamic

Port Mode

Active

Use this Block to react on time events, like every working day at 7am.


The Block has two PortOuts for the trigger and an Enable PortIn. Extended configuration is accessible at the cogwheel icon. The trigger configuration is displayed as main setting in the Block's header.

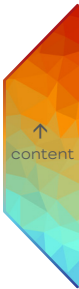
- Define date and time of the time triggers in the extended configuration window.

Category	Description
Enable	A passively received bool value to dis- or enable the Block.
Next Event	An actively sent dynamic value of the next trigger event as defined in the extended configuration window.

Defining date and time triggers in the extended configuration window

Define at what date and at what time the trigger is scheduled in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .
2. Select one of the four date options at the Date frame:



Specific date: Use the calendar at the icon or enter manually a particular day for a trigger.

Daily: Enter a number for the daily interval in which you want to send a trigger, e.g. enter "2" for triggering every other day. Compared to the "0" with immediate effect, when entering a "1", the day count starts with the day after configuring the Time Trigger block.

Weekly: Enter a number for the weekly interval in which you want to send a trigger, e.g. enter "1" for triggering every week. Compared to the "0" with immediate effect, when entering a "1", the week count starts with the Monday of the next week at the time of configuring the Time Trigger block. Further select the weekdays.

Monthly: Select one or multiple months. Also enter specific days or weekdays. The Trigger configuration is of immediate effect. You will have to set at least the days in the month, otherwise no event will be fired.

3. Define a time span at the Time frame. SYMPHOLIGHT scans in the given time span if the configured trigger has been sent, e.g. at a belated project start. If not, the trigger will be sent immediately.

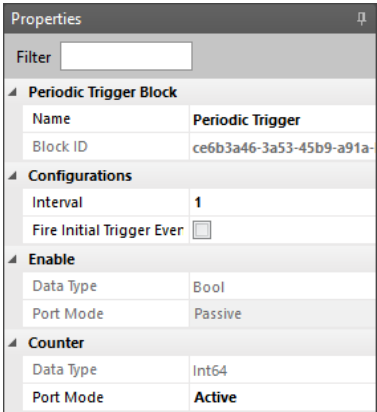
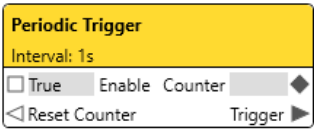
Latest start time: The latest time that the trigger will fire if the Enable EventIn changes in this time span from False to True.

4. Click the "OK" button to save the configuration and close the window.

The Time Trigger block has now the trigger configuration displayed as main setting in the Block's header. It displays the next date and time of sending a trigger.



11.15.4 Periodic Trigger



Use this Block to trigger actions periodically, e.g. to execute a script at a regularly occurring interval.

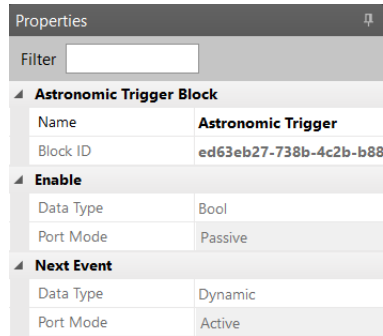
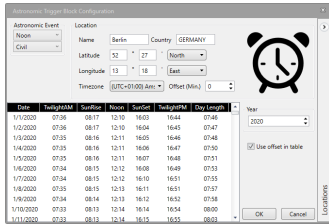
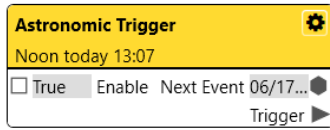
The Block has one EventOut for the trigger, one PortOut for the trigger counter, one EventIn for resetting the counter and an Enable PortIn. The trigger configuration is displayed as main setting in the Block's header.

- Define the interval of the triggering.

Category	Description
Interval	Enter the trigger interval in seconds.
Fire Initial Trigger Event	Enable to fire an initial start trigger without waiting for the first elapsed time period.
Enable	A passively received bool value to dis- or enable the Block.
Counter	An actively sent Int64 value of the accumulated number of sent triggers in total. You can cchange the Port Mode.



11.15.5 Astronomic Trigger



Use this Block to react on astronomic events, like Dawn, Dusk, Sunrise and Sunset.

The Block has two PortOuts for the trigger and an Enable PortIn. Extended configuration is accessible at the cogwheel icon. The trigger configuration is displayed as main setting in the Block's header.

- Define the astronomic event and the reference location in the extended configuration window.

Category Description


Enable A passively received bool value to dis- or enable the Block.

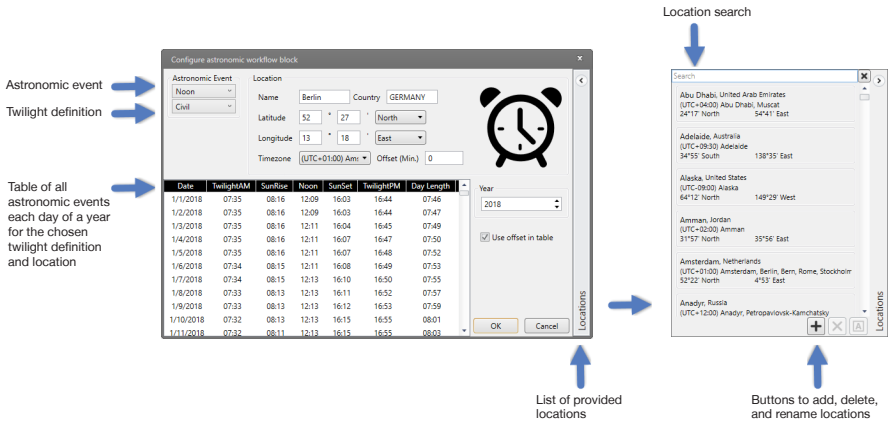
Next Event An actively sent dynamic value of the next trigger event as defined in the extended configuration window.



Defining the astronomic event and the reference location in the extended configuration window

Define the astronomic event and the reference location in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



2. Select an astronomic event at the top drop-down menu on the left side. Available events are: TwilightAM, TwilightPM, Sunrise, Sunset, and Noon.
3. Select a twilight definition at the bottom drop-down menu on the left side to define the referenced time for all astronomic event. Available twilight definitions are: Civil, Nautical, and Astronomic.
Civil: Civil twilight is defined when the sun is 6 degrees below the horizon. Illumination is sufficient, under good weather conditions, for terrestrial objects to be clearly distinguished.
Nautical: Nautical twilight is defined when the sun is 12 degrees below the horizon. General outlines of ground objects may be distinguishable, but detailed outdoor operations are not possible, and the horizon is indistinct.
Astronomic: Astronomical twilight is defined when the sun is 18 degrees below the horizon. For a considerable interval after the beginning of morning twilight and before the end of evening twilight, sky illumination is so faint that it is practically imperceptible.
4. Change the reference location for the event either by clicking a provided location from the list on the right side.
5. Or manually enter a location in the middle section. Click the **+** button to add the location to the provided list. You can delete the entry (**X**) or rename it (**A**). If you have offset, choose whether to use the offset in the table by checking the box "Use offset in table".
6. Click the "OK" button to save the configuration and close the window.

The Astronomic Trigger block has now the trigger configuration displayed as main setting in the Block's header. It displays the event's next date and time of sending a trigger.

- Search for provided locations using the search entry field at the top of the locations list.

Considered Settings

The computation of the astronomic trigger depends on three settings:

1. The time zone specified in the web interface of SYMPHOLIGHT (local time of SYMPHOLIGHT)
2. The location specified in the dialog (with latitude and longitude)
3. The time zone specified in the dialog

The meaning of these settings regarding the computation of the astronomic trigger is the following:

1. Any version of SYMPHOLIGHT, no matter if SYMPL Fusion or SYMPL Core S(p) or Windows application, allows to set a time zone via the web interface. This time zone is used for the calculation of the local time inside of SYMPHOLIGHT. The default time zone setting for the SYMPL Fusion or SYMPL Core S(p) is UTC (Universal Time Coordinated). The default time zone for SYMPHOLIGHT on Windows is the time zone configured in the underlying Windows OS.
2. The location where the times of astronomic events will be calculated for. This is in fact a UTC time.
3. The time zone setting inside the dialog defines how the UTC time calculated based on the location is visualized. This means, the UTC time of an astronomic event is converted to the time equivalent in the specified time zone.

- When does SYMPHOLIGHT raise the astronomic trigger?

SYMPHOLIGHT evaluates the time you can see in the table generated when configuring the astronomic trigger and compares this value with the local time of SYMPHOLIGHT. The local time of SYMPHOLIGHT depends on the time zone setting made in the web interface or the default time zone value as described in [1](#). If this local time equals the time in the dialog, the trigger will be fired.

For productive use, the local time zone of SYMPHOLIGHT should match the time zone specified in the dialog.

- Examples

Local astronomic event

Imagine having an installation in Berlin demonstrating the day-night cycle of Berlin. We want our SYMPHOLIGHT located in Berlin to fire when it is sunset in Berlin. We are having a SYMPHOLIGHT configured with the time zone of "(UTC+1) Berlin, Amsterdam,



Wien” via web interface. We have specified the location Berlin inside of the dialog with the correct coordinates. In the dialog we specified the time zone to be “(UTC+1) Berlin, Amsterdam, Wien”. In this scenario we will have the trigger firing when we see the sunset out of our Berlin office windows.

Distant astronomic event

Imagine having an installation in Berlin demonstrating the day-night cycle of Auckland. We want our SYMPHOLIGHT located in Berlin to fire when it is sunset in Auckland, New-Zealand. We are having a SYMPHOLIGHT configured with the time zone of “(UTC+1) Berlin, Amsterdam, Wien” via web interface. We have specified the location Auckland inside of the dialog with the correct coordinates. In the dialog we specified the time zone to be “(UTC+1) Berlin, Amsterdam, Wien”. In this scenario we will have the trigger firing when the sun sets in Auckland. This could be at the early morning in Berlin, but in Auckland it is sunset time.

Testing a local astronomic event at a different place on the world

Imagine having an installation in Berlin demonstrating the day-night cycle of Berlin but designing the project in Auckland, New Zealand. We want to test it.

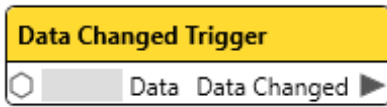
We want our SYMPHOLIGHT located in Berlin to fire when it is sunset in Berlin. We assume our SYMPHOLIGHT has the time zone “(UTC+12) Auckland” configured in the web interface. The test approach is setting the time zone of SYMPHOLIGHT to “(UTC+1) Berlin, Amsterdam, Wien” via web interface. The trigger will now be firing when the sun sets in Berlin. This might be some time in the night in Auckland time.

To avoid waiting for the trigger to fire while testing, you can modify the offset. The offset will be directly applied to the times of astronomic events in the time table.

Attention: Changing the offset will modify the project. After testing the offset should be reverted.



11.15.6 Data Changed Trigger



Properties	
Filter	<input type="text"/>
Data Changed Trigger Block	
Name	Data Changed Trigger
Block ID	94c86594-31db-4211-b6a3-
Data	
Data Type	Dynamic
Port Mode	Passive

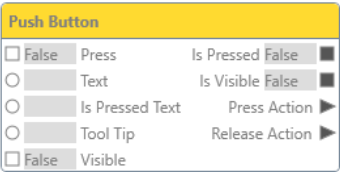
Use this Block to react on any changes that are related to a dedicated port, e.g. if a Sequence Opacity changes. Creates an Event, if a value of any data type including string changed.

The Block has one PortIn for the incoming value and one EventOut for the trigger.

Category	Description
Data	By default, the value is passively caught from the counterpart Block.



11.15.7 Push Button



Properties	
Filter	<input type="text"/>
Push Button Block	
Name	Push Button
Block ID	0571c9b8-b007-41b5-8a45-
Target	
Target	<not set> <input type="button" value="X"/> <input type="button" value="v"/>
Configurations	
Changes from ActionP	<input checked="" type="checkbox"/>
React only on Value Cl	<input checked="" type="checkbox"/>
Press	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Text	
Data Type	String
Input Type	Port
Port Mode	Passive
Is Pressed Text	
Data Type	String
Input Type	Port
Port Mode	Passive
Tool Tip	
Data Type	String
Input Type	Port
Port Mode	Passive
Visible	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Is Pressed	
Data Type	Bool
Port Mode	Active
Is Visible	
Data Type	Bool
Port Mode	Active

Use this Block to use button press and release events to trigger elements in the Workflow Designer, e.g. if you want to start a Sequence from an ActionPad Button.

This Block is target based. The Target value is displayed as main setting in the Block’s header.

When orphan, the Block has all the properties of a bound Block available.

The Block has five PortIns to control the Button and its visibility, two PortOuts for the current Button state and its state of visibility, and two EventOuts of the Button press state.

The Block can be bound to a Button from the ActionPad.

- Define the Target Property.
- Define the configuration of the PortIns.

Category	Property	Description
Target		Define the target to bind the Block to a Push Button from the ActionPad.
Configurations		Configure which incoming values are to be forwarded.
	Changes from ActionPad only	Enable to forward only values of the bound ActionPad target, not PortIn values. Disable to forward PortIn values as well. By default enabled.
	React only on Value Changes	Enable to forward only value changes from connected Blocks at the PortIns. Disable to forward also identical, unchanged values from the connected Blocks at the PortIns. By default enabled. For this property to be considered, “Changes from ActionPad only” must be disabled.



Press

Configure the Press state (pressed / not pressed) of the Push Button. By default, the bool value is passively received from the counterpart Block.

Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (pressed) if checked. An unchecked Value sets the PortIn to FALSE (not pressed).

Text

Define the Text on the Push Button when not pressed. By default, the string value is passively received from the counterpart Block.

Set the Input Type to Property to manually define a static value.

Is Pressed Text

Define the Text on the Push Button when pressed. By default, the string value is passively received from the counterpart Block.

Set the Input Type to Property to manually define a static value.

Tool Tip

Define the Tool Tip for the Push Button when the cursor hovers over the Push Button. By default, the string value is passively received from the counterpart Block.

Set the Input Type to Property to manually define a static value.



Visible

Configure the visibility of the Push Button. By default, the bool value is passively received from the counterpart Block.

Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).

Is Pressed

An actively sent bool value whether the Push Button is pressed in the ActionPad or not.

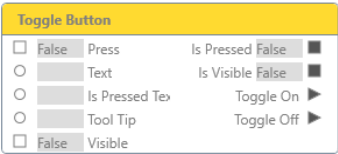
Is Visible

An actively sent bool value whether the Push Button is visible (displayed) in the ActionPad or not.

- The properties “Text”, “Is Pressed Text”, “Tool Tip”, and “Visible” relate to the same properties of the target value that are configured in the ActionPad Designer. When differing, the Properties configuration of the ActionPad is temporarily overruled by the Workflow.



11.15.8 Toggle Button



Properties

Filter

Toggle Button Block

Name: Toggle Button

Block ID: b96e6fa0-c50b-4021-b067-4

Target

Target: <not set> X

Configurations

Changes from ActionP: ☒

React only on Value Cl: ☒

Press

Data Type: Bool

Input Type: Port

Port Mode: Passive

Text

Data Type: String

Input Type: Port

Port Mode: Passive

Is Pressed Text

Data Type: String

Input Type: Port

Port Mode: Passive

Tool Tip

Data Type: String

Input Type: Port

Port Mode: Passive

Visible

Data Type: Bool

Input Type: Port

Port Mode: Passive

Is Pressed

Data Type: Bool

Port Mode: Active

Is Visible

Data Type: Bool

Port Mode: Active

Use this Block to use toggle on and off events to trigger elements in the Workflow Designer, e.g. if you want to start and stop a Sequence from Toggle Button.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has five PortIns to control the Toggle Button and its visibility, two PortOuts for the current Toggle Button state and its state of visibility, and two EventOuts of the Toggle state.

The Block can be bound to a Toggle Button from the ActionPad.

- Define the Target Property.
- Define the configuration of the PortIns.

Category	Property	Description
Target		Define the target to bind the Block to a Toggle Button from the ActionPad.
Configurations		Configure which incoming values are to be forwarded.
	Changes from ActionPad only	Enable to forward only values of the bound ActionPad target, not PortIn values. Disable to forward PortIn values as well. By default enabled.
	React only on Value Changes	Enable to forward only value changes from connected Blocks at the PortIns. Disable to forward also identical, unchanged values from the connected Blocks at the PortIns. By default enabled. For this property to be considered, "Changes from ActionPad only" must be disabled.



Press

Configure the Toggle state (pressed / not pressed) of the Toggle Button. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (pressed) if checked. An unchecked Value sets the PortIn to FALSE (not pressed).

Text

Define the Text on the Toggle Button when not pressed. By default, the string value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value.

Is Pressed Text

Define the Text on the Toggle Button when pressed. By default, the string value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value.

Tool Tip

Define the Tool Tip for the Toggle Button when the cursor hovers over the Toggle Button. By default, the string value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value.



Visible	<p>Configure the visibility of the Toggle Button. By default, the bool value is passively received from the counterpart Block.</p> <p>Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).</p>
---------	--

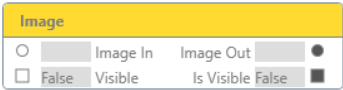
Is Pressed	<p>An actively sent bool value whether the Toggle Button is pressed in the ActionPad or not.</p>
------------	--

Is Visible	<p>An actively sent bool value whether the Toggle Button is visible (displayed) in the ActionPad or not.</p>
------------	--

- The properties “Text”, “Is Pressed Text”, “Tool Tip”, and “Visible” relate to the same properties of the target value that are configured in the ActionPad Designer. When differing, the Properties configuration of the ActionPad is temporarily overruled by the Workflow.



11.15.9 Image



Properties	
Filter	
Image Block	
Name	Image
Block ID	a0ad3a3d-d32e-4ca1-af9a-
Target	
Target	<not set> X
Image In	
Data Type	String
Input Type	Port
Port Mode	Passive
Visible	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Image Out	
Data Type	String
Port Mode	Active
Is Visible	
Data Type	Bool
Port Mode	Active

Use this block to set the image item.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has two PortIns to control the Image and the visibility and two PortOuts for the current Image and the state of visibility.

The Block can be bound to an Image from the ActionPad.

- Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to an Image from the ActionPad.



Image In

Define the Image by stating the name of an Image as it is called in the Media Library (Content Tab). By default, the string value is passively received from the counterpart Block.

Set the Input Type to Property to manually define a static value.

Visible

Configure the visibility of the Image. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).

Image Out

An actively sent string value of the Image.

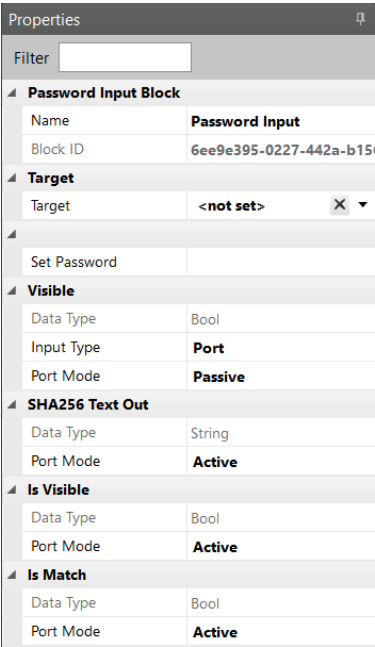
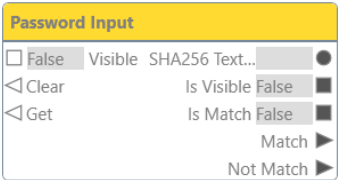
Is Visible

An actively sent bool value whether the Image is visible (displayed) in the ActionPad or not.

- The property “Visible” relates to the same property of the target value that is configured in the ActionPad Designer. When differing, the Property configuration of the ActionPad is temporarily overruled by the Workflow.



11.15.10 Password Input



Use this block to define the password for an ActionPad's Password Input element in order to create a password protection. React on password entries at the ActionPad in the Workflow Designer, e.g. if you want to hide ActionPad content when the ActionPad user is not verified.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has one PortIn to control the visibility, two EventIns for the entered password, three PortOuts and two EventOuts for the verification status, the SHA256 hash value and the state of visibility.

The Block can be bound to a Password Input from the ActionPad.

- Define the Target Property.



Category	Description
Target	Define the target to bind the Block to a Password Input from the ActionPad.
Set Password	Define the password which has to be entered in the ActionPad in the target bound Password Input.
Visible	Configure the visibility of the Password Input. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).
SHA256 Text Out	An actively sent string value of the in the ActionPad entered text as a SHA256 hash text.
Is Visible	An actively sent bool value whether the Password Input is visible (displayed) in the ActionPad or not.
Is Match	An actively sent bool value whether the entered value in Password Input is identical with the value defined in the "Set Password" property.

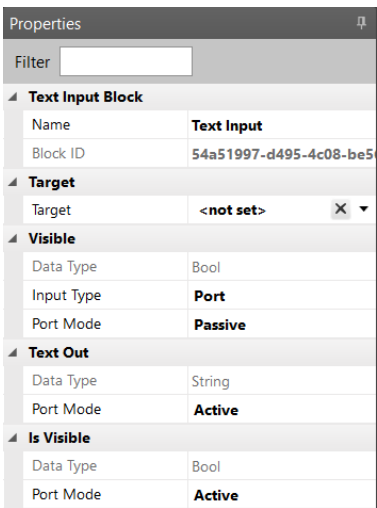
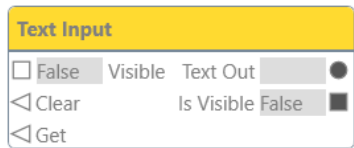
- Using the SHA256 text you can react differently on various password entries (=several user accounts).
- The property "Visible" relates to the same property of the target value that is configured in the ActionPad Designer. When differing, the Property configuration of the ActionPad is temporarily overruled by the Workflow.
- Steps to setup a password input:
 1. ActionPad Designer - add a Password Input element.
 2. WFD - add the corresponding target bound Password Input block.
 3. WFD - define a password at the "Set password" property.
 4. ActionPad - enter a text in the Password Input element. If the text matches the defined password, in the Workflow the trigger "Match" is fired and the property "Is Match" is set to TRUE. If the text differs from the defined password, in the Workflow the trigger "Not Match" is fired and the property "Is Match" is set to FALSE.
- You can use the password protection to, for example, control the visibility of ActionPad elements or pages.



- The EventIn “Get” catches the in the ActionPad entered password and forwards it to the property “Text Out”. If the entered password has not changed or has been forwarded due to the ActionPad property “Update On Enter or Focus Change”, the EventIn “Get” does not forward the password again.



11.15.11 Text Input



Use this Block if you want to use the text of an ActionPad's Text Input in the Workflow Designer, e.g. if you want to write a log entry from within the ActionPad or set a Fader value by entering a value.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has one PortIn to control the visibility, two EventIns for the entered text, and two PortOuts for the entered text value and the state of visibility.

The Block can be bound to a Text Input from the ActionPad.

- Define the Target Property.

Category	Description
----------	-------------

Target	Define the target to bind the Block to a Text Input from the ActionPad.
--------	---



Visible	Configure the visibility of the Text Input. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).
---------	---

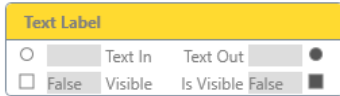
Text Out	An actively sent string value of the entered text.
----------	--

Is Visible	An actively sent bool value whether the Text Input is visible (displayed) in the ActionPad or not.
------------	--

- The property “Visible” relates to the same property of the target value that is configured in the ActionPad Designer. When differing, the Property configuration of the ActionPad is temporarily overruled by the Workflow.
- The EventIn “Get” catches the in the ActionPad entered text and forwards it to the property “Text Out”. If the entered text has not changed or has been forwarded due to the ActionPad property “Update On Enter or Focus Change”, the EventIn “Get does not forward the text again.



11.15.12 Text Label



Properties	
Filter	<input type="text"/>
Text Label Block	
Name	Text Label
Block ID	34552b99-8b0e-464a-93de-
Target	
Target	<not set> ✕
Text In	
Data Type	String
Input Type	Port
Port Mode	Passive
Visible	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Text Out	
Data Type	String
Port Mode	Active
Is Visible	
Data Type	Bool
Port Mode	Active

Use this Block if you want to change the text of ActionPad labels from the Workflow Designer, e.g. if you want to display different messages on the ActionPad depending on the daytime.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has two PortIns to control the Text and the visibility and two PortOuts for the current Text and the state of visibility.

The Block can be bound to a Label from the ActionPad.

- Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to a Label from the ActionPad.

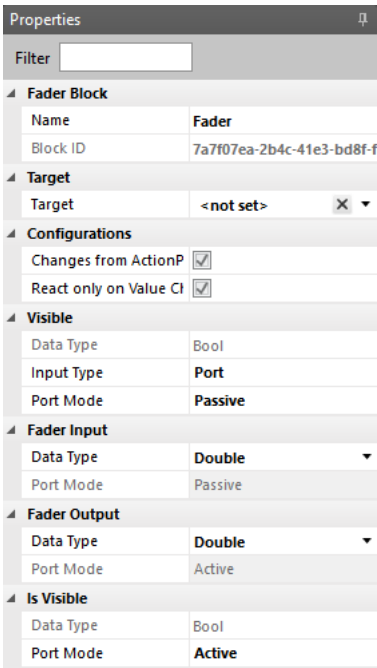
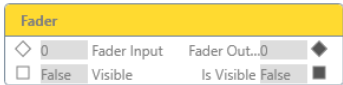


Text In	Configure the incoming Text of the Label. By default, the string value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value.
Visible	Configure the visibility of the Label. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).
Text Out	An actively sent string value of the Text from the Label.
Is Visible	An actively sent bool value whether the Label is visible (displayed) in the ActionPad or not.

- The properties “Text” and “Visible” relate to the same properties of the target value that are configured in the ActionPad Designer. When differing, the Properties configuration of the ActionPad is temporarily overruled by the Workflow.



11.15.13 Fader



Use this Block if you want to set or receive values of ActionPad Faders in the Workflow Designer, e.g. if you want to control the Opacity of a Sequence.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has two PortIns to control the Fader value and its visibility and two PortOuts for the current Fader value and state of visibility.

The Block can be bound to a Fader from the ActionPad.

- Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to a Fader from the ActionPad.

Configurations	Configure which incoming values are to be forwarded.
Changes from ActionPad only	Enable to forward only values of the bound ActionPad target, not PortIn values. Disable to forward PortIn values as well. By default enabled.
React only on Value Changes	Enable to forward only value changes from connected Blocks at the PortIns. Disable to forward also identical, unchanged values from the connected Blocks at the PortIns. By default enabled. For this property to be considered, “Changes from ActionPad only” must be disabled.
Visible	Configure the visibility of the Fader. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).
Fader Input	Define the data type of the PortIn. The Fader value can either be forwarded as Int64 or as Double.
Fader Output	Define the data type of the PortOut. The Fader value can either be forwarded as Int64 or as Double.
Is Visible	An actively sent bool value whether the Fader is visible (displayed) in the ActionPad or not.



- The property “Visible” relates to the same property of the target value that is configured in the ActionPad Designer. When differing, the Properties configuration of the ActionPad is temporarily overruled by the Workflow.
- Use the “Manual Changes Only” option when multiple faders control



11.15.14 Inline Frame



Properties	
Filter <input type="text"/>	
Inline Frame Block	
Name	Inline Frame
Block ID	fdfc2319-21c7-4cb6-bfe3-6:
Target	
Target	<not set> <input type="button" value="X"/>
Url In	
Data Type	String
Input Type	Port
Port Mode	Passive
Visible	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Url Out	
Data Type	String
Port Mode	Active
Is Visible	
Data Type	Bool
Port Mode	Active

Use this block if you want to change the URL for the Frame content.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has two PortIns to control the URL and the visibility and two PortOuts for the current URL and the state of visibility.

The Block can be bound to an Inline Frame from the ActionPad.

- Define the Target Property.



Category	Property	Description
Target		Define the target to bind the Block to an Inline Frame from the ActionPad.
URL In		Define the URL by stating the full format, e.g. "http://www.ecue.com". By default, the string value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value.
Visible		Configure the visibility of the Inline Frame. By default, the bool value is passively received from the counterpart Block. Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).
URL Out		An actively sent string value of the URL.
Is Visible		An actively sent bool value whether the Inline Frame is visible (displayed) in the ActionPad or not.

- The properties "URL In" and "Visible" relate to the same properties of the target value that are configured in the ActionPad Designer. When differing, the Properties configuration of the ActionPad is temporarily overruled by the Workflow.



11.15.15 Color Box / Color Wheel

Color Box			
	0;0;0	Color In	Color Out 0;0;0
	0	Red In	Red Out 0
	0	Green In	Green Out 0
	0	Blue In	Blue Out 0
	0	Alpha In	Alpha Out 0
<input type="checkbox"/>	False	Visible	Is Visible False

Color Wheel			
	0;0;0	Color In	Color Out 0;0;0
	0	Red In	Red Out 0
	0	Green In	Green Out 0
	0	Blue In	Blue Out 0
	0	Alpha In	Alpha Out 0
<input type="checkbox"/>	False	Visible	Is Visible False

Properties	
Filter	<input type="text"/>
Color Box Block	
Name	Color Box
Block ID	1b608de5-0ad6-43db-bc9e
Target	
Target	<not set>
Configurations	
Changes from ActionP	<input checked="" type="checkbox"/>
React only on Value Cl	<input checked="" type="checkbox"/>
Visible	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Color In	
Data Type	ProPhotoColor
Port Mode	Passive
Red In	
Data Type	Int64
Port Mode	Passive
Green In	
Data Type	Int64
Port Mode	Passive
Blue In	
Data Type	Int64
Port Mode	Passive
Alpha In	
Data Type	Int64
Port Mode	Passive
Color Out	
Data Type	ProPhotoColor
Port Mode	Active
Red Out	
Data Type	Int64
Port Mode	Active
Green Out	
Data Type	Int64
Port Mode	Active
Blue Out	
Data Type	Int64
Port Mode	Active
Alpha Out	
Data Type	Int64
Port Mode	Active
Is Visible	
Data Type	Bool
Port Mode	Active



Use the Color Box block or the Color Wheel block to set color.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has six PortIns to control the color and the visibility and six PortOuts for the current color and the state of visibility.

The Color Box block can be bound to a Color Box from the ActionPad.

The Color Wheel block can be bound to a Color Wheel from the ActionPad.

— Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to a Color Box or a Color Wheel from the ActionPad.
Configurations		Configure which incoming values are to be forwarded.
	Changes from ActionPad only	Enable to forward only values of the bound ActionPad target, not PortIn values. Disable to forward PortIn values as well. By default enabled.
	React only on Value Changes	Enable to forward only value changes from connected Blocks at the PortIns. Disable to forward also identical, unchanged values from the connected Blocks at the PortIns. By default enabled. For this property to be considered, "Changes from ActionPad only" must be disabled.

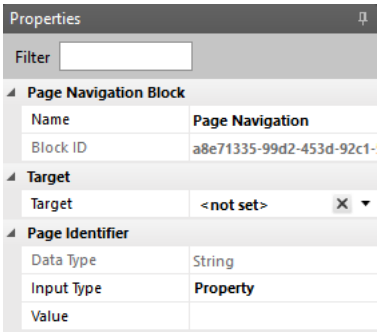
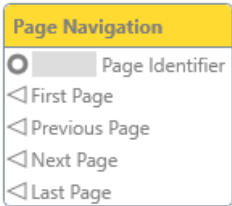


Visible	<p>Configure the visibility of the Color Box / Wheel. By default, the bool value is passively received from the counterpart Block.</p> <p>Set the Input Type to Property to manually define a static value. The Value property sets the value of the PortIn to TRUE (visible) if checked. An unchecked Value sets the PortIn to FALSE (invisible).</p>
Color In	<p>A passively received ProPhotoColor value for the color.</p> <p>You can change the Port Mode.</p>
Red In	<p>A passively received Int64 value (between 0 and 255) for the color channel.</p> <p>You can change the Port Mode.</p>
Green In	
Blue In	
Alpha In	
Color Out	<p>An actively sent ProPhotoColor value of the color.</p> <p>You can change the Port Mode.</p>
Red Out	<p>An actively sent Int64 value of the color channel.</p> <p>You can change the Port Mode.</p>
Green Out	
Blue Out	
Alpha Out	
Is Visible	<p>An actively sent bool value whether the Color Box / Wheel is visible (displayed) in the ActionPad or not.</p>

- The property “Visible” relates to the same property of the target value that is configured in the ActionPad Designer. When differing, the Properties configuration of the ActionPad is temporarily overruled by the Workflow.



11.15.16 Page Navigation



Use this Block if you would like to switch to a certain ActionPad Page on all devices that are actually showing the corresponding ActionPad, e.g. if you want to show a specific page in case an emergency comes up.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has one PortIn and four EventIns to navigate around the ActionPad Pages.

The Block can be bound to a Target Device (Desktop, iPad, Android) from the ActionPad.

- Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to a Target Device from the ActionPad.

Page Identifier	Define the target Page to which you want to switch. By default, the string value is set to the Input Type "Property" to manually define a static value. Set the Input Type to Port to passively receive a value from the counterpart Block.
-----------------	--

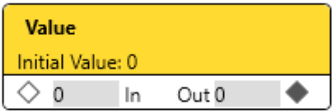
- The First Page EventIn switches to the Page tab with the index 0.
- The Previous Page EventIn switches to the Page tab with the preceding index.
- The Next Page EventIn switches to the Page tab with the following index.



- The Last Page EventIn switches to the Page tab with the highest index.
- Connect a Button block to the Page Navigation block to switch ActionPad Pages upon a Button press. You can also connect different block types to the Page Navigation block.
- All instances of one Target Device's ActionPad that are run in browsers switch to the same triggered target Page, independently of their current Page.
- In comparison to the manual selection of Pages using the ActionPad, Workflow actions concern every open ActionPad on all devices that are actually showing the corresponding ActionPad.



11.15.17 Value



Properties	
Filter	
Value Block	
Name	Value
Block ID	6be67979-108e-4c40-9913-;
Configurations	
Is event based	<input type="checkbox"/>
Settings	
Support Persistence	<input type="checkbox"/>
Initial Value	0
Data Type	Double
Input Port	
Data Type	Double
Port Mode	Passive
Output Port	
Data Type	Double
Port Mode	Active

Use this Block to add a constant value to your workflow, e.g. you want to trigger a scene of a DALI Group. The Value block can be configured to be triggered manually.

The Block has one PortIn and one PortOut for the value. The Initial Value is displayed as main setting in the Block's header.

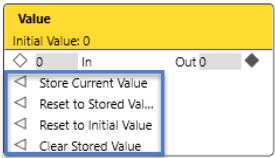
- Define the Data Type.
- Define the Initial Value.

Category	Property	Description
Is event based		Enable to execute this block on event trigger.
Settings	Support Persistence	Configure the constant value and whether to cache it or not. Enable to store the value locally to the persistent memory. Note that the memory is not saved inside SYMPHOLIGHT. A project transferred to another device does not contain the persistent memory data.



Initial Value	The initial value of the PortOut. For data type Bool, check the initial value to be TRUE. If "Store Every Value Change" is enabled, the initial value is the last stored value on loading the project.
Data Type	The data type for the PortIn and PortOut.
Store Every Value Change	Only for "Support Persistence": Enable to save the last value change to the persistent memory.
Input Port	By default, the value is passively caught from the counterpart Block.
Output Port	The value, by default sent actively.

- A possible replacement for a Property PortIn: Connect a Value block to the Port PortIn.
- You can use this Block to cache data and to continue a project from previous data state. Enable "Support Persistence". The Block gets four additional EventIns:



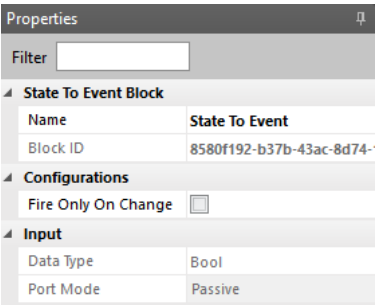
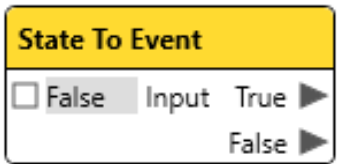
- Store Current Value:** When triggered, saves the current value of the Block to the persistent memory.
- Reset to Stored Value:** When triggered, sets the PortOut to the last stored value of the persistent memory.
- Reset to Initial Value:** When triggered, sets the PortOut to the Initial Value.
- Clear Stored Value:** When triggered, clears the persistent memory of its value data.

- Is the Out PortOut already connected to a counterpart Block and you then set an initial value, the value will be forwarded immediately.
- Use the Converter block if you want e.g. to do a DateTime to double conversion.



Every Link that is connected to the Block will be dismissed on changing the data type.

11.15.18 State To Event



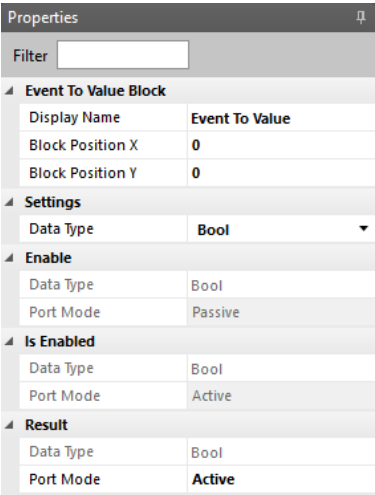
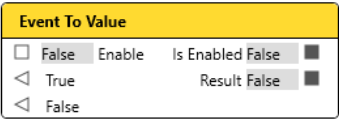
Use this Block to react on bool state changes, e.g. if a value changes from TRUE to FALSE. Creates an Event, if a bool value changed.

The Block has one PortIn for the incoming value and two EventOuts for each value state.

Category	Description
Fire Only On Change	Enable to fire events only on bool state changes, e.g. if a value changes from TRUE to FALSE. Disabled, events are fired every time a bool value was received.
Input	By default, the value is passively caught from the counterpart Block.



11.15.19 Event To Value



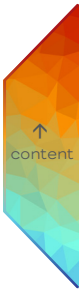
Use this Block if you want to output statically defined values on an event.

The Block has initially two EventIns for the incoming events, an Enable PortIn, one PortOut for the value and an Is Enabled PortOut.

- Define the data types of the incoming Ports.

Category	Property	Description
Settings		Define the EventIns for the incoming events.
	Data Type	Define the source data type of the incoming values. Choose between bool, double, and string.
	Number of Inputs	Define the amount of EventIns to create (max. 32).

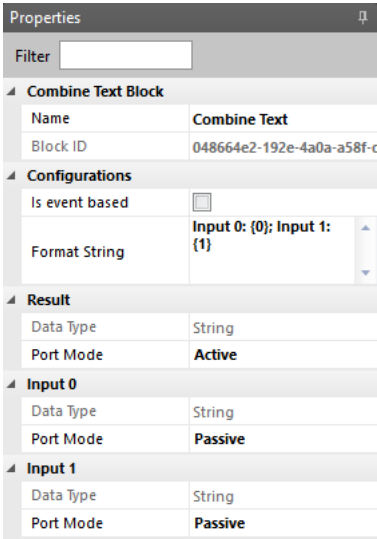
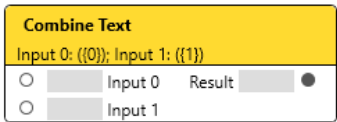
Output Values	Define each value that is forwarded from the PortOut when the corresponding EventIn is triggered.
---------------	---



Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Result	The latest incoming EventIn as a string value, by default sent actively.



11.15.20 Combine Text



Use this block to combine two or more texts.

The Block has initially two PortIns for the incoming text parts and one PortOut for the combined text. The Format String is displayed as main setting in the Block's header.

- Define how the combined text is constituted. The text is built with a format string (Format String), where {0} and {1}, etc. are the input values.

Category	Description
Is event based	Enables this block to be executed on event trigger.
Format String	<p>The format string to be created. It can contain free text as well as numbers in curly brackets (e.g. {0}). Each number in curly brackets defines a PortIn for a text part. The same number can occur multiple times.</p> <p>Example: "Here are the values of port0: {0} and port1: {1} and port0 again: {0}."</p>
Result	The combined text, by default sent actively.

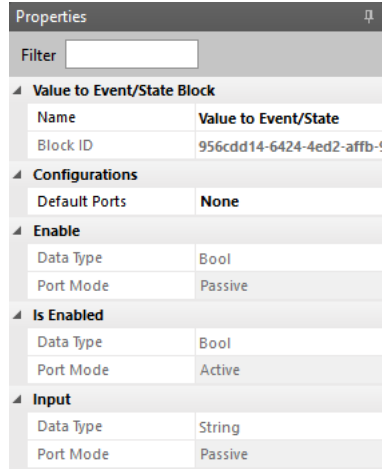
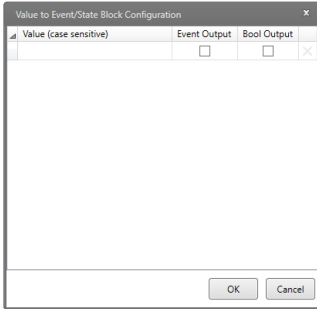
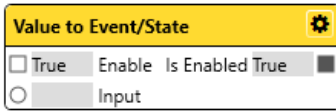


Text part (e.g. Input 0) Each text part has its own PortIn. The number of Ports is defined in the format string. You can change the Port mode for each Input PortIn.

- You can use string in the Format String e.g. to declare units.
Example: "{0} Watt".
- Use the keys Shift + Return in the Format String to insert a line break.



11.15.21 Value to Event / State



Use this Block to catch defined incoming Strings and to send a notification about that event.

The Block has one PortIn for the incoming string, an Enable PortIn, and an Is Enabled PortOut. Extended configuration is accessible at the cogwheel icon.


- Define string values in the extended configuration window and how the event of a matching string value is forwarded. A trigger EventOut and a bool PortOut are available for each string value.
- Configure the handling of non-matching strings.



Category	Property	Description
Default Ports		Define how to handle non-matching strings.
	Value	<p>Configure which type of default PortOut(s) shall be available. The default PortOut(s) is activated whenever none of the input values is matching the conditions.</p> <p>None: No action for non-matching strings. Bool Out: Creates the bool PortOut “Default” for all non-matching strings. FALSE: the incoming string matches one condition. TRUE: the incoming string does not match any condition. Event Out: Creates the EventOut “Default” for all non-matching strings. Whenever an incoming string does not match any condition, an trigger event is fired. Event & Bool Out: Creates the EventOut “Default” and the bool PortOut “Default”.</p>
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not.
Input		The PortIn that passively receives the string value. The Port mode of the counterpart Block should be set to “active”.

Defining string values in the extended configuration window

Define string values in the extended configuration window and how the event of a matching string value is forwarded:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .
2. Enter the string value you want to catch in the column “Match Expression”.
3. Either check the Event Output to create an EventOut, or check the Bool Output to create a bool PortOut, or check both options. At least one box needs to be checked.



The EventOut will send a trigger event whenever the incoming string matches the entered string value in “Match Expression”.

The bool PortOut will send a bool data: FALSE when the incoming string does not match the entered string value; TRUE when the incoming string matches the entered string value.

4. Click the “OK” button to save the configuration and close the window.

The Value to Event / State block has now additional PortOuts according to the extended configuration and named with the string values.

- To add another string value condition, double-click into the column “Match Expression” and press Enter. Alternatively, press the Tab key five times.
- Multi-select of rows is possible. You can also select all rows by clicking the triangle icon at the left top corner of the extended configuration window.
- To delete a string value condition, click the cross button of the row on the right.

Example: The three string values “ExampleText” with a bool PortOut, “AnotherExample” with both EventOut and bool PortOut, and “Sequence#1Play” with an EventOut are defined. Non-matching string values are not reported (Default Ports Value: None):

Value to Event/State

☐ True
 ☐ Enable
 ☐ Is Enabled
 ☐ True

☐ Input
 ExampleText False
 AnotherExa...False
 AnotherExample
 Sequence#1Play

Value to Event/State Block Configuration

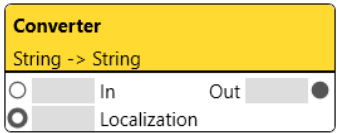
Value (case sensitive)	Event Output	Bool Output	
ExampleText	<input type="checkbox"/>	<input checked="" type="checkbox"/>	×
AnotherExample	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	×
Sequence#1Play	<input checked="" type="checkbox"/>	<input type="checkbox"/>	×
	<input type="checkbox"/>	<input type="checkbox"/>	×

OK

Cancel



11.15.22 Converter



Properties	
Filter <input type="text"/>	
Converter Block	
Name	Converter
Block ID	d12c6c2c-643f-4842-bdaf
Input Port	
Data Type	String X ▾
Data Type	String
Port Mode	Passive
Output Port	
Data Type	String X ▾
Data Type	String
Port Mode	Active
Localization	
Data Type	String
Input Type	Property
Value	

Use this Block to localize the date and time values (language and order of elements) and to extract specific date and time values with the DateTime data type.
As a general data converter this block is obsolete since v3.0 because the Workflow Designer Blocks are converting data types automatically.

The Block has two PortIns for incoming values and one PortOut for the converted value.
The Data Type Conversion defines the title of the Block.

- Define the data types of the incoming and of the outgoing Ports.

Category	Property	Description
Input Port		An initially passively received value to be converted into another data type.
	Data Type	Define the source data type of the incoming value.
	Port Mode	Define the Port mode of the Input Port.



Output Port		An initially actively sent value of the conversion's result.
	Data Type	Define the target data type of the outgoing value.

	Port Mode	Define the Port mode of the Output Port.
--	-----------	--

Configurations	Rounding Rule	Only for Data Type "Double": Select the rounding method.
----------------	---------------	--

String to DateTime Conversion	Input Date Order	Only for Input Port - Data Type "String" and Output Port - Data Type "DateTime". Define the order in which the String value is converted to a DateTime value regarding the Day, Month and Year values.
-------------------------------	------------------	--

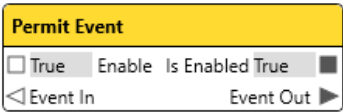
DateTime to String Conversion	Format	Only for Input Port - Data Type "DateTime" and Output Port - Data Type "String". Define how the DateTime value is converted to a String value. The list of valid values and format examples are in the description box
-------------------------------	--------	---

Localization		Only for Data Type "DateTime". Define a local time format and language that is applied on conversion. Set the Input Type to Property to manually define a static value. The list of valid values is in the description box. For using the current system location, leave the Value empty.
--------------	--	--

- The availability of data types depends on the counterpart Port. E.g. after choosing a data type for the PortIn, the PortOut offers a changing set of data types (same goes when first setting the PortOut data type.). To retrieve the full list of (all) data types, you have to select "String" as data type on either Port. Then the other Port provides all data types again.



11.15.23 Permit Event



Properties	
Filter	<input type="text"/>
Permit Event Block	
Name	Permit Event
Block ID	d40d1c53-6715-4e41-a3e1-
Is Enabled	
Data Type	Bool
Port Mode	Active
Enable	
Data Type	Bool
Port Mode	Passive

Use this Block to decide whether to pass an incoming event or to block it.

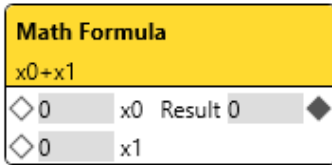
The Block has one EventIn for the incoming event, an Enable PortIn, one EventOut for the permitted event and an Is Enabled PortOut.

Category	Description
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.

Enable	A passively received bool value to dis- or enable the Block.
--------	--



11.15.24 Math Formula



Properties	
Filter	<input type="text"/>
Math Formula Block	
Name	Math Formula
Block ID	5e80bd66-a2bb-4ac8-b6df-
Configurations	
Is event based	<input type="checkbox"/>
Formula	x0+x1
x0	
Data Type	Double
Input Type	Port
Port Mode	Passive
x1	
Data Type	Double
Input Type	Port
Port Mode	Passive
Result	
Data Type	Double
Port Mode	Active

Use this Block for advanced mathematical functionalities, e.g. combine arithmetic calculations.

The Block has PortIns for the variables of the formula and one PortOut for the result of the calculation. The equation is displayed as main setting in the Block's header.

- Define a mathematical formula.

Category	Description
Is event based	Enables this block to be executed on event trigger.
Formula	Enter the mathematical function as a string. Any formula is possible (all math methods such as "pow(x,y)": the number x raised to the power y) and any variable names (any string separated by an operator is a new variable / Port).



Variable (e.g. x0)	<p>A variable starts with any letter and can contain numbers and any amount of digits.</p> <p>By default, the content is passively caught from the counterpart Block.</p> <p>Set the Input Type to Property to manually define a static value.</p>
--------------------	--

Result	<p>The actively sent result from the calculation.</p> <p>You can change the Port Mode.</p>
--------	--

- Available mathematical functions to use in the Formula are:
 - abs:** returns the absolute value of a specified number.
 - cos:** returns the cosine of the specified radians.
 - cosh:** returns the hyperbolic cosine of the specified radians.
 - arccos:** returns the inverse cosine of the specified radians.
 - sin:** returns the sine of the specified radians.
 - sinh:** returns the hyperbolic sine of the specified radians.
 - arcsin:** returns the inverse sine of the specified radians.
 - tan:** returns the tangent of the specified radians.
 - tanh:** returns the hyperbolic tangent of the specified radians.
 - arctan:** returns the inverse tangent of the specified radians.
 - sqr:** returns the square root of a specified number.
 - rem:** returns the remainder that results after division of a by b.
 - root:** returns the specified root of a specified number.
 - pow:** returns a specified number raised to the specified power.
 - exp:** returns e raised to the specified power.
 - log:** returns the logarithm of a specified number.
 - round:** rounds a value to the nearest integer or to the specified number of fractional digits.
 - truncate:** calculates the integral part of a number.
 - floor:** returns the largest integral value less than or equal to the specified number.
 - ceiling:** returns the smallest integral value greater than or equal to the specified number.
 - sign:** returns an integer that indicates the sign of a number.
- Available absolute terms to use in the Formula are:
 - pi**
 - tao** = $2.0 * \pi$
 - e**
 - phi** = 1.61803398874989
 - major** = 0.61803398874989
 - minor** = 0.38196601125011



$\text{pitograd} = 180.0 / \pi$

$\text{piofgrad} = \pi / 180.0$

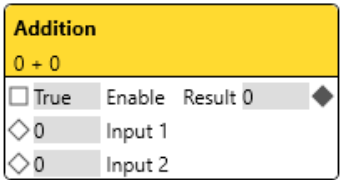
- Use absolute terms without brackets; for functions use brackets. Example:

Formula	Result
phi	1.61803398874989
2phi	3.23606797749978
sqrt(4)	2

- Truth values are supported, e.g. " $x0 > 2$ " where $x0$ is a variable.



11.15.25 Addition



Properties	
Filter	<input type="text"/>
Addition Block	
Name	Addition
Block ID	42926f67-f8be-4968-bdbd-
Configurations	
Is event based	<input type="checkbox"/>
Input Count	2
Input 2	
Data Type	Double
Input Type	Port
Port Mode	Passive
Enable	
Data Type	Bool
Port Mode	Passive
Input 1	
Data Type	Double
Port Mode	Passive
Result	
Data Type	Double
Port Mode	Active

Use this Block to add two or more values.

The Block has initially two PortIns for the incoming values, an Enable PortIn, and one PortOut for the result. The equation is displayed as main setting in the Block's header.

- Define the numbers of terms (Input Count).

Category	Description
Is event based	Enables this block to be executed on event trigger.
Input Count	Count of available PortIns / terms (min. 2 values, max. 20 values).
PortIn (e.g. Input 2)	By default, the value is passively caught from the counterpart Block. Set the Input Type to Property to manually define a static value.



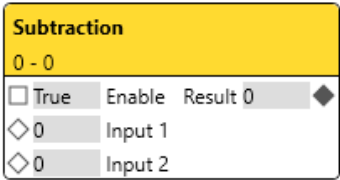
Enable A passively received bool value to dis- or enable the Block.

Input 1 A received value.
You can set the Port Mode.

Result The result of the addition, by default sent actively.



11.15.26 Subtraction



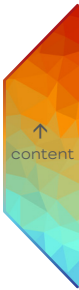
Properties	
Filter	<input type="text"/>
Subtraction Block	
Name	Subtraction
Block ID	0b29c7ad-2565-4d8d-a5f6-
Configurations	
Is event based	<input type="checkbox"/>
Input Count	2
Input 2	
Data Type	Double
Input Type	Port
Port Mode	Passive
Enable	
Data Type	Bool
Port Mode	Passive
Input 1	
Data Type	Double
Port Mode	Passive
Result	
Data Type	Double
Port Mode	Active

Use this Block to subtract two or more values.

The Block has initially two PortIns for the incoming values, an Enable PortIn, and one PortOut for the result. The equation is displayed as main setting in the Block's header.

- Define the numbers of terms (Input Count).

Category	Description
Is event based	Enables this block to be executed on event trigger.
Input Count	Count of available PortIns / terms (min. 2 values, max. 20 values).
PortIn (e.g. Input 2)	By default, the value is passively caught from the counterpart Block. Set the Input Type to Property to manually define a static value.



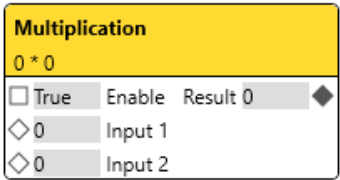
Enable A passively received bool value to dis- or enable the Block.

Input 1 A received value.
You can set the Port Mode.

Result The result of the subtraction, by default sent actively.



11.15.27 Multiplication



Properties	
Filter	
Multiplication Block	
Name	Multiplication
Block ID	69ecadac-0afe-40a9-ac76-fl
Configurations	
Is event based	<input type="checkbox"/>
Input Count	2
Input 2	
Data Type	Double
Input Type	Port
Port Mode	Passive
Enable	
Data Type	Bool
Port Mode	Passive
Input 1	
Data Type	Double
Port Mode	Passive
Result	
Data Type	Double
Port Mode	Active

Use this Block to multiply two or more values.

The Block has initially two PortIns for the incoming values, an Enable PortIn, and one PortOut for the result. The equation is displayed as main setting in the Block's header.

- Define the numbers of factors (Input Count).

Category	Description
Is event based	Enables this block to be executed on event trigger.
Input Count	Count of available PortIns / factors (min. 2 values, max. 20 values).
PortIn (e.g. Input 2)	By default, the value is passively caught from the counterpart Block. Set the Input Type to Property to manually define a static value.

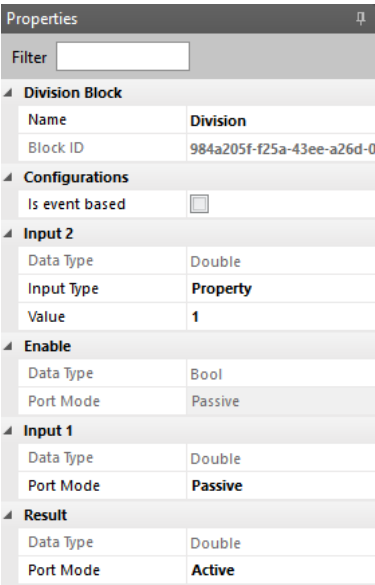
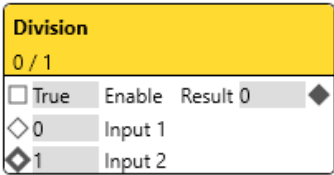
Enable A passively received bool value to dis- or enable the Block.

Input 1 A received value.
You can set the Port Mode.

Result The result of the multiplication, by default sent actively.



11.15.28 Division



Use this Block to divide two values.

The Block has two PortIns for the incoming values, an Enable PortIn, and one PortOut for the result. The equation is displayed as main setting in the Block's header.

- Define the numbers of terms (Input Count).

Category	Description
Is event based	Enables this block to be executed on event trigger.
Input 2	The divisor value. You can set the Port Mode. By default it is a Property PortIn. The value 0 is invalid.
Enable	A passively received bool value to dis- or enable the Block.

Input 1	The dividend value. You can set the Port Mode.
Result	The result of the division, by default sent actively.



11.15.29 Moving Average



Properties	
Filter <input type="text"/>	
Moving Average Block	
Name	Moving Average
Block ID	ae0ed04f-eca8-4fde-857c-b
Configurations	
Is event based	<input type="checkbox"/>
Calculation Mode	Input Count
Start Processing Imme	<input type="checkbox"/>
Included Input Values	5
Default Value	0
Input Value	
Data Type	Double
Port Mode	Passive
Moving Average	
Data Type	Double
Port Mode	Active

Use this Block if you want to create a moving average value based on a number of received values or based on values that have been received within a certain time.

The Block has one PortIn for the received values and one PortOut for the result of the moving average.

- Define the Calculation Mode.
- Define the usage of values.

Category	Description
Is event based	Enables this block to be executed on event trigger.
Calculation Mode	The work mode of the block. Time: Calculates the moving average based on the PortIn values received within the specified time. Input Count: Calculates the moving average based on the specified number of lastly received values.
Start Processing Immediately	When enabled, calculates the moving average also when less values than specified have been received or less time than specified has past.

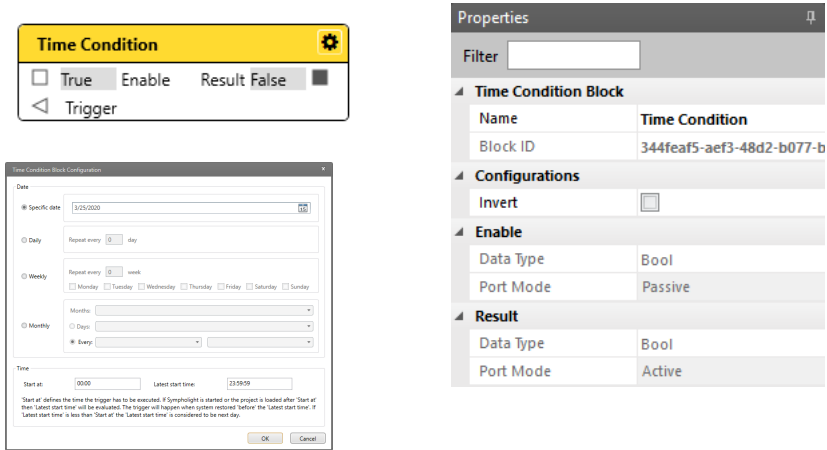


Included Input Values	For Input Count as Calculation Mode only: Specifies the count of lastly received values which is used to calculate the moving average. Default: 5 values.
Covered Timespan	For Time as Calculation Mode only: Specifies the elapsed timespan for which the received PortIn values will be used to calculate the moving average (in seconds).
Default Value	The value which is used in the Moving Average PortOut while not enough PortIn values have been received.
Input Value	The passively received values to calculate the moving average from.
Moving Average	The actively sent moving average from the calculation.

Please note that while assembling your Workflow, the Moving Average resets with every newly added connection to the “Input Value” PortIn. This does not lead to any loss of functionality when running a project. Recommendation: Connect blocks (e.g. Value Blocks) to the PortIns first and then define or change the values of the connected blocks (e.g. Value Blocks).



11.15.30 Time Condition



Use this Block to react on time events, but providing a bool value instead of a trigger. Use this Block if you want to limit the execution of other triggers to a specific time range, e.g. only allow to switch on the lights if it is after 7am.

The Block has one EventIn for the incoming trigger, an Enable PortIn, and a bool PortOut. Extended configuration is accessible at the cogwheel icon. The condition configuration is displayed as main setting in the Block's header.


- Define date and time of the time condition in the extended configuration window.
- Connect a trigger Link to the EventIn.

Category	Description
Invert	Inverts the bool value of the PortOut. Use this setting to get a FALSE PortOut value whenever the time condition is met, e.g. to disable a button for a certain time span.
Enable	A passively received bool value to dis- or enable the Block.
Result	An actively sent bool value whether the time condition is met or not.



Defining date and time in the extended configuration window

Define date and time in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .
2. Select one of the four date options at the Date frame:

Specific date: Use the calendar at the icon or enter manually a particular day.

Daily: Enter a number for the daily interval, e.g. enter "2" for triggering every other day. Compared to the "0" with immediate effect, when entering a "1", the day count starts with the day after configuring the block.

Weekly: Enter a number for the weekly interval, e.g. enter "1" for triggering every week. Compared to the "0" with immediate effect, when entering a "1", the week count starts with the Monday of the next week at the time of configuring the block. Further select the weekdays.

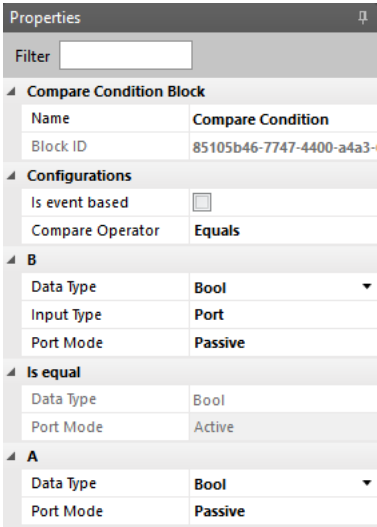
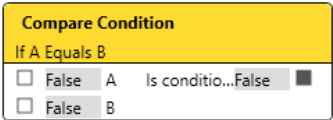
Monthly: Select one or multiple months. Also enter specific days or weekdays. The configuration is of immediate effect.

3. Define a time span at the Time frame. During the time span the condition is met (TRUE PortOut value).
4. Click the "OK" button to save the configuration and close the window.

The Time Condition block has now the condition configuration displayed as main setting in the Block's header. It displays the next date and time of being TRUE.



11.15.31 Compare Condition



Use this Block to compare two values with each other.

The Block has two PortIns for the values that are to be compared, and a bool PortOut. The condition configuration is displayed as main setting in the Block's header.

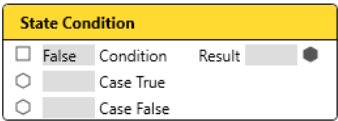
- Define how to compare two values with the Compare Operator.
- Define the two PortIns for the values you want to compare.



Category	Property	Description
Is event based		Enables this block to be executed on event trigger.
Compare Operator		The operator the block needs to compare the two values.
	Value	<p>Equals: The two values match.</p> <p>SmallerThan: When A is smaller than B.</p> <p>GreaterThan: When A is greater than B.</p> <p>SmallerThanOrEquals: When A is smaller than or matches B.</p> <p>GreaterThanOrEquals: When A is greater than or matches B.</p>
B		B is the value to be compared to. Adjust the data type, input type and port mode for the B PortIn according to your needs. You can set the B PortIn as a Property with a fixed value.
Is condition met		An actively sent bool value whether the compare condition is met or not.
A		Adjust the data type and the Port mode for the A PortIn according to your needs.



11.15.32 State Condition



Properties	
Filter	
State Condition Block	
Name	State Condition
Block ID	6a539d65-ce28-40f0-999b-4
Case True	
Data Type	Dynamic
Input Type	Port
Port Mode	Passive
Case False	
Data Type	Dynamic
Input Type	Port
Port Mode	Passive
Result	
Data Type	Dynamic
Port Mode	Active
Condition	
Data Type	Bool
Port Mode	Passive

Use this Block if you would like to switch between two different scenarios depending on one single value, e.g. change the text of a button if another button has been pressed. Depending on the Condition PortIn either the value of the Case True PortIn or of the Case False PortIn is routed to the Result PortOut.

The Block has three PortIns for the state and the cases that are to be caught, and a dynamic PortOut. The condition configuration is displayed as main setting in the Block's header.

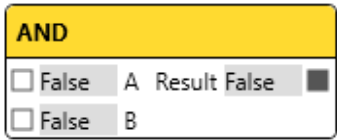
- Define the two Cases (Case True and Case False)



Category	Description
Case True	<p>The value which is forwarded at the PortOut if the value of the bool Condition PortIn is TRUE.</p> <p>As a dynamic data type Port, any PortOut is connectable.</p> <p>By default, the content is actively caught from the counterpart Block.</p> <p>Set the Input Type to Property to manually define a static value.</p> <p>The Value property sets the value of the PortIn to TRUE if checked. An unchecked Value sets the PortIn to FALSE.</p>
Case False	<p>The value which is forwarded at the PortOut if the value of the bool Condition PortIn is FALSE.</p> <p>As a dynamic data type Port, any PortOut is connectable.</p> <p>By default, the content is actively caught from the counterpart Block.</p> <p>Set the Input Type to Property to manually define a static value.</p> <p>The Value property sets the value of the PortIn to TRUE if checked. An unchecked Value sets the PortIn to FALSE.</p>
Result	<p>The value of the Case which matches the Condition as a dynamic PortOut.</p> <p>You can change the Port Mode.</p>
Condition	<p>The bool value that decides which Case's value is forwarded in the Result PortOut.</p> <p>You can change the Port Mode.</p>



11.15.33 AND



Properties	
Filter	
AND Block	
Name	AND
Block ID	bee9a10b-16c8-4941-ae14-
Configurations	
Number Of Input Port:	2
Invert Output	<input type="checkbox"/>
A	
Data Type	Bool
Input Type	Port
Port Mode	Passive
B	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Result	
Data Type	Bool
Port Mode	Active

Use this Block if you want to combine two or more conditions: only if all inputs are TRUE, the output will be TRUE. A logical AND function with two or more inputs.

The Block has initially two PortIns for the incoming values and one PortOut for the result.

- Define the numbers of PortIns (Number of Input Ports).

Category	Description
Number of Input Ports	Count of available PortIns (min. 2 values).

Invert Output	Inverts the bool value of the PortOut. Use this setting to get a FALSE PortOut value whenever the AND condition is met. Enable to use the block as AND gate (output which is FALSE only if all its inputs are TRUE).
---------------	--

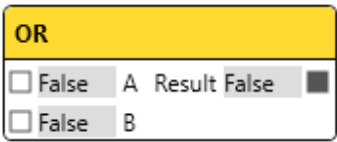


PortIn (e.g. A) By default, the value is passively caught from the counterpart Block.
Set the Input Type to Property to manually define a static value.

Result The result of the combination, by default sent actively.



11.15.34 OR



Properties	
Filter	<input type="text"/>
OR Block	
Name	OR
Block ID	6b90daed-2789-4ace-8b5a-
Configurations	
Number Of Input Port:	2
Invert Output	<input type="checkbox"/>
A	
Data Type	Bool
Input Type	Port
Port Mode	Passive
B	
Data Type	Bool
Input Type	Port
Port Mode	Passive
Result	
Data Type	Bool
Port Mode	Active

Use this Block if you want to combine two or more conditions: only if one or more inputs are TRUE, the output will be TRUE. A logical OR function with two or more inputs.

The Block has initially two PortIns for the incoming values and one PortOut for the result.

- Define the numbers of PortIns (Number of Input Ports).

Category	Description
Number of Input Ports	Count of available PortIns (min. 2 values).
Invert Output	Inverts the bool value of the PortOut. Use this setting to get a FALSE PortOut value whenever the OR condition is met. Enable to use the block as NOR gate (output which is TRUE only if all its inputs are FALSE).



PortIn (e.g. A) By default, the value is passively caught from the counterpart Block.
Set the Input Type to Property to manually define a static value.

Result The result of the combination, by default sent actively.



11.15.35 NOT



Properties	
Filter	<input type="text"/>
NOT Block	
Name	NOT
Block ID	522a609f-a572-49a9-82fb-2
In	
Data Type	Bool
Port Mode	Passive
Result	
Data Type	Bool
Port Mode	Active

Use this Block to negate a boolean value: if the input is TRUE, the output will be FALSE and vice versa. A logical NOT function for one value.

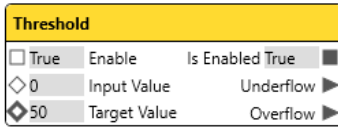
The Block has one PortIn for the incoming value and one PortOut for the result.

Category	Description
In	By default, the value is passively caught from the counterpart Block.

Result	The result of the combination, by default sent actively.
--------	--



11.15.36 Threshold



Properties	
Filter	<input type="text"/>
Threshold Block	
Name	Threshold
Block ID	ffe249f2-931c-41c4-b939-f0
Configurations	
Initial State	DoNothing
Tolerance	0
Underflow Delay	0
Overflow Delay	0
Target Value	
Data Type	Int64
Input Type	Property
Value	50
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Input Value	
Data Type	Int64
Port Mode	Passive

Use this Block if you want to turn off the light if the light sensor value is above a threshold level and to turn on the light if the sensor value is below, using triggers.

The Block has an Enable PortIn, two PortIns for the light sensor value and the threshold, two EventOuts to switch the light on / off, and an Is Enabled PortOut.

- Define the Target Light Level.
- Define Tolerance, Underlow and Overflow Delay, and Initial State where applicable.

Category	Description
----------	-------------

Initial State	<p>Define whether to send a trigger upon initializing the SYMPHOLIGHT project:</p> <p>SetLightOn: The Light On EventOut is fired.</p> <p>SetLightOff: The Light Off EventOut is fired.</p> <p>DoNothing: No trigger is fired.</p>
---------------	--

Tolerance	Define the value range around the Target Light Level that is ignored for sending triggers.
Underflow Delay	Define the time in seconds to elapse between receiving a sensor value below the threshold Target Light Level value and firing the Light On EventOut trigger. During the delay time, incoming sensor values are ignored.
Overflow Delay	Define the time in seconds to elapse between receiving a sensor value above the threshold Target Light Level value and firing the Light Off EventOut trigger. During the delay time, incoming sensor values are ignored.
Target Value	The threshold value to which the incoming sensor value is compared. Define whether the threshold value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Input Value	A passively received Int64 value of the light sensor.



11.15.37 Fade

Fade			
<input type="checkbox"/>	True	Enable	Is Enabled True
<input type="text" value="0"/>	Lower Target	Is fading Up	False
<input type="text" value="100"/>	Upper Target	Is fading Down	False
<input type="text" value="5"/>	Fade Time (s)	Value	0
<input type="text" value="0"/>	External value	Inverted Value	100
< Fade Up		Fade Up Stopped >	
< Fade Down		Fade Down Stopped >	
< Pause			
< Reset			
< Immediate Up			
< Immediate Down			

Properties	
Filter	<input type="text"/>
Fade Block	
Name	Fade
Block ID	d6aa5f1e-652a-4ed0-8334--
Behavior	
Trigger ports	Dedicated event ports
Retrigger Behavior	Continue fading
Opposite direction trigger	Pause current fade
Lower Target	
Data Type	Double
Input Type	Property
Value	0
Upper Target	
Data Type	Double
Input Type	Property
Value	100
Fade Time (s)	
Data Type	Double
Input Type	Property
Value	5
External value	
Data Type	Double
Input Type	Property
Value	0
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Is fading Up	
Data Type	Bool
Port Mode	Active
Is fading Down	
Data Type	Bool
Port Mode	Active
Value	
Data Type	Double
Port Mode	Active
Inverted Value	
Data Type	Double
Port Mode	Active



Use this Block to fade between two values. E.g. fade the opacity of a Sequence or the luminance of a DALI group.

The Block has initially four PortIns for the opacity, six EventIns to control the play state of the Sequence, an Enable PortIn, four PortOuts for opacity and playback, two EventOuts of the play state, and an Is Enabled PortOut.

Category	Property	Description
Behavior	Trigger Ports	Choose a set of ports which trigger fadings.
	Retrigger Behavior	Define the behavior of the block when a fade in the same direction is triggered while the block is already fading.
	Opposite Direction Trigger	Define the behavior of the block when a fade in the opposite direction is triggered while the block is already fading in one direction.
Lower Target		Defines the minimum value where the fade down ends.
	Input Type	Define whether the target value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").
	Value	Define the target value of the lower target.
Upper Target		Defines the maximum value where the fade up ends.
	Input Type	Define whether the target value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").



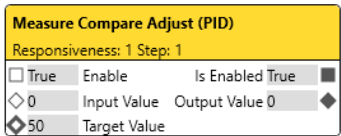
	Value	Define the target value of the upper target.
Fade Time		The maximum duration of the fade from Lower Target to Upper Target.
	Input Type	Define whether the target value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").
	Value	Define the target value of the lower target.
External Value		Define an initially set or externally set value. Also used for the Reset Trigger event.
	Input Type	Define whether the target value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").
	Value	Define the target value of the lower target.
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Is Fading Up		A bool value about whether the block is currently fading up. You can define the Port Mode.
Is Fading Down		A bool value about whether the block is currently fading down. You can define the Port Mode.



Value	A double value of the fading value. You can define the Port Mode.
Inverted Value	A double value that inverts the fading value. You can define the Port Mode.
<ul style="list-style-type: none">External value: Helps, when a target value can be controlled from multiple sources (Fade block, ActionPad Fader, etc.). Other sources can be connected to this port of the Fade block instead of the target directly, and there should not be much conflict between fading and other sources.Immediate up/down events: E.g. set a fade to a Glass Touch button short push and the immediate jump to the double click or long press. For users, who are in a hurry.Fade up/down stopped events: To be notified on a fade stop and completion.Fade time can be a port. Changing the fade time during a fade is also supported. E.g. if someone wants to trigger a “fast forward”.	



11.15.38 Measure Compare Adjust (PID)



Properties	
Filter	
Measure Compare Adjust (PID) Block	
Name	Measure Compare Adjust (
Block ID	af46388b-82cd-4019-9e06-;
Block Configuration	
Min. Output Value	0
Max. Output Value	100
Enable Expert Mode	<input type="checkbox"/>
Calculation Settings	
Responsiveness	1
Step	1
Target Value	
Data Type	Double
Input Type	Property
Value	50
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Input Value	
Data Type	Double
Port Mode	Passive
Output Value	
Data Type	Double
Port Mode	Active

Use this Block if you want to control the fixture light level depending on a certain sensor target value, e.g. in Daylight Harvesting scenarios.

The Output Value is calculated in accordance to the Calculation Settings and controls the light level of fixtures. The Output Value is adjusted until the light level that is produced in assistance of the fixtures and measured by the sensor equals the Target value.

The Block has an Enable PortIn, two PortIns for the light sensor value and the sensor target value, one PortOut of the current light value, and an Is Enabled PortOut.

- Define the light adjustment method and the forwarded light value limits at the Block Configuration.
- Define how to calculate the light adjustment at the Calculation Settings.
- Define the Target value of the light to be achieved.

Category	Property	Description
Block Configuration		Define the light limits of the forwarded light value and the light adjustment method.
	Min. Target Level	Define the minimum value the forwarded light value (Output Value) can be.
	Max. Target Level	Define the maximum value the forwarded light value (Output Value) can be.
	Enable Expert Mode	<div>Define the light adjustment method: Enable to change the Calculation Settings to the three PID terms K_p, K_i, and K_d using the formula</div> <div>$u(t) = K_p e(t) + K_i \int_0^t e(r) dr + K_d \frac{de(t)}{dt}$</div> <div>When disabled, the simple mode is activated with K_p as Responsiveness, K_i as Step, and $K_d=0$.</div>



Calculation Settings	The adjustment calculation is based on the terms of the Calculation Settings.	
	Step	Define the size of the adjustment step between 1 as low impact and 100 as high impact (high risk of oversteering). $K_i = \text{Step} / 100$.
	Responsiveness	Define the speed of the adjustment between 1 as slowest and 100 as fastest. $K_p = \text{Responsiveness} * 0.01$.
	K_p	The proportional gain of the adjustment.
	K_i	The integral gain of the adjustment.
	K_d	The derivative gain of the adjustment.
Target Value	The value to which the incoming sensor value is compared. The Output Value is adjusted until the light level that is produced in assistance of the fixtures and measured by the sensor equals the Target Value.	
	Input Type	Define whether the target value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port"). Use Port for defined scenarios like a connected Multi Delay block or Human Centric Lighting block.
	Value	Define the target value of light. The target value is limited by the Min. Target Level and the Max. Target Level.
Enable	A passively received bool value to dis- or enable the Block.	



Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Sensor Value	A passively received double value of the light sensor.
Output Value	A double value that is calculated to control the fixtures in order to achieve the Target Value measured by the sensor. Connect to fixtures. You can define the Port Mode. The Output value is limited by the Min. Target Level and the Max. Target Level.

- The Block is designed for the use case of sensor and fixtures being placed in one area.
- When switching between Expert Mode and Simple Mode in the Block Configuration, the Calculation Settings are reset.
- The configuration of the Measure Compare Adjust (PID) block is subject to the sensors and fixtures in use as well as the location of the devices. Test the installation in the field and for Daylight Harvesting settings under various scenarios to determine whether the sensor is eligible and to define the Calculation Settings. Use a second sensor to check the light value of the target space, e.g. table surface.
- To determine the Min. Target Level for Daylight Harvesting scenarios, test the installation on a very bright day. Control the light value until the target value of the second sensor is achieved. Measure the light value of the fixtures at that moment. Set this value as Min. Target Level.
- To determine the Max. Target Level for Daylight Harvesting scenarios, test the installation at a very dark situation, e.g. at night. Turn all fixtures off. Measure the light value of the fixtures at that moment. Set this value as Max. Target Level.



11.15.39 Human Centric Lighting

What is Human Centric Lighting - An Introduction

Human centric lighting is the art of creating lighting that mimics the natural daylight that drives our bodily functions by changing the color temperature and intensity to stimulate natural light colors. It enhances human performance, comfort, health and wellbeing. Human Centric Lighting solutions essentially contribute to levels of well-being.

The most obvious effect of light on humans, is vision. It enables us to identify brightness, shapes, colors, images, and perceive information and contrast. But light also impacts our biology; it affects our hormones, alertness, attention and fatigue, and also determines our body clock and circadian rhythm. Human centric lighting takes these effects into consideration to provide a holistic and application oriented approach to lighting for humans. It balances visual, emotional and biological needs of humans in lighting application. Dynamic white light with brightness levels and light colours modified according to the course of day support our natural biological rhythm to increase our well-being and performance capability, as well as achieving optimum synchronisation with the outdoor environment.

To implement an adequate Human Centric Lighting solution install a time-controlled lighting system with a circadian-friendly light sequence, meaning the luminaires adapt their light colour and intensity during the course of the day.

Based upon time, date and geographical location, the Human Centric Lighting block of SYMPHOLIGHT can calculate sunrise and sunset for the given location at any time of the year. Transition values are calculated to continuously and accordingly adapt the color temperature and intensity.

The Human Centric Lighting block is user configurable, the default configuration is: At 7 AM, the Block starts adapting the light temperature setting from 2700 K to its maximum of 6500 K over the course of four hours. The light will stay at a fixed temperature level of 6500 K until the 2 PM. Then, the Block starts the inverse transition process, adapting the light temperature back to 2700 K, reaching 2700 K at 7 PM.



Human Centric Lighting

☐

True

Enable

Is Enabled

True

Start Preview

Color Tem...2700

Stop Preview

Intensity 0

Human Centric Lighting Block Configuration

100 %

50 %

0 %

0

3

6

9

12

15

18

21

24

Values

Time Of Day

Settings

Color Temperature

Intensity

Min Color Temperature [K]

2700

Line Color

Max Color Temperature [K]

6500

Color

FFBA0000

Values

Time

Value

00:00

2700.00

+

-

01:00

2700.00

+

-

02:00

2700.00

+

-

03:00

2700.00

+

-

04:00

2700.00

+

-

05:00

2700.00

+

-

06:00

2700.00

+

-

07:00

3000.00

+

-

08:00

3500.00

+

-

OK

Cancel

Apply

Properties

Filter

Human Centric Lighting Block

Name

Human Centric Lighting

Block ID

842b0f08-54e0-4f81-98c5-2

Configurations

Execution Interval

60

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Color Temperature

Data Type

Int64

Port Mode

Active

Intensity

Data Type

Double

Port Mode

Active

Use this Block to create human centric lighting features like defining a color temperature curve depending on the time of day.

The Block has an Enabled PortIn, two EventIns to control the Preview, an Is Enabled PortOut, and two PortOuts for the values of the human centric lighting features. Extended configuration is accessible at the cogwheel icon.

- Define the human centric lighting features in the extended configuration window.
- Define the Execution Interval.

Category	Properties	Description
Execution Interval		Define the interval in seconds in which the block calculates its outputs. Minimum time is 0.1 seconds.

Enable		A passively received bool value to dis- or enable the Block.
--------	--	--

232

Is Enabled

An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.

Color
Temperature

The color temperature value that is calculated by the block according to the human centric lighting settings.
You can change the Port Mode.

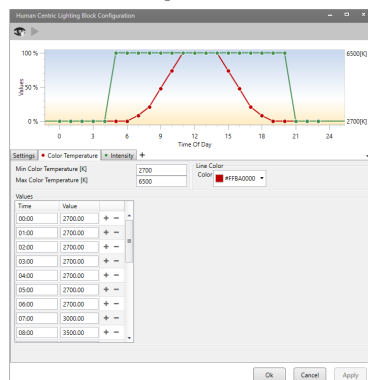
Intensity

The light intensity value that is calculated by the block according to the human centric lighting settings.
You can change the Port Mode.

Defining the human centric lighting features in the extended configuration window

Define the human centric lighting features by setting the location and the desired lighting features in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window:



The graph represents the configuration of the tabs. The scale at the right displays the minimum and maximum values of the selected lighting feature (tab). The scale at the left displays the percent with the minimum value as 0% and the maximum value as 100%. The background color displays the color gradient between the minimum and the maximum value of the Color Temperature. Use the calendar at the top of the graph to display the astronomic settings for a certain day.

2. Define the location in the Visual Settings tab to adjust the astronomic times as reference of the astronomic events on site. You can select the astronomic events that you want to be displayed in the graph and define offset times.

3. Define the human centric lighting features. Each tab represents a lighting feature. By default, the two features Color Temperature in Kelvin and Intensity in percent are predefined.




You can define the minimum and maximum limits for the values that can be used in the table, a line color for the graph, and values for specific times. The number of time spots is also adjustable, by default one dot every hour. Each time spot with its value is represented by a colored dot in graph.

You can change the name of the lighting feature that is also displayed as PortOut name in the block by using the context menu of the tab.

4. You can manually add (double-click in the graph) and adjust (drag and drop) dots for a selected lighting feature (tab) in the graph for rough settings.
5. Click the "OK" button to save the configuration and close the window. Click "Apply" to save the changes, but continue working in the configuration window.

The Human Centric Lighting block has now the lighting features configured. All features are implemented with Ports in the Block.

Preview the Human Centric Lighting configuration

Use the Preview for testing the human centric lighting configuration. The Preview is a fast run-through of one day, displayed on your fixtures. Define the duration of one run-through and the interval for value changes during this time in the extended configuration window, Visual Settings tab. You can run the Preview in loop and define a time frame. You can either start the Preview from inside the extended configuration window, enabling the Preview with the  icon and clicking the play icon . Or you can start and control the Preview using the EventIns of the Block. When using the  icon, the icon turns red to indicate the Preview mode of the block. Additionally, a time bar appears in the graph for visual feedback in the graph when the Preview is running. This time bar can be dragged to manually change the starting point of the Preview. When a Preview is running or enabled from inside the extended configuration window, you cannot change any configuration in the extended configuration window.

- The transition values between the defined time spots are linear calculated / interpolated.
- Due to the high amount of data, the light values on the fixtures might vary slightly during Preview from the live execution values.
- Hover over a dot in the graph for information on its values.
- Time values will be sorted automatically after re-selecting the tab if entered in a "wrong" chronological order.
- When changing the minimum and maximum limits of a lighting feature in the extended configuration window and values exceed these limits, these values are automatically adjusted to the new limits. E.g. you have value of 6500 K used in the Color Temperature

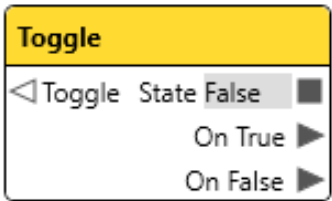


feature. Now you set the maximum limit for Color Temperature to 6000 K; the value is automatically changed to 6000 K.

- Each additional tab for a new lighting attribute creates a corresponding PortOut in the Block.
- Combine the Human Centric Lighting block with Daylight Threshold and sensor blocks to achieve a flawless energy balance while avoiding energy input.



11.15.40 Toggle



Properties	
Filter	<input type="text"/>
Toggle Block	
Name	Toggle
Block ID	5f40af8e-c390-4440-929b-8
Configurations	
Initial State	<input type="checkbox"/>
State	
Data Type	Bool
Port Mode	Active

Use this Block to achieve a toggle functionality, e.g. first button press: light on, second button press: light off.

The Block has a Trigger EventIn, two EventOuts to switch the light on / off, and a State PortOut.

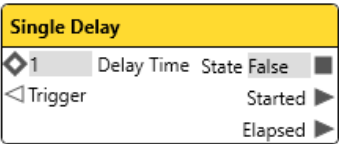
- No configuration is needed.

Category	Description
Initial State	Enable to fire the Triggered EventOut on project start.
State	An actively sent bool value that toggles on every incoming trigger, alternating between TRUE and FALSE.

- The Triggered EventOut is fired when the State is changed to TRUE.
- The Released EventOut is fired when the State is changed to FALSE.



11.15.41 Single Delay



Properties	
Filter	<input type="text"/>
Single Delay Block	
Name	Single Delay
Block ID	a6186471-db70-4415-8a08-
Delay Time	
Data Type	Double
Input Type	Property
Value	1
State	
Data Type	Bool
Port Mode	Active

Use this Block to react on an event a certain time after it has been triggered.

The Block has a Trigger EventIn, a PortIn for the delay time, two EventOuts to switch the light on / off, and a State PortOut.

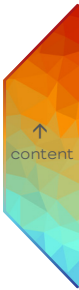
- Define the Delay Time.

Category	Description
Delay Time	Define the time after which the State value will be reset to FALSE after it has been triggered to TRUE.

State

An actively sent bool value that becomes TRUE on receiving the incoming trigger and turns back to FALSE after the defined Delay Time.

- The Started EventOut is fired as soon as a Trigger EventIn is received; the State is TRUE.
- The Elapsed EventOut is fired after the defined Delay Time and when the State is changed to FALSE. If another Trigger EventIn is received before the Delay Time has passed, the Elapsed EventOut will not be fired and the State will not change to FALSE.



11.15.42 Multi Delay

Multi Delay

<input type="checkbox"/> True	Enable	Is Enabled True	<input type="checkbox"/>
1	T0 Delay	Value 0	
0	T0 Value	T0 State False	
1	T1 Delay	T1 State False	
0	T1 Value	T2 State False	
1	T2 Delay	Elapsed	▶
0	T2 Value	T0 Started	▶
◀ Trigger		T1 Started	▶
		T2 Started	▶

Multi Delay Block Configuration

Trigger Times				
Name	Start Time	Duration	Value	Trigger
T0	00:00:00.000	00:00:01.0	0	<input checked="" type="checkbox"/> + -
T1	00:00:01.000	00:00:01.0	0	<input checked="" type="checkbox"/> + -
T2	00:00:02.000	00:00:01.0	0	<input checked="" type="checkbox"/> + -

Ok Cancel Apply

Properties

Filter

Multi Delay Block

Name	Multi Delay
Block ID	6df29704-d99f-4de2-a61a

Retrigger Behavior

Retrigger Behavior	Restart From Begin
--------------------	--------------------

T0 Delay

Data Type	Double
Input Type	Property
Value	1

T0 Value

Data Type	Int64 ▼
Input Type	Property
Value	0

T1 Delay

Data Type	Double
Input Type	Property
Value	1

T1 Value

Data Type	Int64 ▼
Input Type	Property
Value	0

T2 Delay

Data Type	Double
Input Type	Property
Value	1

T2 Value

Data Type	Int64 ▼
Input Type	Property
Value	0

Enable

Data Type	Bool
Port Mode	Passive

Is Enabled

Data Type	Bool
Port Mode	Active

...



Use this Block to react on an event a certain time after it has been triggered. Multiple delays can be specified.

The Block has an Enabled PortIn, three Delay and three Value PortIns for the delay configuration, an Is Enabled PortOut, a Trigger EventIn, a Value PortOut, three State PortOuts, an Elapsed EventOut for the completed execution, and three Started EventOuts. Extended configuration is accessible at the cogwheel icon.

- Define the multiple trigger sequence in the extended configuration window.
- Connect a Link to the Trigger EventIn.

Category	Property	Description
Retrigger Behavior		Define how the Block should react when the Block is triggered while the trigger sequence is still running: Restart From Begin, Restart Last Trigger, Restart One After End, Ignore Trigger.
Delay (e.g. T0)		Define the Delay Time. Each trigger has its own Delay Time.
	Input Type	Define whether the Delay Time is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").
	Value	Define the time after which the State value of a trigger will be reset to FALSE after it has been triggered to TRUE.




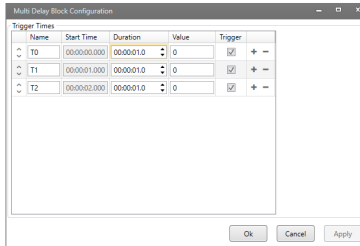
Value (e.g.T0)		Define the value that is added to a trigger. Each trigger has its own value.
	Date Type	The data type for the PortIn. Please note that all Value properties, PortIns and PortOut, are automatically set to the same Data Type.
	Input Type	Define whether the added trigger value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").
	Value	Define the value that is sent along with the trigger at the Value PortOut.
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Value		The value that is sent when its event has been triggered. You can change the Port Mode. You can change the Data Type. Please note that all Value properties, PortIns and PortOut, are automatically set to the same Data Type.
State (e.g. T0)		An actively sent bool value that becomes TRUE when the trigger is activated and turns back to FALSE after the defined Delay Time.






Defining the multiple trigger sequence in the extended configuration window

Define the multiple trigger sequence with the number of triggers, their order, and each trigger's delay time and value, in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



2. Define the number of triggers with the icons  for adding and  for removal. You can rename each trigger.
3. Define the delay time (Duration) and the Value of each trigger.
4. Define the sequence of the triggers: The order of their entries in the configuration window equals the chronology of the triggers. The top most trigger is fired first, the second row is fired afterwards and so on. Use the arrows at the front to change the sequence: .

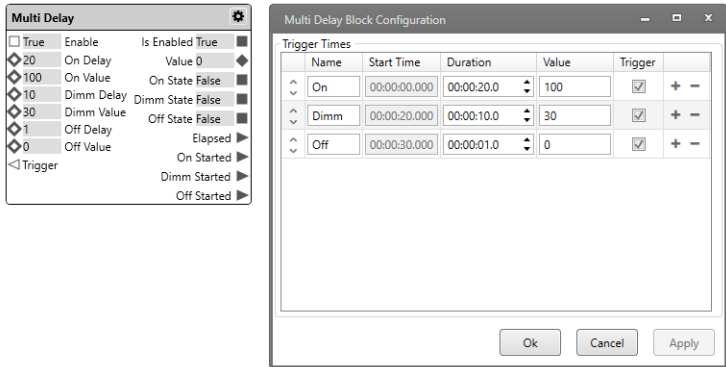
The Start Time value indicates the absolute amount of time between the block is triggered and the specific trigger will be fired.

5. Dis- or enable whether to have the Started EventOut for a trigger displayed in the block. Uncheck the "Trigger" box to remove the Started EventOut.
6. Click the "OK" button to save the configuration and close the window. Click "Apply" to save the changes, but continue working in the configuration window.

The Multi Delay block has now the trigger sequence configured. All settings are implemented with Ports in the Block.



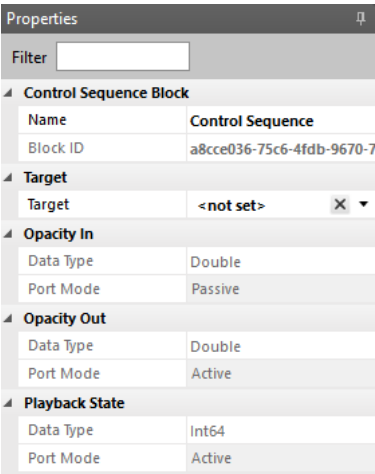
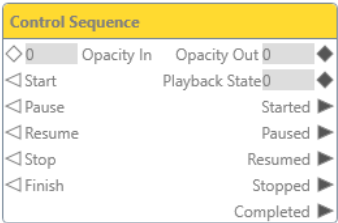
Example:



Here the Multi Delay block is used for a stepwise dimming of fixtures. The Value PortOut provides the light level for fixtures. First “On” is triggered with its Value of 100. For 20 seconds, the light is on at its maximum capacity. During this period, the On State PortOut is TRUE. After the Delay Time of 20 seconds has passed, “Dimm” is triggered. The light level is reduced to 30%. The On State value turns to FALSE and the Dimm State value becomes TRUE. And after the “Dimm” Delay Time of 10 seconds has passed, “Off” is triggered at last that turns the light off. Only Off State is TRUE for 1 second.

- The trigger properties Delay Time and Value can be defined in the Properties window of the Workflow Designer as well as in the extended configuration window of the Multi Delay block when set to the Input Type “Property”. Properties of the Input Type “Port” are not editable in the extended configuration window.
- A Started EventOut is fired as soon as a Trigger EventIn for this trigger is received; the State is TRUE.
- The Elapsed EventOut is fired after a defined Delay Time and when a State is changed to FALSE. If another Trigger EventIn is received before the Delay Time has passed, the Elapsed EventOut will not be fired and the State will not change to FALSE.

11.15.43 Control Sequence



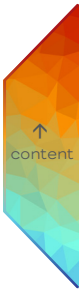
Use this Block to control the opacity and play state of a sequence or to react on play state changes, e.g. to start, stop or pause a Sequence.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available. The Block has a PortIn for the opacity, five EventIns to control the play state of the Sequence, two PortOuts for opacity and playback, and five EventOuts of the play state. The Block can be bound to a Sequence from the Content Tab.

— Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to a Sequence.
Opacity In		A passively received double value of the opacity for the Sequence.
Opacity Out		An actively sent double value of the opacity of the Sequence.



Playback State

An actively sent Int64 value of the play state of the Sequence:

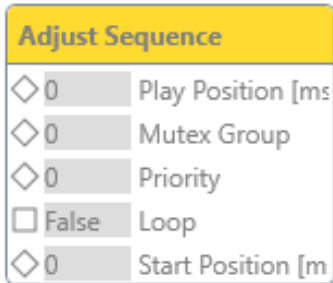
0: Sequence does not run (use trigger Stop or Finish)

1: Sequence is running (use trigger Start or Resume)

2: Sequence is paused (use trigger Pause).



11.15.44 Adjust Sequence



Properties	
Filter	<input type="text"/>
Adjust Sequence Block	
Name	Adjust Sequence
Block ID	07bb9b92-aac3-46d7-b704-
Target	
Target	<not set> ✕ ▼
Play Position [ms]	
Data Type	Int64
Port Mode	Passive
Mutex Group	
Data Type	Int64
Port Mode	Passive
Priority	
Data Type	Int64
Port Mode	Passive
Loop	
Data Type	Bool
Port Mode	Passive
Start Position [ms]	
Data Type	Int64
Port Mode	Passive

Use this Block to control some advanced parameters of a Sequence, e.g. the priority value of a Sequence.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has five PortIns to control the advanced parameters of the Sequence.

The Block can be bound to a Sequence from the Content Tab.

- Define the Target Property.

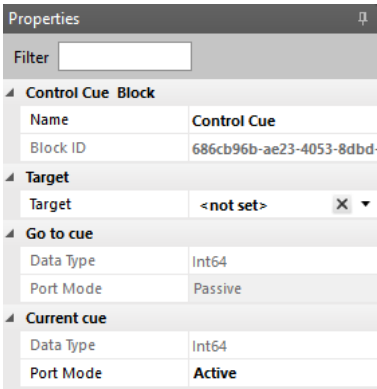
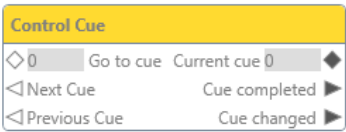
Category	Property	Description
Target		Define the target to bind the Block to a Sequence.

Play Position	A passively received Int64 value to jump to a specific time point of the running Sequence in milliseconds. Refers to the total playing time, e.g. looping time included.
MutexGroup	A passively received Int64 value to change the Mutex Group of the Sequence. This change does not affect the Sequence configuration in the Content Tab.
Priority	A passively received Int64 value to change the Priority for the Sequence.
Loop	A passively received bool value to dis- or enable looping the Sequence. If FALSE, but Loop Segment property in Sequence Properties is enabled, then the Sequence loops nonetheless.
Start Position	A passively received Int64 value for the Start Position of the Sequence in milliseconds.

- All parameters are Properties of the Sequence that are configured in the Content Tab. When differing, the Properties configuration of the Content Tab is temporarily overruled by the Workflow.



11.15.45 Control Cue



Use this Block to control Sequences based on Cues, e.g. skip to the next or previous Cue or react to completed cues of a Sequence.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has one PortIn and two EventIns to control the played Cue of the Sequence, one PortOut for the currently played Cue, and two EventOuts of Cues events.

The Block can be bound to a Sequence from the Content Tab.

— Define the Target Property.

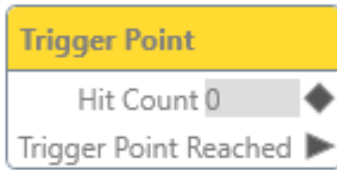
Category	Property	Description
Target		Define the target to bind the Block to a Sequence.
Go to cue		A passively received Int64 value to jump to a specific Cue of the Sequence.
Current cue		An actively sent Int64 value of the current Cue. You can change the Port Mode. When the Sequence is completed, the value 1 is sent.



- The EventOut “Cue completed” is also fired when a Sequence without any Cues is completed.
- The Next Cue EventIn jumps to the next Cue. When the Sequence is configured to loop, it restarts with the first Cue after the last Cue.
- The Previous Cue EventIn jumps to the previous Cue. Even when the Sequence is configured to loop, it jumps to the first Cue at most - it does not continue with the last Cue.



11.15.46 Trigger Point



Properties	
Filter	<input type="text"/>
Trigger Point Block	
Name	Trigger Point
Block ID	21be0ff0-06a9-4f08-b402-d
Target	
Target	<not set> <input type="button" value="X"/>
Hit Count	
Data Type	Int64
Port Mode	Active

Use this Block to react on Trigger Points that have been assigned to a specific Sequence. Whenever the Sequence play time is passing a trigger point an event will be fired and the number of crossings is available.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has one PortOut for the Hit Count and one EventOut of the Trigger Point passing.

The Block can be bound to a Trigger Point of a Sequence from the Content Tab.

- Define the Target Property.

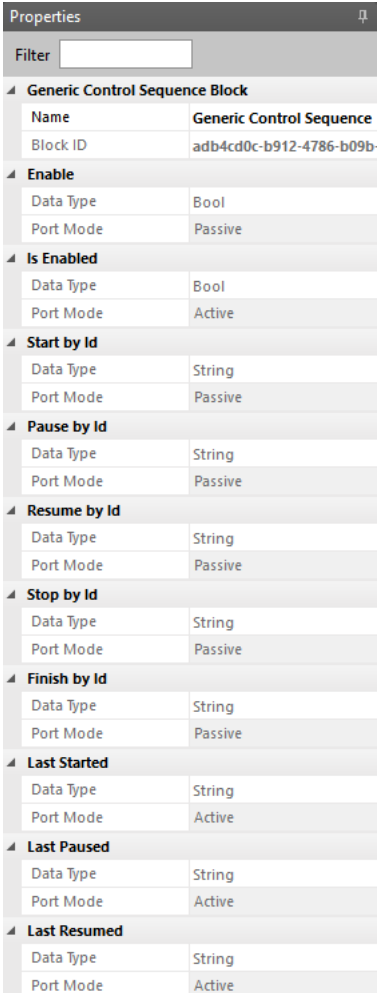
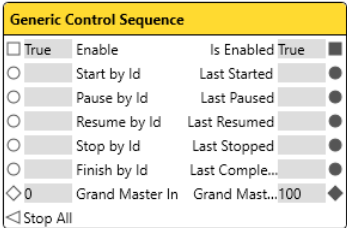
Category	Property	Description
Target		Define the target to bind the Block to a Trigger Point of a Sequence.

Hit Count		An actively sent Int64 value of the number of Trigger Point crossings.
-----------	--	--

- Please note that Trigger Points only perform when triggered from the Output tab.



11.15.47 Generic Control Sequence



Use this Block to control multiple Sequences at once and get notifications of them at one place.

The Block has four six PortIns and one EventIns to control the Sequences, an Enable PortIn, six PortOuts to get notifications of the Sequences' activities, and an Is Enabled PortOut.

- The following controls are available: start, pause, resume, stop, and finish any Sequence by sending the Sequence Identifier (string value) to the block. Set the opacity of the Grand Master in the Output Tab to a specific value with the Grand Master In PortIn. Stop all running Sequences with the Stop All EventIn.
- The following notifications are available: the last started, the last paused, the last resumed, the last stopped, and the last completed Sequence as Sequence Identifier (string value). Get the opacity of the Grand Master in the Output Tab with the Grand Master Out PortOut.
- Use this Block when you work with many Sequences to reduce the scale of the Workflow Designer.
- Use this Block to trigger Sequences with the network, e.g. connecting the Receive from Network block.



11.15.48 Mutex Group

Mutex Group

☐ True

Enable

Is Enabled

☒ True

Index

Index

☒ False

Continue With Next Sequence

Identifier

☒ False

Loop Mutex Group

Display Name

Properties

Filter

Mutex Group Block

Name

Mutex Group

Block ID

a00491f1-733d-43ac-8db0-4

Target

Target

<not set>

X

Loop Configuration

Data Type

Bool

Input Type

Property

Continue With Next Sequence

☐

Data Type

Bool

Input Type

Property

Loop Mutex Group

☐

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Index

Data Type

Int64

Port Mode

Passive

Data Type

Int64

Port Mode

Active

Identifier

Data Type

String

Port Mode

Active

Display Name

Data Type

String

Port Mode

Active

Use this Block to control Mutex Groups, e.g. to jump to the next Sequence within the same Mutex Group.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has an Enable PortIn, three PortIns to control the Mutex Group, six EventIns to control the play state of the Mutex Group, three PortOuts about the current Sequence and an Is Enabled PortOut.

The Block can be bound to a Mutex Group. A Mutex Group is the abbreviation for Mutual Exclude Collection ID which is defined in the Properties of a Sequence in the Content Tab. With Mutex you can easily define a set of Sequences of which only one can run at a time. Assign the same Mutual Exclude Collection ID to all the Sequences you want to group. If the ID is zero, no mutex grouping happens.

— Define the Target Property.

Category	Property	Description
Target		Define the target to bind the Block to a Mutex Group.
Loop Configuration		Define the behavior of the Mutex Group.
	Input Type	Define whether the Continue With Next Sequence value is a fixed bool value (Input Type "Property") or a received bool Input value (Input Type "Port").
	Continue With Next Sequence	Enable to continue with the next Sequence when the current Sequence has finished. Continues until the last Sequence of the Mutex Group.
	Input Type	Define whether the Loop Mutex Group value is a fixed bool value (Input Type "Property") or a received bool Input value (Input Type "Port").
	Loop Mutex Group	Enable to loop the entire Mutex Group when the last Sequence has finished. Works only when "Continue With Next Sequence" is enabled.

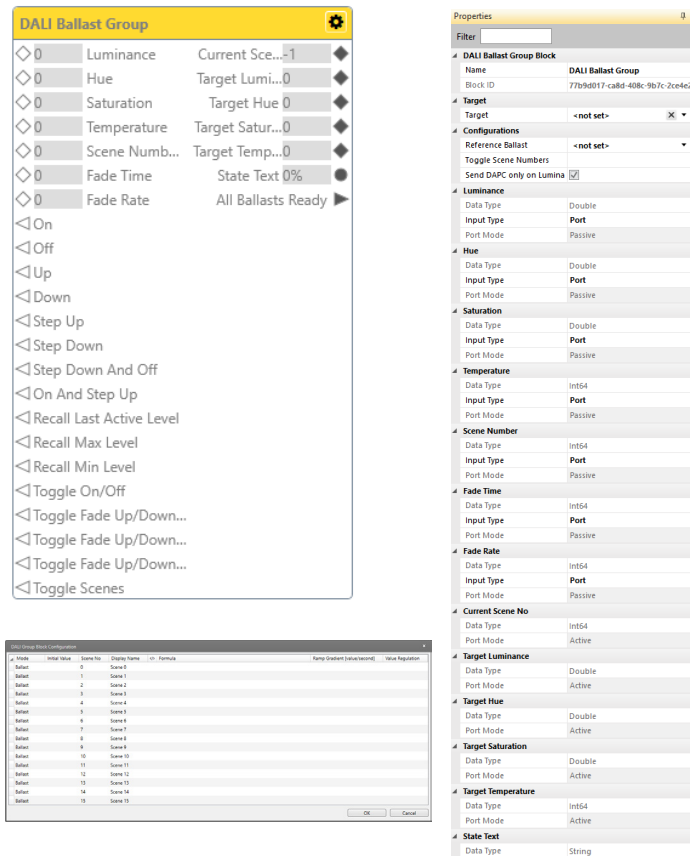


Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the “Enable” PortIn.
Index	A passively received Int64 value to control the Sequence within the Mutex Group that is to be played. And an actively sent Int64 value of the current Mutex Group’s Index. You can find the Index value in the Content Tab: The Properties of a Sequences contain the entry “Mutex Group Index”.
Identifier	An actively sent string value of the current Sequence’s Identifier. You can find the Identifier value in the Content Tab: The Properties of a Sequences contain the entry “Identifier”.
Display Name	An actively sent string value of the current Sequence’s Display Name. You can find the Display Name value in the Content Tab: The Properties of a Sequences contain the entry “Display Name”.



- When a Sequence has the Property “Loop Sequence” in the Content Tab enabled, the Mutex Group does not go to the next Mutex Group Sequence even though “Continue With Next Sequence” is enabled. The Mutex Group loops the enabled Sequence.
- Please note that all Sequence Properties, such as “Fade In Time”, “Fade Out Time”, and “Start Position” are taken into account when running a Sequence from within the Mutex Group block.
- When the current Sequence is stopped, the PortOuts are updated: Index = 0, Identifier and Display Name are left empty.
- Is the received PortIn Index 0, then the mutex group is stopped.
- Is the received PortIn Index less than or equal to 0, then the mutex group is stopped.
- Is the received PortIn Index greater than the mutex group’s index count, then the largest index is played.

11.15.49 Ballast Group and Daylight Harvesting



Use this Block to control your DALI ballast Groups. You can also configure Daylight Harvesting.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has double PortIns and PortOuts for incoming and forwarded properties of light and EventIns to trigger light values. Extended configuration is accessible at the cogwheel icon.

The Block can be bound to a DALI Ballasts group as defined in the Setup Tab.

- Define the Target.
- Define the Reference Ballast that act as representative for DALI Settings. Important when one of these properties differ within the group's ballasts.
- Define dynamic scenes for Daylight Harvesting in the extended configuration window.

In the following list, the properties without a Port are explained:

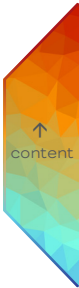
Category	Properties	Description
Target		Define the target to bind the Block to a DALI group.
Reference Ballast		Define a ballast whose Fade Time and Fade Rate settings (Setup Tab - Properties of a virtual ballast: DALI Settings) are attributed to the whole group.
Toggle Scene Numbers		Define the scenes between which you want to switch when triggering the Toggles Scenes EventIn. The scenes are defined in the Content tab - Scene Editor, a maximum of 16 scenes with 0 as the first scene. Use the numbers of the scenes, separated by commas. Valid are numbers between -1 and 15; the number "-1" switches the group off. The order of the numbers defines the sequence of the scenes. Example: "0, 1, 2, -1". Go to scene 0, go to scene 1, go to scene 2 and switch the group off.



Send DAPC only on Luminance changes	<p>Enable to use direct Arc Power Control (DAPC). DAPC commands will only be sent for changed Luminance values. If there are no changes on the Luminance input, the same Luminance value is not sent to the ballast.</p> <p>Disabled: Every incoming Luminance value causes a DAPC to be sent. This can stress the system.</p>
-------------------------------------	--

In the following list, all PortIns, PortOuts EventIns and EventOuts are explained. Please note that the “light value” includes the luminance, hue, saturation, and temperature as applicable to the device type of the group. Please note as well that most of the EventIns are only applicable for DALI DT8 devices:

Port	Description
Luminance	A passively received double value for the light intensity in percent. When defined as Property, configure a value between 0 and 100.
Hue	A passively received double value for the hue in degrees. When defined as Property, configure a value between 0 and 360. Only relevant for DT8 or a group containing a Channel R ballast, a Channel G ballast, and a Channel B ballast (Setup Tab - Properties of a virtual ballast: Color Type).
Saturation	A passively received double value for the saturation in percent. When defined as Property, configure a value between 0 and 100. Only relevant for DT8 or a group containing a Channel R ballast, a Channel G ballast, and a Channel B ballast (Setup Tab - Properties of a virtual ballast: Color Type).
Temperature	A passively received Int64 value for the color temperature in Kelvin. When defined as Property, configure a value between 2,000 and 20,000 in Kelvin. Only relevant for DT8 or a group containing a Channel Warm ballast and a Channel Cold ballast (Setup Tab - Properties of a virtual ballast: Color Type).



Scene Number	<p>A passively received Int64 value for the scene number. The scenes are defined in the Content tab - Scene Editor, a maximum of 16 scenes with 0 as the first scene and 15 as the last, or in the extended configuration window.</p> <p>When defined as Property, configure a value between 0 and 15 to select a specific scene.</p>
--------------	---



Fade Time

A passively received Int64 value for the fade time in seconds. A valid representative value needs to be received for the Block to execute the according fade time in seconds.

Seconds	Representative value
0	0
0.7	1
1	2
1.4	3
2	4
2.8	5
4	6
5.7	7
8	8
11.3	9
16	10
22.6	11
32	12
45.3	13
64	14
90.5	15

Please note: Using this PortIn overrules the fade time of the internal ballast settings (Setup Tab - Properties of the virtual ballast).

When defined as Port: If a ballast is a member of several groups, it can receive new Fade Time values when calling each group. The last received value is valid.

When defined as Property, configure a representative value. Only set to Property when all ballasts of the group belong to this group only.



Fade Rate

A passively received Int64 value for the fade rate in steps per second. A valid representative value needs to be received for the Block to execute the according fade rate in seconds.

Steps per seconds	Representative value
358	0
253	1
179	2
126	3
89	4
63	5
45	6
32	7
22	8
16	9
11	10
8	11
6	12
4	13
2.8	14

Please note: Using this PortIn overrules the fade rate of the internal ballast settings (Setup Tab - Properties of the virtual ballast).

When defined as Port: If a ballast is a member of several groups, it can receive new Fade Rate values when calling each group. The last received value is valid.

When defined as Property, configure a representative value. Only set to Property when all ballasts of the group belong to this group only

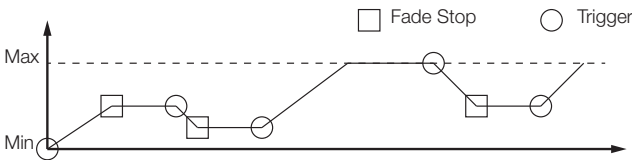
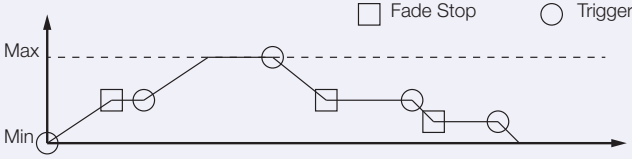
On

The group is switched on to the last active light value (value when the Off was triggered), respecting the Fade Time. When triggered after a system start, the maximum value (Setup Tab - Properties of the virtual ballast: Max Level) of the reference ballast is used.



Off	<p>Turns the group off.</p> <p>Please note that the group can not be turned on again using Toggle Fade triggers.</p>
Up	<p>Increases the light value up to the maximum value (Setup Tab - Properties of the virtual ballast: Max Level), honoring the Fade Rate of the group.</p>
Down	<p>Decreases the light value down to the minimum value (Setup Tab - Properties of the virtual ballast: Min Level), honoring the Fade Rate of the group. The group does not turn off as the minimum value cannot be zero.</p>
Step Up	<p>A fade up, honoring the Fade Rate starts for 300 ms. If retrigged during this period, another 300 ms fade continues. The maximum value is the Max Level.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>
Step Down	<p>A fade down, honoring the Fade Rate starts for 300 ms. If retrigged during this period, another 300 ms fade continues. The minimum value is the Min Level.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>
Step Down And Off	<p>A fade down, honoring the Fade Rate starts for 300 ms. If retrigged during this period, another 300 ms fade continues. Following the minimum value, the group is turned off.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>
On And Step Up	<p>The group is turned on and a fade up, honoring the Fade Rate starts for 300 ms. If retrigged during this period, another 300 ms fade continues.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>



Recall Last Active Level	The group is switched on to the last active light value (value when the Off was triggered), respecting the Fade Time. When triggered after a system start, the maximum value (Setup Tab - Properties of the virtual ballast: Max Level) of the reference ballast is used.
Recall Max Level	The group is switched to the Max Level value (Setup Tab - Properties of the virtual ballast: Max Level), respecting the Fade Time.
Recall Min Level	The group is switched to the Min Level value (Setup Tab - Properties of the virtual ballast: Min Level), respecting the Fade Time.
Toggle On / Off	Switches between turning the group on and off.
Toggle Fade Up / Down Start by Direction	<div></div> <p>The group fades until a fade stop (Toggle Fade Up / Down Stop) happens or Min / Max Level is reached. After another trigger, the fading changes direction until a fade stop comes or Min / Max Level is reached.</p>
Toggle Fade Up / Down Start by Limit	<div></div> <p>The group fades until a fade stop (Toggle Fade Up / Down Stop) happens or Min / Max Level is reached. After another trigger, the fading continues until Mix / Max Level or another fade stop is reached. The direction does not change.</p>
Toggle Fade Up / Down Stop	Stops the fading of Toggle Fade Up / Down Start by Direction or by Limit.

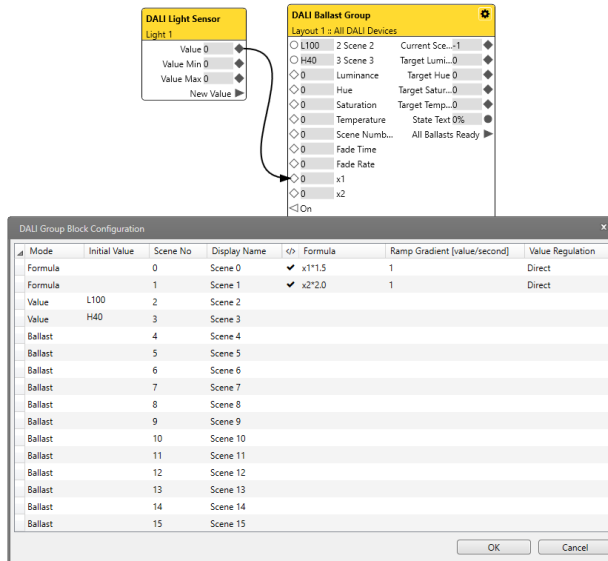
Toggle Scenes	Switches between the scenes that are defined in the Toggle Scene Numbers property.
Current Scene No	An actively sent Int64 value of the currently running scene number. The scenes are defined in the Content tab - Scene Editor, a maximum of 16 scenes with 0 as the first scene and 15 as the last. When light values are changed by another Port than Toggle Scenes or Scene Numb, e.g. Step Up, the current scene number is always "-1".
Target Luminance	An actively sent Int64 value of the target luminance value (in percent) to be reached or already in place by the reference ballast. Fades are not respected. 0 is off.
Target Hue	An actively sent Int64 value of the target hue value (between 0 and 360) to be reached or already in place. Fades are not respected.
Target Saturation	An actively sent Int64 value of the target saturation value (in percent) to be reached or already in place. Fades are not respected.
Target Temperature	An actively sent Int64 value of the target temperature value in Kelvin to be reached or already in place. Fades are not respected.
State Text	An actively sent string value of the group's state. When a scene is running, the state text equals the Display Name of the running scene number as defined in the extended configuration window. When the Current Scene No is "-1", the Target Luminance percentage is used.
All Ballasts Ready	Sends a trigger when the initializing is completed after a system start. Fires only once.



Defining scenes for Daylight Harvesting in the extended configuration window



SYMPHOLIGHT supports Daylight Harvesting for DALI devices. Daylight harvesting systems use daylight to reduce the amount of electric lighting needed to properly light a space, in order to reduce energy consumption. You will need a light sensor to measure the amount of natural light, the received value is used to control the intensity for the DALI Group. The formulas to calculate the target light values are to be defined in the extended

configuration window behind the cogwheel icon of the DALI Group block. The scenes defined are dynamic and replace the static scenes from the Content Tab.



Define scenes for Daylight Harvesting in the extended configuration window:

Column	Description
Mode	<p>Define the source of the value.</p> <p>By ballast: Define an initial value for all ballasts in the group in the column “Initial Value”. Only executed on the initial project start.</p> <p>By formula: Define a formula using the columns “</>”, “Formula”, “Ramp Gradient” and “Value Regulation”. Executed as soon as the formula has a valid value.</p> <p>By value: The scene value is available as a Port PortIn. The value can also be set from within SymphoCity (MQTT). Define an initial value in the column “Initial Value”. Only executed when a scene is called.</p>

Initial Value	<p>Define an initial value.</p> <p>For a luminance value, put an "L" in front of the value.</p> <p>For a hue value, put an "H" in front of the value.</p> <p>For a saturation value, put an "S" in front of the value.</p> <p>For a temperature value, put an "T" in front of the value.</p> <p>For a combined value (LTHS) use commas.</p> <p>Is the initial value empty, then no changes happen on selecting the scene.</p> <p>Format:</p> <p>The port accepts Luminance (L), Temperature (T), Hue (H), Saturation (S) values.</p> <p>Place a letter in front of the value to define its type, e.g. "S70".</p> <p>For decimal places use a point ".", e.g. "L50.4".</p> <p>For setting multiple values (LTHS) use commas, e.g. "L50,H30".</p> <p>Enter up to four numbers without preceding letters and they will be automatically be assigned to the LTHS order, e.g. "50,2500" -> "L50,T2500".</p> <p>If one of the values should not be set in a numbers only entry, use an asterisk, e.g. "50,*,30,70" -> "L50,H30,S70".</p>
Scene No	<p>The scene number, ranging between 0 and 15. The scenes are otherwise defined in the Scene Editor of the Content Tab.</p>
Display Name	<p>Define a name for the Daylight Harvesting scene. The Display Name is forwarded in the State Text PortOut when the scene is running.</p>
</>	<p>Gives information about the Formula:</p> <p> : The Formula is valid. PortOuts for the variables of the Formula are created and the scene with the Formula are executed when selected.</p> <p> : The Formula is erroneous. PortOuts for the variables of the Formula are not created. When enabled and the scene selected, the Formula is not executed.</p>




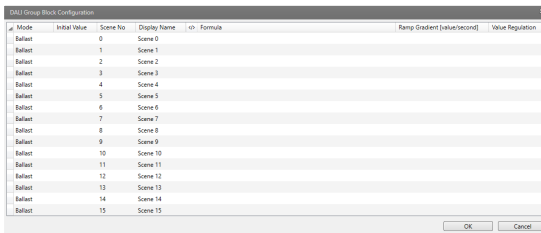
Formula Define the mathematical function for Daylight Harvesting. All mathematical functions, see „[Available mathematical functions to use in the Formula are:](#)“ on page 198, are valid. Every variable has to start with “x” followed by a number, e.g. x1. Further letters are possible, e.g. x1Ceiling. The same variable can be used in multiple scenes.

Ramp Gradient When “RampByGradient” in Value Regulation is selected, define a luminance value (percent) per second. The higher the Ramp Gradient value, the faster the target light values will be reached. Please be aware that this might stress the DALI bus.

Value Regulation Define the way of dimming the Daylight Harvesting scene:
Direct: The target light values will be set with the Fade Time configured in the ballasts of the group.
RampByGradient: A fading time longer than the maximum duration of 90 ms is possible using RampByGradient. SYMPHOLIGHT is calculating immediate steps, considering the Ramp Gradient in (percent) values per second, until the target light values are reached.



1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



2. Define the Modes.
3. When set to “Value”, define the Values.
4. When set to “Formula”, define the Formulas. Use variables representing PortIns. Variable names have to start with x and a following number. If not correct, a warning icon instead of a check mark appears. The scene is still enabled but is neither executed nor new PortIns created.
5. Define the Value Regulation as RampByGradient for a fading longer than 90 ms. If the Value Regulation is defined as RampByGradient, define the Ramp Gradient.
6. Enable scenes as desired by clicking the Enabled check box. For each enabled variable a double PortIn is created.

7. Click the “OK” button to save the configuration and close the window.

The DALI Group block has now the Daylight Harvesting scenes configured and the PortIns for each enabled variable accordingly available:

Category	Description
x1	<p>Additional PortIn for controlling the light value, created in the extended configuration window for Daylight Harvesting. Use variables starting with x and a following number in the formulas to create additional inputs.</p> <p>When defined as Property, configure the data type (double or Int64) and the value.</p>
0 Scene 0	<p>Additional PortIn for controlling the light value of a scene whose mode is set to “Value” in the extended configuration window. Here scene number 0.</p> <p>Format:</p> <p>The port accepts Luminance (L), Temperature (T), Hue (H), Saturation (S) values.</p> <p>Place a letter in front of the value to define its type, e.g. “S70”.</p> <p>For decimal places use a point “.”, e.g. “L50.4”.</p> <p>For setting multiple values (LTHS) use commas, e.g. “L50,H30”.</p> <p>Enter up to four numbers without preceding letters and they will be automatically be assigned to the LTHS order, e.g. “50,2500” -> “L50,T2500”.</p> <p>If one of the values should not be set in a numbers only entry, use an asterisk, e.g. “50,*,30,70” -> “L50,H30,S70”.</p>

- The PortOuts for light values are not updated as long as the group consists of only virtual devices and no real devices are matched.
- SYMPHOLIGHT only processes changes in values. Has the same value been incoming, a second calculation or processing does not take place; the first processed value is still valid.
- The Target Luminance might vary for decimal places from the Luminance PortIn depending on the chosen DALI settings. The Target Luminance is the exact technical value the DALI system has produced at the Reference Ballast.
- For Hue values, relate to the following gradient:



- In Daylight Harvesting, the same variable can be used in multiple scenes. The Current Scene No determines which formula the variable is allocated to and the result is accordingly calculated.
- Please configure all DALI ballast of one group identical at the following properties (Setup Tab - Properties of the virtual ballasts: DALI Settings): Fade Rate, Fade Time, Min Level, Max Level, and Dimming Curve. Otherwise irregular light values within the group will occur. Further take care to only group ballast of compatible types: Intensity and DT8; Channel Warm, Channel Cold, and Color Temperature DT8; Channel Red, Channel Green, Channel Blue, Color RGB DT8, and Color RGBW DT8.
- Please consider that the fading process is regulated by the reference ballast.
- Imitate an RGB DT8 ballast by configuring three ballasts with Channel Red, Channel Green, and Channel Blue respectively.
- Imitate a Temperature DT8 ballast by configuring two ballasts with Channel Cold and Channel Warm respectively.
- When ballasts are configured for DT8 or its equivalent by imitation, the PortOuts for Target Hue, Target Saturation, and / or Target Temperature are set to the respective minimum values.
- Please consider that enabled scenes from the extended configuration window of the DALI Group block overrule scene values defined in the Content Tab or in the ballasts of the fixtures.
- Please note that the Reference Ballast has to be online for the Block to be fully testable.
- When scenes values out of range are used, the corresponding max or min value is used.
- Scene value changes are persisted. After SYMPHOLIGHT is restarted, these values are still available. Initial values remain in the extended configuration window.
- Current ("cache") and initial values are saved in *.hfp project file. When exporting the project file to another Sympholight instance, the current scene values are also transferred.

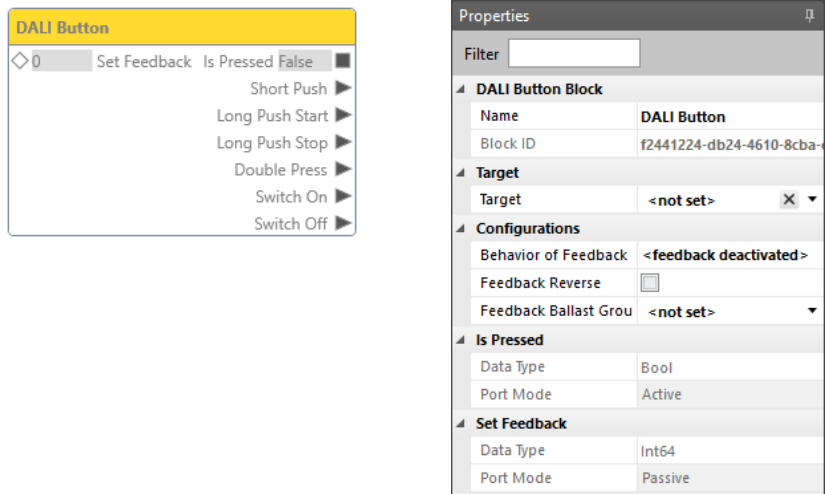
Implementing Daylight Harvesting behavior:

- Add a light sensor to the project.
- Add a DALI Light Sensor block to the Workflow.
- Add a real and a virtual DALI ballast to the project (necessary: License dongle, SYMPL dali Node) and match them.
- Add a DALI Group block to the Workflow.
- Define a scene in the extended configuration window for Daylight Harvesting, enable it, and click OK.
- Connect the sensor's Light Value PortOut to the new PortIn for the Daylight Harvesting scene, here x1.

The implementation of Daylight Harvesting is completed.



11.15.50 Button



Use this Block to include DALI Buttons in the Workflow Designer, e.g. to switch on/off lights based on button presses.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has an Is Pressed PortOut and six EventOuts.

The Block can be bound to a DALI Button (Fixture Library in Setup Tab: Generic and Osram).

- Define the Target.
- Define the Behavior of Feedback.
- Configure the DALI Button in the Setup Tab.

Category	Description
Target	Define the target to bind the Block to a DALI Button.

Behavior of Feedback	<p>Define how the Block should get feedback:</p> <p><feedback deactivated>: The Feedback function is disabled and the Feedback channel is off.</p> <p>By DALI Ballast Group: The feedback is defined by the luminance status or the scene value of the reference ballast of a DALI Ballast Group. Set the DALI Group at "Feedback Ballast Group". Target luminance values > 0 activate the feedback channel.</p> <p>By Button Switch On/Off Event: Only for a DALI Button configured as a switch. On a "Switch On" command, the feedback is activated. On a "Switch Off" command, the feedback is stopped.</p> <p>By Feedback In Port: For using the "Set Feedback" PortIn. Incoming values != 0 activate the feedback channel. Incoming value= 0 deactivates feedback.</p>
Feedback Reverse	<p>Enable to inverse the Feedback commands: on an "activate feedback" command, the feedback is stopped. On a "stop feedback" command, the feedback is activated.</p>
Feedback Ballast Group	<p>Only for "Behavior of Feedback - By DALI Ballast Group". Define a DALI ballast group that has an associated (i.e. target based) DALI Group Block in the Workflow and whose reference ballast and its luminance or scene number is to determine the feedback status..</p>
Is Pressed	<p>An actively sent bool value whether the bound target is being pressed or not.</p>
Set Feedback	<p>Only for "Behavior of Feedback - By Feedback In Port". A passively received Int64 value to send a value to the Feedback channel of the DALI button.</p> <p>When the incoming value is != 0, the feedback is on (= active).</p> <p>When the incoming value is =0, the feedback is off (= stopped).</p> <p>Note that there are no parameter validity checks.</p>



Event Port	Description
Short Push	Sends a trigger after a short push of the button. Short push: a single push and release within a period that is shorter than the defined Short Timer period.
Long Push Start	Sends a trigger after the long push start of the button. Long push start: a single push that exceeds the defined Short Timer period.
Long Push Stop	Sends a trigger after the long push stop of the button. Long push stop: a release after a single long push.
Double Press	Sends a trigger after a double press of the button. Double press: two push releases within a period that is shorter than the defined Double Timer period.
Switch On	Fires when the state is TRUE.
Switch Off	Fires when the state is FALSE.

- In order to receive Double Press events, you need to enable the event and to set a valid value > Double Timer Min for the Double Timer. The Double Timer min is Device specific. Refer to the documentation of the button.
- Connect the DALI Button block to the Toggle, a Single Delay or a Multi Delay block to further elaborate the handling of DALI Button events.
- Osram DALI Button: You can configure the Osram DALI Button and its Block either for push buttons sending Push and Press commands OR for switches using Switch behavior. The Block always shows all EventOuts, although some are idle due to this configuration. Define the configuration in the Setup Tab: In the Properties of the DALI Button, disable “Plug and Play”. You can now select three different Press events for push buttons OR the Switch behavior. SYMPHOLIGHT prevents you to select from both types.
- The block does not check if the associated input device offers button signaling. The DALI device ignores commands that are not implemented for ACTIVATE_FEEDBACK or STOP_FEEDBACK.
- All configurations are executed on the next incoming command.
- For the SYMPHO DALI2 Control Panels (e.g. SYMPHO DALI2 Panel 4-Key): each button owns two Feedback channel instances (two LEDs of different color). Only one color is addressed by the DALI Button Block. To access the second color / LED, see the User

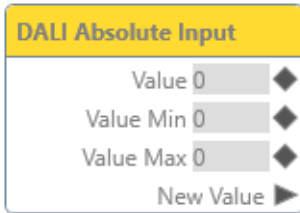


Instructions of the product for the LED's instance number. Then use the DALI Feedback Channel Line Block with the LED's address and instance number.

- For a comprehensive overview over DALI feedback functions in SYMPHOLIGHT see „[17 DALI Feedback Function](#)“ on page 461.



11.15.51 Absolute Input



Properties	
Filter	<input type="text"/>
DALI Absolute Input Block	
Name	DALI Absolute Input
Block ID	089fb766-74f3-4653-9721-9
Target	
Target	<not set> X ▾
Configurations	
Value in Percent	<input checked="" type="checkbox"/>
Only Value Changes	<input checked="" type="checkbox"/>
Show Event Value	<input type="checkbox"/>
Value	
Data Type	Double
Port Mode	Active
Value Min	
Data Type	Double
Port Mode	Active
Value Max	
Data Type	Double
Port Mode	Active

Use this Block to include DALI Absolute Inputs in the Workflow Designer, e.g. to get values from absolute instances.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has three PortOuts for light values and one EventOut for new value triggers.

The Block can be bound to a DALI Absolute Input (Fixture Library in Setup Tab: Generic).

- Define the Target.
- Configure the DALI Absolute Input in the Setup Tab.

Category	Description
----------	-------------

Target	Define the target to bind the Block to a DALI Absolute Input.
--------	---

Value in Percent	Enable to show the Value, the Value Min, and the Value Max in percent. The percentages are based on the resolution, predetermined by the device. Disabled, the PortOut "Resolution" is created.
------------------	--

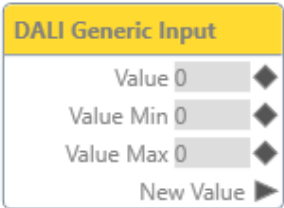


Only Value Changes	Enable to fire events at the New Value EventOut only on value changes. Disabled, events are fired every time a value was received.
Show Event Value	Enable to create the PortOut "Event Value".
Event Value	An actively sent Int64 value of the DALI Absolute Input. The value is actively sent by the device (event based). The original bit value (max. 10 bits).
Value	An actively sent double value of the DALI Absolute Input. The value is actively sent by the device (event based). Scaled to the resolution (max. 24 bits) where applicable.
Value Min	An actively sent double value of the minimum value. Defined by the device.
Value Max	An actively sent double value of the maximum value. Defined by the device.
Resolution	An actively sent Int64 value of the resolution. Defined by the device. Created, when the PortOut "Value in Percent" is disabled.

- The New Value EventOut is sent after all value PortOuts have been updated/ sent. It is recommended to use this trigger e.g. to start a script.
- Event based values are output twice: at the PortOuts Event Value and Value.
- It is recommended to use the Event Value PortOut only for binary defined states/ analyses (protocol near).



11.15.52 Generic Input



Properties

Filter

DALI Generic Input Block

Name DALI Generic Input

Block ID 254fb6e1-f48a-4f6a-a1ec-c

Target

Target <not set> X

Configurations

Value in Percent ☒

Only Value Changes ☒

Show Event Value ☐

Value

Data Type Double

Port Mode Active

Value Min

Data Type Double

Port Mode Active

Value Max

Data Type Double

Port Mode Active

Use this Block to include DALI Generic Inputs in the Workflow Designer, e.g. to get values from generic instances.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has three PortOuts for light values and one EventOut for new value triggers.

The Block can be bound to a DALI Generic Input (Fixture Library in Setup Tab: Generic).

- Define the Target.
- Configure the DALI Generic Input in the Setup Tab.

Category	Description
Target	Define the target to bind the Block to a DALI Generic Input.
Value in Percent	Enable to show the Value, the Value Min, and the Value Max in percent. The percentages are based on the resolution, predetermined by the device. Disabled, the PortOut "Resolution" is created.

Only Value Changes	Enable to fire events at the New Value EventOut only on value changes. Disabled, events are fired every time a value was received.
Show Event Value	Enable to create the PortOut "Event Value".
Event Value	An actively sent Int64 value of the DALI Generic Input. The value is actively sent by the device (event based). The original bit value (max. 10 bits).
Value	<p>An actively sent double value of the DALI Generic Input. Depending on the predetermined device configuration (whether it sends events automatically or not), this is either an event based value or a polled value.</p> <p>Event based: The value is actively sent by the device. Scaled to the resolution (max. 24 bits) where applicable.</p> <p>Polled: If the device does not send events automatically, set the Value Poll Rate*) of the device in the Setup Window - Properties Window > 0. Unscaled, in max. resolution.</p>
Value Min	An actively sent double value of the minimum value. Defined by the device.
Value Max	An actively sent double value of the maximum value. Defined by the device.
Resolution	An actively sent Int64 value of the resolution. Defined by the device. Created, when the PortOut "Value in Percent" is disabled.

*) It cannot be guaranteed that the poll rate is executed at the defined time. The poll rate is the time which has to pass at least before the next poll inquiry is available at the earliest.

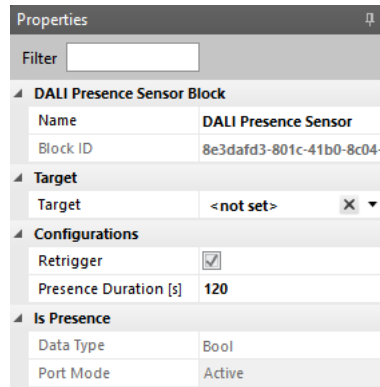
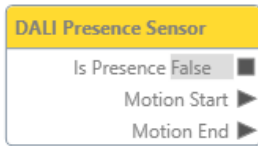
- The New Value EventOut is sent after all value PortOuts have been updated/ sent. It is recommended to use this trigger e.g. to start a script.
- When polling is required because the device does not send events automatically, the polled values are only output at the Value PortOut - not at the Event Value PortOut. It is recommended to use the Event Value Block here only in case of binary defined analyses (protocol near).
- Event based values are output twice: at the PortOuts Event Value and Value.



- It is recommended to use the Event Value PortOut only for binary defined states/analyses (protocol near).



11.15.53 Presence Sensor



Use this Block to include DALI presence detection sensors in the Workflow Designer, e.g. to switch on/off lights based on presence information.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has an Is Presence PortOut and two EventOuts for motion start and end points.

The Block can be bound to a DALI Presence Sensor (Fixture Library in Setup Tab: Generic and Osram).

- Define the Target.
- Configure the DALI Presence Sensor in the Setup Tab.

Category	Description
Target	Define the target to bind the Block to a DALI Presence Sensor.
Retrigger	Enable to trigger a Motion End event if no new Motion Start event occurs within the Presence Duration time.
Presence Duration	Define the duration time in seconds for the Retrigger property after which a Motion End event is sent if no new Motion Start event occurred.



Is Presence An actively sent bool value whether movement is detected or not. Is Presence value depends on the bound target.
TRUE between Motion Start and Motion End. FALSE otherwise.

- Motion Start fires when a motion is detected. Motion End fires when no more motion is detected in the Presence Duration and Retrigger is active.
- When presence is detected, a Motion Start gets fired and the Is Presence PortOut changes to TRUE.

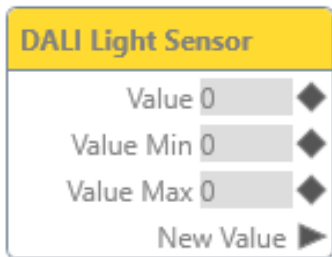
After the duration time has passed with no further presence detection, a Motion End gets fired and the Is Presence PortOut changes to FALSE.

If during the duration time another presence is detected (retriggered), the duration time is reset to countdown again from the beginning and the Is Presence PortOut remains TRUE.

- Take care that the recurrence of presence can be detected by the sensor's properties (Hold Timer, etc.) before the Presence Duration elapses. The Presence Duration value needs to be larger.
- For Generic DALI Presence Sensor: For the DALI Presence Sensor Block to work, the DALI Presence Sensor properties "Occupied Event" and "Movement Event" in the Setup Tab must be enabled. Also note, that the "Movement Event" only works when the "Occupied Event" is enabled as well.
- Is a greater variety of trigger options required, use the Presence Sensor Line Block where every DALI v2 trigger is available.



11.15.54 Light Sensor



Properties	
Filter	<input type="text"/>
DALI Light Sensor Block	
Name	DALI Light Sensor
Block ID	65af7be3-c8a5-4517-8865
Target	
Target	<not set> <input type="button" value="X"/>
Configurations	
Value	Percent
Only Value Changes	<input checked="" type="checkbox"/>
Value	
Data Type	Double
Port Mode	Active
Value Min	
Data Type	Double
Port Mode	Active
Value Max	
Data Type	Double
Port Mode	Active

Use this Block to process DALI light sensor values in the Workflow Designer, e.g. for Daylight Harvesting.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has three PortOuts for light values and one EventOut for new value triggers.

The Block can be bound to a DALI Light Sensor (Fixture Library in Setup Tab: Generic and Osram).

- Define the Target.
- Configure the DALI Light Sensor in the Setup Tab.

Category	Description
Target	Define the target to bind the Block to a DALI Light Sensor.



Value	<p>Select the output type of the value.</p> <p>Percent: Shows the Value, the Value Min, and the Value Max in percent. The percentages are based on the resolution, predetermined by the device.</p> <p>Absolute: Shows the values in bit. The PortOut “Resolution” is created.</p> <p>Lux: Shows the values in lux. The light values in lux are defined in the DALI Settings of the DALI Light Sensor's properties in the Setup Tab.</p>
-------	---

Only Value Changes	Enable to fire events at the New Value EventOut only on value changes. Disabled, events are fired every time a value was received.
--------------------	--

Value	An actively sent double value of the light value. The value is actively sent by the device (event based). Scaled to the resolution (max. 24 bits) where applicable.
-------	---

Value Min	An actively sent double value of the minimum value. Defined by the Properties of the DALI Light Sensor in the Setup Tab.
-----------	--

Value Max	An actively sent double value of the maximum value. Defined by the Properties of the DALI Light Sensor in the Setup Tab.
-----------	--

Resolution	An actively sent Int64 value of the resolution. Refers to the lux values of the sensor in the Properties, in the Setup Tab.
------------	---

- Define the DALI settings such as the sensor values in lux and the send intervals of the DALI Light Sensor in the Properties, in the Setup Tab. Enter the values as stated in the data sheet of the sensor.
- The New Value EventOut is sent after all value PortOuts have been updated/ sent. It is recommended to use this trigger e.g. to start a script.



11.15.55 Ballast Line

DALI Ballast Line

- ◊ 0 Luminance
- ◊ 0 Hue
- ◊ 0 Saturation
- ◊ 0 Temperature
- ◊ 0 Scene Num...
- ◊ 0 Fade Time
- ◊ 0 Fade Rate
- ⊗ ColorRgb Color Control Mode
- ⊗ Logarithmic Dimming Curve
- ⊗ Broadcast Address Type
- ◊ 0 Address
- ◁ Off
- ◁ Up
- ◁ Down
- ◁ Step Up
- ◁ Step Down
- ◁ Step Down And Off
- ◁ On And Step Up
- ◁ Recall Last Active Level...
- ◁ Recall Min Level
- ◁ Recall Max Level

Properties

Filter

▲ **DALI Ballast Line Block**

Name	DALI Ballast Line
Block ID	eb990ca5-c3cd-48ce-9ecf-5

▲ **Target**

Target	<not set> <input type="button" value="X"/> ▼
--------	--

▲ **Color Control Mode**

Data Type	DaliBallastColorControlMo
Input Type	Property
Value	RGB

▲ **Dimming Curve**

Data Type	DaliBallastDimmingCurveTy
Input Type	Property
Value	Logarithmic

▲ **Address Type**

Data Type	DaliBallastAddressType
Input Type	Property
Value	Broadcast

▲ **Address**

Data Type	Int64
Input Type	Port
Port Mode	Passive

▲ **Luminance**

Data Type	Double
Port Mode	Passive

▲ **Hue**

Data Type	Double
Port Mode	Passive

▲ **Saturation**

Data Type	Double
Port Mode	Passive

▲ **Temperature**

Data Type	Int64
Port Mode	Passive

▲ **Scene Number**

Data Type	Int64
Port Mode	Passive

▲ **Fade Time**

Data Type	Int64
Port Mode	Passive

▲ **Fade Rate**

Data Type	Int64
Port Mode	Passive



Use this block to control DALI ballasts of a line in the direct way by ballast short address, group address, and Broadcast. No matching of ballasts in the Setup Tab required.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has double PortIns for incoming properties of light and EventIns to trigger light values.

The Block can be bound to a DALI Line from the Setup Tab.

— Define the Target.

Category	Property	Description
Target		Define the target to bind the Block to a DALI Line.
Color Control Mode		Define the color control mode. By default, the value is set to the Input Type "Property" to manually define a static value. Set the Input Type to Port to passively receive an Int64 value from the counterpart Block. The respective valid numbers are described in the related Property "Value" below.
	Value	Select the color control mode: RGB: Generates RGB color DALI commands . When the Dimming Curve is set to Port, invoked by "0". RGBW: Generates RGBW color DALI commands. When the Dimming Curve is set to Port, invoked by "1".



Dimming Curve

Define the dimming mode for fading calculations. By default, the value is set to the Input Type "Property" to manually define a static value. Set the Input Type to Port to passively receive an Int64 value from the counterpart Block. The respective valid numbers are described in the related Property "Value" below.

Value	Select the dimming mode: Logarithmic: When the Dimming Curve is set to Port, invoked by "0". Linear: When the Dimming Curve is set to Port, invoked by "1".
-------	---

Address Type

Define the address mode of the Block. By default, the value is set to the Input Type "Property" to manually define a static value. Set the Input Type to Port to passively receive an Int64 value from the counterpart Block. The respective valid numbers are described in the related Property "Value" below.

Value	Select the address mode: Single: A single ballast is addressed. When the Address Type is set to Port, invoked by "0". Group: A group of ballasts is addressed. When the Address Type is set to Port, invoked by "1". Broadcast: All ballasts on the Line are addressed. When the Address Type is set to Port, invoked by "2".
-------	---



Address

A passively received Int64 value for the address of the ballast or the group.

When defined as Property, configure a value.

Value

Define an address. Dependent on the selected Address Type, enter the address for either a single ballast or a group:

For the Address Type "Single", enter a value between 0 and 63. For the Address Type "Group", enter a value between 0 and 15. For the Address Type "Broadcast" not applicable.

Luminance

A passively received double value for the light intensity in percent.

When defined as Property, configure a value between 0 and 100.

Hue

A passively received double value for the hue in degrees

When defined as Property, configure a value between 0 and 360.

Only relevant for DT8 or a group containing a Channel R ballast, a Channel G ballast, and a Channel B ballast (Setup Tab - Properties of a virtual ballast: Color Type).

Saturation

A passively received double value for the saturation in percent.

When defined as Property, configure a value between 0 and 100.

Only relevant for DT8 or a group containing a Channel R ballast, a Channel G ballast, and a Channel B ballast (Setup Tab - Properties of a virtual ballast: Color Type).



Temperature	<p>A passively received Int64 value for the color temperature in Kelvin.</p> <p>When defined as Property, configure a value between 2,000 and 20,000 in Kelvin.</p> <p>Only relevant for DT8 or a group containing a Channel Warm ballast and a Channel Cold ballast (Setup Tab - Properties of a virtual ballast: Color Type).</p>
Scene Number	<p>A passively received Int64 value for the scene number. The scenes are defined in the Content tab - Scene Editor, a maximum of 16 scenes with 0 as the first scene and 15 as the last.</p> <p>When defined as Property, configure a value between 0 and 15.</p>



Fade Time

A passively received Int64 value for the fade time in seconds.

Seconds	Representative value
0	0
0.7	1
1	2
1.4	3
2	4
2.8	5
4	6
5.7	7
8	8
11.3	9
16	10
22.6	11
32	12
45.3	13
64	14
90.5	15

Please note: Using this PortIn overrules the fade time of the internal ballast settings (Setup Tab - Properties of the virtual ballast).

When defined as Port: If a ballast is a member of several groups, it can receive new Fade Time values when calling each group. The last received value is valid.

When defined as Property, configure a representative value. Only set to Property when all ballasts of the group belong to this group only.



Fade Rate

A passively received Int64 value for the fade rate in steps per second.

Steps per seconds	Representative value
358	0
253	1
179	2
126	3
89	4
63	5
45	6
32	7
22	8
16	9
11	10
8	11
6	12
4	13
2.8	14

Please note: Using this PortIn overrules the fade rate of the internal ballast settings (Setup Tab - Properties of the virtual ballast).

When defined as Port: If a ballast is a member of several groups, it can receive new Fade Rate values when calling each group. The last received value is valid.

When defined as Property, configure a representative value. Only set to Property when all ballasts of the group belong to this group only.

The EventIns of the Block in detail:

Please note that the "light value" includes the luminance, hue, saturation, and temperature as applicable to the device type of the ballasts.



Event Port	Description
Off	<p>Turns the group off.</p> <p>Please note that the group can not be turned on again using Toggle Fade triggers.</p>
Up	<p>Increases the light value up to the maximum value (Setup Tab - Properties of the virtual ballast: Max Level), honoring the Fade Rate of the group.</p>
Down	<p>Decreases the light value down to the minimum value (Setup Tab - Properties of the virtual ballast: Min Level), honoring the Fade Rate of the group. The group does not turn off as the minimum value cannot be zero.</p>
Step Up	<p>A fade up, honoring the Fade Rate starts for 300 ms. If retrIGGERED during this period, another 300 ms fade continues. The maximum value is the Max Level.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>
Step Down	<p>A fade down, honoring the Fade Rate starts for 300 ms. If retrIGGERED during this period, another 300 ms fade continues. The minimum value is the Min Level.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>
Step Down And Off	<p>A fade down, honoring the Fade Rate starts for 300 ms. If retrIGGERED during this period, another 300 ms fade continues. Following the minimum value, the group is turned off.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>
On And Step Up	<p>The group is turned on and a fade up, honoring the Fade Rate starts for 300 ms. If retrIGGERED during this period, another 300 ms fade continues.</p> <p>For continuously fading please use the Toggle Fade Up/Down by Limit EventIn instead.</p>



Recall Last Active Level	The group is switched on to the last active light value (value when the Off was triggered), respecting the Fade Time. When triggered after a system start, the maximum value (Setup Tab - Properties of the virtual ballast: Max Level) of the reference ballast is used. Works only on DALI 2 ballasts!
Recall Min Level	The group is switched to the Min Level value (Setup Tab - Properties of the virtual ballast: Min Level), respecting the Fade Time.
Recall Max Level	The group is switched to the Max Level value (Setup Tab - Properties of the virtual ballast: Max Level), respecting the Fade Time.

- Please note that for linear dimming purposes DALI ballasts of DT6 with linear dimming curve support are required.
- Please note that for color control mode support DALI ballasts of DT8 Color Temperature are required. Or when changing Hue, Saturation or Temperature.
- Please use the DALI Line block only for DALI Lines that contains unmatched ballasts only to avoid controlling conflicts.
- Please note that short address or group address of unmatched ballasts will only be displayed in a later release of SYMPHOLIGHT.
- For Hue values, relate to the following gradient:



- For each DALI Line you can use as many target based DALI Ballast Line Blocks as needed to filter and select the inputs in various ways.
- To select and filter devices on the line, use the PortIns Address and Address Type (AND condition). The block then communicates exclusively with the filtered devices.



11.15.56 Generic Input Line

DALI Generic Input Line			
<input type="radio"/>	Address	Address	0
<input type="radio"/>	Instance No	Instance No	0
<input checked="" type="radio"/>	Instance Type	Instance Type	0
		Event Value	0
		Value	0
		Value Min	0
		Value Max	0
		Resolution	0
		New Value	▶

Properties	
Filter	<input type="text"/>
DALI Generic Input Line Block	
Name	DALI Generic Input Line
Block ID	489de389-6d98-468c-adb9
Target	
Target	<not set> <input type="button" value="X"/> ▼
Configurations	
Value in Percent	<input checked="" type="checkbox"/>
Only Value Changes	<input checked="" type="checkbox"/>
Address	
Data Type	String
Input Type	Property
Value	
Data Type	Int64
Port Mode	Active
Instance No	
Data Type	String
Input Type	Property
Value	
Data Type	Int64
Port Mode	Active
Instance Type	
Data Type	DaliInputInstanceTypeFile
Input Type	Property
Value	All Instance Types (32)
Data Type	Int64
Port Mode	Active
Event Value	
Data Type	Int64
Port Mode	Active
Value	
Data Type	Double
Port Mode	Active
Value Min	
Data Type	Double
Port Mode	Active
Value Max	
Data Type	Double
Port Mode	Active
Resolution	
Data Type	Int64
Port Mode	Active



Use this Block to control selected DALI inputs of a line in the direct way by short address, instance number, and / or instance type.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available. The Block has three PortIns for the target configuration, eight PortOuts for the configuration information and light values and one EventOut for new value triggers. The Block can be bound to a DALI Line from the Setup Tab.

- Define the Target.
- Define the short address, instance number, and / or instance type to select inputs from the line.

Category	Property	Description
Target		Define the target to bind the Block to a DALI Line.
Value in Percent		Enable to show the Value, the Value Min, and the Value Max in percent. The percentages are based on the resolution, predetermined by the device.
Only Value Changes		Enable to fire events at the New Value EventOut only on value changes. Disabled, events are fired every time a value was received.



Address	<p>A string value for the short address(es) of the line.</p> <p>PortIn: To filter devices of the DALI Line to which the Block is bound by short address. Set the Input Type. When defined as Property, configure a value.</p> <p>PortOut: The short address of the device, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.</p>
Value	<p>Define a short address value between 0 and 63. Select multiple devices by comma separation. Leave empty to get all devices on the Line.</p>
Instance No	<p>A string value for the instance number(s) (=input instance of max. 32 instances inside the input device) of the line.</p> <p>PortIn: To filter instances of the DALI Line to which the Block is bound by instance number. Set the Input Type. When defined as Property, configure a value.</p> <p>PortOut: The instance, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.</p>
Value	<p>Define an instance number value between 0 and 31. Select multiple instances by comma separation. Leave empty to get all instances of a device.</p>



Instance Type	<p>An Int64 value for the instance type(s) (=instance type of the input instances inside the input device) of the line</p> <p>PortIn: To select instances of the DALI Line to which the Block is bound by instance type. Set the Input Type. When defined as Property, configure a value. When defined as Port, the receiving value has to be between 0 and 31. To get all instance types, the value 32 is required.</p> <p>PortOut: The instance type, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.</p>
Value	<p>Define an instance type. Select an instance type from the list.</p>
Event Value	<p>An actively sent Int64 value of a DALI Generic Input on the DALI line. The value is actively sent by the device (event based). The original bit value (max. 10 bits).</p>
Value	<p>An actively sent double value of a DALI Generic Input on the DALI line. Depending on the predetermined device configuration (whether it sends events automatically or not), this is either an event based value or a polled value.</p> <p>Event based: The value is actively sent by the device. Scaled to the resolution (max. 24 bits) where applicable.</p> <p>Polled: If the device does not send events automatically, set the Value Poll Rate of the device in the Setup Window - Properties Window > 0. Unscaled, in max. resolution.</p>



Value Min	An actively sent double value of the minimum value. Defined by the device.
-----------	--

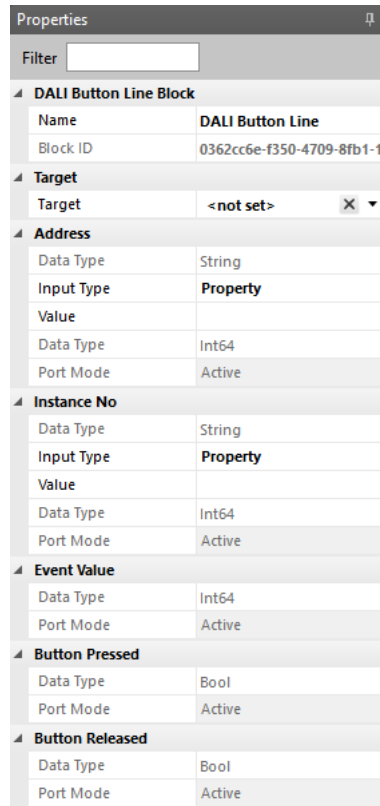
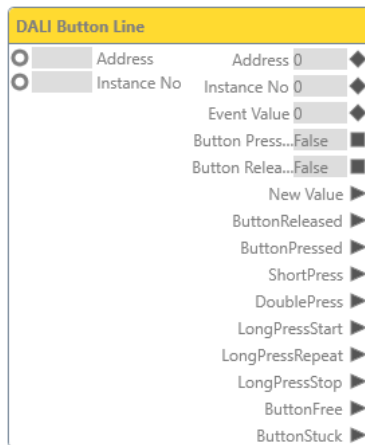
Value Max	An actively sent double value of the maximum value. Defined by the device.
-----------	--

Resolution	An actively sent Int64 value of the resolution. Defined by the device.
------------	--

- For each DALI Line you can use as many target based DALI Generic Input Line Blocks as needed to filter and select the inputs in various ways.
- To select and filter devices on the line, use the PortIns Address, Instance No, and Instance Type (AND condition). The block then communicates exclusively with the filtered devices.
- The New Value EventOut is sent after all value PortOuts have been updated/ sent. It is recommended to use this trigger e.g. to start a script.
- When polling is required because the device does not send events automatically, the polled values are only output at the Value PortOut - not at the Event Value PortOut. It is recommended to use the Event Value Block here only in case of binary defined analyses (protocol near).
- Event based values are output twice: at the PortOuts Event Value and Value.
- It is recommended to use the Event Value PortOut only for binary defined states/ analyses (protocol near).



11.15.57 Button Line



Use this Block to control DALI push button inputs of a line in the direct way by short address and instance number. Only for DALI v2 button input devices.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has two PortIns for the target configuration, five PortOuts for the configuration information and button states and eight EventOuts for button triggers. The Block can be bound to a DALI Line from the Setup Tab.

- Define the Target.
- Define the short address and / or instance number to select button inputs from the line.



↑
content

Category	Property	Description
Target		Define the target to bind the Block to a DALI Line.
Address		<p>A string value for the short address(es) of the line.</p> <p>PortIn: To filter devices of the DALI Line to which the Block is bound by short address. Set the Input Type. When defined as Property, configure a value.</p> <p>PortOut: The short address of the device, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.</p>
	Value	Define a short address value between 0 and 63. Select multiple devices by comma separation. Leave empty to get all devices on the Line.
Instance No		<p>A string value for the instance number(s) (=input instance of max. 32 instances inside the input device) of the line.</p> <p>PortIn: To filter instances of the DALI Line to which the Block is bound by instance number. Set the Input Type. When defined as Property, configure a value.</p> <p>PortOut: The instance, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.</p>
	Value	Define an instance number value between 0 and 31. Select multiple instances by comma separation. Leave empty to get all instances of a device.



Event Value	An actively sent Int64 value of a DALI Absolute Input on the DALI line. The value is actively sent by the device (event based). The original bit value (max. 10 bits).
Button Pressed	An actively sent bool value whether a button is being pressed or not, according to DALI2 standard. "ShortPress" is not supported.
Button Released	An actively sent bool value whether a button is being released or not, according to DALI2 standard.
Event Port	Description
New Value	Sends a trigger when any of the PortOuts value has changed, that is when the defined information occurs.
ButtonReleased	Sends a trigger when a button is released.
ButtonPressed	Sends a trigger when a button is pressed.
ShortPress	Sends a trigger after a short push of a button. Short push: a single push and release within a period that is shorter than the defined Short Timer period.
DoublePress	Sends a trigger after a double press of a button. Double press: two push releases within a period that is shorter than the defined Double Timer period.
LongPressStart	Sends a trigger after the long push start of a button. Long push start: a single push that exceeds the defined Short Timer period.
LongPressRepeat	Sends trigger repeatedly after a long push start event of a button each time the Repeat Timer period is exceeded as long as the button is not released.



LongPressStop Sends a trigger after the long push stop of a button.
Long push stop: a release after a single long push.

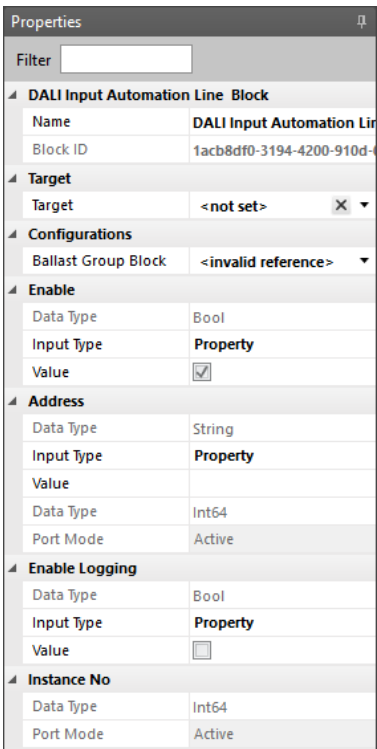
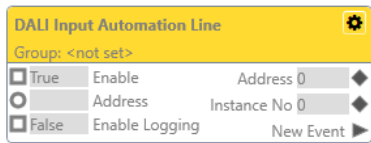
ButtonFree Sends a trigger when the button is released after a Button Stuck event.

ButtonStuck Sends a trigger after a very long push that will exceed the Stuck Timer period.

- For each DALI Line you can use as many target based DALI Button Line Blocks as needed to filter and select the inputs in various ways.
- To select and filter devices on the line, use the PortIns Address and Instance No (AND condition). The block then communicates exclusively with the filtered devices.
- The New Value EventOut is sent after all value PortOuts have been updated/ sent. It is recommended to use this trigger e.g. to start a script.
- It is recommended to use the Event Value PortOut only for binary defined states/ analyses (protocol near).
- The EventOuts are constructed according to the DALI standard.
- Events will be passed through from the button instance without consistency check: each device may send different signals as long as conform to the specification.
- You might need to enable events and set appropriate Timer values > 0 or > the corresponding Timer Min value which is device dependent. Refer to the device documentation.



11.15.58 Input Automation Line



Use this Block to control a DALI Group Block by inputs like buttons or sensors of a line in the direct way by input address / instance number and execute DALI group actions on incoming events. The Block receives all button instances events for a defined input device short address. This works similar to the DALI Button Line Block to filter events on a line and to externally control a DALI ballast group by DALI events.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has three PortIns for the target address and logging, two PortOuts for the address information and presence states and one EventOuts for new event triggers. Extended configuration is accessible at the cogwheel icon.

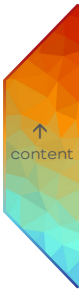
The Block can be bound to a DALI Line from the Setup Tab.



Prerequisite: DALI input instances must be recognized during a scan and present as real input instances.

- Define the Target.
- Define the DALI Ballast Group block.
- Define the short address to select inputs from the line.
- Define whether Block is enabled and whether its events are logged.
- Define triggers and conditions for feedback channels in the extended configuration window.

Category	Property	Description
Target		Define the target to bind the Block to a DALI Line.
Configurations		Define a DALI group block (on the target line) that will perform the event action commands.
Enable		A passively received bool value to dis- or enable the Block. When defined as Property, configure a value.
	Value	Define whether the block is enabled or disabled.



Address		<p>A string value for the short address(es) of the line.</p> <p>PortIn: To filter devices of the DALI Line to which the Block is bound by short address. Set the Input Type. When defined as Property, configure a value.</p> <p>PortOut: The short address of the device, that passed all filter conditions and has sent the last received and filtered input event.</p>
	Value	<p>Define a short address value between 0 and 63. Select multiple devices by comma separation. Leave empty to get all devices on the Line.</p>
Enable Logging		<p>A passively received bool value to dis- or enable the logging of event actions to the SYMPHOLIGHT log. When defined as Property, configure a value.</p>
	Value	<p>Define whether the logging is enabled or disabled.</p>
Instance No		<p>An Int64 value for the instance number(s) (=input instance of max. 32 instances inside the input device) of the line.</p> <p>PortOut: The instance, that passed all filter conditions and has sent the last received and filtered input event.</p>
	Value	<p>Define an instance number value between 0 and 31. Select multiple instances by comma separation. Leave empty to get all instances of the device.</p>




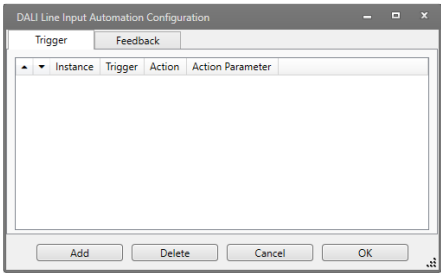
Event Port	Description
New Event	Sends a trigger when the defined trigger information occurs once the update of PortOut of the address and instance number is complete.

- The reference is to the DALI Ballast Group block, not a layout group. The associated target based DALI Ballast Group Block needs to be in the Workflow.
- When an input is received on the Instance Automation line, the DALI Ballast Group Block receives the port inputs (and MQTT commands) accordingly.
- Check if the DALI real device has the function enabled or configured in the Setup Tab.
- For each DALI button and input, include the virtual device in the project (not just the real device). Only then can the DALI configuration be saved.
- You can use multiple Input Automation Line Blocks for one DALI Ballast Group.
- The Block works directly with the DALI line. Therefore, there are no available instance names.

Defining triggers and feedback in the extended configuration window

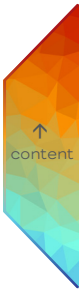
Define reactions on input events and feedback conditions for controlling the feedback channels by the assigned DALI Ballast group block in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .

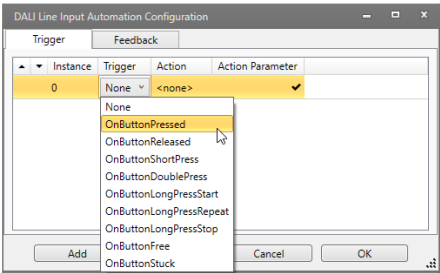


Configure reactions on input events in the Trigger tab.

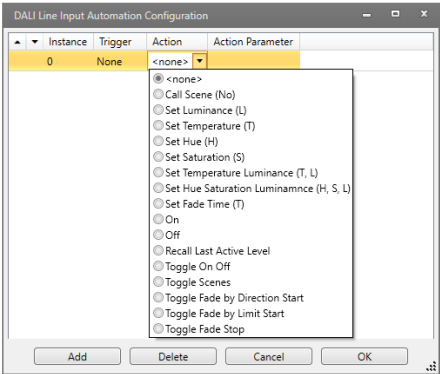
2. Add events and configure each entry:
3. Set an instance number of the instance that fires an event.



4. Define the trigger type (Trigger) that needs to be fired by the instance.



5. Define the reaction type upon the trigger (Action) that will be execute on the connected DALI Ballast Group block.



6. Define the reaction value (Action Parameter).

7. Click the “OK” button to save the configuration and close the window.

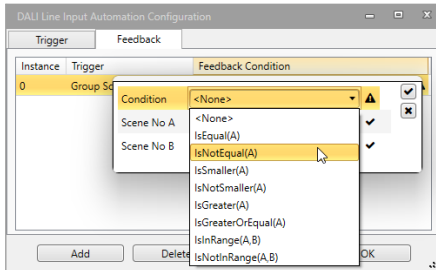
The DALI Input Automation Line Block now listens for DALI Line events. When an trigger that is defined in the table is fired, the assigned action will be executed (e.g. a scene gets called). The corresponding PortOut or EventOut of the associated DALI Ballast Group block indicates this action as well.

- Clicking on the “Action Parameter” cell in the table opens an Action Parameter dialog in the context of the currently selected Action.
- Duplicate entries: Select (multiple) entries. Clicking “Add” will copy & paste the selected entry(entries).
- Multiple commands for one instance get executed by order (top down). Change the order by using the arrows in left columns.

Configure feedback conditions for controlling the feedback channels by the assigned DALI Ballast Group block in the Feedback tab.

The block listens for changes in the DALI Ballast Group block value and compares it to the action parameter. It can consider state changes on DALI Group target values and sequence state changes. When the feedback condition is true, the feedback channel is activate; otherwise, it is deactivated.

8. Add events and configure each entry:



9. Set an instance number of the instance that fires an event.

10. Define the trigger type (Trigger) that needs to be fired by the instance

11. Define the feedback condition that, when met, will activate the feedback channel.

The DALI Input Automation Line Block now listens for DALI Line events. When a feedback condition that is defined in the table is met, the assigned instance will activate its feedback channel.

- Clicking on the “Feedback Condition” cell in the table opens an Feedback Condition dialog in the context of the currently selected Trigger.
- Duplicate entries: Select (multiple) entries. Clicking “Add” will copy & paste the selected entry(entries).
- When property “B” (e.g. “Play State B” or “Luminance B”) is not needed by the condition (condition does not include a “B”), then property “B” will still be visible (not be disabled or hidden). It will not have an impact, however



11.15.59 Presence Sensor Line

DALI Presence Sensor Line

<input type="radio"/>	Address	Address 0	◆
<input type="radio"/>	Instance No	Instance No 0	◆
	Event Value	0	◆
	Movement	False	■
	Vacant	False	■
	Still Vacant	False	■
	Occupied	False	■
	Still Occupied	False	■
	Presence Se...	False	■
	Movement S...	False	■
	New Value		▶
	No Movement		▶
	Movement		▶
	Vacant		▶
	Repeat		▶
	Occupied		▶

Properties

Filter

DALI Presence Sensor Line Block

Name	DALI Presence Sensor Line
Block ID	cd2cd6c5-b472-46dd-a9d5-

Target

Target	<not set> X ▼
--------	--

Address

Data Type	String
Input Type	Property
Value	
Data Type	Int64
Port Mode	Active

Instance No

Data Type	String
Input Type	Property
Value	
Data Type	Int64
Port Mode	Active

Event Value

Data Type	Int64
Port Mode	Active

Movement

Data Type	Bool
Port Mode	Active

Vacant

Data Type	Bool
Port Mode	Active

Still Vacant

Data Type	Bool
Port Mode	Active

Occupied

Data Type	Bool
Port Mode	Active

Still Occupied

Data Type	Bool
Port Mode	Active

Presence Sensor

Data Type	Bool
Port Mode	Active

Movement Sensor

Data Type	Bool
Port Mode	Active



Use this Block to control DALI occupancy inputs (presence sensors) of a line in the direct way by short address and instance number.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available. The Block has two PortIns for the target configuration, ten PortOuts for the configuration information and presence states and six EventOuts for presence triggers. The Block can be bound to a DALI Line from the Setup Tab.

- Define the Target.
- Define the short address and / or instance number to select presence sensor inputs from the line.

Category	Property	Description
Target		Define the target to bind the Block to a DALI Line.
Address		A string value for the short address(es) of the line. PortIn: To filter devices of the DALI Line to which the Block is bound by short address. Set the Input Type. When defined as Property, configure a value. PortOut: The short address of the device, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.
	Value	Define a short address value between 0 and 63. Select multiple devices by comma separation. Leave empty to get all devices on the Line.



Instance No	<p>A string value for the instance number(s) (=input instance of max. 32 instances inside the input device) of the line.</p> <p>PortIn: To filter instances of the DALI Line to which the Block is bound by instance number. Set the Input Type. When defined as Property, configure a value.</p> <p>PortOut: The instance, that passed all filter conditions and has sent the current data values of the other PortOuts of the Block.</p>
Value	Define an instance number value between 0 and 31. Select multiple instances by comma separation. Leave empty to get all instances of the device.
Event Value	An actively sent Int64 value of an DALI Absolute Input on the DALI line. The value is actively sent by the device (event based). The original bit value (max. 10 bits).
Movement	An actively sent bool value of "Movement" according to the DALI2 standard.
Vacant	An actively sent bool value of "Vacant" according to the DALI2 standard.
Still Vacant	An actively sent bool value of "Still vacant" according to the DALI2 standard.
Occupied	An actively sent bool value of "Occupied" according to the DALI2 standard.
Still Occupied	An actively sent bool value of "Still occupied" according to the DALI2 standard.



Presence Sensor

An actively sent bool value of “Presence sensor” according to the DALI2 standard.

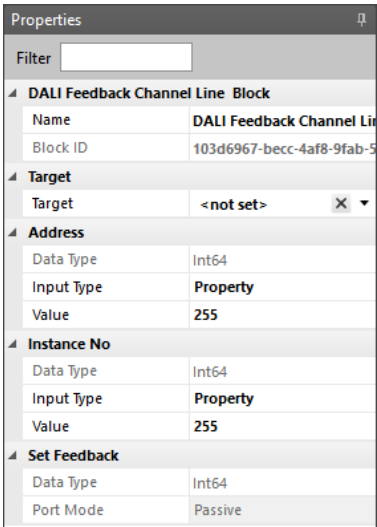
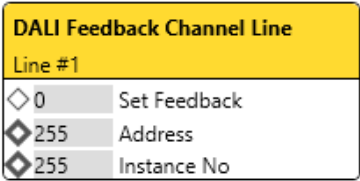
Movement Sensor

An actively sent bool value of “Movement sensor” according to the DALI2 standard.

- The Repeat EventOut is fired as soon as Still Occupied or Still Vacant is TRUE.
- The availability of DALI commands depends on the implementation by the manufacturer.
- For each DALI Line you can use as many target based DALI Presence Sensor Line Blocks as needed to filter and select the inputs in various ways.
- To select and filter devices on the line, use the PortIns Address and Instance No (AND condition). The block then communicates exclusively with the filtered devices.
- The New Value EventOut is sent after all value PortOuts have been updated/ sent. It is recommended to use this trigger e.g. to start a script.
- It is recommended to use the Event Value PortOut only for binary defined states/ analyses (protocol near).



11.15.60 Feedback Channel Line



Use this Block to control DALI feedback channels of a line in the direct way by input address and instance number. The Block sends DALI commands to the associated line, device and instance number(s). Only for DALI v2 Feedback functionality supporting devices.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has three PortIns for the feedback and the target configuration.

The Block can be bound to a DALI Line from the Setup Tab.

- Define the Target.
- Define the short address and / or instance number to select button inputs from the line.

Category	Property	Description
Target		Define the target to bind the Block to a DALI line.

Address	<p>An Int64 value for the short address(es) of the line.</p> <p>To filter devices of the DALI Line by short address. Set the Input Type. When defined as Property, configure a value.</p>
Value	<p>Define a short address value between 0 and 63. Select multiple devices by comma separation. Leave empty to get all devices on the Line.</p> <p>255 (initial value) - Feedback disabled</p>

Instance No	<p>An Int64 value for the instance number(s) (input instance of max. 32 instances inside the input device) of the line.</p> <p>To filter instances of the DALI Line by instance number. Set the Input Type. When defined as Property, configure a value.</p>
Value	<p>Define an instance number value between 0 and 31. Select multiple instances by comma separation. Leave empty to get all instances of a device.</p> <p>255 (initial value) - Feedback disabled</p>

Set Feedback	<p>A passively received Int64 value to send a value to the Feedback channel of the DALI button.</p> <p>When the incoming value is != 0, the feedback is on (= active).</p> <p>When the incoming value is =0, the feedback is off (= stopped).</p> <p>Note that there are no parameter validity checks.</p>
--------------	--

- Only use this Block for feedback channels with very special behavior that are not matched to a button instance.
- On sending feedback activate/stop command, there are no checks for valid parameters.
- You can use multiple blocks of the same target.



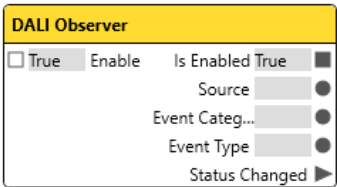
- All configurations are implemented on the next incoming command.
- For the SYMPHO DALI2 Control Panels (e.g. SYMPHO DALI2 Panel 4-Key): each button owns two Feedback channel instances (two LEDs of different color). Only one color is addressed by the DALI Button Block. To access the second color / LED, see the User Instructions of the product for the LED's instance number. Then use the DALI Feedback Channel Line Block with the LED's address and instance number.

Example: You want to use the second Feedback LED for Button #1 (Address Info: I 1 0) of a SYMPHO DALI2 Panel 4-Key.

- For a comprehensive overview over DALI feedback functions in SYMPHOLIGHT see „[17 DALI Feedback Function](#)“ on page 461.



11.15.61 DALI Observer



Properties	
Filter	
DALI Observer Block	
Name	DALI Observer
Block ID	8b7ca644-40e1-43da-ba4a-
Event Configuration	
Event Category	Any
Device Configuration	
Block Target	All DALI Devices
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Source	
Data Type	String
Port Mode	Active
Event Category	
Data Type	String
Port Mode	Active
Event Type	
Data Type	String
Port Mode	Active

Use this Block to process DALI information that can be collected from DALI fixtures.

The Block has one Enable PortIn, four PortOuts for the DALI information and one EventOut for changed status triggers.

The Block can be assigned to a target. Targets are virtual DALI fixtures.

- Define the DALI devices to listen to at the Block Target.
- Define the Event Configuration.



Category	Property	Description
Event Configuration		Define what type of information you want to retrieve from the DALI fixture. Set an Event Category and, where applicable, the information type.
	Event Category	Define the event type to which this Block should react to. The PortOut "Event Category" sends the chosen type: Any: All available event types are considered. Connection Type: Events of scanning the system. Warning Type: Events of minor negative impact. Error Type: Events of major negative impact.
	Connection Type	Define the information type for connection events. "Any" considers all available information types.
	Warning Type	Define the information type for warning events. "Any" considers all available information types.
	Error Type	Define the information type for error events. "Any" considers all available information types.



Device Configuration	Define the DALI targets.
Block Target	Define the selection mechanism for DALI fixtures and specify the DALI fixtures. All DALI Devices: All connected DALI fixtures. DALI Device: Listen to one single DALI fixture. DALI Group: Listen to a group of DALI fixtures.
DALI Device	Block Target is DALI Device: Select from the drop-down list of available DALI fixtures.
Layout	Block Target is DALI Group: Select from the drop-down list of available Layouts.
DALI Group	Block Target is DALI Group: Select from the drop-down list of available Groups.
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not.
Source	An actively sent string value of the DALI ballast's Display Name. The Display Name is configured in the Setup Tab - Properties of the virtual ballast: Device Information.
Event Category	An actively sent string value of the event category.
Event Type	An actively sent string value of the event type.



Port	Description
Status Changed	Sends a trigger when any of the PortOuts value has changed, that is when the defined information occurs.

- Please note that most information types are only discovered after a sent request of the DALI system. The timing of sending requests depends on the traffic and the amount of ballasts in the DALI system. It might take 10 minutes until a request is sent and an event information is received after happening. Missing LAN connectivity is, as exception, discovered and received immediately (Connection event - Device Lost).
- Use a Combine Text block to attach all string PortOuts and forward the combined information.



11.15.62 Build RDM Command



Properties	
Filter	<input type="text"/>
Build RDM command Block	
Name	Build RDM command
Block ID	4dec22b8-c920-4295-a552-4
Configurations	
Is event based	<input checked="" type="checkbox"/>
RDM Command Library	
Library	RDM Standard ▼
Command	Get Communication S ▼
RDM Command	
Data Type	Dynamic
Port Mode	Active

Use this Block to process RDM information that can be collected from RDM fixtures, e.g. temperatures.

Please consider the following preconditions:

- Before working with the RDM system in the Workflow Designer, do not forget to enable RDM for the used DMX port in the RDM subwindow at the button of the Setup window. See „9.10 RDM Devices“ on page 80 for details.

Use this Block to build simple RDM commands that can be send to RDM devices using the Send RDM Block or scripting.

The Block has initially one EventIn and one PortOut for the command.

- Define the Library of RDM commands.
- Define the Command.

Category	Property	Description
Is event based		Enables this block to be executed on event trigger.
RDM Command Library		Define the library and the specific RDM command from within the chosen library.
	Library	Select the library to choose the RDM command from.

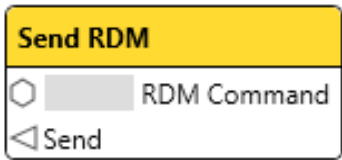


Command	Select the RDM command to create.
RDM Command	The actively sent dynamic command.

- For detail information about RDM and using RDM in SYMPHOLIGHT, please see „16 RDM in Detail“ on page 452.



11.15.63 Send RDM



Properties	
Filter	<input type="text"/>
Send RDM Block	
Name	Send RDM
Block ID	aa43766a-83b9-4b45-a286-
Configurations	
Is event based	<input checked="" type="checkbox"/>
Target Configuration	
Block Target	All Controllers
Destination UID	Broadcast (FFFF:FFFF) ▾
RDM Command	
Data Type	Dynamic
Port Mode	Passive

Use this Block to process RDM information that can be collected from RDM fixtures, e.g. temperatures.

Please consider the following preconditions:

- Before working with the RDM system in the Workflow Designer, do not forget to enable RDM for the used DMX port in the RDM subwindow at the button of the Setup window. See „9.10 RDM Devices“ on page 80 for details.

The Send RDM block is only functional if RDM is enabled in Setup.

Use this Block to send RDM commands to specified destination remote devices.

The Block has initially one EventIn and one PortIn for receiving the RDM command.

- Define the Target Configuration.



While working with the Send RDM block, we strongly suggest to configure the Block while the real fixture(s) is/are connected to the DMX port. While doing so, the vendor specific parameters are scanned from the fixture(s) directly and the RDM Parameters are filled with valid entries only.

Category	Property	Description
Is event based		Enables this block to be executed on event trigger.
Target Configuration		Define the library and the specific RDM command from within the chosen library.

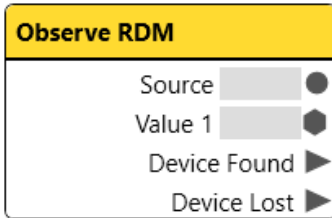


Block Target	Select the target type to send the RDM command.
Destination UID	The identifier of the device to be handled. Format: ManufacturerId:DeviceId:SubDeviceIndex E.g.: 45a3:1234567:1 or 45a3:1234567:65535 to perform SUB_DEVICE_ALL_CALL
Controller	Select the RDM controller to be used for sending of RDM requests.
Port	Select the RDM controller port to be used for sending of RDM requests.
RDM Command	The passively received dynamic command.

- For detail information about RDM and using RDM in SYMPHOLIGHT, please see „16 RDM in Detail“ on page 452.



11.15.64 Observe RDM



Properties	
Filter	<input type="text"/>
Observe RDM Block	
Name	Observe RDM
Block ID	4ad2bd46-b306-484c-884
Block Configuration	
Block Target	Any
Number Of Attributes	1
Parameter Configuration 1	
Parameter Text 1	
Parameter Identifier 1	<not set>
Attribute Identifier 1	<not set>
Source Of Value 1	Change
Condition Configuration 1	
Minimum Value 1	
Maximum Value 1	
Source	
Data Type	String
Port Mode	Active
Value 1	
Data Type	Dynamic
Port Mode	Active

Use this Block to process RDM information that can be collected from RDM fixtures, e.g. temperatures.

Please consider the following preconditions:

- Before working with the RDM system in the Workflow Designer, do not forget to enable RDM for the used DMX port in the RDM subwindow at the bottom of the Setup window. See „9.10 RDM Devices“ on page 80 for details.

The Observe RDM block is only functional if RDM is enabled in Setup.

The Block has initially four PortOuts for the RDM information.

The Block can be assigned to a target. Targets are real RDM fixtures that are connected to the project (see “Implementing RDM fixtures” the end of the chapter). When assigned to a target, more configuration options are available.

- Define the RDM fixtures to listen to at the Block Target.
- Define the Parameters.



While working with the Observe RDM block, we strongly suggest to configure the Block while the real fixture(s) is/are connected to the DMX port. While doing so, the vendor specific parameters are scanned from the fixture(s) directly and the RDM Parameters are filled with valid entries only.

Category	Property	Description
Block Configuration		Define the RDM targets and the number of available information represented by PortOuts.
	Number of Parameters	Define the number of parameters for the targets of the Block. Each information type is an attribute of a parameter.
	Block Target	Define the selection mechanism for RDM fixtures. Specify the RDM fixtures at Device Configuration. Any: All connected RDM fixtures. Single Device: Listen to one single RDM fixture. Layout Group: Listen to a group of RDM fixtures.



Device Configuration

Depending on the Block Target settings, either both Layout properties are to be defined or the Single Device Identifier. Specify the RDM fixtures to listen to.

Layout

Block Target is Layout Group: Select the Layout that contains the desired RDM fixtures. Prerequisite: real RDM fixtures have to be matched to virtual fixtures.

Layout Group

Block Target is Layout Group: Select a group within the Layout. Prerequisite: real RDM fixtures have to be matched to virtual fixtures.

Single Device Identifier

Block Target is Single Device: Select from the drop-down list of available RDM fixtures. Prerequisite: RDM has to be enabled in the Setup Tab.

Parameter Configuration

Each type of information is an attribute of a parameter. Set the Parameter Identifier first, then the Attribute Identifier.

Parameter Text

The Port description in the block. By default the Parameter Identifier name.

Parameter Identifier

Select the identifier of the RDM parameter from the drop-down list. Listed from the RDM scan of the devices. The list is also displayed in the Setup Tab: select the RDM fixture in the RDM pane. In the Properties window, Supported Parameters (Standard Parameters) lists the RDM parameters.

Attribute Identifier

Select the identifier of the attribute from the drop-down list. The list varies according to the selected parameter.



Condition	Some attributes need a minimum value and a maximum value to set a range for valid values. The unit of the values depend on the selected attribute. When defined, two additional EventOuts for the value limits are created.	
Configuration		
	Minimum Value	Define a minimum value. When the received value from the fixture is smaller, the Underflow EventOut is triggered.
	Maximum Value	Define a maximum value. When the received value from the fixture is larger, the Overflow EventOut is triggered.
Source	An actively sent string value of the RDM device name. The device name is displayed in the Setup Tab - RDM pane - Device; or Properties of the virtual device: Device Info - Device ID. Always the latest fixture that had sent a changed information.	
Value 1	An actively sent value of the data type dynamic containing the information about the RDM fixture as defined in the Parameter Configuration. The PortOut is named as defined at Parameter Text.	

Port	Description
Device Found	Sends a trigger when a new RDM fixture is discovered.
Device Lost	Sends a trigger when an RDM fixture has gone missing.

- It is recommended to preconfigure the Block with real devices connected.
- Please note that the available identifiers are the union of all RDM fixtures' parameters when the Block Target is "Any" or "Layout". To ensure identical information behavior, the RDM fixtures should be of the same device type, the same firmware version, and of the same RDM Personality.
- The Parameter Identifier are based on the RDM standard. For further information on the topic please refer to the standard. Also consult the manual of the RDM fixtures.



Implementing RDM fixtures:

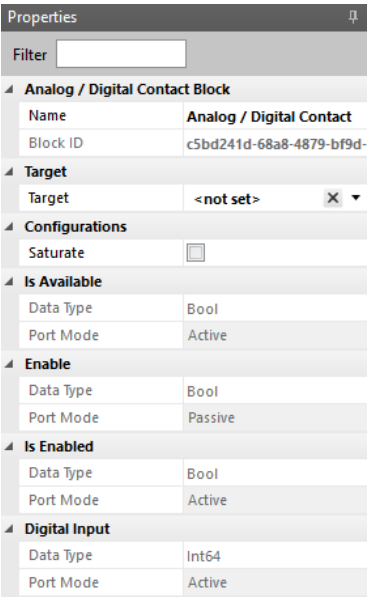
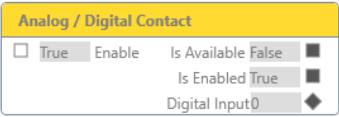
- You need a License dongle and a SYMPHOLIGHT project running in Full Mode.
- Add an RDM capable controller to the project: Butler S2, Butler XT2, Butler Pro, SYMPL dmx Node or SYMPL e:pix Node. Define the DMX universes at the Properties window - Address Mapping (Universe = Port).
- Add virtual RDM fixtures to the project (customized fixtures) and map DMX universes and addresses to them at their Properties.
- Connect the real RDM fixtures to the controller where applicable.
- Go to the RDM pane in the RDM / DALI window. Enable RDM for Ports of the controller in its context menu (right-click). A full heart icon indicates the enabled RDM. The scanning of RDM capable devices and retrieving of information from the fixtures starts (stethoscope icon). A list of discovered fixtures is added to each Port.
- The Device column displays the Device ID needed for Single Device target assigning of the Observe RDM block. You can change the Label (DeviceLabel) for additional identification. The label is stated before the device name in the Single Device Identifier of the Observe RDM block.
- Map DMX addresses to the RDM fixtures: Select an RDM fixture from the list. In the Properties window, define the DmxStartAddress. Use the Port / DMX universe and the DMX address of the virtual fixture you want to map the RDM fixture to. This step is necessary for SYMPHOLIGHT when using the Layout target assignment of the Observe RDM block.
- Define groups of RDM fixtures in the 3D canvas as desired; these groups are available for Layout target assignment. Consider the implications for parameter configuration when creating a heterogeneous group.

The implementation of RDM fixtures is completed.

- For more detail information about RDM and using RDM in SYMPHOLIGHT, please see [„16 RDM in Detail“ on page 452](#).



11.15.65 Analog / Digital Contact



Use this Block to react on the values of analog inputs of e.g. the SYMPL input Node and process these values in the Workflow Designer.

This Block is target based. The Target value is displayed as main setting in the Block's header and its type defines the Block title.

When initially orphan, the Block has an Enable PortIn, an Is Available PortOut, an Is Enabled PortOut, and one PortOut for a received digital input value. When orphan after being target based once, the Block keeps the target based Ports facilitating an exchange of targets.

The Block can be bound to a SYMPL input Node.

- Define the Target.

Category	Description
Target	Define the target to bind the Block to a SYMPL input Node.

Saturate	Enable to saturate underflow / overflow values to the Target value range. The Port values will be limited so that overflow and underflow values cannot exceed the Target value range.
----------	---

Is Available	An actively sent bool value whether the bound target is available or not. For SYMPL nodes, the condition is automatically TRUE when a real Node is matched.
--------------	---

Enable	A passively received bool value to dis- or enable the Block.
--------	--

Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
------------	---

Digital Input	An actively sent Int64 value that is composed from the digital representation of all input values. Base are the threshold settings in the input Nodes' properties in the Setup Tab.
---------------	---

- For each target value, the Analog Input properties in the Setup Tab need to be configured. Please define for every available Analog Input port the Digital Threshold Low, the Digital Threshold High and the Event Delta. Please consider that this configuration has influence on the Digital Input port of the Block as well as the visualization of the port LEDs on the SYMPL input Node.



SYMPL input Node based:

Analog / Digital Contact			
Input			
<input type="checkbox"/> True	Enable	Is Available	False
		Is Enabled	True
Digital Input 0			
Input 1 0			
Input 2 0			
Input 3 0			
Input 4 0			
Input 5 0			
Input 6 0			
Input 7 0			
Input 8 0			
Underflow Input 1			
Overflow Input 1			
Underflow Input 2			
Overflow Input 2			
Underflow Input 3			
Overflow Input 3			
Underflow Input 4			
Overflow Input 4			
Underflow Input 5			
Overflow Input 5			
Underflow Input 6			
Overflow Input 6			
Underflow Input 7			
Overflow Input 7			
Underflow Input 8			
Overflow Input 8			

Properties	
Filter	<input type="text"/>
Analog / Digital Contact Block	
Name	Analog / Digital Contact
Block ID	c5bd241d-68a8-4879-bf9
Target	
Target	Input <input type="button" value="X"/>
Configurations	
Saturate	<input type="checkbox"/>
Input 1	
Input 1 Minimum Source	0
Input 1 Maximum Source	10000
Input 1 Minimum Target	0
Input 1 Maximum Target	10
Data Type	Double
Port Mode	Active
Input 2	
Input 2 Minimum Source	0
...	
Input 8	
Input 8 Minimum Source	0
Input 8 Maximum Source	10000
Input 8 Minimum Target	0
Input 8 Maximum Target	10
Data Type	Double
Port Mode	Active
Is Available	
Data Type	Bool
Port Mode	Active
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Digital Input	
Data Type	Int64
Port Mode	Active



In addition to the properties of an orphan Analog / Digital Contact block, the SYMPL input Node based Block has eight PortOuts for the received input values and 16 EventOuts for overflow and underflow triggers.

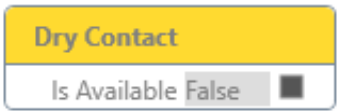
In addition to the properties of an orphan block:

- Define the Source Value range.
- Define the Target value range.

Category	Property	Description
Input		An actively sent double value that is received from the connected sensor.
	Input Minimum Source Value	The minimum allowed source value in millivolt. Any value below will fire an underflow event.
	Input Maximum Source Value	The maximum allowed source value in millivolt. Any value above will fire an overflow event.
	Input Minimum Target Value	The minimum allowed value in unit of the sensor. The minimum source value will be mapped to this value.
	Input Maximum Target Value	The maximum allowed value in the unit of the sensor. The maximum source value will be mapped to this value.



11.15.66 Dry Contact



Properties	
Filter	<input type="text"/>
Dry Contact Block	
Name	Dry Contact
Block ID	fdc49d85-74f6-43e9-a8ad-e
Target	
Target	<not set> <input type="button" value="X"/>
Is Available	
Data Type	Bool
Port Mode	Active

Use this Block to receive dry contact triggers from, e.g. the LCE3.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has an Is Available PortOut.

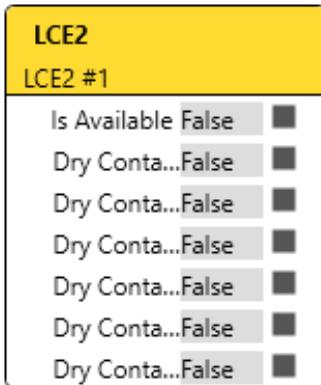
The Block can be bound to an LCE2 or LCE3 (see „LCE2 and LCE3 based:“ on page 331), a Butler XT, or a Butler XT2 (see „Butler XT and Butler XT2 based:“ on page 332).

- Define the Target.

Category	Description
Target	Define the target to bind the Block to: an LCE2, an LCE3, a Butler XT, or a Butler XT2.
Is Available	An actively sent bool value whether the bound target is available or not.



LCE2 and LCE3 based:



Properties	
Filter	
Dry Contact Block	
Display Name	LCE2
Block Position X	0
Block Position Y	0
Target	
Value	LCE2 #1
Is Available	
Data Type	Bool
Port Mode	Active
Dry Contact 1	
Data Type	Bool
Port Mode	Active
Dry Contact 2	
Data Type	Bool
Port Mode	Active
Dry Contact 3	
Data Type	Bool
Port Mode	Active
Dry Contact 4	
Data Type	Bool
Port Mode	Active
Dry Contact 5	
Data Type	Bool
Port Mode	Active
Dry Contact 6	
Data Type	Bool
Port Mode	Active

In addition to the properties of an orphan Dry Contact block, the LCE2 / LCE3 based Block has six PortOuts for six dry contacts.

- Connect the PortOuts to counterpart Blocks.

Category	Description
----------	-------------

Dry Contact	An actively sent bool value of the state of the dry contact (FALSE: low voltage or open; TRUE: high voltage).
-------------	---

Butler XT and Butler XT2 based:

The Butler XT based Block and the Butler XT2 based Block are identical to the LCE2 / LCE3 based Block, but with eight PortOuts for eight dry contacts.

- Connect the PortOuts to counterpart Blocks.



11.15.67 Relay

Relay

☐ True

Enable

Is Available

False

☐

Is Enabled

True

☐

Properties

Filter

Relay Block

Name

Relay

Block ID

266a8fcc-e78e-4ebb-b1ce-1

Target

Target

<not set>

X

Is Available

Data Type

Bool

Port Mode

Active

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Use this Block to control the relay outputs of e:cue devices, e.g. the SYMPL relay Node.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has an Enable PortIn, an Is Available PortOut, and an Is Enabled PortOut.

The Block can be bound to an LCE2, an LCE3, or a SYMPL relay Node. When target based, the Block has additionally EventIns for the relay controls:

Relay Output

LCE2

☐ False

Enable

Is Available

False

☐

Relay 1 NC

Is Enabled

False

☐

Relay 1 NO

Relay 2 NC

Relay 2 NO

— Define the Target Property.

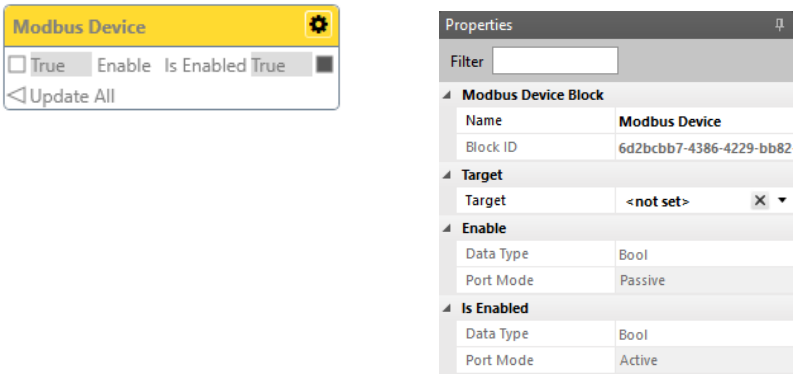
Category	Property	Description
Target		Define the target to bind the Block to an LCE2, an LCE3, or a SYMPL relay Node.

Is Available	An actively sent bool value whether the bound target is available or not.
Enable	A passively received bool value to dis- or enable the Block.

- Is Enabled
- An actively sent bool value whether the Block is enabled or not at the “Enable” PortIn.
- Trigger the NC EventIn to close the NC contact and trigger the NO EventIn to close the NO contact.



11.15.68 Modbus



Use this Block to communicate with an external mapped Modbus device.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When initially orphan, the Block has an Enable PortIn, an Update All EventIn, and an Is Enabled PortOut.

The Block has bool PortIns for the relays and Int64 PortOuts for the measured data as defined in the extended configuration window. Extended configuration is accessible at the cogwheel icon. When orphan after being configured once, the Block keeps the configured Ports facilitating an exchange of targets.

The Block can be bound to a Serial Modbus Device.


- Define the Target.
- Define and configure the PortIns and PortOuts for each Modbus device port in the extended configuration window.

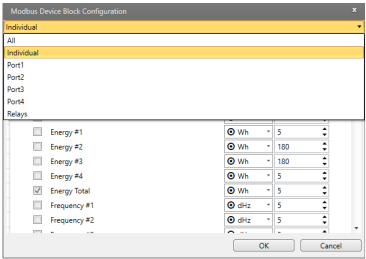
Category	Description
Target	Define the target to bind the Block to a Serial Modbus Device.
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.



Defining PortIns and PortOuts for each Modbus device port in the extended configuration window

Define and configure the PortIns and PortOuts for each Modbus device port in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



2. Select which Modbus device ports and data are available as PortIns or PortOuts: Define at the drop-down menu at the top which ports are pre-selected.

All: All entries are enabled.

Individual: Manually select entries.

Port#: All Modbus device port no.# related entries are enabled.

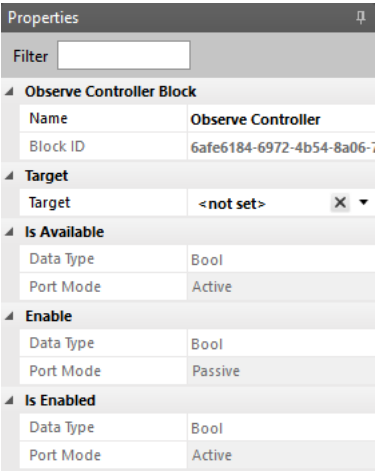
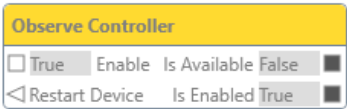
Relays: The cumulated current, total energy and power , and all “Power Active” and “Status” entries are enabled.

3. You can adjust the Units and the Refresh Rate of each entry. To disable internal value reading from the Modbus device, set the Refresh Rate to 0.
4. Click the “OK” button to save the configuration and close the window.

The Modbus block has now the PortIns and PortOuts for each Modbus device port configured and accordingly available.



11.15.69 Observe Controller



Use this Block to control and fetch information from e:net devices.

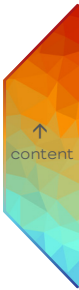
This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has an Enable PortIn, one EventIn, an Is Available PortOut and an Is Enabled PortOut.

The Block can be bound to a controller.

- Define the Target.

Category	Property	Description
Target		Define the target controller to bind the Block to.
Is Available		An actively sent bool value whether the bound target is available or not.
Enable		A passively received bool value to dis- or enable the Block.



Is Enabled

An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.

- For each target controller the Block is bound to different PortOuts are available. They do not need configuration.



11.15.70 Observe System

Observe System	
Instance Name	ecueT...
Model Name	ecue...
Software Version	5.5 (5...
License Level Restriction	
Is Muted	False
Engine FPS	30
Uptime	3292
CPU Load	0.13
Memory Usage	0.7267
Active Network	172.2...
Project Name	
Project Save	
Total Controller	0
Online Controller	0
Offline Controller	0
Sequences Total	2
Sequences Running	0
Last Offline	
Last Online	
Is Backup Master	False
Is Backup Slave	False

Properties	
Filter	
Observe System Block	
Name	Observe System
Block ID	14523f92-cec0-4adb-9eb1
Instance Name	
Data Type	String
Port Mode	Active
Model Name	
Data Type	String
Port Mode	Active
Software Version	
Data Type	String
Port Mode	Active
License Level	
Data Type	String
Port Mode	Active
Is Muted	
Data Type	Bool
Port Mode	Active

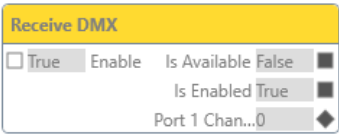
Sequences Running	
Data Type	Int64
Port Mode	Active
Last Offline Controller	
Data Type	String
Port Mode	Active
Last Online Controller	
Data Type	String
Port Mode	Active
Is Backup Master	
Data Type	Bool
Port Mode	Active
Is Backup Slave	
Data Type	Bool
Port Mode	Active

Use this Block to retrieve various information about the running instance and the project.

The Block has 20 PortOuts. All of them are fixed.



11.15.71 Receive DMX



Properties	
Filter	
Receive DMX Block	
Name	Receive DMX
Block ID	e750f102-a215-4337-92dc-5
Target	
Target	<not set> X
Configurations	
Number Of Output Po	1
Output port 1	
Port Index	1
Channel Index	1
Data Type	Int64
Port Mode	Active
Is Available	
Data Type	Bool
Port Mode	Active
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active

Use this Block to receive DMX values from an external system, like a SYMPL dmx Node, and process them in the Workflow Designer, e.g. to map a DMX channel value to a Sequence Opacity.

This Block is target based. The Target value is displayed as main setting in the Block's header and its type defines the Block title.

When orphan, the Block has all the properties of a bound Block available. The Block has an Enable PortIn, an Is Available PortOut, an Is Enabled PortOut, and initially one PortOut for the received DMX value.

The Block can be bound to a SYMPL dmx Node or a SYMPL e:pix Node.

- Add a (virtual) SYMPL dmx or e:pix Node to the project. In the Setup Tab, set at least one DMX "Direction" property of the SYMPL node to "Input". The SYMPL node is now available as a target value for the DMX Input block.
- Define the Target.
- Define the number Output Port(s).

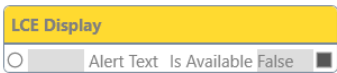


Category	Property	Description
Target		Define the target to bind the Block to: a SYMPL dmx Node or a SYMPL e:pix Node.
Number of Output Ports		Define the number of PortOuts to be available at the block. Each DMX channel of each DMX input universe has its own PortOut assigned. Define the configuration and mapping in the dedicated “Output port” properties.
Output port (e.g. 1)	Port Index	Define the DMX Port (universe) of the Target value to listen to.
	Channel Index	Define the DMX channel of the selected universe to listen to. The Output Ports are per default consecutively mapped to DMX channels.
	Data Type	Define the data type of the PortOut. The DMX value can either be forwarded as Int64 or as Double.
Is Available		An actively sent bool value whether the bound target is available or not. For SYMPL nodes, the condition is automatically TRUE when a real Node is matched.
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not at the “Enable” PortIn.

- You can completely preconfigure the Receive DMX block without an attached SYMPL node in the project.



11.15.72 LCE2 Display



Properties

Filter

LCE Display Block

Name	LCE Display
Block ID	9223e14d-7c95-4b4f-b62e

Target

Target	<not set>	X
--------	-----------	---

Configurations

Timeout	5
---------	---

Is Available

Data Type	Bool
Port Mode	Active

Alert Text

Data Type	String
Port Mode	Passive

Use this Block to show a specific message on the display of an e:cue Lighting Control Engine device.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has a PortIn for the Alert Text on the LCE display and an Is Available PortOut.

The Block can be bound to an LCE2 or an LCE3.

- Define the Target Property.
- Define the Timeout.

Category	Property	Description
Target		Define the target to bind the Block to an LCE.
Timeout		Define the timeout of the alert in seconds. After the timeout, the LCE display returns to display its system pages
Is Available		An actively sent bool value whether the bound target is available or not.

Alert Text

A passively received string value of the alert text to be displayed on the LCE.

- While an alert text is displayed on the LCE, the OK button of the cursor control keys on the LCE is flashing. Press the OK button to dismiss the alert text before the timeout as an act of acknowledgement.
- To display alert text over more than one row / display line, use the Combine Text block. Put each curly bracket / PortIn in a new line in the Format String property of the Combine Text block and connect the alert text as parts to the PortIns.
- Please note that the LCE display supports the standard characters from ASCII 32 to ASCII 126 (Hex 20 to Hex 7E).



11.15.73 Glass Touch T6

Glass Touch T6

Glass Touch T6 #1

☐ True

Enable

Is Available False

Is Enabled True

On/Off Is...False

Dim- Is Pr...False

Dim+ Is Pr...False

1 Is Pressed False

2 Is Pressed False

3 Is Pressed False

4 Is Pressed False

Dim- Short Push

Dim+ Short Push

1 Short Push

2 Short Push

3 Short Push

4 Short Push

On/Off Toggle On

On/Off Toggle Off

Glass Touch T6 Block Configuration

☐ Is Toggle Btn.
Is Pressed.Shc

☐ Is Toggle Btn.
Is Pressed.Shc

☐ Is Toggle Btn.
Is Pressed.Shc

☐ Is Toggle Btn.
Is Pressed.Shc

Dim-

Dim+

1

2

3

4

On/Off Toggle

☐ Is Toggle Btn.
Is Pressed.Shc

☐ Is Toggle Btn.
Is Pressed.Shc

☐ Is Toggle Btn.
Is Pressed.Shc

☐ Is Toggle Btn.
Is Pressed.Shc

OKCancel

Properties

Filter

Glass Touch T6 Block

NameGlass Touch T6

Block IDd4b223f6-0ae0-4da1-b19

Target

Target<not set>

Configurations

Beep On Change

Long Push Threshold500

Button On/Off

LED Represents Inter

Button Dim-

LED Represents Inter

Button Dim+

LED Represents Inter

Button 1

LED Represents Inter

Is Available

Data TypeBool

Port ModeActive

Enable

Data TypeBool

Port ModePassive

Is Enabled

Data TypeBool

Port ModeActive

Button On/Off Is Pressed

Data TypeBool

Port ModeActive

Dim- Is Pressed

Data TypeBool

Port ModeActive

Dim+ Is Pressed

Data TypeBool

Port ModeActive

1 Is Pressed

Data TypeBool

Port ModeActive

Use this Block to control your installation with a Glass Touch T6 User Terminal.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has bool PortOuts and EventOuts for each button as defined in the extended configuration window. Extended configuration is accessible at the cogwheel icon.

The Block can be bound to a Glass Touch T6.

- Define the Target.
- Define the available functionalities as PortOuts of each button in the extended configuration window.


Category	Description
Target	Define the target to bind the Block to a Glass Touch T6.
Beep on Change	Enable to get an acoustical feedback of the Glass Touch on every event. The Glass Touch will beep on pushing the button and on releasing the button.
Button (e.g. On / Off, Dim+/-, 1)	<p>Enable to display the internal state of the button with the button's LED. The LED is on when the button is being pressed or is toggled on. The LED is off when the button is not being pressed or is toggled off.</p> <p>Disable to create a passive Int64 PortIn named "LED (e.g. On / Off, Dim+/-, 1)". This PortIn controls the LED of the button:</p> <p>0: off</p> <p>1: on</p> <p>3: flashing.</p>
Is Available	An actively sent bool value whether the bound target is available or not.
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.

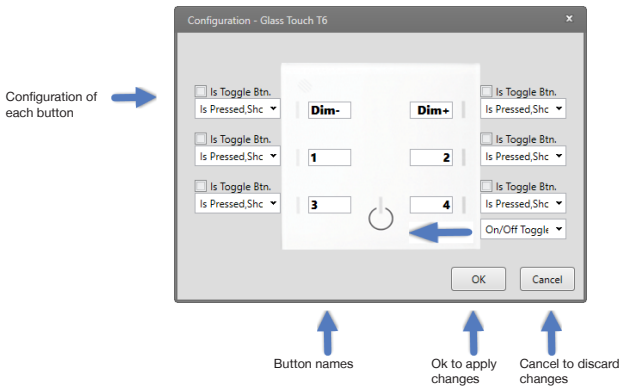


Button Is Pressed An actively sent bool value whether the button is being pressed or (e.g. On / Off, not.
Dim+/-, 1)

Defining the available functionalities of each button in the extended configuration window

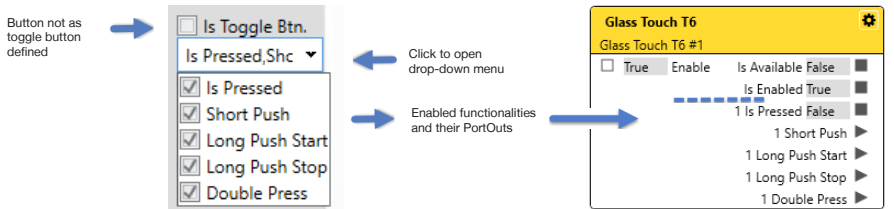
Define the available functionalities of each button in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



2. Configure each button. Select which functionalities are available:
First, define whether to handle the button as a toggle button. Check the box "Is Toggle Button" at the top of the drop-down menu to enable as toggle button.
Second, enable functionalities in the drop-down menu. Each functionality will have a separate PortOut in the Glass Touch block.

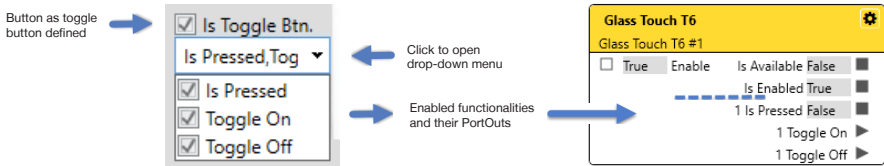
Is NOT Toggle Button:



Is Pressed: The boolean TRUE is forwarded for the duration of the button pressing.
Short Push: A trigger is sent when the button is pressed for a short time.
Long Push Start: A trigger is sent at the beginning of a longer button press.
Long Push Stop: A trigger is sent at the end of a longer button press.

Double Press: A trigger is sent when the button is pressed shortly twice.

Is Toggle Button:



Is Pressed: The boolean TRUE is forwarded for the duration of the button pressing.

Toggle On: A trigger is sent when the button is pressed to switch to the state "Toggle On". If "LED Represents Internal State" is enabled for the Button On / Off, the LED is on at the Toggle On state.

Toggle Off: A trigger is sent when the button is pressed to switch to the state "Toggle Off". If "LED Represents Internal State" is enabled for the Button On / Off, the LED is off at the Toggle Off state.

3. Click the "OK" button to save the configuration and close the window.

The Glass Touch T6 block has now the functionalities of the buttons configured and the PortOuts accordingly available.

- Please do not use the Glass Touch button as a toggle button if more than one button is connected to the same functionality. The Glass Touch toggle button has no back channel to remotely change its state.
- The On / Off Toggle button does not represent a power switch. The Power button of the Glass Touch does not need to be pressed on for the other buttons to be executed.
- You can rename buttons in the extended configuration window. Note that the customized name is used for the PortOut names only, not at the Properties window.
- You can completely preconfigure the Glass Touch block without an attached Glass Touch user terminal in the project and in Trial Mode.
- Please note that a Glass Touch is only responsive (LED indication, sound) after adding its target based Workflow block to the Workspace.



11.15.74 Glass Touch T6R

Glass Touch T6R

Glass Touch T6R

- ☐ True Enable Is Available True
- Is Enabled True
- On/Off Is... False
- 1 Is Pressed False
- 2 Is Pressed False
- 3 Is Pressed False
- 4 Is Pressed False
- 5 Is Pressed False
- 6 Is Pressed False
- Play/Pause... False
- Jog Position 0
- Jog Incre... 0
- Jog Incre... 0
- Jog Push ▶
- 1 Short Push ▶
- 2 Short Push ▶
- 3 Short Push ▶
- 4 Short Push ▶
- 5 Short Push ▶
- 6 Short Push ▶
- On/Off Toggle On ▶
- Play/Pause Toggle... ▶
- On/Off Toggle Off ▶
- Play/Pause Toggle... ▶

Properties

Filter

▲ **Glass Touch T6R Block**

Name	Glass Touch T6R
Block ID	3aef6e13-f38d-4b1c-818f-

▲ **Target**

Target	<not set> <input type="button" value="X"/>
--------	--

▲ **Configurations**

Beep On Change	<input type="checkbox"/>
Long Push Threshold	500

▲ **Button On/Off**

LED Represents Inter	<input checked="" type="checkbox"/>
----------------------	-------------------------------------

▲ **Jog Increment Abs**

Enable Increment Of	<input type="checkbox"/>
Min Value	-1000
Max Value	1000
Data Type	Double
Port Mode	Active

▲ **Button 1**

LED Represents Inter	<input checked="" type="checkbox"/>
----------------------	-------------------------------------

Is Available	
Data Type	Bool
Port Mode	Active
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Button On/Off Is Pressed	
Data Type	Bool
Port Mode	Active
1 Is Pressed	
Data Type	Bool

Button Play/Pause	
Data Type	Bool
Port Mode	Active

Jog Position	
Data Type	Double
Port Mode	Active

Jog Increment Rel	
Data Type	Double
Port Mode	Active



Use this Block to control your installation with a Glass Touch T6R User Terminal.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available.

The Block has bool PortOuts and EventOuts for each button as defined in the extended configuration window. Extended configuration is accessible at the cogwheel icon.

The Block can be bound to a Glass Touch T6R.

- Define the Target.
- Define the available functionalities as PortOuts of each button in the extended configuration window.

Category	Property	Description
Target		Define the target to bind the Block to a Glass Touch T6R.
Beep on Change		Enable to get an acoustical feedback of the Glass Touch on every event. The Glass Touch will beep on pushing the button and on releasing the button.



Jog Increment Abs		An actively sent double value of the jog position movement. The Jog Increment Abs value is the absolute increment of the jog position of all motions, from the very first motion to the last position of the latest motion. Clockwise motion adds to the value, counterclockwise motion subtracts from the value. One turn equals the value 100.
	Enable Increment Offset	Enable to create a PortIn which will be taken as start value for Jog Increment Abs.
	Min Value	Define the minimum value the Jog Increment Abs can have. E.g. when -300, three full turns counterclockwise on the jog wheel are needed to reach the minimum value starting from zero.
	Max Value	Define the maximum value the Jog Increment Abs can have. E.g. when 300, three full turns clockwise on the jog wheel are needed to reach the maximum value starting from zero.
Button (e.g. On / Off, 1)	LED Represents Internal State	<p>Enable to display the state of the button with the LED of the button. The LED is on when the button is being pressed or is toggled on. The LED is off when the button is not being pressed or is toggled off.</p> <p>Disable to create a passive Int64 PortIn named "LED (e.g. On / Off, 1)". This PortIn controls the LED of the button:</p> <p>0: off 1: on 3: flashing.</p>

Is Available	An actively sent bool value whether the bound target is available or not.
--------------	---



Enable	A passively received bool value to dis- or enable the Block.
--------	--

Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
------------	---

Button Is Pressed (e.g. On / Off, Play / Pause, 1)	An actively sent bool value whether the button is being pressed or not.
--	---


Jog Position	An actively sent double value of the place where the jog wheel is touched. The Jog Position value is "0" at the 12 o'clock position and increases clockwise up to a maximum value of "100" at 360°.
--------------	---

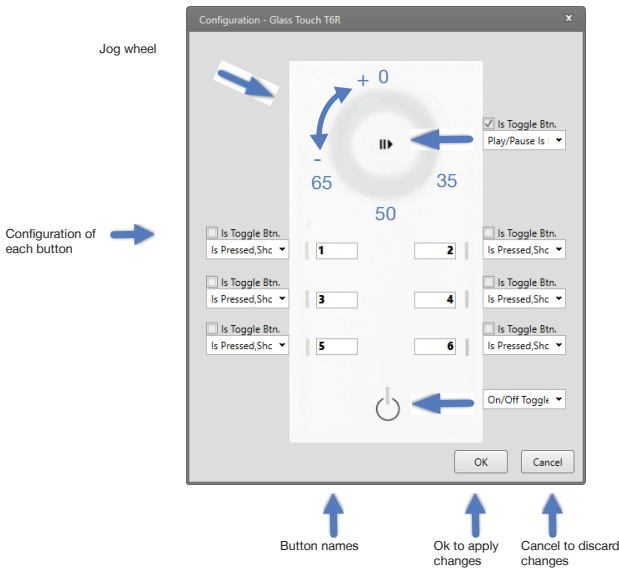
Jog Increment Rel	An actively sent double value of the jog position movement. The Jog Increment Rel value is the difference of the jog position from one motion, from the starting point to the end point of a motion.
-------------------	--



Defining the available functionalities of each button in the extended configuration window

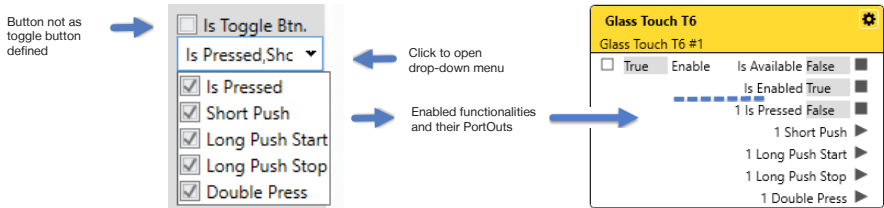
Define the available functionalities of each button in the extended configuration window:

- 1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



- 2. Configure each button. Select which functionalities are available:
First, define whether to handle the button as a toggle button. Check the box "Is Toggle Button" at the top of the drop-down menu to enable as toggle button.
Second, enable functionalities in the drop-down menu. Each functionality will have a separate PortOut in the Glass Touch block.

Is NOT Toggle Button:

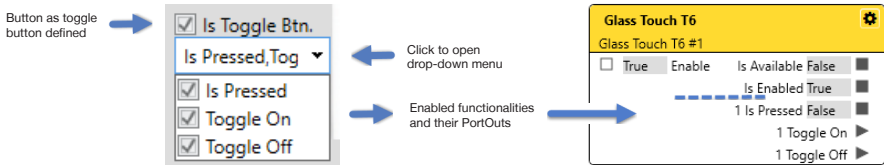


- Is Pressed:** The boolean TRUE is forwarded for the duration of the button pressing.
- Short Push:** A trigger is sent when the button is pressed for a short time.
- Long Push Start:** A trigger is sent at the beginning of a longer button press.

Long Push Stop: A trigger is sent at the end of a longer button press.

Double Press: A trigger is sent when the button is pressed shortly twice.

Is Toggle Button:



Is Pressed: The boolean TRUE is forwarded for the duration of the button pressing.

Toggle On: A trigger is sent when the button is pressed to switch to the state "Toggle On". If "LED Represents Internal State" is enabled for the Button On / Off, the LED is on at the Toggle On state.

Toggle Off: A trigger is sent when the button is pressed to switch to the state "Toggle Off". If "LED Represents Internal State" is enabled for the Button On / Off, the LED is off at the Toggle Off state.

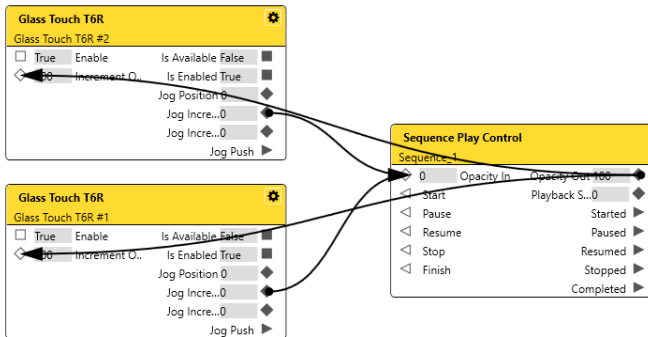
3. Click the "OK" button to save the configuration and close the window.

The Glass Touch T6R block has now the functionalities of the buttons configured and the PortOuts accordingly available.

- The On / Off Toggle button does not represent a power switch. The Power button of the Glass Touch does not need to be pressed on for the other buttons to be executed.
- You can rename a button in the extended configuration window. Note that the customized name is used for the PortOut names, not at the Properties window.
- Use the jog wheel PortOuts for dimming purposes, e.g. connect an ActionPad fader block or a DALI - Group Control block. Use it also to control Key Framings of Effects by connecting a Change Object Property block. Use the Jog Increment Abs PortOut for dimming purposes with more fine-tuning: increase the limits of the Min and Max value to enable fine adjustment when calibrating. It is not possible to control hue. Use the Jog Push PortOut for triggers that are sent upon a short push on the jog wheel.
- You can completely preconfigure the Glass Touch block without an attached Glass Touch user terminal in the project and in Trial Mode.
- Please note that a Glass Touch is only responsive (LED indication, sound) after adding its Workflow block to the Workspace ().

Implementing classic jog wheel behavior

- To avoid a jump from 100% to 0% in opacity when crossing the 12 o'clock position on the jog wheel, use the Jog Increment Abs PortOut.
- If you want to use several Glass Touches T6R to control the same sequence, you need to pass the jog value through to all Glass Touches T6R. Enable the “Enable Increment Offset” of the blocks and connect the opacity of the sequence to the Increment Offset PortIn of the Glass Touch blocks. The offset will be considered by the Jog Increment Abs PortOut.
- To get the classic behavior of the jog wheel, use the following workflow:



11.15.75 Glass Touch T12

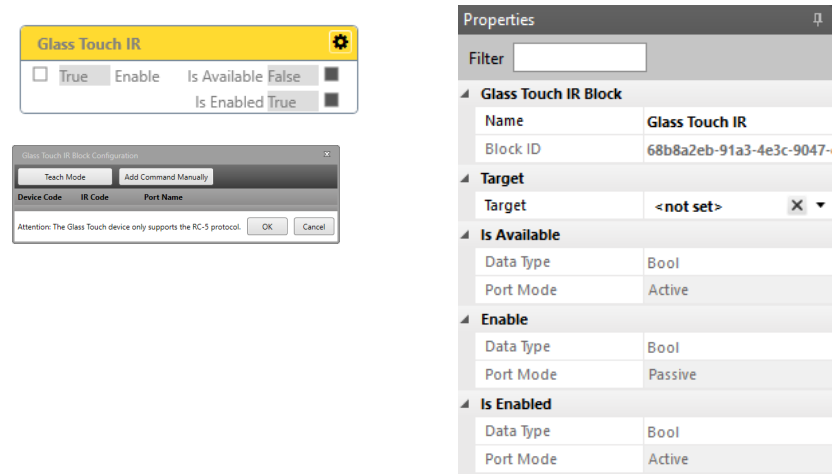
The Glass Touch T12 based Block is identical to the Glass Touch T6 based Block, but with 12 buttons and one Button On / Off.

See „[11.15.73 Glass Touch T6](#)“ on [page 344](#) for details on the Block.

The Block can be bound to a Glass Touch T12.



11.15.76 Glass Touch IR



Use this Block to process IR commands from a remote control that are received by any Glass Touch User Terminal. Make sure the IR-Remote supports RC5.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has all the properties of a bound Block available. The Block has bool PortOuts and EventOuts for each IR command as defined in the extended configuration window. Extended configuration is accessible at the cogwheel icon.

The Block can be bound to a Glass Touch T6, Glass Touch T6R and Glass Touch T12.

- Define the Target.
- Define the available EventOuts of IR commands in the extended configuration window.

Category	Description
Target	Define the target to bind the Block to a Glass Touch.


Is Available An actively sent bool value whether the bound target is available or not.

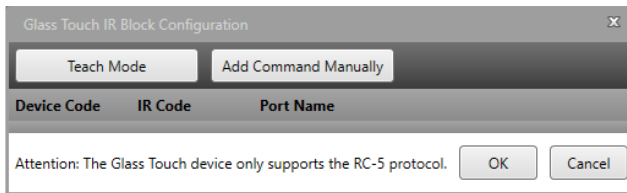
Enable	A passively received bool value to dis- or enable the Block.
--------	--

Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
------------	---

Defining the available EventOuts of IR commands in the extended configuration window

Define the available EventOuts of IR commands in the extended configuration window:

1. Click the cogwheel icon at the right top corner of the Block to open the extended configuration window: .



The configuration of an IR command consists of three elements:

Device Code: A selectable identification number for the remote control device which sends the IR command. A device with a device code "0" is not being progressed.

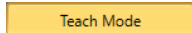
IR Code: The IR code sent by the remote control.

Port Name: Define a name for the EventOut that will be created for the IR command.

2. Add an IR command using either the Teach Mode or the manually adding option.

Teach Mode: Configure IR commands using remote controls.

Click the "Teach Mode" button. When activated, the button is highlighted:



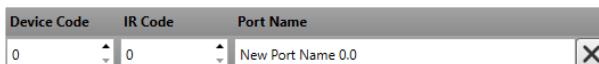
Point the remote control towards the Glass Touch while pressing the button on the remote control you want to configure. When the IR frequency is received by the Glass Touch, enter a name for the EventOut in the pop up entry field.

Click "OK" to add the IR command to the configuration. A new line is added to the extended configuration window that consists of the automatically entered device code, IR code and the user defined Port name.

To add further IR commands, press another button on the remote control.

Add Command Manually: Configure IR commands without using remote controls. You need to know the IR codes to be able to use this option.

Click the "Add Command Manually" button to add a new line to the extended configuration window:




Define the three elements of the IR command.

To add further IR commands, click the button again.

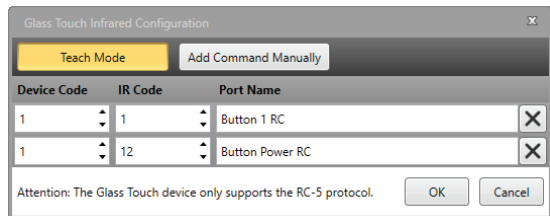
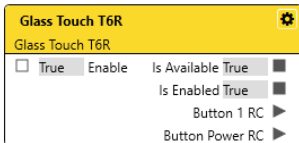


3. Click the “OK” button to save the configuration and close the window.

The Glass Touch IR block has now the IR commands configured and the EventOuts accordingly available.

- You can preconfigure the Glass Touch block for IR commands while in Trial Mode and without an attached Glass Touch user terminal in the project as long as you configure the commands manually.
- You can manually change and rename any IR command.
- To delete an IR command, click the cross button at the end of the line: .
- To configure identical IR commands for multiple Glass Touch IR blocks, first configure one target based Block. Secondly, delete the target value of the Block. Copy and paste the now orphan Block. Set the target values for the copied Blocks as desired. All configured IR commands are multiplied and available.

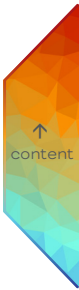
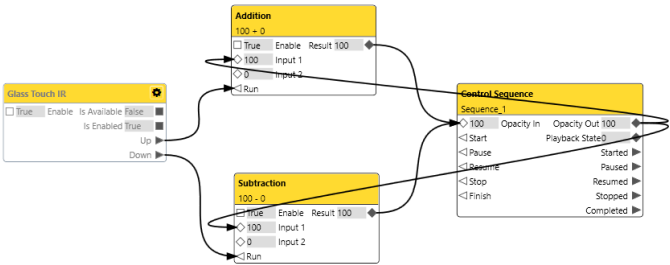
Example: Two IR commands for pressing button 1 and the Power button on the remote control are defined using Teach Mode in the extended configuration window. Two EventOuts for the IR commands are created in the Glass Touch IR block:



Implementing classic dimming behavior using remote control

- To dim a sequence using a remote control, create the two IR commands Up and Down.
- Connect an Add Numbers block to the Up EventOut, setting the Block to “Is event based” and Input 2 to Property with a value of 10. Using values lower than ten slows down dimming, greater than ten makes dimming faster (depending on the repeat rate of the used IR control).
- Connect the Result to the Opacity In PortIn of the Sequence Play Control block and the Opacity Out PortOut of the sequence to the Input 1 PortIn of the Add Numbers block.

- Connect a Subtract Numbers block to the Down EventOut and repeat all settings from the Add Numbers block.



11.15.77 Sympl Fusion Action Buttons

Sympl Fusion Action Buttons

☐ True

Enable

Is Enabled True

Is Available False

1. Is Down False

2. Is Down False

1. Down

1. Up

1. Short Pressed

1. Long Press Start

1. Long Pressed

2. Down

2. Up

2. Short Pressed

2. Long Press Start

2. Long Pressed

Properties

Filter

Sympl Fusion Action Buttons Block

Name

Sympl Fusion Action Butto

Block ID

42da869d-6171-4780-8a1c-

Configurations

Long Press Delay 1. Bu

500

Long Press Delay 2. Bu

500

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Is Available

Data Type

Bool

Port Mode

Active

1. Is Down

Data Type

Bool

Port Mode

Active

2. Is Down

Data Type

Bool

Port Mode

Active

Use this Block to trigger automation from the Action buttons of a SYMPL Fusion device.

The Block has an Enable PortIn, four PortOuts for the block and button states and ten EventOuts for button triggers.

- Define the long press delays.

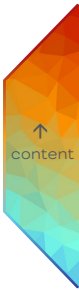
Category	Property	Description
Configurations		Configure the definition of a long press.

	Long Press Delay 1. Button	Define the amount of time that should elapse after pressing the Action 1 button before it is considered a long press. Value range from 100 to 5000 ms. Default= 500 ms.
--	----------------------------	---

	Long Press Delay 2. Button	Define the amount of time that should elapse after pressing the Action 2 button before it is considered a long press. Value range from 100 to 5000 ms. Default= 500 ms.
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Is Available		An actively sent bool value whether the action buttons of a SYMPL Fusion are available or not. Depends on which instances has the control over the modules and inputs from a SYMPL Fusion. When FALSE, the button states and actions cannot be processed.
1. Is Down		An actively sent bool value whether the Action 1 button is currently being pressed or not.
2. Is Down		An actively sent bool value whether the Action 2 button is currently being pressed or not.
Event Port	Description	
1. Down	Sends a trigger when the Action 1 button is down.	
1. Up	Sends a trigger when the Action 1 button is up.	
1. Short Pressed	Sends a trigger after a short press of the Action 1 button. Short press: a single push and release within a period that is shorter than the defined Long Press Delay 1 Button period.	



- | | |
|---------------------|--|
| 1. Long Press Start | Sends a trigger after the long press start of the Action 1 button.
Long press start: a single press that exceeds the defined Long Press Delay 1 Button period. |
| 1. Long Pressed | Sends a trigger after the long press stopped of the Action 1 button.
Long press stop: a release after a single long press. |
| 2. Down | Sends a trigger when the Action 2 button is down. |
| 2. Up | Sends a trigger when the Action 2 button is up. |
| 2. Short Pressed | Sends a trigger after a short press of the Action 2 button.
Short press: a single push and release within a period that is shorter than the defined Long Press Delay 1 Button period. |
| 2. Long Press Start | Sends a trigger after the long press start of the Action 2 button.
Long press start: a single press that exceeds the defined Long Press Delay 1 Button period. |
| 2. Long Pressed | Sends a trigger after the long press stopped of the Action 2 button.
Long press stop: a release after a single long press. |
- When the access to the Action buttons is redirected to somewhere else, then no events or state changes are done. So the last known states are kept. When the access to the Action buttons is given back, then the state of the Action buttons are updated. And the all events that fit to the state are fired. So the correct update to the actual state is provided. The Block is updated to the current/actual state.
 - When enabling the Block, it behaves in the same way as it does when giving back access to the Action buttons (see above).



11.15.78 Sympl Fusion Dry Contacts

Sympl Fusion Dry Contacts

☐ True

Enable

Is Enabled True

Is Available True

1. Is Closed False

2. Is Closed False

3. Is Closed False

4. Is Closed False

1. Closed

1. Opened

1. Short Pressed

1. Long Press Start

1. Long Pressed

2. Closed

2. Opened

2. Short Pressed

2. Long Press Start

2. Long Pressed

3. Closed

3. Opened

3. Short Pressed

3. Long Press Start

3. Long Pressed

4. Closed

4. Opened

4. Short Pressed

4. Long Press Start

4. Long Pressed

Properties

Filter

Sympl Fusion Dry Contacts Block

Name

Sympl Fusion Dry Contacts

Block ID

fc16a315-1b97-473b-a237-5

Configurations

Long Press Delay 1. Dr

500

Long Press Delay 2. Dr

500

Long Press Delay 3. Dr

500

Long Press Delay 4. Dr

500

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Is Available

Data Type

Bool

Port Mode

Active

1. Is Closed

Data Type

Bool

Port Mode

Active

2. Is Closed

Data Type

Bool

Port Mode

Active

3. Is Closed

Data Type

Bool

Port Mode

Active

4. Is Closed

Data Type

Bool

Port Mode

Active

Use this Block to trigger automation from the digital inputs of a SYMPL Fusion device.

The Block has an Enable PortIn, four PortOuts for the block and inputs states and ten EventOuts for digital input triggers.

- Define the long press delays.

Category	Property	Description
Configurations		Configure the definition of a long press.
	Long Press Delay 1. Dry Contact (respectively 2, 3, or 4)	Define the amount of time that should elapse after the digital input 1 (respectively 2, 3, or 4) closes before it is considered a long press. Value range from 100 to 5000 ms. Default= 500 ms.
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
Is Available		An actively sent bool value whether the digital inputs of a SYMPL Fusion are available or not. Depends on which instances has the control over the modules and inputs from a SYMPL Fusion. When FALSE, the inputs states and actions cannot be processed.
1. Is Closed (respectively 2, 3, or 4)		An actively sent bool value whether the digital input 1 (respectively 2, 3, or 4) is closed or not.
Event Port	Description	
1. Closed (respectively 2, 3, or 4)	Sends a trigger when the digital input 1 (respectively 2, 3, or 4) is closed.	
1. Opened (respectively 2, 3, or 4)	Sends a trigger when the digital input 1 (respectively 2, 3, or 4) is opened.	



1. Short Pressed (respectively 2, 3, or 4)	<p>Sends a trigger after a short press of the digital input 1 (respectively 2, 3, or 4).</p> <p>Short press: a single close and open within a period that is shorter than the defined Long Press Delay 1 (respectively 2, 3, or 4) Dry Contact period.</p>
---	--

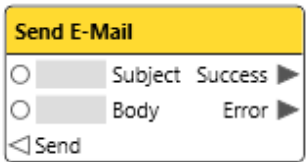
1. Long Press Start (respectively 2, 3, or 4)	<p>Sends a trigger after the long press start of the digital input 1 (respectively 2, 3, or 4).</p> <p>Long press start: a single close that exceeds the defined Long Press Delay 1 (respectively 2, 3, or 4) Dry Contact period.</p>
--	---

1. Long Pressed (respectively 2, 3, or 4)	<p>Sends a trigger after the long press stopped of the digital input 1 (respectively 2, 3, or 4).</p> <p>Long press stop: an open after a single long press.</p>
--	--

- When the access to the digital inputs is redirected to somewhere else, then no events or state changes are done. So the last known states are kept. When the access to the digital inputs is given back, then the state of the digital inputs are updated. And the all events that fit to the state are fired. So correct update to the actual state is provided. The Block is updated to the current/actual state.
- When enabling the Block, it behaves in the same way as it does when giving back access to the digital inputs (see above).



11.15.79 Send E-Mail



Properties	
Filter	<input type="text"/>
Different Blocks	
Name	Send E-Mail
Block ID	953c341d-33d0-455d-ad40-
Configurations	
Is event based	<input checked="" type="checkbox"/>
Mail Settings	
Encryption	None
Host	
Port	25
Password	
From	
To	
CC	
BCC	
Subject	
Data Type	String
Input Type	Port
Port Mode	Active
Body	
Data Type	String
Input Type	Port
Port Mode	Active

Use this Block if you want to send e-mails with customizable messages.

The Block has two PortIns for the content of the e-mail, is event triggered by default, and has two EventOuts to give feedback on its execution.

- Define an SMTP server data (Host, Encryption, and Port).
- Enter sender address (From, Password) and recipient address (To). Optionally, add recipients in CC and BCC.
- Define the content of the Subject and the Body of the e-mails.

Category	Property	Description
	Is event based	Enabled. This Block is executed on event trigger.

Mail Settings

Define the mail settings.

Encryption	Choose between None, SSL, STARTTLS according to your email provider specifications. Required.
Host	Enter the SMPT server address according to your e-mail provider specifications. Required.
Port	Enter the SMPT port according to your e-mail provider specifications. Required.
Password	Enter the password of your e-mail account. Required.
From	Enter your e-mail address. Required.
To	Enter the recipient address. Required.
CC	Optional.
BCC	Optional.

Subject

Configure how the content for the subject of the e-mail message is constituted.

By default, the content is actively caught from the counterpart Block. Set the Input Type to Property to manually define a static value.



Body

Configure how the content of the body of the e-mail message is constituted.


By default, the content is actively caught from the counterpart Block. Set the Input Type to Property to manually define a static value.

- The Success EventOut is fired when the e-mail has been sent successfully; the Error EventOut is fired when the e-mail could not be sent.



11.15.80 Send MQTT

Send MQTT

☐ True Enable Is Enabled True 

MQTT Block Configuration

Data Type	Identifier	Topic	Retain
-----------	------------	-------	--------

Add Item OK Cancel

Properties

Filter

Send MQTT Block

Name

Send MQTT

Block ID

ee49b5f7-14bb-4290-9221

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Use this Block to send MQTT data, e.g. when using a SYMPHOLIGHT Gateway BACnet.



To actually output MQTT data, the Gateway Communication credit is required (see [“Features” on page 19](#)).

The Block has initially an Enable PortIn and an Is Enabled PortOut.
The Block has PortIns and EventIns for each MQTT data point as defined in the extended configuration window. Extended configuration is accessible at the cogwheel icon.

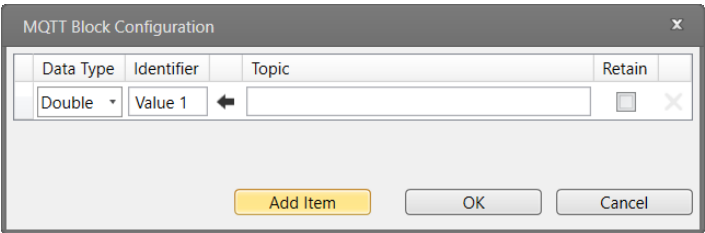
- Define the MQTT data points in the extended cogwheel window..

Category	Description
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the “Enable” PortIn.

Defining MQTT data points in the extended configuration window

Define and configure the MQTT data points that are to be send in the extended configuration window:

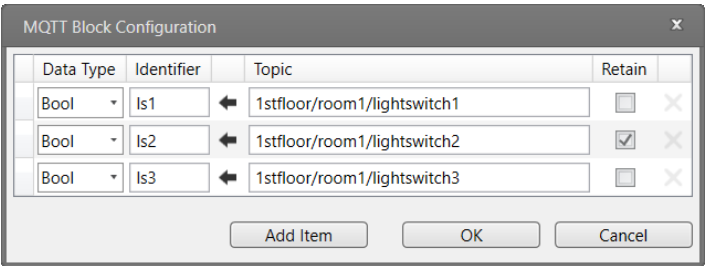
Add for each MQTT data point an item in the list. Configure the items accordingly (Data Type, Topic, Retain).



The screenshot shows the 'MQTT Block Configuration' dialog box. It contains a table with the following structure:

Data Type	Identifier	Topic	Retain
Double	Value 1		<input type="checkbox"/>


Below the table are three buttons: 'Add Item' (highlighted in yellow), 'OK', and 'Cancel'. A left-pointing arrow is located between the Identifier and Topic columns.




The screenshot shows the 'MQTT Block Configuration' dialog box with three items configured in the table:

Data Type	Identifier	Topic	Retain
Bool	Is1	1stfloor/room1/lightswitch1	<input type="checkbox"/>
Bool	Is2	1stfloor/room1/lightswitch2	<input checked="" type="checkbox"/>
Bool	Is3	1stfloor/room1/lightswitch3	<input type="checkbox"/>

Below the table are three buttons: 'Add Item', 'OK', and 'Cancel'. A left-pointing arrow is located between the Identifier and Topic columns.

- Click on  to copy the Topic entry to the Identifier entry if desired.
- Note that the Topic must comply with the naming conventions. The Topic entry appears in the MQTT broker. The Identifier entry names the PortIn only.
- MQTT data points of the bool type will become EventIns, all other types become according PortIns.




The MQTT interface is in the testing phase and should not be used in production environments. It is subject to ongoing changes without backward compatibility. Currently, the interface is intended exclusively for use with SymphoCity. Only available when the Gateway Communication feature is activated.



11.15.81 Receive MQTT

Receive MQTT

☐ True Enable Is Enabled True 

MQTT Block Configuration

Data Type	Identifier	Topic	Retain
-----------	------------	-------	--------

Add Item OK Cancel

Properties

Filter

Receive MQTT Block

Name	Receive MQTT
Block ID	57bca8f1-33b7-483f-a9e9

Enable

Data Type	Bool
Port Mode	Passive

Is Enabled

Data Type	Bool
Port Mode	Active

Use this Block to receive MQTT data, e.g. when using a SYMPHOLIGHT Gateway BACnet.



To actually input MQTT data, the Gateway Communication credit is required (see [“Features” on page 19](#)).

The Block has initially an Enable PortIn and an Is Enabled PortOut.
The Block has PortIns and EventIns for each MQTT data point as defined in the extended configuration window. Extended configuration is accessible at the cogwheel icon.

- Define the MQTT data points in the extended cogwheel window..

Category	Description
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the “Enable” PortIn.

Defining MQTT data points in the extended configuration window

Define and configure the MQTT data points that are to be received in the extended configuration window:

Add for each MQTT data point an item in the list. Configure the items accordingly (Data Type, Topic).

MQTT Block Configuration

Data Type	Identifier	Topic
Double	Value 1	

Add Item

OK

Cancel


MQTT Block Configuration


Data Type	Identifier	Topic	Retain
Bool	Is1	1stfloor/room1/lightswitch1	<input type="checkbox"/>
Bool	Is2	1stfloor/room1/lightswitch2	<input checked="" type="checkbox"/>
Bool	Is3	1stfloor/room1/lightswitch3	<input type="checkbox"/>

Add Item

OK

Cancel

- Click on  to copy the Topic entry to the Identifier entry if desired.
- Note that the Topic must comply with the naming conventions. The Topic entry appears in the MQTT broker. The Identifier entry names the PortIn only.
- MQTT data points of the bool type will become EventIns, all other types become according PortIns.



The MQTT interface is in the testing phase and should not be used in production environments. It is subject to ongoing changes without backward compatibility. Currently, the interface is intended exclusively for use with SymphoCity. Only available when the Gateway Communication feature is activated.



11.15.82 Send UDP

Send UDP			
127.0.0.1:1337 (ASCII)			
<input type="checkbox"/> True	Enable	Is Enabled True	<input type="checkbox"/>
<input type="radio"/>	Packet Content	Packet Sent	<input type="button" value="▶"/>

Properties	
Filter	<input type="text"/>
Send UDP Block	
Name	Send UDP
Block ID	abab9aa4-29e9-4150-92f6-
UDP Settings	
Encoding	ASCII
IP Address	127.0.0.1
Port	1337
Sender Port	-1
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Packet Content	
Data Type	String
Port Mode	Passive

Use this Block to send UDP network triggers to a given IP address of external systems, e.g. with a Constant Value Block.

The Block has one PortIn for the message to be sent, an Enable PortIn, an Is Enabled PortOut, and an EventOut for feedback that the message was sent.

- Define the UDP Settings.



Category	Property	Description
UDP Settings		Define the receiver / destination and the encoding of the UDP packets.
	Encoding	<p>The encoding how to interpret the Packet Content.</p> <p>Choose between ANSI, ASCII, UTF8, and Binary.</p> <p>Using Binary, you can combine hex chars, decimal chars and text for input, output and filtering in any way:</p> <p>13 10 "Hello" 0x0D 0x0A \$D \$A means \\r\\nHello\\r\\n\\r\\n.</p> <p>E.g. specify each character one by one and combine it with ANSI text in quotes.</p> <p>Use \" to have a quote sign inside the quotes.</p>
	IP Address	<p>The destination IP Address of the receiver of the UDP packets to send to.</p> <p>Default: 127.0.0.1 which is the IP address of the localhost (= the computer running SYMPHOLIGHT); the computer itself receives the sent message.</p>
	Port	<p>The destination Port of the receiver that receives the UDP packets sent from the SYMPHOLIGHT instance (from 0 to 65535).</p>
	Sender Port	<p>If -1, a random port is chosen from which SYMPHOLIGHT is sending. If not -1, define a port from 0 to 65535 to ensure that the peer receives the packet from that port.</p>

Enable	A passively received bool value to dis- or enable the Block.
--------	--



Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.
------------	---

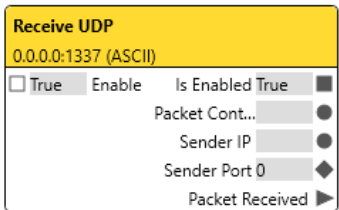
Packet Content	The content of the UDP packet, a string value, is actively forwarded to the counterpart Block.
----------------	--

- For mutual, bidirectional communication with an external system, use the Send UDP and the Receive UDP blocks. The IP Addresses in both UDP Settings have to be identical.
- Please make sure to avoid well known ports. Only use unassigned UDP ports.



↑
content

11.15.83 Receive UDP



Properties	
Filter	
Receive UDP Block	
Name	Receive UDP
Block ID	8f9ea1e8-6bfb-4af1-8dce-6
Packet Filter	
Filter	
UDP Settings	
Encoding	ASCII
IP Address	0.0.0.0
Port	1337
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active
Packet Content	
Data Type	String
Port Mode	Active
Sender IP	
Data Type	String
Port Mode	Active
Sender Port	
Data Type	Int64
Port Mode	Active

Use this Block to receive external UDP network triggers that can be processed in other blocks, e.g. the Value to Event / State Block. Creates an event and provides the content of a received UDP packet.

The Block has an Enable PortIn, one PortOut for the content, one EventOut and an Is Enabled PortOut. The UDP Settings are displayed as main setting in the Block's header.

- Define the UDP settings.



Category	Property	Description
Packet Filter		Define if only UDP packets with a specified value are to be received. When empty, all UDP packets are received.
UDP Settings		Define the sender and the encoding of the UDP packets.
	Encoding	<p>The encoding how to interpret the Packet Content.</p> <p>Choose between ANSI, ASCII, UTF8, and Binary.</p> <p>Using Binary, you can combine hex chars, decimal chars and text for input, output and filtering in any way:</p> <p>13 10 "Hello" 0x0D 0x0A \$D \$A means \r\nHello\r\n\r\n.</p> <p>E.g. specify each character one by one and combine it with ANSI text in quotes. Use \" to have a quote sign inside the quotes.</p>
	IP Address	The IP Address of the sender from which it is allowed to send UDP packets to the SYMPHOLIGHT instance. Use 0.0.0.0 if all senders are allowed to send UDP packets.
	Port	The Port of the receiver / destination (= SYMPHOLIGHT) of the UDP packets to bind to (from 0 to 65535).
Enable		A passively received bool value to dis- or enable the Block.
Is Enabled		An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.



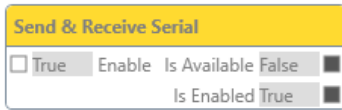
Packet Content

The content of the UDP packet, a string value, is actively received from the network (e.g. a Send UDP block).

- For mutual, bidirectional communication with an external system, use the Send UDP and the Receive UDP blocks. The IP Addresses in both UDP Settings have to be identical.
- Please make sure to avoid well known ports. Only use unassigned UDP ports.



11.15.84 Send & Receive Serial



Properties	
Filter	<input type="text"/>
Send & Receive Serial Block	
Name	Send & Receive Serial
Block ID	df53e991-4bf1-435e-9ea6-3
Target	
Target	<not set> ✕ ▼
Send Buffer	
Enable Buffering	<input type="checkbox"/>
Buffer Size In Bytes	40000
Is Available	
Data Type	Bool
Port Mode	Active
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active

Use this Block to communicate with external systems using serial ports, like the SYMPL serial Node or a Local Serial Port. Use this Block to send data to the serial ports and to receive data from the serial ports.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has an Enable PortIn, an Is Available PortOut and an Is Enabled PortOut.

The Block can be bound to a Local Serial Port with RS232 interface (see „[Local Serial Port based:](#)“ on page 381) or a SYMPL serial Node with RS232 and RS485 interfaces (see „[SYMPL serial Node based:](#)“ on page 383).

- Define the Target.

Category	Property	Description
Target		Define the target serial port to bind the Block to a SYMPL serial Node or a Local Serial Port.



Send Buffer	Define whether to buffer data that is to be sent to the target serial port.
Enable Buffering	Enable to buffer the data of the project that is to be sent to the target serial port if the target is currently unavailable. An additional EventIn named "Clear Send Buffer" is created. Clear Send Buffer: When triggered, clears the buffer of its value data.
Buffer Size in Bytes	Only for enabled "Enable Buffering": Define the size of the send buffer in bytes.
Is Available	An actively sent bool value whether the bound target is available or not. For a Local Serial Port, configure Serial Settings properties in the Setup Tab. For SYMPL Nodes, the condition is automatically TRUE when a real Node is matched.
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.

- For each target serial port, the Serial Settings properties in the Setup Tab need to be configured. Please define the Baudrate, DataBits, Parity, and Stop Bits properties in accordance to the serial ports. Conformity of the sender and the receiver ports is essential.



Local Serial Port based:

Serial Sender / Receiver			
Local Serial Port #1			
<input type="checkbox"/> True	Enable	Is Available	False
<input type="radio"/> Send 1 (RS2...		Is Enabled	True
	Receive 1 (...)		

Properties	
Filter	<input type="text"/>
Send & Receive Serial Block	
Name	Local Serial Port
Block ID	7c26a137-b93f-49c8-b955-a
Target	
Target	Serial Port <input type="button" value="X"/>
Send Buffer	
Enable Buffering	<input type="checkbox"/>
Buffer Size In Bytes	40000
Send 1 (RS232)	
EncodingName	ANSI
Data Type	String
Port Mode	Passive
Receive 1 (RS232)	
EncodingName	ANSI
Search String	
Separator String	\n
Received Bytes Thresh	512
Receive Timeout	500
Data Type	String
Port Mode	Active
Is Available	
Data Type	Bool
Port Mode	Active
Enable	
Data Type	Bool
Port Mode	Passive
Is Enabled	
Data Type	Bool
Port Mode	Active

In addition to the properties of an orphan Send & Receive Serial block, the Local Serial Port based Block has one PortIn for the content to be sent to the target serial port and one PortOut for the received message.

In addition to the properties of an orphan Send & Receive Serial block:

- Define the encoding of the outgoing messages (Send 1 (RS232)).
- Define the settings for the received PortOut messages (Receiver 1 (RS232)).



Category	Property	Description
Send 1 (RS232)		A passively received string value that is sent to the target serial port (RS232 interface).
	Encoding Name	Select an encoding scheme for the outgoing string value: ANSI, ASCII, UTF8, or Binary.
Receive 1 (RS232)		An actively sent string value that was received from the target serial port (RS232 interface).
	Encoding Name	Select an encoding scheme for the received string value: ANSI, ASCII, UTF8, or Binary. Use the encoding scheme of the target serial port where specified.
	Search String	Define the string value that defines whether a message is forwarded or not. Only messages that contain the search string (multiple entries allowed) are processed.
	Separator String	Define a string value. When the specified string has been received, the message will be processed. Not part of the forwarded content. Programming language, e.g. line feed “\n”, is not supported.
	Received Bytes Threshold	Define a number in bytes. When the specified number has been received, the message will be processed. Largest possible received number should be encompassed as minimum size.
	Receive Timeout	Define a time in milliseconds (ms). When the specified period has elapsed, the message will be processed. The time metering starts on the first received value. A message is only created when any value has been received in the specified time.



- A message is processed when the condition of the separator string OR of the bytes threshold OR of the timeout is met. The search string does not have any influence on the length of a message.
- The threshold and the timeout start over after a message is processed.
- Please take the serial port specifications into account when configuring the message processing scheme in SYMPHOLIGHT.
- The encoding scheme can be predetermined by the target serial port. Select this encoding scheme in the “Encoding Name” properties.
- The string value can be predetermined by the target serial port.
- Please note that SYMPHOLIGHT has an internal clock generator of 100 ms. A receive timeout might deviate for about 100 ms or for about 5 characters.
- To set the “Is Available” property to TRUE, go to the Setup Tab properties of the Local Serial Port controller. Enable the “Is Active” property and define a “Serial Port”.
- For the Encoding Name properties: choose between ANSI, ASCII, UTF8, and Binary. Using Binary, you can combine hex chars, decimal chars and text for input, output and filtering in any way:
 13 10 “Hello” 0x0D 0x0A \$D \$A means \r\nHello\r\n\r\n.
 E.g. specify each character one by one and combine it with ANSI text in quotes. Use “ to have a quote sign inside the quotes.

SYMPL serial Node based:

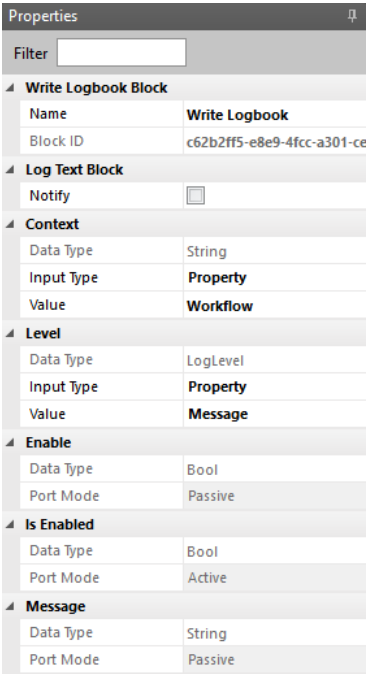
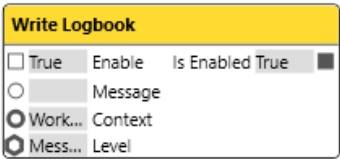
The SYMPL serial Node based Block is identical to the Local Serial Port based Block, but with an additional RS485 PortIn and an additional RS485 PortOut.

In addition to the properties of an orphan Send & Receive Serial block:

- Define the encoding of the outgoing messages (Send 1 (RS232) and Send 2 (RS485)). See the Local Serial Port based RS232 properties for Send 1 details. They also apply for the Send 2 (RS485) interface.
- Define the settings for the received PortOut messages (Receiver 1 (RS232) and Receiver 2 (RS485)). See the Local Serial Port based RS232 properties for Receive 1 details. They also apply for the Receive 2 (RS485) interface.
- The “Is Available” property is TRUE when the target serial port is matched to a real SYMPL serial Node.



11.15.85 Write Logbook






Use this Block to write a logbook message to the Log Tab, e.g. to check if the triggers work.

The Block has three PortIns for the content of the message, an Enable PortIn, and an Is Enabled PortOut.

- Define the message, the context and level.

Category	Description
Log Text Block	Enable to additionally display the message in a toast box.
Context	Define with which context the message is entered in the logbook (third column). If the Input type is set to Port, a Link at the PortIn Context is required for the Block to execute.

Level	<p>Define with which level the message is entered in the logbook (first column).</p> <p>The visibility of the according level has to be enabled in the Log Tab for the messages to be displayed.</p> <p>Define the Input Type. Set as Property, choose a Value:</p> <p>Debug: Use for Support only. The message will not be visible in the Log Tab, only written in the log file. Text Block notifications are not displayed.</p> <p>Message: Uses the info icon .</p> <p>Warning: Uses the Warning icon .</p> <p>Error: Uses the Error icon .</p>
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not.
Message	The content of the logbook message, a string value, is passively received from the counterpart Block.



11.15.86 Send to Network

Send to Network

True

Enable

Is Enabled

True

Target IP Address:

5984

Port

Network Transfer Block configuration

Port Identifier

Contains Data

Group Name

Import...

Export...

OK

Cancel

Properties

Filter

Send to Network Block

Name

Send to Network

Block ID

bd687d77-8faf-40d0-a819-...

Configurations

Encryption Key

Target IP Address(es)

Data Type

String

Input Type

Property

Value

Port

Data Type

Int64

Input Type

Property

Value

5984

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Use this Block to send data from configured ports to the network.

The Block has initially two PortIns for the target network, an Enable PortIn, and an Is Enabled PortOut. Extended configuration is accessible at the cogwheel icon.

- Define Target IP Address(es).
- Define the Port.
- You can configure additional PortIns for values and EventIns (uncheck "Contains Data"), group them, and import and export CSV sheets in the extended cogwheel window.

Category	Property	Description
Configurations	Encryption Key	A key to encrypt the transferred data. Sender and receiver must use the same key for successful transfer. Leave empty to transfer the data unencrypted.

Target IP Address(es) Define the target IP address(es) to communicate with.

Input Type Define whether the Target IP address(es) value is a fixed bool value (Input Type "Property") or a received bool Input value (Input Type "Port").

Value Set one or more (comma-separated without space) IP addresses here to unicast to these addresses. Leave empty to use broadcast.

Port

Input Type Define whether the port value is a fixed Int64 value (Input Type "Property") or a received Int64 Input value (Input Type "Port").

Value Define the port to send to. The port value must be the same port value as configured at the receiver.

Enable A passively received bool value to dis- or enable the Block.

Is Enabled An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.



11.15.87 Receive from Network

Receive from Network

True

Enable

Is Enabled

True

5984

Port

Network Transfer Block configuration

Port Identifier

Contains Data

Group Name

Import...

Export...

OK

Cancel

Properties

Filter

Receive from Network Block

Name

Receive from Network

Block ID

69a09683-87ef-4a59-a8a4-6

Configurations

Encryption Key

Port

Data Type

Int64

Input Type

Property

Value

5984

Enable

Data Type

Bool

Port Mode

Passive

Is Enabled

Data Type

Bool

Port Mode

Active

Use this Block to receive data from the network and dispatch it to configured ports.

The Block has initially one PortIn for the target network, an Enable PortIn, and an Is Enabled PortOut. Extended configuration is accessible at the cogwheel icon.

- Define the Port.
- You can configure additional PortOuts for values and EventOuts (uncheck “Contains Data”), group them, and import and export CSV sheets in the extended cogwheel window.

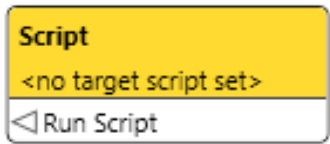
Category	Property	Description
Configurations	Encryption Key	A key to encrypt the transferred data. Sender and receiver must use the same key for successful transfer. Leave empty to transfer the data unencrypted.

Port

Input Type	Define whether the port value is a fixed Int64 value (Input Type "Property") or a received Int64 Input value (Input Type "Port").
Value	Define the port to receive from. The port value must be the same port value as configured at the sender.
Enable	A passively received bool value to dis- or enable the Block.
Is Enabled	An actively sent bool value whether the Block is enabled or not at the "Enable" PortIn.



11.15.88 Script



Properties	
Filter	
Script Block	
Name	Script
Block ID	ec2c9d9b-42c4-46d8-b55b-
Configurations	
Is event based	<input checked="" type="checkbox"/>
Target Class Function	
Class	<not set> X
Function	<not set> X
Script Settings	
Timeout	0

Use this Block to add Scripts to the Workflow Designer, e.g. if you want to start Scripts based on ActionPad button presses.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has no Ports. After defining the Target Settings, the Block has Ports as defined in the script.

The Block can be bound to a Script from the Scripting tab.

- Define the Target.

Category	Property	Description
Target		Define the target to bind the Block to a script.
Is event based		Enabled. This Block is executed on event trigger.
Script Settings	Timeout	Define the maximum run time of the script for one execution after triggering in milliseconds. 0 means disabled.

- Define further properties as designed in each script.

11.15.89 Set Property



Properties	
Filter	<input type="text"/>
Set Property Block	
Name	Set Property
Block ID	57bbac25-b6c0-4055-8bff-5
Configurations	
Is event based	<input type="checkbox"/>
Target Settings	
Object	<not set> <input type="button" value="X"/> <input type="button" value="v"/>

Use this Block if you want to change the value of an object property. An object can for example be an opacity value of an effect or the master fader.

Example: assign a fader on an ActionPad page to the Grand Master Fader (Target Object: ProjectInformation, Target Property: Opacity).

This Block is target based. The Target value is displayed as main setting in the Block's header.

After defining the Target Settings, the Block has one PortIn for the new value to which the defined property of the object is to be changed to.

The Block can be bound to a Timeline element (Content), an ActionPad element or a general property.

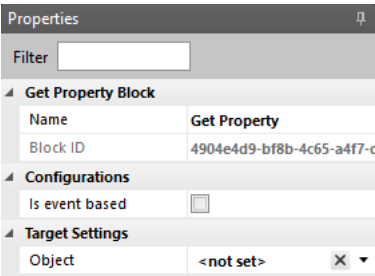
- Define the target (Property "Object").
- Define the Target Property.
- Define the new value.

In addition, select whether the new value is a fixed value (Input Type "Property") or a received Input value (Input Type "Port"). When set to Input Type "Property", the Block gets the additional EventIn "Perform Change" so that the change can be applied.

- To be able to set property information of an element, the element needs a defined Identifier name for the Block to be able to bind to the element. Most elements automatically have a defined Identifier. Effect components do not have a defined Identifier. See the „[Example: Getting Effect Opacity Value](#)“ on page 393 on how to bind Blocks to a property of an element with undefined Identifier.



11.15.90 Get Property



Use this Block if you want to get the value of an object. The Block gets a property from an object and provides it as value. An object can for example be an opacity value of an effect or the master fader.

Example: fetch the color selection from a Color Picker on an ActionPad page and provide the color value as ProPhotoColor value.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has no Ports. After defining the Target Settings, the Block has one PortOut for the defined property.

The Block can be bound to a Timeline element (Content), an ActionPad element or a general property.

- Define the target (Property "Object").
- Define the Target Property.

Category	Property	Description
Target Settings		Define the target to bind the Block to (object) and the property you want to get from the target.
	Object	Define the target to bind the Block to. Objects are all editable properties of the currently running SYMPHOLIGHT project.
	Target Property	Define the property of the selected target to receive. Depending on the selected type of object, the available list of properties varies.

Output Value

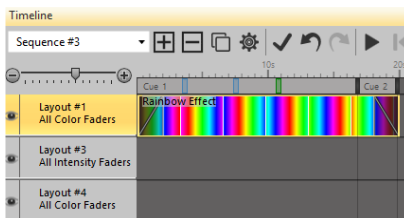
The data type and Port mode of the created PortOut, defined by SYMPHOLIGHT.

- To be able to get property information of an element, the element needs a defined Identifier name for the Block to be able to bind to the element. Most elements automatically have a defined Identifier. Effect components do not have a defined Identifier. See the example below on how to bind Blocks to a property of an element with undefined Identifier:

Example: Getting Effect Opacity Value

Get the opacity of an Effect, e.g. Circular Rainbow, by defining the Identifier of the Effect in the Content Tab first to make the Effect available as an Object. Now bind the Get Property block to the property of the Effect and you receive the current opacity value:

1. Go to the Content Tab and select the Effect of which you want to get the opacity value. In this example, select the Circular Rainbow effect in Sequence #3:



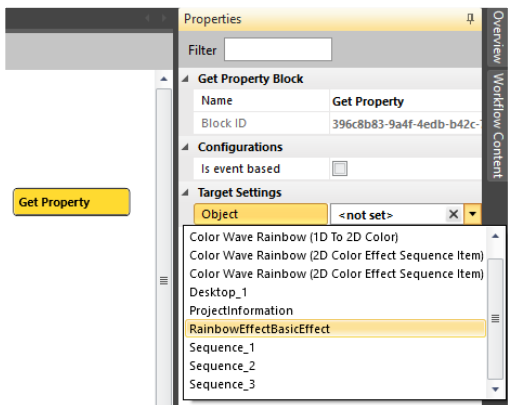
2. In the Properties window of the Effect, go to the Identifier property of the first Effect (Blended 2D Color Effect) and define the name / value. Here, the given Identifier value is RainbowEffectBasicEffect:

Effect	Blended 2D Color Effect
Display Name	
Identifier	RainbowEffectBasicEffect
Opacity	100
BlendMode	Normal
Display Name	
Identifier	

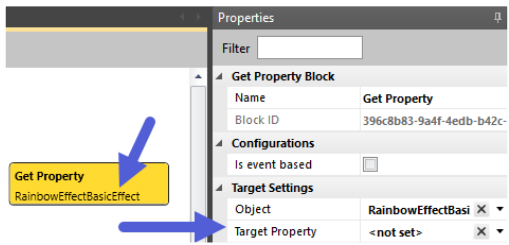
3. Go to the Workflow Designer, Automation Tab and add the Get Property block to the Workspace.



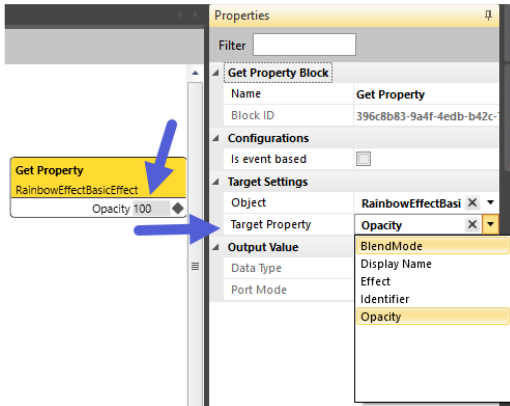
4. In the Properties window of the Block, select the RainbowEffectBasicEffect in the Object property:



Now the Block is bound to this Content element and provides a list of available properties at Target Property:



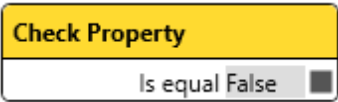
5. Select Opacity as Target Property. Now the Block has a PortOut that provides the current opacity value of the Rainbow Effect:



The Get Property block now provides the current opacity value of the Rainbow Effect. To change this value, use the Change Object Property block and to check the value by comparison with another value, use the Check Object Property block.



11.15.91 Check Property



Properties

Filter

Check Property Block

Name	Check Property
Block ID	e143796a-2470-4cae-9b01-

Configurations

Is event based ☐

Target Settings

Object <not set> X

Is equal

Data Type	Bool
Port Mode	Active

Use this Block if you want to compare the value of an object property. The Block compares a property of an object against a value. It provides a bool value as result from the comparison. An object can for example be an opacity value of an effect or the master fader.

This Block is target based. The Target value is displayed as main setting in the Block's header.

When orphan, the Block has no Ports. After defining the Target Settings, the Block has one PortIn for the value to which the defined property of the object is to be compared and one bool PortOut for the result of the comparison.

The Block can be bound to a Timeline element (Content), an ActionPad element or a general property.

- Define the target (Property "Object").
- Define the Target Property.
- Define the value to which the defined property of the object is to be compared (Category "Comparand").

In addition, select whether the comparand is a fixed value (Input Type "Property") or a received Input value (Input Type "Port").

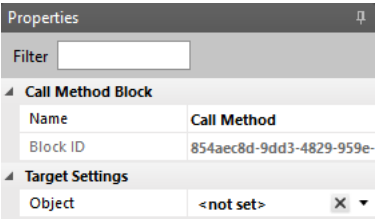
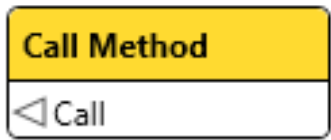
Category	Property	Description
Is event based		Enable to execute this Block on event triggers.



- The Is condition met PortOut forwards the result of the comparison: TRUE when the condition of the comparand gets identical to the Target Property.
- To be able to get property information of an element, the element needs a defined Identifier name for the Block to be able to bind to the element. Most elements automatically have a defined Identifier. Effect components do not have a defined Identifier. See the „[Example: Getting Effect Opacity Value](#)“ on page 393 on how to bind Blocks to a property of an element with undefined Identifier.



11.15.92 Call Method



Use this Block to call a function of an object.

Example: Play/Pause/Stop of a Sequence, or Cue Control for the Sequence.

This Block is target based. The Target value is displayed as main setting in the Block's header.

After defining the Target Settings, the Block has where applicable one PortIn for the value of the called method and one EventIn to trigger the method call.

The Block can be bound to a Sequence or to a Mutex Group.

- Define the target (Property "Object").
- Define the Target Method.
- Define the new parameter's PortIn value where applicable.
- Please note: To be able to call a method of an element, the element needs a defined Identifier name for the Block to be able to bind to the element. Most elements automatically have a defined Identifier. Effect components do not have a defined Identifier. See the on how to bind Blocks to a property of an element with undefined Identifier.



11.16 Scripting



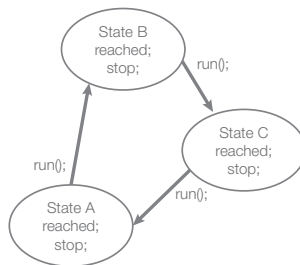
Using Rest API and Scripting requires prior knowledge of these topics.

In the Scripting tab, you can create macros in the C# language. The macros are compiled and can be executed in the Workflow Designer. For example allows the Scripting RDM access, video import or sending custom DMX commands / values to fixtures, overwriting Content / Output Windows' values. In general, the full C# language scope can be used. There are some limitations for Scripts in SYMPHOLIGHT regarding the use and runtime behavior, compared to standalone C# applications in Windows:

- Do not use threading. You cannot control threads that are created by your Script.
- Do not create never ending program types (like for(;;)), as they may consume too much memory and there is no real control over your Script.

The best method to create Scripts in SYMPHOLIGHT is to use the concept of state machines.

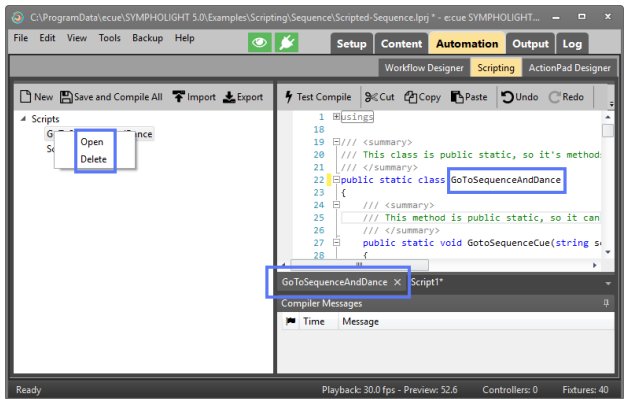
A Script starts to execute and does only the necessary things. Then it stops until it is called again to do the next things. This minimizes the runtime for the Script and allows better control over your Script.



To create and edit Scripts, use the Scripting tab in the Automation Window.



To create a new script, click  New.

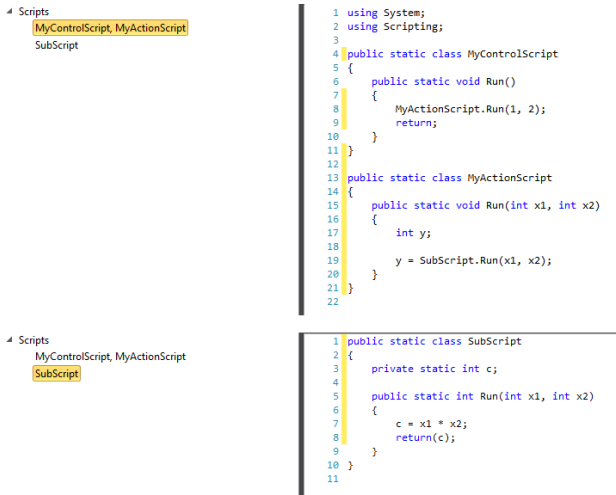



To edit a Script, double-click it in the list of Scripts, select Open in the context menu, or select its tab. The tabs for the opened scripts are at the bottom of the editor window, you can also close a script with the cross in the tab line.


You can also import Scripts and export Scripts. Click  Import or  Export in the tool bar.

Select Delete in the context menu to delete a script.

One Script may contain more than one class. You can also define classes in other Scripts and refer to them:




To test a Script for errors (syntax check of the script), click  **Test Compile** in the top menu. In the Compiler Messages you find the debugging feedback. The Script is not applied to the project yet.

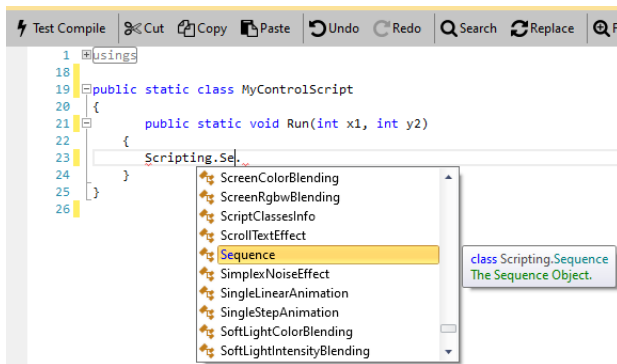
To compile, save, and apply a Script, click  **Save and Compile All**. After a Script compiled successfully, the Script is available in the Workflow Designer, where the defined input and output parameters are used as PortIns and PortOuts.



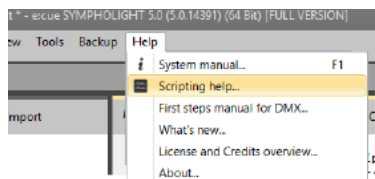
Please note that the scripting compiles all scripts into one assembly to allow the scripts to reference each other. Therefore it is not possible to compile only working scripts. If you have an erroneous script amongst working scripts, you will be informed about the problem and the workflow uses the previous (working) state.

At first, the name of the Script will be system-defined. Change the name of the class inside the Script to a new unique name. After a first  **Save and Compile All**, the name will be changed in the list of Scripts also.

Access to the objects of SYMPHOLIGHT happens via interface classes. While you type, you will get an overview of objects, methods or devices that are available at this point. Press **Ctrl + Space** to get a help overview.



The Help Reference is available at F1 from within the editor or at any time at Scripting Help under Help in the main menu.



Run a script, and a method also, directly from the Workflow Designer by using the Script bound Scripting blocks.



For older Scripting versions, simply add the using Ecue.Sympholight.Scripting.Api.

11.17 Examples for SYMPHOLIGHT Scripting

Example files of Scripting are stored at

C:\ProgramData\ecue\SYMPHOLIGHT 5.0\Examples\Scripting.

Check also the download area of the e:cue website.



When you copy and paste the examples shown below into a source code editor, you need to replace the quotation marks.

11.17.1 Adding an Entry in the SYMPHOLIGHT Logs

This example is the default code in every newly generated script.

Create a new script by clicking New in the left subwindow of the Scripting Editor. Rename the public static class Script1 to public static class DebugTest. This is the name with which this script will be available in the Workflow Editor later. Use the following code:

```
// using Scripting; // The deprecated scripting API necessary to access
// ActionPad and Effects.
using Ecue.Sympholight.Scripting.Api;
using System;

/// <summary>
/// This class is public static, so it's methods can be accessed from
/// the workflow designer.
/// </summary>
public static class DebugTest
{
    /// <summary>
    /// This method is public static, so it can be accessed from the
    /// workflow designer.
    /// </summary>
    public static void Run()
```

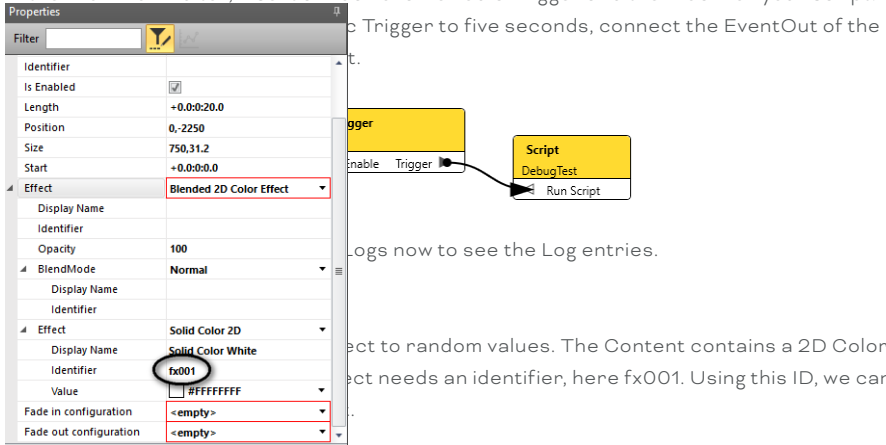



```

{
    // Use "ecue." to access SYMPHOLIGHT scripting functionality
    ecue.Logging.Info("This is a test log message generated by
your script.");
}
}

```

In the Workflow Editor, insert a Block of a Periodic Trigger and the Block for your script.



The screenshot shows the Workflow Editor interface. On the left, the 'Properties' panel is open, displaying the settings for a selected block. The 'Effect' property is set to 'Blended 2D Color Effect'. The 'BlendMode' is set to 'Normal'. The 'Effect' is set to 'Solid Color 2D'. The 'Display Name' is 'Solid Color White'. The 'Identifier' is 'fx001'. The 'Value' is '#FFFFFF'. The 'Fade in configuration' and 'Fade out configuration' are both set to '<empty>'. On the right, the workflow diagram shows a 'Trigger' block (yellow) connected to a 'Script' block (yellow). The 'Script' block is labeled 'DebugTest' and 'Run Script'. The 'Trigger' block is labeled 'Trigger' and 'Enable Trigger'. The 'Script' block is connected to the 'Trigger' block via a line with a dot at the end.

```

using System;
using Scripting;
using Ecue.Scripting.Compatibility;

```

```

public static class ColorFx

```

```

{
    public static void Run()
    {
        SolidColor2DEffect effect =
            ContentObjectAccessor.GetObject("fx001")
            as SolidColor2DEffect;
        if (effect != null) // if successful
        {
            Random rgbval = new Random();
            // RGBA, floats between 0 and 1
            effect.Value = new ProPhotoColor(
                (float)rgbval.NextDouble(),
                (float)rgbval.NextDouble(),
                (float)rgbval.NextDouble(),

```



```

        1.0f); // -> alpha is constant
    }
}

```

11.17.3 Handling UDP Messages

This script receives a command via a UDP message and controls a Sequence. See comments for details.

```

using Ecue.Contracts;
using Ecue.Contracts.Workflow;
using Ecue.Contracts.UserAccess;
using Ecue.Contracts.Scheduler;
using Ecue.Contracts.Rdm;
using Ecue.Contracts.Logging;
using Ecue.Contracts.Dmx;
using Ecue.Contracts.Exceptions;
using Ecue.Contracts.Devices;
using Ecue.Contracts.Content;
using Ecue.Contracts.Content.Effects;
using Ecue.Sympholight.Scripting.Api;
using System;

public static class UDP_Handler
{
    // Evaluates strings received
    // Received strings must have the following structure:
    // Action-SequenceIdentifier,
    // where "Action" is a Sequence action (Play, Pause, Stop)
    // and SequenceIdentifier is a valid Sequence ID
    //
    // Further error handling (e.g., handling of wrong or invalid
    // parameters) is recommended but not part of this script sample
    public static string Run(String UDP_Content)
    {
        string retVal = "";
        if (UDP_Content == "")
            // check if submitted string is empty,
            // if empty a warning is returned.
            // if message is valid, further processing will be done
        {
            retVal = "An empty message has been received. Please
            check.";
        } else
        {
            string[] splittedString = UDP_Content.Split('-');
            // Split string at each "-", the string before "-"

```




```

// is saved in splittedString[0] and the other string is
// saved in splittedString[1]
if (splittedString.Length < 2)
{
    //without 2 Strings we cannot proceed
    retVal = "Could not split the string \"" + UDP_Content +
    "\"" into two parts. Please check.";
} else
{
    if (splittedString.Length > 2)
    {
        // we can proceed, but something unexpected happened.
        retVal = "The string \"" + UDP_Content + "\"" has been
        splitted into " + splittedString.Length + " parts.
        \n\r" +
        "Proceeding with the script. Please check.";
    }

    var seq = ecue.Sequences.GetSequenceByIdentifier
    (splittedString[1]);
    if (seq == null)
    {
        // we have to check, if there is a Sequence with the
        passed name available
        retVal += "could not find Sequence \"" + splitted
        String[1] + "\". Please check.";
    }
    else
    {
        switch(splittedString[0])
        // check and perform the action that has been
        // submitted in the string
        {
            case "Start":
                seq.Play();
                retVal += splittedString[1] + " started";
                break;
            case "Pause":
                seq.Pause();
                retVal += splittedString[1] + " paused";
                break;
            case "Stop":
                seq.Stop();
                retVal += splittedString[1] + " stopped";
                break;
            default:
                retVal += "invalid string submitted";
                break;
        }
    }
}
}

```

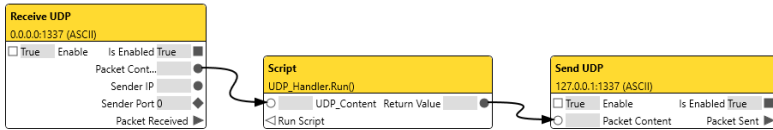



```

    }
    return retVal;
}
}

```

This is the configuration in the Workflow Designer:



11.18 Supported Function Param Types

The following list contains all supported function parameter types for the Scripting Run(...) method.

```

using System;
using Scripting;

```

```

/// <summary>
/// This script test all supported function param types for scripting Run(...) method
/// </summary>

```

```

public static class TestInOut
{
    public static bool Run(
        bool vBool,
        out bool ovBool,
        ref bool rvBool,

        byte vByte,
        out byte ovByte,
        ref byte rvByte,

        char vChar,
        out char ovChar,
        ref Char rvChar,

        DateTime vDateTime,
        out DateTime ovDateTime,
        ref DateTime rvDateTime,

```




```
float vFloat,  
out float ovFloat,  
ref float rvFloat,  
  
Int32 vInt32,  
out Int32 ovInt32,  
ref Int32 rvInt32,  
  
Int64 vInt64,  
out Int64 ovInt64,  
ref Int64 rvInt64,  
  
double vDouble,  
out double ovDouble,  
ref double rvDouble,  
  
string vString,  
out string ovString,  
ref string rvString)  
{  
  
    vBool = !vBool;  
    ovBool = vBool;  
    rvBool = vBool;  
  
    vByte++;  
    ovByte = vByte;  
    rvByte = vByte;  
  
    vChar++;  
    ovChar = vChar;  
    rvChar = vChar;  
  
    vDateTime = DateTime.Now;  
    ovDateTime = vDateTime;  
    rvDateTime = vDateTime;  
  
    vFloat++;  
    ovFloat = vFloat;  
    rvFloat = vFloat;
```




```
    vInt32++;  
    ovInt32 = vInt32;  
    rvInt32 = vInt32;  
  
    vInt64++;  
    ovInt64 = vInt64;  
    rvInt64 = vInt64;  
  
    vDouble++;  
    ovDouble = vDouble;  
    rvDouble = vDouble;  
  
    vString += "1";  
    ovString = vString;  
    rvString = vString;  
  
    Logger.Info("TestInOut:"+vString);  
    return true;  
}  
}
```

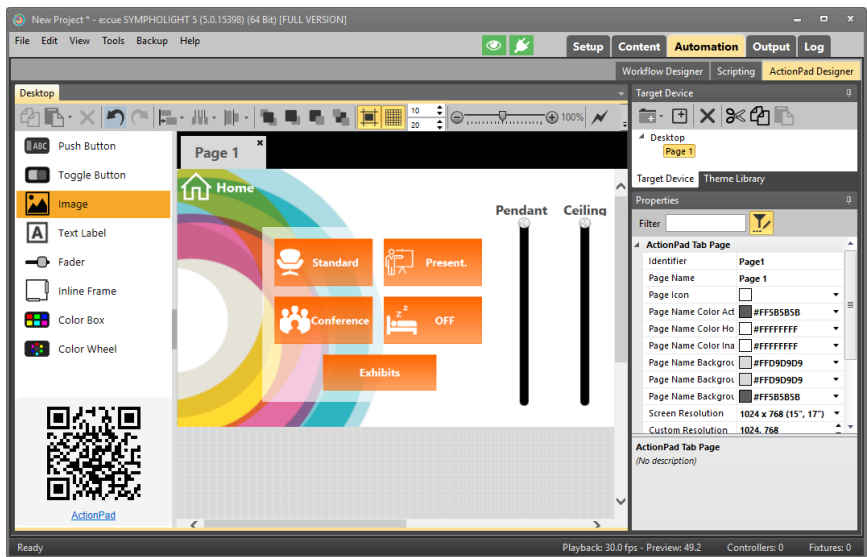


11.19 ActionPad Designer

With the ActionPad Designer, you can build a graphical user interface for SYMPHOLIGHT, called the ActionPad. You can connect to SYMPHOLIGHT via the build-in HTTP server and control SYMPHOLIGHT over a LAN or WiFi connection.



Make sure that the web server is enabled in the Tools menu of SYMPHOLIGHT. Also adjust the network parameters in the main menu of SYMPHOLIGHT in Tools - Application Settings.



Target selection and New Theme editing
Page Control to structure the Workspace

Target Device,
Theme Library

Element
Library
Control
Elements

Workspace
with ActionPad Pages

Properties
Edit element
properties like
appearance

QR Code

To place Control Elements on the Page, drag-and-drop Elements from the Element Library onto the Workspace. Resize it to fit your needs. Also adjust its properties in the Properties Window as desired.

A special Element is the Inline Frame. After placing it on the Page and selecting it, enter an URL in the Properties for the Inline Frame. The content of the web page will be displayed in the frame.

These Elements are available in the Element Library:

ActionPad Element	Description	Link to corresponding Workflow Block
Push Button	A button that sends an event in the Workflow Designer when pressed and when released.	p.156
Toggle Button	A button that locks and releases when pressed. Sends an Toggle On and Toggle Off event.	p.160
Image	An image from the Media Library. You can upload a new image in the dialogue when selecting the Path.	p.164
Text Label	A text Element.	p.171
Fader	A fader Element, horizontal or vertical orientation. It has a PortIn to set the fader, and a PortOut to provide the fader position (only sent on change).	p.173
Inline Frame	An area on the desktop of the Page. Content is fetched from a web site with the URL defined in its properties.	p.176
Color Box	A color picker as square. Select with Mapping in the properties the target of the Color Box, usually an SYMPHOLIGHT Solid Color effect, e.g. Solid Color White from the Factory Effects Library.	p.178
Color Wheel	Like Color Box, but as a wheel.	p.178



Text Input A text input field. Content is entered in the ActionPad and processed via the Workflow Designer. [p.169](#)

Password Input A password input field to verify an entry. Characters can be hidden and entry obligatory. [p.166](#)

At the Page Control bar in the top menu of the ActionPad Designer, you can align selected Elements or copy and paste Elements. Test the look and feel of your ActionPad within SYMPHOLIGHT by clicking the flash icon in the top toolbar. Clicking on the earth icon, your standard browser will be opened and redirected to this ActionPad Page. These Page Control functions are available:



Copy selected Elements. (Ctrl + C)



Paste the copied or cut Elements into the Workspace. (Ctrl + V)



Delete selected Elements.



Undo last change.



Redo last Undo command.



Align selected Elements. Choose from the drop-down menu the direction of alignment: left, right, top, bottom, horizontal center, vertical center.



Space selected Elements evenly. Choose from the drop-down menu the axis of spacing: horizontal, vertical.





Distribute selected Elements. Choose from the drop-down menu the reference point of distribution: left edge, right edge, top edge, bottom edge, horizontal center, vertical center.



Bring Element to the front.



Bring Element forward.



Send Element backward.



Send Element to the back.



Enable / Disable snapping to grid: toggles snapping of Elements to the grid in the Workspace.



Enable / Disable displaying the grid.

Change grid width and height with the entry fields.




Zoom in and out of the Workspace (min. 10%, max. 400%).




Switch between build mode and view mode. Use view mode to test the look and feel of the ActionPad within SYMPHOLIGHT.



Open the ActionPad Page in the standard browser.

To create a new Page for the ActionPad Designer, click on the  icon in the Target Device Window. You can now set all other properties for the Page here, like background color or grid usage. Depending on the selected element in the ActionPad Designer, the Properties Window will reflect the properties of the selected element.

To create Pages for different Target Devices (devices on which the ActionPad is used on), click on the  icon in the Target Device Window and select the target display. A new Target Device including one Page is created.



These Target Device functions are available:



Create new Target Device. Select a device type: Desktop, iPad or Android Tablet.



Create new ActionPad Page.



Cut selected Pages. (Ctrl + X)



Delete selected Target Devices or Pages.



Copy selected Target Devices or Pages. (Ctrl + C)



Paste the copied or cut Target Devices or Pages into the Target Device Window. (Ctrl + V)

The Theme Library, sharing the place with the Target Devices, offers predefined design themes of ActionPad properties and the possibility to create your own designs. When creating a new Theme, you can configure the standard appearance of the following Elements: Push Button, Toggle Button, Text Label, Fader, Color Box, Color Box, Color Wheel, the ActionPad Tab Page and Background. Decide on which Pages and Elements to apply a Theme. Theme designs are device type independent: they can be applied on any Target Device. ActionPad Theme files use the extension *.appt.



Create new Theme template. The new Theme template opens in a new tab.



Import Theme template.



Export Theme template.



Delete selected Theme designs.





Edit Theme template. The Theme design opens in a new tab.



Create a new Page in the selected Theme design. Only applied on ActionPad Tab Page and Background properties.



Apply selected Theme design on currently selected Page. Only applied on ActionPad Tab Page and Background properties.



Apply selected Theme design on selected Elements.



Apply selected Theme design on all Pages and Elements of the selected Target Device.

All Target Devices and Themes in editing mode have their own tabs, accessible at the Target Selection / New Theme Editing bar above the Page Control bar.

11.19.1 Element Properties

All properties regarding the look and feel, like colors, font types and sizes, orientation and more are self-explaining due to the property names. Additionally, there are help text comments in the small window below the properties. As the ActionPad uses HTML5 to create the browser content, there are many options for styles and behavior. Some properties are not self-explaining, so here is more information about these special properties.

Text End Ellipsis	If the text in a button is too long, the name will be shortened and ends with "...".
Tool Tip	A text that gets displayed in the web browser as help when the user hovers over a button.
Range Step	Property for faders, allows to increase the positions in defined steps only.

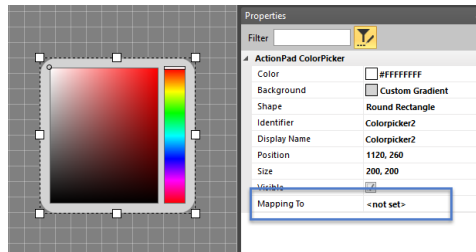
When editing text like button text, **Shift + Enter** will create a line break. **Enter** alone will commit the change.



11.19.2 Assigning Actions to GUI Elements

After placing and defining the ActionPad Pages with buttons and faders, you have to assign an action to the Elements in the Page. For buttons and faders, this is done in the Workflow Designer. See the previous chapter about the Workflow Designer. Use the “State configuration” at the “IsPressed” property to link buttons to objects.

The Color Pickers do not create events. In the Properties Window for a Color Picker, use the Mapping property to assign the Color Picker to a Solid Color Effect, e.g. Solid Color White from the Factory Effects Library in the Timeline of a Sequence. For Color Pickers and Color Picker Wheels, the Workflow Designer provides also Workflow Blocks.



11.19.3 Accessing the ActionPad on Mobile Devices

In the lower left part of the ActionPad Designer, there is a QR field with the URL of the ActionPad on this server.



Scan the QR code with an iOS or Android application and the mobile device can open the ActionPad by the given URL.

If you already plan to use a specific device for remote control, select the right desktop sizes in the Target Device list of the ActionPad Designer. There are lots of predefined desktop sizes in the ActionPad Designer.

If more than one device project has been created, the individual devices can be accessed by using the correct URL. The scheme is as follows:

`http://[IP]:[Port]/ActionPad2/ActionPad2.html?idx=0`

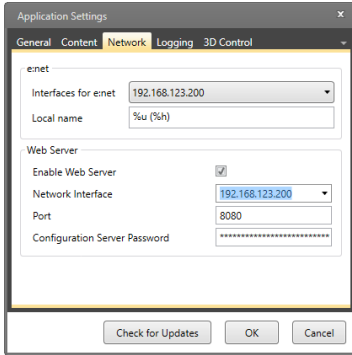
Where [IP] is in the following format: 192.168.123.101, [Port] is the same value as configured in the SYMPHOLIGHT Settings and the value for idx (zero based) is matching the position of the device in the Target Device Window of the ActionPad Designer Window.

11.20 SYMPHOLIGHT Remote Console

You can also access SYMPHOLIGHT and the ActionPad via the Remote Console. Use the values defined in SYMPHOLIGHT's system settings.



With your web browser, use the system's IP address and port as URL and enter it in the



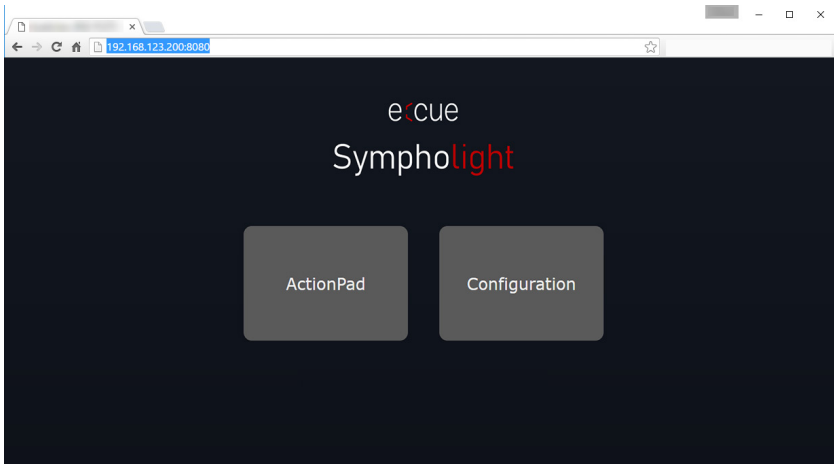
browser's address line. Here:

`http://192.168.123.200:8080`

The Server Password defined in the settings is required to access the Remote Console. The default setting is "ecue".

Log-in to the Remote Console with user = admin and the given password.

Next the Remote Console gets shown.

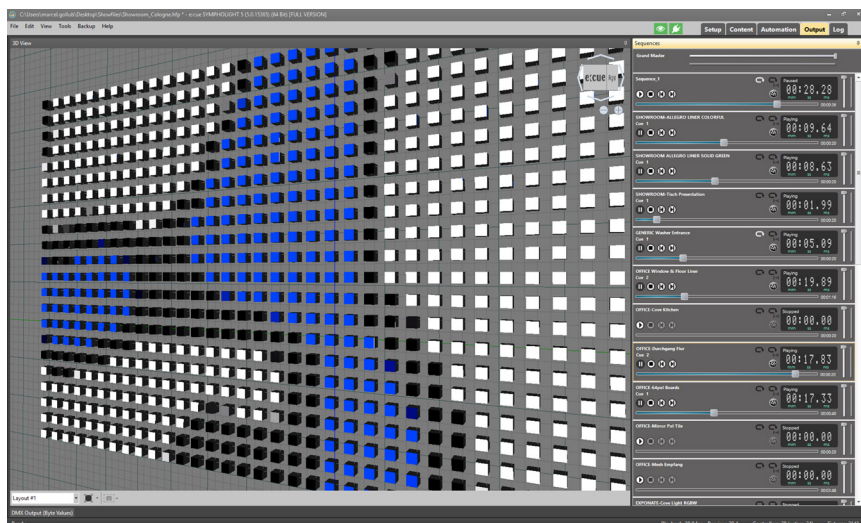


From here, you can call the ActionPad, or set some system settings, or start and stop Sequences. This is in general the same user interface as for the SYMPL Fusion or SYMPL Core S(p). See „13 SYMPL Fusion & SYMPL Core S(p)“ on page 420 for more details.

12 Output Window

12.1 Principles

The Output window shows the real results being sent to the Layouts, which are the fixtures respectively.



In the center, there is a preview of the currently running and viewed Layout. If there are several Layouts, you can select the one to view. In the Preview area, you can zoom in and out with the mouse wheel. Below the Output view, the DMX data is displayed. You can select either byte values or raw RGB. Selection is made with the two tabbed areas below the DMX Output window.

On the right side window, there is the Play Control Window. You can start, pause or stop the Sequences. This allows checking and testing of shows and configurations. Additionally you can set Opacity in every run control with the fader on the right. Above all play controls, the grand master sets intensity for all fixtures and Sequences.

You can rearrange the player fields by keeping a player clicked and moving it to another place. A yellow bar shows the resulting position.

The Loop icons on the right side of the players allows looping. There is one icon for Sequence Loop and another icon for Sequence Segment Loop. Both icons are linked with the Sequence Properties of the individual sequences. If you want to use the Loop Segment feature, please make sure the Loop Segment is enabled and defined in the respective Sequence Property. You can access the Sequence Property in the Content Tab. Both Loop Properties are saved in and loaded from the Project File.

The Finish icon disables all loops in the Sequence until the Sequence reaches its end and stops.

12.2 Player Window Icons, Commands and Functions

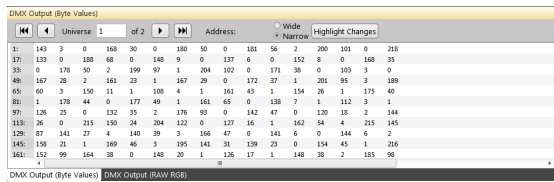


- 1 Play controls, play/pause, stop, skip to previous Cue, skip to next Cue.
- 2 Current position in Sequence, move to change position.
- 3 Total length of Sequence.
- 4 End loop when reaching end of Sequence.
- 5 Enable Sequence Loop mode.
- 6 Enable Loop Sequence Segment.
- 7 Submaster with visualization for this Sequence. The visualization is showing the live value of the Submaster, for example if it is influenced by the Workflow Designer and/or a fader on in the GUI Editor.
- 8 Grand Master with visualization. The visualization is showing the live value of the Grand Master, for example if it is influenced by the Workflow Designer and/or a fader on in the GUI Editor.
- 9 Current play position of the Sequence. The time format will be adapted automatically for Sequences longer than 60 minutes and longer.

12.3 Z-Position and Priority

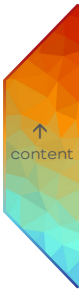
The position of a player determines its priority. The higher the position of the player, the higher the priority of the generated output.

12.4 DMX Output Window



Address	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	143	3	0	168	30	0	180	50	0	181	56	2	200	101	0	218
17	133	0	188	68	0	143	9	0	137	6	0	132	8	0	168	35
33	0	178	50	2	199	97	1	204	102	0	171	38	0	103	3	0
49	167	28	2	161	23	1	167	29	0	172	37	1	201	95	3	189
65	60	3	150	11	1	108	4	1	161	43	1	154	29	1	173	49
81	1	178	44	0	177	49	1	161	65	0	138	7	1	112	3	1
97	126	25	0	132	35	2	178	93	0	142	47	0	120	18	2	144
113	26	0	215	150	24	204	122	0	127	16	1	162	14	4	215	145
129	87	141	27	4	140	39	3	166	47	0	141	6	0	144	6	2
145	158	21	1	169	46	3	195	141	31	139	23	0	154	45	1	216
161	152	99	164	18	0	148	20	1	126	17	1	148	35	2	165	98

When viewing byte values, you can see the real byte values sent as DMX values. Select the currently viewed addresses and universe with the option left above the window. Select a wide or narrow display of data. In narrow display, the rows are limited to 16 byte ranges. You can also highlight changed values by displaying them in red.

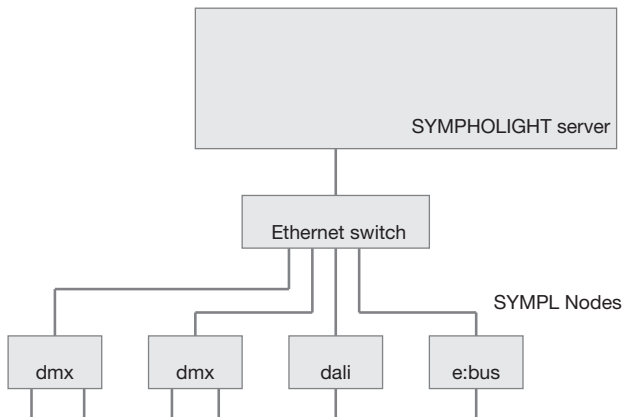


13 SYMPL Fusion & SYMPL Core S(p)

SYMPL Fusion and SYMPL Core S(p) are SYMPHOLIGHT instances which run the same SYMPHOLIGHT application, i.e. project, as a server but in a different environment. They cannot be used to develop a project, but run a completed project in a runtime-only environment. The main use of SYMPL Fusion and SYMPL Core S(p) are applications that do not require the performance of a PC-based SYMPHOLIGHT server but the same functionality. As there is no user interface like on a SYMPHOLIGHT server, the SYMPL Fusion and SYMPL Core S(p) are controlled via the Action Pad, connected input devices, UDP, REST API or the Remote Control in a browser (Web Interface).

13.1 System Overview

For commissioning, connect the SYMPL Fusion or SYMPL Core S(p) to the network of the SYMPHOLIGHT server.



After booting, the SYMPL Fusion or SYMPL Core S(p) will become visible in the Sympholights list of SYMPHOLIGHT. The SYMPL Fusion or SYMPL Core S(p) is not listed as an interface device like the other SYMPL Nodes and Butlers, but has an own chapter in the Sympholights list.

Like for any other device, the main properties for the SYMPL Fusion or SYMPL Core S(p) are editable in the Properties Window. Only values in black can be changed, like the name and network properties. The SYMPL Fusion or SYMPL Core S(p) default IP address is 192.168.123.100/24, if available. An active external DHCP server has precedence. The fallback to a random IP in the 192.168.123.0/24 subnet is used as a last resort. IP

addresses can be set to manual configuration, in this case be aware of possible IP address conflicts.

A right-click on the SYMPL Fusion or SYMPL Core S(p) in the Sympholights list opens a context menu for additional function for the SYMPL Fusion or SYMPL Core S(p)



Before assigning a project and setting all properties of the SYMPL Fusion or SYMPL Core S(p) you have to configure the network and assign a Password. With the context menu of the SYMPL Fusion or SYMPL Core S(p) in the Device Window of the Setup tab open a browser for this purpose.

Over the context menu of the SYMPL Fusion or SYMPL Core S(p) the following functions are available:

Download project

Download the current project file from the SYMPL Fusion or SYMPL Core S(p). The project remains unchanged on the SYMPL Fusion or SYMPL Core S(p). The device must support version 5.5 or higher.

Upload project

Upload a project to the SYMPL Fusion or SYMPL Core S(p). Set a password in the Web Interface beforehand, otherwise the upload will not run. Background images of Layouts are not included when uploading a project onto a SYMPL Fusion or SYMPL Core S(p). It is therefore recommended to keep a local project backup from the original SYMPHOLIGHT instance which contains the background images, at a safe place. Click the Handover & Close button when you want to immediately start the project on the SYMPL Fusion or SYMPL Core S(p) and its devices.

Delete currently loaded project

Deletes the currently loaded project from the SYMPL Fusion or SYMPL Core S(p)

Redirect modules and input events to this instance

SYMPL Fusion only: Hand over the control over the modules and inputs to the local instance.



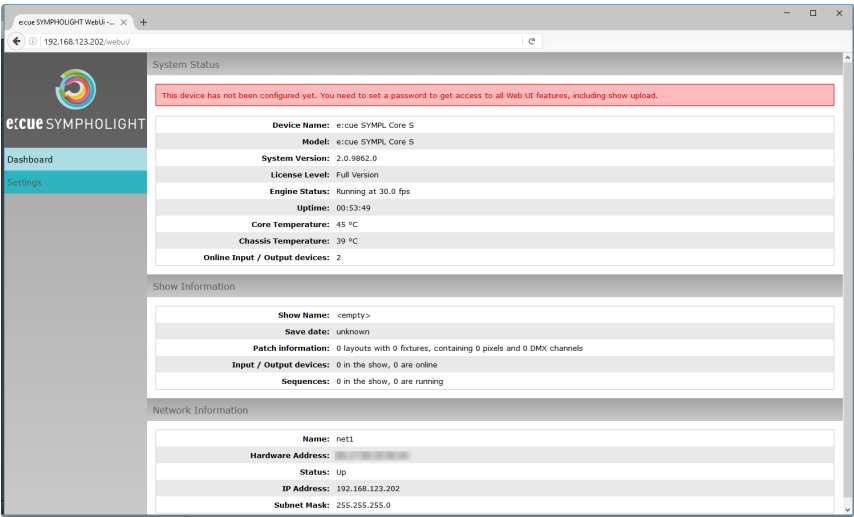
Give back the modules and input access to the device	SYMPL Fusion only: Return the control over the modules and inputs back to the SYMPL Fusion.
Mute controller detection	Select to become invisible for other SYMPHOLIGHT instances in the network. Click again to unmute.
Share License	Use the SYMPL Fusion or SYMPL Core S(p) as License Key. For details see „ 13.6 Using SYMPL Fusion or SYMPL Core S(p) as License Key for SYMPHOLIGHT “ on page 432.
Restart controller	Restart the SYMPL Fusion or SYMPL Core S(p) e.g. in case of project problems.
Reset controller to factory defaults	Reset all settings and configuration to the factory defaults. This will also delete the project.
Download log files	Download the log files of the SYMPL Fusion or SYMPL Core S(p) via a web browser.
Open Web UI in Browser	Configure and control the SYMPL Fusion or SYMPL Core S(p) in a web browser.
Open Rest API (Swagger) in Browser	Configure and control the SYMPL Fusion or SYMPL Core S(p) in a Rest API (Swagger).
Open Scheduler in Browser	Configure the Scheduler of the SYMPL Fusion or SYMPL Core S(p) in a web browser.
Configure Network	Configure the network settings of the SYMPL Fusion or SYMPL Core S(p).
Copy Instance ID	Copy the SYMPHOLIGHT instance ID to the clipboard.
Upload software	Upload a firmware file (e.g. updated version) of the SYMPL Fusion or SYMPL Core S(p). The device must support version 5.5 or higher.
Update software in browser	Update the firmware of the SYMPL Fusion or SYMPL Core S(p) via a web browser.



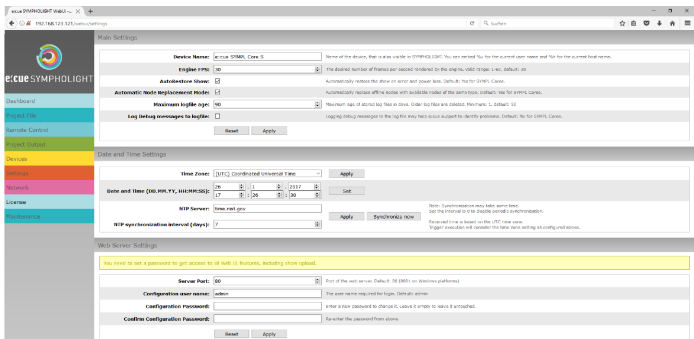
Before using the SYMPL Fusion or SYMPL Core S(p) in full functionality, select “Open Web UI in Browser” to start the default web browser on the SYMPHOLIGHT server.

13.2 Configuring the SYMPL Fusion or SYMPL Core S(p)

With a properly configured network, you will see the start screen. Select Dashboard from the left menu.

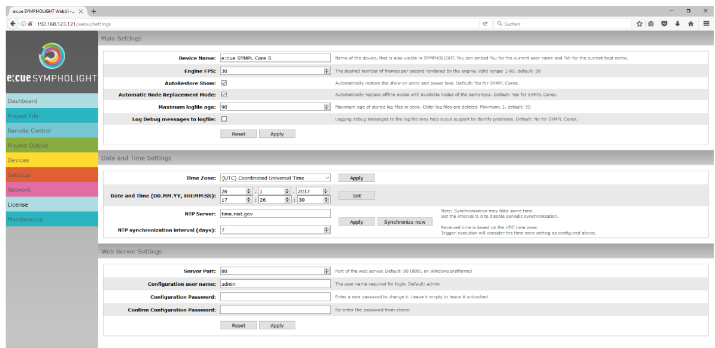


In the Dashboard, you see an overview about the SYMPL Fusion or SYMPL Core S(p) and the uploaded project. As no project is loaded when initially accessing the Dashboard, this part is empty. As the SYMPL Fusion or SYMPL Core S(p) is fresh out of the box, the system reminds you on assigning a password to this SYMPL Fusion or SYMPL Core S(p). Click on Settings to do so.



The editable parameters in the Settings window all have a description on the right side. Edit the basic system settings and click **Apply** to commit the changes. Assign a password to

this SYMPL Fusion or SYMPL Core S(p) and Click Apply. The system has now its basic configuration and the complete configuration is possible.



The full list of menu entries at the left:

Dashboard	Overview of the SYMPL Fusion or SYMPL Core S(p), firmware version, uptime, project details, input and output devices and more.
Project file	Upload and download a project to this SYMPL Fusion or SYMPL Core S(p) by drag-and-drop or upload. *.lprj and *.hfp formats are supported. By uploading the project from any source, no SYMPHOLIGHT server is necessary. Use an USB memory stick. Background images of Layouts are not included when uploading a project onto a SYMPL Fusion or SYMPL Core S(p). It is therefore recommended to keep a local project backup from the original SYMPHOLIGHT instance which contains the background images, at a safe place.
Remote Control	Other SYMPHOLIGHT instances from which remote control is possible. You can execute basic operations from Remote Control instances, for example to start and stop Sequences.
Project Output	Overview over running Sequences. Work with Sequences and Sequence opacity.

DALI	Indicator when the DALI system of a standalone SYMPL Fusion or SYMPL Core Sp is ready to be used. Only the expanded line is scanned and updated. Refresh automatically: rate is every 10 sec. Faster/ manual refresh: click “Expand” again.
Devices	SYMPHOLIGHT instances and devices, available or included, with updating option.
Test Mode	SYMPL Fusion only: Run tests on the connected modules.
Settings	Basic SYMPL Fusion or SYMPL Core S(p), date and time, web server, Rest API and backup settings. MQTT and external MQTT broker settings are only available when the Gateway Communication feature is activated.
Network	Basic and advanced network settings. Art-Net and sACN settings - you will need one credit for each physical DMX universe.
License	Order and upload credits for and to the SYMPL Fusion or SYMPL Core S(p). For more information see „4.2 Credit System“ on page 18 .
Maintenance	Reset, restart, update SYMPL Fusion or SYMPL Core S(p). Access log files. Capture network packets.



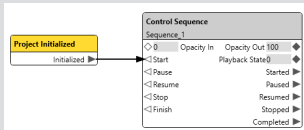
The MQTT interface is in the testing phase and should not be used in production environments. It is subject to ongoing changes without backward compatibility. Currently, the interface is intended exclusively for use with SymphoCity. Only available when the Gateway Communication feature is activated.

After assigning a SYMPHOLIGHT project to the SYMPL Fusion or SYMPL Core S(p) via the web interface, the SYMPL Fusion or SYMPL Core S(p) will begin executing the project. As long as the SYMPHOLIGHT server runs, the SYMPL Fusion or SYMPL Core S(p) cannot get SYMPL Nodes or other interfaces. As soon as the SYMPHOLIGHT server releases the devices, the SYMPL Fusion or SYMPL Core S(p) will take over the devices and integrate them as configured in the running project.





If you want the SYMPL Fusion or SYMPL Core S(p) or SYMPHOLIGHT server to start a Sequence or do anything else after system start, do not forget to include the Project Initialized Block in the Workflow Designer.



To control the SYMPL Fusion or SYMPL Core S(p), use the Action Pad or the Remote Control in a web browser.

13.3 SYMPL Fusion network configuration

The SYMPL FUSION Base Device provides two Ethernet ports that can be operated individually and configured for different subnets, or interconnected in Bridge mode.

Bridge mode (Daisy chain) - default

Use the Bridge mode to daisy chain SYMPL FUSION Base Devices or other SYMPL devices. The maximum number of devices is 10. In the Bridge mode of the SYMPL FUSION Base Device (factory default), its two Ethernet ports are connected via the internal network bridge. LAN 1: daisy chain IN, LAN 2: daisy chain OUT.



Design your network with a central switch. Do not design your network in a way that simply connects one SYMPL FUSION Base Device to another.

Use “Static” if you have a network plan where the IP addresses are predefined. Use “DHCP Client” if you use a central DHCP server for network management. Select “Automatic DHCP client or server” if you want the SYMPL FUSION Base Device to act as the DHCP server when no external DHCP server is present.



If you decide to use “Automatic DHCP client or server”, enable this mode on one of the SYMPL FUSION Base Device / SYMPL Core Sp devices. Having multiple SYMPL FUSION Base Devices with “Automatic DHCP client or server” in one network is not supported. The IP addresses provided by the integrated DHCP server cannot be configured further (e.g., static leases). Use an external DHCP server if you have this requirement.

Single mode (Standalone configuration)

Use the Single mode to use separate networks. In the Single mode of the SYMPL FUSION Base Device, its two Ethernet ports must be configured to use two non-overlapping subnets. Otherwise, routing errors will occur.



13.3.1 Default network properties

Bridge network:

A SYMPL FUSION Base Device has the following network configuration in the factory-new state:

- Bridge Mode is enabled. Any port can be used for the initial setup.
- DHCP is enabled.
- If DHCP is unavailable, the SYMPL FUSION Base Device uses the IP address 192.168.123.100 and the subnet mask 255.255.255.0.
- If the IP address 192.168.123.100 is occupied by another device, the SYMPL FUSION Base Device searches for a free IP address in the 192.168.123.x range with a subnet mask of 255.255.255.0.



Use Sympholight, a network sniffer, or your router's dashboard to find the Fusion in the network.

LAN 1:

Network Bridge member.

LAN 2:

Network Bridge member.

13.3.2 Network configuration with the web interface

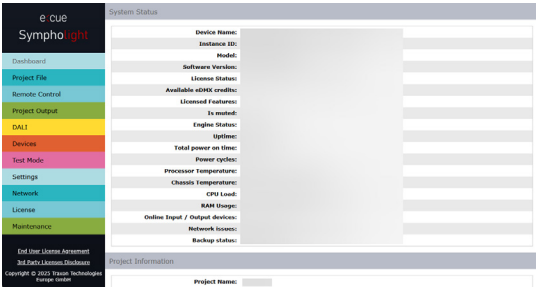
1. For the initial setup, locate the SYMPL FUSION Base Device on the network.
2. Open a common web browser on a connected PC. Enter the SYMPL FUSION Base Device's IP address into the address bar:
e.g. `http://192.168.123.100`.

The remote console of the device is displayed:



3. Select "Configuration" and the dashboard of the SYMPL FUSION Base Device is displayed:





4. On first access, define your login credentials for the web interface of the SYMPL FUSION Base Device under “Settings”.

Web Server Settings

You need to set a password to get access to all Web UI features, including project upload.

Server Port:	<input type="text" value="80"/>	Port of the web server. Default: 80 (8081 on Windows platforms)
Configuration user name:	<input type="text" value="admin"/>	The user name required for login. Default: admin
Configuration Password:	<input type="password"/>	Enter a new password to change it. Leave it empty to leave it untouched.
Confirm Configuration Password:	<input type="password"/>	Re-enter the password from above.

5. Configure the network settings to match your network topology and subnets.

Bridged Network configuration

Network Configuration:	<input type="text" value="Automatic DHCP Client or v"/>
Static IP Address:	<input type="text" value="0.0.0.0"/>
Static Subnet Mask:	<input type="text" value="0.0.0.0"/>
Static Gateway:	<input type="text" value="0.0.0.0"/>

LAN 1 [eth0] configuration

Network Configuration:	<input type="text" value="Network Bridge member v"/>
Static IP Address:	<input type="text" value="0.0.0.0"/>
Static Subnet Mask:	<input type="text" value="0.0.0.0"/>
Static Gateway:	<input type="text" value="0.0.0.0"/>

LAN 2 [eth1] configuration

Network Configuration:	<input type="text" value="Network Bridge member v"/>
Static IP Address:	<input type="text" value="0.0.0.0"/>
Static Subnet Mask:	<input type="text" value="0.0.0.0"/>
Static Gateway:	<input type="text" value="0.0.0.0"/>

- 5.1. Decide whether to use the bridge.
- 5.2. Define the setup for either the bridge or the individual networks. If you use the bridge, configure the network on the bridge and only configure the other ports as

bridge members. If you do not have a bridge, configure the networks individually (i.e. DHCP or static).

- If you deactivate the bridge, which is equivalent to there being no longer a bridge member, you must follow this sequence:
 1. Configure the port used to access the SYMPL FUSION Base Device to a new IP address that does not correspond to the bridge IP address.
 2. Reopen the web interface under the newly configured IP.
 3. Configure the other network port individually.
 4. Restart the SYMPL FUSION Base Device.
- If you activate the bridge after having two separate networks you must follow this sequence:
 1. Configure the port that you are not currently using to access the SYMPL FUSION Base Device as a bridge member, and set its other IP addresses to 0.0.0.0.
 2. Restart the SYMPL FUSION Base Device. The bridge will then appear for configuration.
 3. Configure the bridge as desired.
 4. Configure the port that you are currently using to access the SYMPL FUSION Base Device as a bridge member.

5.3. Click **Apply** to apply the settings.

The network configuration of the device is completed.



Do not connect a SYMPL FUSION Base Device configured for “Automatic DHCP client or server” to a company network or another network with critical infrastructure. In some cases, the SYMPL FUSION Base Device may not detect external DHCP servers and may start its own. This could result in severe network malfunction and significant recovery efforts.



Note down all employed and, if applicable, reserved IP addresses. Keep the information for future network configurations.

You can change the login password to the SYMPL FUSION Base Device's configuration page and its device name at “Settings” page. Do not forget the new password.

13.4 Art-Net and sACN on SYMPL Fusion or SYMPL Core S(p)

The SYMPL Fusion or SYMPL Core S(p) supports Art-Net and sACN output. You will need one credit for each physical DMX universe.

- To order credits, open the web interface and go to “License”. Send your order and upload the new credits to empower the Art-Net and sACN output.



e cue
Sympho
light

Dashboard

Project File

Remote Control

Project Output

DALI

Devices

Test Mode

Settings

Network

License

Maintenance

End User License Agreement
2nd Party License Disclosure
Copyright © 2019 Sympho Technologies
All Rights Reserved

License Overview

License Status: Full version without content management.

License Source: e-cue SYMPL Core Hardware Protection

License ID:

Available eMMS credits: 0

Licensed Features:

Order additional credits

To order additional credits for your system, enter the order details below and click Proceed.

Your Name:

Your Company:

Your e-mail address:

Your phone number:

Additional credits:

Additional features:

SymphoCity Scheduler

Content Exchange

Gateway Communication

SymphoCity Play Board

SymphoCity Energy Board

SymphoCity Fixture Board

Order

Order

Order

Order

Order

Order

Reset

Proceed

API to schedule events.

API to replace media files.

Gateway communication over MQTT

License for the SymphoCity Play Board

License for the SymphoCity Energy Board

License for the SymphoCity Fixtures Board

Upload ordered credits

After we have processed your credit order, you will receive a file, which applies the additional credits to your device. This file can be used only once and only for the device you entered the credits for.

Upload ordered credits: No file selected.

- To enable Art-Net and sACN output, go to “Network” and configure their settings accordingly.

Art-Net Settings

Enable Art-Net:

Enable ArtSync packets:

Optimize timing:

Apply

sACN Settings

Enable Streaming ACN:

Optimize timing:

Apply

13.5 Date, Time & NTP settings

In the web interface, set the date, time and NTP settings of the SYMPL Fusion or SYMPL Core S(p) Go to the Settings tab and set the properties according to your requirements.

Calibrate the hardware clock on NTP update: ☒

Calibrate the hardware clock on manual time set: ☐

Reset

Apply

Date and Time Settings

Time Zone: (UTC) Coordinated Universal Time

Apply

Date and Time (DD.MM.YY, HH:MM:SS):

Set

NTP Server:

Apply

Synchronize now

Manual update interval (minutes):

If you need to change both the time zone and the time, proceed as follows:

1. Change the time zone.
2. Restart the SYMPL Fusion or SYMPL Core S(p) device.
3. Adjust the time.

The SYMPL Fusion or SYMPL Core S(p) runs on an integrated clock that needs to be configured/calibrated. This is done via an NTP server. On the initial start-up of the SYMPL Fusion or SYMPL Core S(p) connect it to an NTP server (e.g. internet). Set the NTP server settings in the web interface of the Fusion.



- It is recommended to have an NTP server permanently present in the installation, especially in time sensitive projects, to ensure the correct time on the SYMPL Fusion or SYMPL Core S(p). The NTP server should also coordinate the full system.
- Otherwise, if less time critical, it is recommended to re-connect the SYMPL Fusion or SYMPL Core S(p) to an external NTP server (e.g. internet) once in a while, e.g. once a year to correct any time deviation.

Enter the IP address or host name of the appropriate NTP server. If the SYMPL Fusion or SYMPL Core S(p) can reach an Internet resource, you can use a public NTP server. Otherwise, an internal NTP resource can be defined. You can set a sync interval (10 ... 1440 min (24h)) to query the NTP server for the current time.

If NTP is not to be used, leave the NTP server entry empty.



- Restart the SYMPL Fusion or SYMPL Core S(p) after changing the time zone.
- Do not use manual update intervals for public NTP servers. This could result in the SYMPL Fusion or SYMPL Core S(p) being blocked or subject to rate limiting. Use short update intervals only with NTP server that accepts them.

When the time on the SYMPL Fusion or SYMPL Core S(p) is synchronized via Network Time Protocol (NTP) and the user manually adjusts the time in the web interface, the internal Real-Time Clock (RTC) drift is calibrated assuming the NTP or user-provided time is correct.

- The recommended approach is to use NTP, which provides a calibrated clock. Enable "Calibrate the hardware clock on NTP update".
- If you do not have access to an NTP server, you can manually calibrate the clock. Set the time as precisely as possible. Enable "Calibrate the hardware clock on manual time set".



- If you use NTP to set the local time to UTC and distribute time zone changes via NTP, disable both “Calibrate the hardware clock on NTP update” and “Calibrate the hardware clock on manual time set”.

13.6 Using SYMPL Fusion or SYMPL Core S(p) as License Key for SYMPHOLIGHT

The SYMPL Fusion or SYMPL Core S(p) can be used as License Key for the SYMPHOLIGHT Software. In order to transfer the License to another SYMPHOLIGHT instance, go to the Setup Tab of the SYMPHOLIGHT the License shall be transferred to. Open the context menu of the SYMPL Fusion or SYMPL Core S(p) in the Devices Window. Select Share License and enter the password for the SYMPL Fusion or SYMPL Core S(p) you’ve configured before. If you did not set a password for the SYMPL Fusion or SYMPL Core S(p) before, please refer to chapter „13.2 Configuring the SYMPL Fusion or SYMPL Core S(p)“ on page 423 for further instructions and set a password.

Once the License has been transferred successfully, the red “Trial Version” label will be removed. This can take a few moments.



Please be aware that the Share License feature is not meant for permanent operation, but is limited for a certain period of time (4 hours). Once the License is expired, it can be renewed by selecting Share License from the context menu again.



Any automated use of the license sharing feature violates the license conditions and is forbidden, including, but not limited to, scripts and UI automation.

13.7 Uploading a Project

Before uploading a project, try the project and its triggers/ automation on a PC running SYMPHOLIGHT.

Upload a project to the SYMPL Fusion or SYMPL Core S(p) via its web interface or its context menu in SYMPHOLIGHT.



Since v5.3 it is only possible to upload a project to the SYMPL Fusion or SYMPL Core S(p) from a SYMPHOLIGHT instance with the exact same build number (=four digits following the version, e.g. 5.3.**3352**.2102).



13.8 Selecting the ActionPad

Look up the IP address of the SYMPL Fusion or SYMPL Core S(p) in the Properties Window in the Setup tab of SYMPHOLIGHT, or call the Remote Control from the SYMPHOLIGHT server's device list. Use the plain IP address in the web browser and you can select between the ActionPad and the Configuration.

For later direct access, bookmark the ActionPad's IP address on this SYMPL Fusion or SYMPL Core S(p).

13.9 Using Video with the SYMPL Fusion or SYMPL Core S(p)

Please be aware that the SYMPL Fusion or SYMPL Core S(p) is capable of decoding a maximum of two videos in parallel.

13.10 Maximum Number of SYMPHOLIGHT Instances

There should not be more than ten instances of SYMPHOLIGHT in a network. Using more than ten instances will decrease the performance of the overall configuration.

13.11 Accessing the Logfiles

You can ccess the log files in two different ways:

- Access or download via the web interface's Maintenance tab.
- Directly enter the following URL in the browser: `http://[IP Address]/logs/`.

13.12 Using the Scheduler

The Scheduler and the logs use the local time configured on a SYMPL Fusion or SYMPL Core S(p). This allows configuring the Scheduler on a desktop SYMPHOLIGHT and use the project on a SYMPL Fusion or SYMPL Core S(p) afterwards.

Note: You will need to set the time zone to a different value and then set it to the desired value for this to work.

13.13 Automatic Node Replacement Mode

Via the web interface or in the Setup properties of a SYMPL Fusion or SYMPL Core S(p) in SYMPHOLIGHT, enable this mode to automatically replace an (offline) node with another, available node of the same type on the SYMPL Fusion or SYMPL Core S(p).



Only replace one Node at a time.

Prerequisites

- A computer to download and upload files and with a supported web browser.
Supported web browsers are: Google Chrome, Microsoft Edge, Mozilla Firefox, Safari.



- A stable network access of the computer to the SYMPL Fusion or SYMPL Core S(p). It is recommended to use a wired network connection.
- The SYMPL Fusion or SYMPL Core S(p) is running.
- On the SYMPL Fusion or SYMPL Core S(p) an installed SYMPHOLIGHT v5.2 or newer.
- The password to the web interface of the SYMPL Fusion or SYMPL Core S(p) is set.

How to perform the replacement

1. If the replacing node is already connected, remove it from the network.
2. Remove the node to be replaced from the network.
3. In the Setup tab of SYMPHOLIGHT select the SYMPL Fusion or SYMPL Core S(p) and enable in the Properties the “Automatic Node Replacement Mode”.

OR

Open the SYMPL Fusion or SYMPL Core S(p) web interface and select “Configuration”. In the “Settings” Tab enable the “Automatic Node Replacement Mode”. The search for an equivalent node automatically starts.

4. Connect the replacing node to the network. The node is integrated into the project. The project works like before the replacement.
5. Disable the “Automatic Node Replacement Mode” in the “Settings” Tab.
6. Go to the “Project File” Tab.
7. Press the “Download project with media (as *.hfp)” button and download the project file.
8. Upload the project file again. Now the replacement implemented.

When you have more than one node to be replaced, proceed with the next replacing node by starting from the first step.

13.14 Updating SYMPHOLIGHT in the SYMPL Fusion or SYMPL Core S(p)

The e:cue SYMPL Fusion or SYMPL Core S(p) runs on an integrated firmware.

Depending on the installed firmware version, there are different ways to update the firmware.

For a detailed description of all update ways apart from updating via the context menu, see the Setup Manual or the Update Instructions of the SYMPL Fusion or SYMPL Core S(p).

Updating via the context menu

Update a SYMPL Fusion or SYMPL Core S(p) in Sympholight. The configuration, i.e. the settings, the Sympholight project file, and any media files, is preserved.

Prerequisites

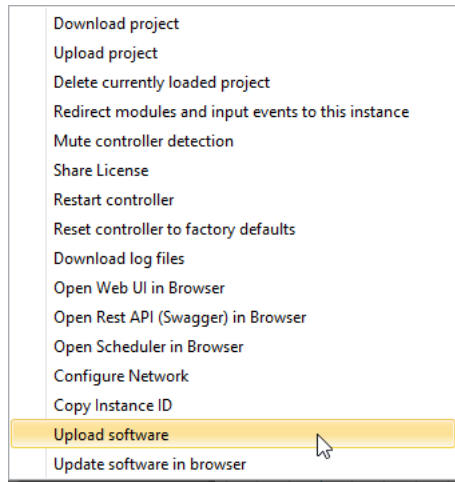
- A computer with a running Sympholight.



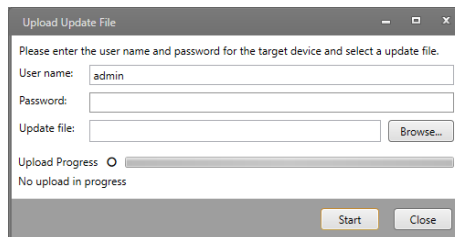
- A stable network access of the computer to the SYMPL Fusion or SYMPL Core S(p). It is recommended to use a wired network connection.
- The SYMPL Fusion or SYMPL Core S(p) is running.
- The password to the web interface of the SYMPL Fusion or SYMPL Core S(p) is set.

How to perform the update

1. Download the new update file from www.ecue.com to the computer. The update file has the extension *.scfs and contains the new software.
2. Connect the computer to the same network as the SYMPL Fusion or SYMPL Core S(p).
3. In Sympholight, go to Setup - Sympholights .
4. Right-click on the SYMPL Fusion or SYMPL Core S(p).
5. In the context menu, select Upload software.



6. Enter user name and password. Browse for the update file. Click the Start button.



The update process starts. It will take some minutes, indicated by a progress bar.



In case of an upload error, try the update again from 4 onwards without restarting the SYMPL Fusion or SYMPL Core S(p).

At the end of the update process, the SYMPL Fusion or SYMPL Core S(p) restarts and resumes normal operation. When the Status LED blinks green slowly, the device has restarted and is operational. The update is now completed.



Do not interrupt the network connection or disconnect the power from the SYMPL Fusion or SYMPL Core S(p) while the update is in progress. Otherwise, the device may fail to start.

Troubleshooting

If the SYMPL Fusion or SYMPL Core S(p) does not restart properly after the update, try the another update option.

13.15 Configuring SYMPL Fusion modules

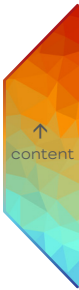
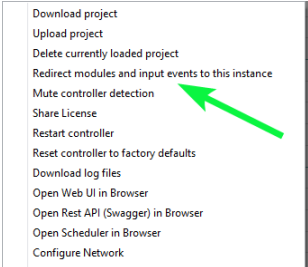
You can pre-arrange a Sympholight project for a SYMPL FUSION Base Device including the modules and digital inputs. The modules are handled the same way as other devices and can be configured in the Device Manager



Starting point

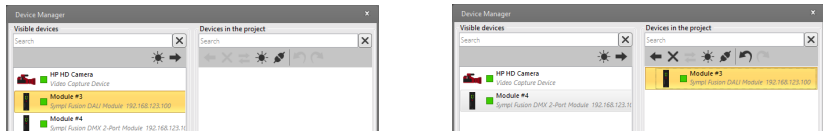
Open the Sympholight software and your project file.

13.15.1 Scenario: Modules from SYMPL FUSION Base Device

- 1. Connect the SYMPL FUSION Base Device.
- 2. Open the context menu of the SYMPL FUSION Base Device.
- 3. Select “Redirect modules and input events to this instance”.



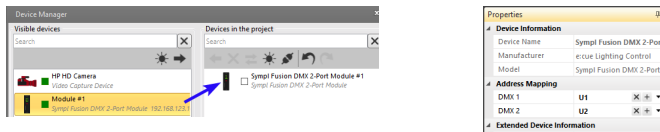
- The status box of the SYMPL FUSION Base Device and the Status LED turn blue indicating the redirection.  **SymplFusion-00000000-0000-4**
Version 
- Drag the recognized modules from "Visible Devices" into the project.



- You can test it.
- Upload the project to the SYMPL FUSION Base Device.
- Return the control of the modules to the SYMPL FUSION Base Device.

13.15.2 Scenario: New modules for not specified slots

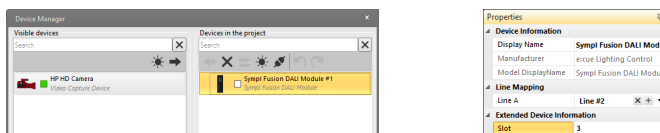
- Create new modules.
- Configure the properties of the modules. See the individual instruction for each module type. Leave the slot number at 0.
- Connect the SYMPL FUSION Base Device.
- Assign the modules with drag-and-drop.



- You can test it.
- Upload the project to the SYMPL FUSION Base Device.
- Return the control of the modules to the SYMPL FUSION Base Device.

13.15.3 Scenario: New modules for designated slots

- Create new modules.
- Configure the properties of the modules. See the individual instruction for each module type.
- Set the slot number in which the corresponding module type is installed in the SYMPL FUSION Base Device.



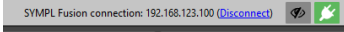
- Connect the SYMPL FUSION Base Device.
- You can test it.



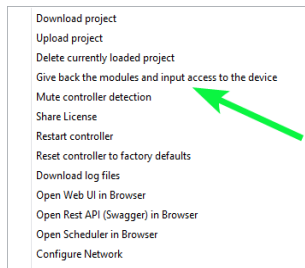
6. Upload the project directly to the SYMPL FUSION Base Device.
7. Return the control of the modules to the SYMPL FUSION Base Device

13.15.4 Return of control

To complete the modules configuration, return the control over them back to the SYMPL FUSION Base Device. To return the control, click “Disconnect” in the main menu bar



Alternatively, open again the context menu of the SYMPL FUSION Base Device and select “Give back the modules and input access to the device”.



Upon uploading a project file to the SYMPL FUSION Base Device and clicking “Handover & Close”, the control over the modules also returns back to the SYMPL Fusion. The status box of the SYMPL FUSION Base Device turns back to green indicating the full control of the device.



If the connection between Sympholight and the SYMPL FUSION Base Device is interrupted while Sympholight has control of the modules, Sympholight will try to re-establish the connection continuously until you return control. While the connection is interrupted, the SYMPL FUSION Base Device will take back control after 30 seconds.

13.16 SYMPL Fusion Test mode

The SYMPL Fusion provides a Test mode for connected fixtures.

Please note that the Test mode overwrites all other output like running project content for the time of testing. A running show is reverted to after exiting the Test mode.

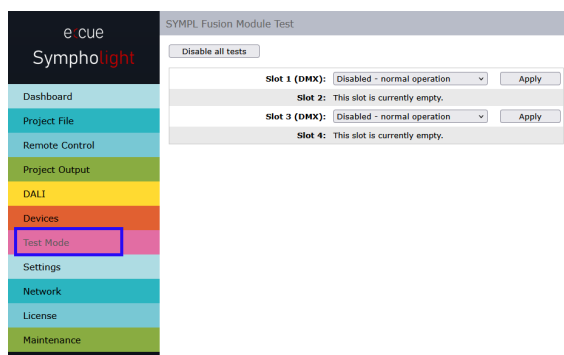
Test mode via web interface

The web interface of the SYMPL Fusion provides four different test patterns.

To access the web interface, open a web browser on a connected PC. Enter the SYMPL Fusion's IP address into the address bar:

e.g. <http://192.168.123.100>.

Open the Test mode page by selecting “TestMode” on the left side of the web interface of the SYMPL Fusion.



Each module is tested individually:

- Select the test pattern. The available test pattern depends on the module type.
- Start and stop the test.

Click “Disable all tests” to stop all running tests and return to normal operation. Tests started using the buttons before show up in the web interface. You can change the patterns and disable the tests there as well.



In case a Test mode is running and a Desktop Sympholight connects to the SYMPL Fusion the Test mode remains active.

13.17

13.18 Updating SYMPL Fusion modules

There are different ways to update the firmware. For a detailed description of all update ways apart from updating via the context menu, see the Setup Manual or the Update Instructions of the SYMPL Fusion.

Updating via the context menu

Update a module in Sympholight to the firmware version that comes along the .SYMPL FUSION Base Device firmware. The configuration is preserved.

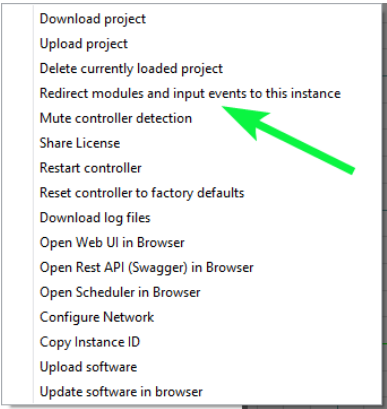
Prerequisites

- A computer with a running Sympholight.
- A stable network access of the computer to the SYMPL FUSION Base Device. It is recommended to use a wired network connection.

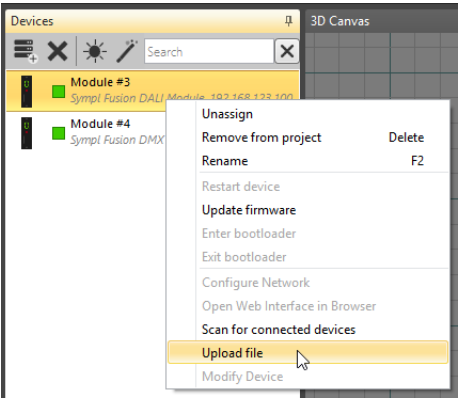
- The SYMPL FUSION Base Device is running.
- The password to the web interface of the SYMPL FUSION Base Device is set.

How to perform the update

1. Connect the computer to the same network as the SYMPL FUSION Base Device.
2. In Sympholight, go to *Setup - Sympholights*.
3. Right-click on the SYMPL FUSION Base Device.
4. In the context menu, select *Redirect modules and input events to this instance*.



5. Go to *Devices*.
6. Right-click on the module which you want to update.
7. In the context menu, select *Update firmware*.



The update process starts, indicated by a red status box.

At the end of the update process, the module resumes normal operation. When the status box changes back from red, the module is operational. The update is now completed. You can hand back the modules to the SYMPL FUSION Base Device.

13.19 Resetting to Factory Default

The Identify button can used, among other things, to reset the device to factory state:

- Keep the button pressed while powering up until Master/Backup and Error/Status LEDs blink alternating (5s), then blink together (5s).
- Releasing the button now resets the device to its factory settings.
- Keeping the button pressed further on, the device proceeds to normal operation. No changes apply.



14 REST API

For an open and secure interaction with other systems (3rd party, cloud, etc.), SYMPHOLIGHT provides a SYMPHOLIGHT (REST) Web API. You can establish a secure connection with the new encrypted (SSL) webserver and send commands or receive status information. This will give you the possibility to handle several SYMPHOLIGHT instances from one central system (PC as well as SYMPL Fusion or SYMPL Core S(p)).



The SYMPL Fusion or SYMPL Core S(p) does not support SSL encryption.



Using Rest API and Scripting requires prior knowledge of these topics.

14.1 REST API Setup

Set the network user name and password in [Tools / Application Settings / Web - REST API](#). They are the same for the Rest API authentication.

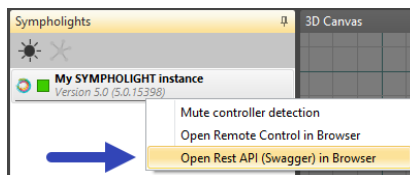
Also configure the Rest API settings at this location. It is recommended to generate an own SSL PFX certificate. The Rest API settings will only take effect after a software restart.

14.2 SYMPHOLIGHT Web API

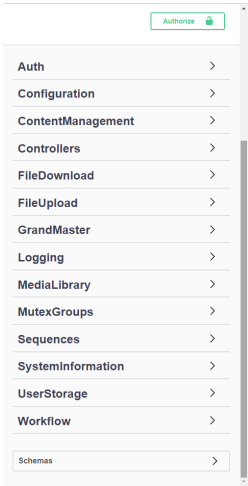
With the SYMPHOLIGHT Rest API it is possible to write your own client that can control a Sympholight instance. The Rest API adheres to the Swagger specification that means that you can use the Swagger tools to generate clients for different programming languages like C#, TypeScript, JavaScript, etc.

To execute the methods and get responses you need to log in:

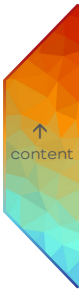
- To open the Rest API documentation (Swagger) in a browser, open the context menu of the SYMPHOLIGHT instance in the Setup window:



The Swagger web page displays all available methods of the SYMPHOLIGHT Rest API:



Auth	Authentication.
Configuration	Gets the Configuration that contains the configuration.
ContentManagement	Get, Change, and Interact with Media Effects (Video and Image). Requires "Content Exchange" license (see „4.2 Credit System“ on page 18).
Controllers	Gets all controllers that are used in the current project.
FileDownload	Get and Interact with downloads.
FileUpload	Upload a file.
GrandMaster	Get and Change the Grand Master.
Logging	Get log files (also filtered).
MediaLibrary	Get and Interact with the Media Library.
MutexGroups	Get and Interact (Play, Pause, next...) with Mutex Groups.



Sequences	Get and Interact with Sequences (incl. opacity) and Cues.
SystemInformation	Get System Status.
UserStorage	Get and Interact with files and directories in the User Data directory.

Workflow Get all Blocks and Write (or get) Values from the “Value Block”.

- Select at Auth the Post method /login:

Auth

POST

/api/auth/login

Called if a user wants to log in to the system. If successful, it should return a valid JWT token that can be used to access other API methods.

- Click the Try it out button:

Auth

POST

/api/auth/login

Called if a user wants to log in to the system. If successful, it should return a valid JWT token that can be used to access other API methods.

Parameters

Try it out

- In the Request body, replace the text String in the username row with your user name as defined in the network settings of SYMPHOLIGHT. Also replace the text String in the password row with your password as defined in the network settings:

Request body required

application/json

The Login object that contains the user credentials.

```
{
  "username": "string",
  "password": "string"
}
```

- Click the Execute button.
- In Responses - Code Details - Response body, copy the token value (excl. the quotation marks):



- # Symphonight Web API

1.0.0 OAS3

/swagger/OpenAPI/swagger.json

With the Symphonight REST API it is possible to write your own client that can control a Symphonight instance. The REST API adheres to the Swagger specification that means that you can use the Swagger tools to generate clients for different programming languages like C#, TypeScript, JavaScript, etc.



Authorize

- Bearer [token value]

Available authorizations

Bearer (apiKey)

Copy 'Bearer' + valid JWT token (retrieved by using "/api/auth/login" endpoint) into the field
Name: Authorization
In: header
Value:

Authorize

Close

- The login is completed. You can now execute the methods.

14.3 Examples

Examples files are stored at

C:\ProgramData\ecue\SYMPHOLIGHT 5.0\Examples\REST API.

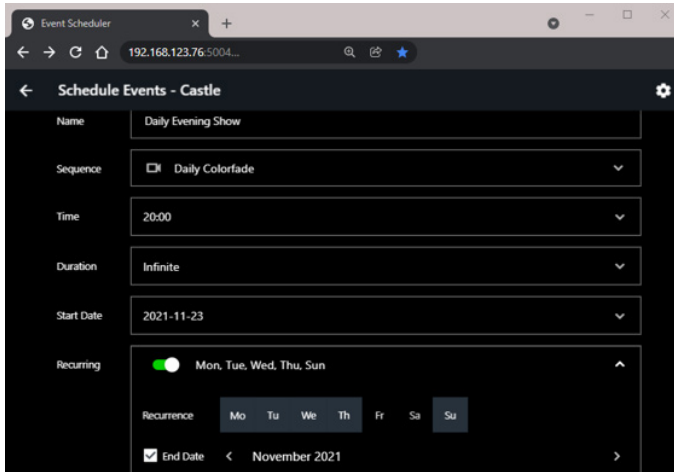
The Media Exchange example shows the authentication in Python language.

The Text Input example offers text input fields in the ActionPad, usable also in the Workflow Designer. It uses the Static File Hosting.



15 Scheduler

The Scheduler is a tool within the SYMPHOLIGHT system to schedule (start and stop) Sequences.



Configure the scheduling of Sequences, so called Events, of a SYMPHOLIGHT project via an easy-to-use web interface with a LAN or WiFi connection. The Scheduler can configure the events of multiple sites and SYMPHOLIGHT instances and is based on the Rest API. After the configuration, the network connection to the web interface is no longer needed because the configured schedules are executed by the SYMPHOLIGHT instances.

You can access the Scheduler with any device that is equipped with a browser, e.g. tablet, smartphone, computer. No SYMPHOLIGHT software needed on the device. The device has to be in the same network as the target SYMPHOLIGHT instance(s).

Note: The Scheduler does not support the Internet Explorer.



For executing a Schedule, a license is necessary. One Schedule requires one Scheduler License (see [“Features” on page 19](#)). Configuring schedules does not require any license.

Preparation / prerequisites:

Before opening the Scheduler,

1. At [Tools / Application Settings / Web - Web Server](#) (see [„Web Server” on page 39](#)), enable the web server and configure the network interface.



- 2. Set the network user name and password at this location. They are the same for the Scheduler web interface authentication.

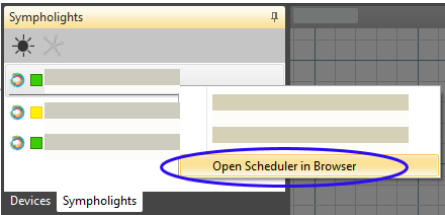
When accessing the Scheduler from the computer running the SYMPHOLIGHT instance:

- 3. Save the project.

When accessing the Scheduler remote:

- 3. At Tools | Application Settings | General (see „General“ on page 35), set for Automatically save the project after a value > 0. The SYMPL Fusion or SYMPL Core S(p) also offers this dialogue under Settings | Scheduler and needs to be set accordingly.
- 4. Save the project.

To open the Scheduler, open the context menu of a SYMPHOLIGHT instance and select “Open Scheduler in Browser”. Or enter the IP address of the SYMPHOLIGHT instance into the address bar with the port 5004 (e.g. “127.0.0.1:5004”).




A new tab of your default browser opens, displaying the web interface of the Scheduler. Add a new Schedule or access an existing Schedule by clicking the **+ New Schedule** button and enter the login credentials. One SYMPHOLIGHT instance can have one Schedule.

Server	
Display Name	Define a name of the Schedule. When re-accessing a Schedule, you can enter any name – it is only important to enter the same IP address.
IP Address	The IP address of the SYMPHOLIGHT instance that the Schedule will be bound to. Enter the same IP address from <u>Application Settings Web - Web Server - Network Interface</u> . It is also identical with the IP address in the address bar of the browser.
Port	Enter “5004”.
Username	Enter the same user name from <u>Application Settings Web - Web Server - User Name</u> .

Password Enter the same password from [Application Settings / Web - Web Server - Password](#).

The credentials are stored in the browser for eased subsequent access from the same device. To prevent this automatic login on the same device (i.e. to “log out”), delete the cache of the browser.

The Settings of a Schedule are identical to the server settings.

Click on a Schedule to configure its triggers: Add new Events for the Schedule () and edit the calendar entries. Schedule Events can be one-time, daily or on recurring weekdays, with a configurable start and end date.

Event

Name Define a name of the Event (max. 32 characters).

Sequence The content of the Event. Select a Sequence from the SYMPHOLIGHT instance.
When simultaneously working with SYMPHOLIGHT, refresh the browser tab to see all recently added Sequences.

Time Define the start time (trigger time) of the Event.

Duration Define the duration of the Event.
Infinite Mode: No stop trigger.
Disable the Infinite Mode to set a specific duration (max. 23:59:59).

Date Define the start date of the Event.



Recurring

Off: one-time trigger. Enable to repeat the Sequence trigger. Select the type of Recurrence:




Daily or by Weekdays: For daily recurrence, select all weekdays. For specific weekdays, select (multiple) weekdays.


Monthly: Select a day of the month to repeat the trigger every month at that day.

Yearly: Select a day of the month and a month to repeat the trigger every year at that date.

Enable an End Date to stop the recurrence:

End Date: Define the last date of the recurring Event.

To confirm the settings of the Schedule Event, click the  button. With immediate effect, the Event is active. To cancel the settings, click the  button. To delete the Schedule Event, click the  button. You can disconnect the device with the browser now. The scheduled Events are triggered from within the SYMPHOLIGHT instances. It is necessary to save a SYMPHOLIGHT project whenever an assigned Schedule - or an Event within - has been created or changed. Confirming a Schedule Event does not save it. Upon loading a project with a Schedule, only Events with an upcoming start time will be triggered.

You can install the Scheduler as an app on your desktop which enables a quick and direct access to the Scheduler - no further need of a browser afterwards. Use the Google Chrome browser, open the Scheduler and click the  icon in the address bar. Install the app and close the browser tab. You can open the Scheduler now by clicking the shortcut on the desktop.

The log files of the Scheduler are integrated part of the log files of AutomationService, located at ProgrammData\ecue\logs\Ecue.Sympholight.AutomationService.

The Scheduler and the logs now use the local time configured on a SYMPL Fusion or SYMPL Core S(p). This allows configuring the Scheduler on a desktop SYMPHOLIGHT and use the project on a SYMPL Fusion or SYMPL Core S(p) afterwards. Note: You will need to set the time zone to a different value and then set it to the desired value for this to work.

Testing

If you want to test your Schedule, it is recommended to only use the following steps. Nonetheless, it cannot be guaranteed that the testing gives an accurate report.

1. Change the system time of the computer running the SYMPHOLIGHT instance. Set a time before the scheduled start trigger time. You should not use dates in the past.



2. Let the Scheduler fire the start trigger(s) and the end trigger(s). It is recommended not to change the system time in between. E.g. for an Event with a 10min duration, wait for these 10 min.
3. Reset the system time after the test run is completed.

Troubleshooting

If current changes, e.g. renamings, are not displayed, reload the Scheduler website in the browser.

If the web interface is showing connection errors when trying to add a new connection, supposing that the IP address and the port are correct, try the following:

- Clear cache of the web browser and reload the Scheduler website.



16 RDM in Detail

This is an explanatory chapter of RDM and using RDM in SYMPHOLIGHT.

16.1 RDM Glossary

Request

A network message send from a SYMPHOLIGHT to a RDM device.

Response

A network message send from a RDM device to SYMPHOLIGHT as reaction to a request.

Command

In the RDM context a command is a type of a message which is send in a request and response.

Parameter

A parameter is a dataset stored on the RDM device. E.g.: Sensor Value

Attribute

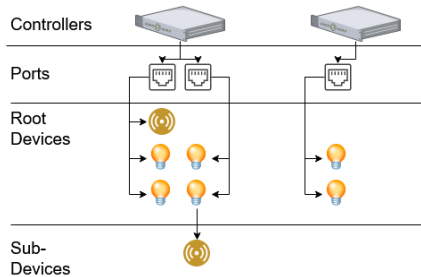
An attribute is a value from within the parameter dataset. E.g.: Sensor Value -> Present Value

16.2 Remote Device Management (RDM)

RDM permits intelligent bi-directional communication between devices from multiple manufacturers utilizing a modified DMX512 data link. RDM permits a controlling device to discover and then configure, monitor and manage intermediate and end-devices connected through a DMX512 network. SYMPHOLIGHT provides the functionality of such a controlling device.

16.2.1 Topology

SYMPHOLIGHT works with a topology containing controllers (like Butler S2, SYMPL dmx Node ...) having different amounts of DMX512 ports. RDM extends this topology with so called root-devices. This can be a luminaire, ballast or similar. Root-devices can have child elements, those devices are called sub-devices. The total topology visualized:



16.2.2 Addressing

RDM devices (root- and sub-devices) have a unique identifier. The UID consists of three parts: 1) The manufacturer ID is unique for each manufacturer of RDM capable devices and is assigned to the manufacturer by the ESTA. 2) The device ID is assigned by the manufacturer within the production process. The combination of manufacturer ID and device ID is globally unique to one root-device. 3) If a root-device has sub-devices, those will have a sub-device index in range 1 - 255.

Name	Size	Range	Note
ESTA Manufacturer ID	16Bit	0x0001 – 0x7FFF	Unique ID per manufacturer (managed by ESTA)
Device ID	32 Bit	1 - 4.294.967.295	Unique ID per root device of the manufacturer
Sub-device Index	16 Bit	0 - 255	Zero for root-devices, otherwise within range

16.2.3 Discovery

To make it possible to communicate with root- or sub-devices, the devices have to be known by the controlling entity (SYMPHOLIGHT). To realize that, the discovery process of the entity will send requests using a binary search algorithm on the addresses. The discovery process proceeds until every root- and sub-device has been found and all the information the controller requests is provided.

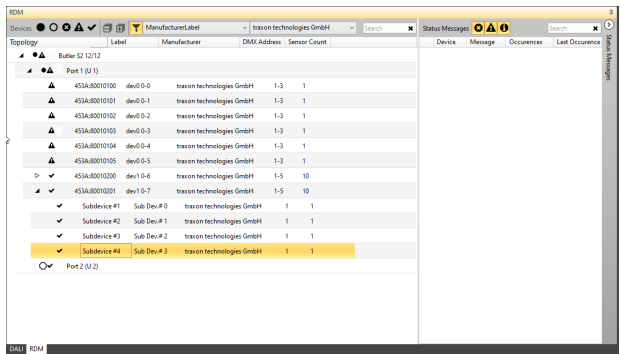
16.2.4 Update Discovery

To make sure SYMPHOLIGHT recognizes when a device goes offline, a provided value of a device changes, or new devices are added, the discovery takes place in the background all the time.

16.3 RDM View in Setup Tab

The complete topology of the discovered RDM devices is visualized in the RDM view. The RDM view is located at the bottom of the Setup tab: choose “RDM” at the right side of “DALI”. An example of a controller (Butler S2 12/12) with two ports:





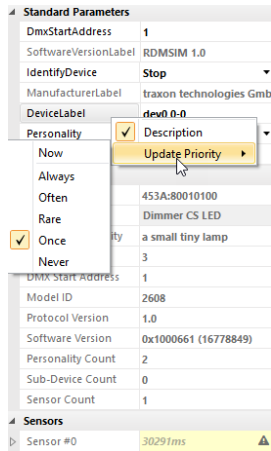
Only Port 1 has RDM enabled. This is visualized with the filled circle icon next to Port 1. Port 2 has RDM disabled, the circle icon is not filled. There are eight root-devices found at Port 1. The last two root-devices do also have sub-devices. The root-devices 1 to 6 have reported a warning which is visualized with the exclamation mark (this happens e.g. when a sensor of the device is not inside its valid range). The refresh icon visualizes that the system is currently updating the attributes of the device. With the toolbar above it is possible to filter the devices based on their status and attribute values. The status messages which are provided by some RDM devices are visualized at the right side

16.3.1 Update Optimization

To improve the performance of the discovery time, SYMPHOLIGHT does not request every information a device provides every round. Some information is not supposed to change every round (e.g. the manufacturer label of a fixture will not change from one round to another). The priority of the information request is pre-configured for all default attributes defined in the RDM specification.

16.3.2 Adjust Attribute Update Priority

Sometimes it is necessary to change the update interval for a specific attribute. This can be done within the property grid of SYMPHOLIGHT:



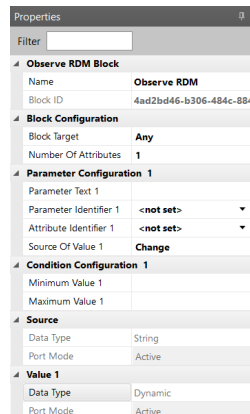
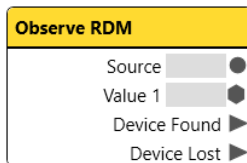
A right-click on the attributes opens the context menu. The context menu provides the possibility to choose between different update intervals. After changing the interval, the selected attribute will be requested one time (as fast as possible). The update priority will then continue with the chosen interval.

16.4 RDM in Workflow-Designer

For monitoring and handling a huge amount of RDM devices, the Workflow-Designer provides blocks which can be used for the automation and monitoring of RDM devices.

16.4.1 Observe RDM

The Observe RDM block provides the possibility to listen for RDM messages received by SYMPHOLIGHT. The simplest use case of the block is the monitoring of device found and device lost events. This is the default configuration.



The Source PortOut of the block will automatically be set to the RDM UID of the device which was found or lost every time when such an event takes place. Since the Block Target is set to Any in the property grid, the block reacts on every device lost / found notification independent from the controller or port where the lost / found device was located.

A more advanced use case is when the Observe RDM block is configured to listen for a single root-device (see Block Target):

Observe RDM

Source 453A:...

SensorValue...31.111

Device Found ▶

Device Lost ▶

SensorValue 0 Underfl... ▶

Observe RDM Block

Name

Observe RDM

Block ID

928f68e9-78a4-4ca9-aa6b-t

Block Configuration

Block Target

Single Device

Number Of Parameters

1

Device Configuration

Single Device Identifier

dev0 0-0 (453A:80010

Parameter Configuration 1

Parameter Text 1

SensorValue 0

Parameter Identifier 1

SensorValue 0 (0xd020

Attribute Identifier 1

PresentValue

Condition Configuration 1

Minimum Value 1

12

Maximum Value 1

Source

Data Type

String

Port Mode

Active

Value 1

Data Type

Dynamic

Port Mode

Active

Define the root-device with its UID at Device Configuration. Furthermore, the block listens only for changes of one attribute from one parameter (see Number Of Parameters). In Parameter Identifier 1, the block is set to listen for changes of the SensorValue 0. Since there is multiple information regarding a sensor (recorded value, lowest value, ...), the Attribute Identifier 1 has to be specified too. Set it to here to "PresentValue". The present sensor value is now available and will update every time the present value is received from the RDM device in the background. To make the block send a trigger e.g. when the present sensor value falls below 12, set the Minimum Value 1 to "12". The event port for the underflow is visible at the block and is evaluated every time the sensor values changes.

16.4.2 Build RDM Command

The Build RDM Command block provides a simple possibility to build RDM commands which can afterwards be send by using the Send RDM block.

Build RDM command

<Create RDM Comm...

Build RDM command Block

Name

Build RDM command

Block ID

e5d8f5ed-a2a9-4fe3-ba57-t

Configurations

Is event based

☒

RDM Command Library

Library

RDM Standard

Command

Get Communication S

RDM Command

Data Type

Dynamic

Port Mode

Active

To specify the command which shall be build, it is necessary to choose the library where the command is located in. All “default” commands are located in the RDM Standard library. Additionally, SYMPHOLIGHT provides an Allegro library which contains custom commands used by Allegro fixtures.

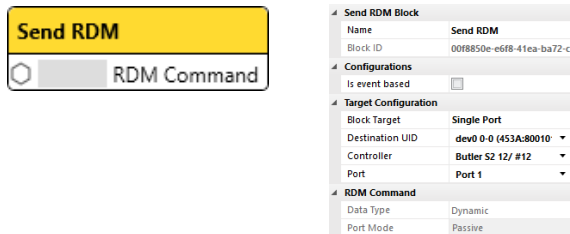
Once a library is selected, the Command drop-down menu contains all commands of chosen the library. To search in the drop-down menu, type in the command property. Some commands are Get commands, others are Set commands. When sending a Get command, use an Observe RDM block and listen for the according attribute of the RDM device. Some Get commands and all Set commands require the specification of some values in order to build a valid RDM command. Specify those values as usual by PortIn values or property configuration.

Advanced:

An RDM device manufacturer may define own commands. To facilitate the usage of those commands in SYMPHOLIGHT, it is possible to define those commands in a own library file as it was done for the Allegro fixtures. To do so, create a *.slrt file complying with the XML format which is defined in the RdmCommandTemplateSchema.xsd. Place the *.slrt file in the „%appdata%\ecue\SYMPHOLIGHT <Version_Number>\RDM Command Templates“ directory. The schema file is located at the same directory.

16.4.3 Send RDM

The Send RDM block forwards commands to RDM devices.



For example, RDM commands which are created by the Build RDM Command block can be given to the Send RDM block. In this block, configure the destination to which the command shall be send. The Block Target property offers All Controllers, Controller Ports and Single Port. In case of All Controllers, it is only possible to send commands to broadcast addresses. In case of Controller Ports, specify a controller which will send those commands. Like the All Controllers setting, the Controller Ports setting allows only broadcast addresses.

The Single Port setting allows the specification of a controller, port and either the UID of a specific RDM device or a broadcast address.

16.5 RDM in Scripting

The management of a big number of RDM devices or the handling of complex rule chains with the workflow designer is best handled with scripts for RDM. The following chapter gives a slight overview about the functions available in scripting. Focus is on current limitations which can be solved by combining scripting with the Workflow Designer. A basic example can be found in the examples folder which is per default installed with SYMPHOLIGHT on the computer. The folder has the name Examples and is located in the installation directory of SYMPHOLIGHT.

16.5.1 Finding RDM Devices

RDM works on basis of the devices described in „[16.2.1 Topology](#)“ on page 452. To find specific devices, there are different possibilities. A) Go through the hierarchy of the devices and try to find the desired device. B) Use existing methods and properties providing simple ways to find a device, for example as shown below.

In case a specific RDM root-device is searched for, but the controller and the port where it is connected are unknown. Only the UID of the root-device is known. In that case, use the following code:

```
var rootDevice = ecue.Rdm.GetRootDeviceByUid(0x453A, 0x80020101);
```

The code consist of the following parts: The first word is the keyword var. It creates a new variable which gets the name rootDevice with the next word. Then the return value of the method ecue.Rdm.GetRootDeviceByUid is assigned to the rootDevice variable. This method requires two parameters. The first one is 0x453A. It is the manufacturer ID of Traxon. The second parameter 0x80020101 is the device ID of the wanted root-device. While typing in the editor, the code completion tool assists with proposing possible functionality. For example, to access a functionality of SYMPHOLIGHT, begin with „ecue.“, then suggestions for the categories which exist are shown. Currently this would be DMX, RDM, Logging, Sequences and many more. Selecting RDM would extend the written code to ecue.RDM while typing a dot would now suggest all the methods and properties accessible for RDM in SYMPHOLIGHT.

16.5.2 Working with RDM Devices

To demonstrate how to work with RDM devices, the example from the previous chapter is taken. Now an RDM command shall be sent which sets a DMX address to the earlier defined root-device. To realize that in a very simple way, a combination of the



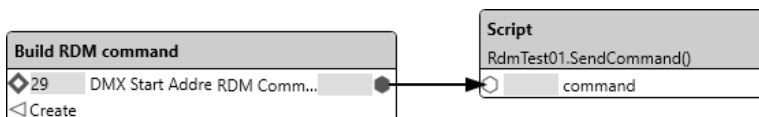
functionalities from the Workflow Designer and scripting is recommended. The start is scripting part; for example with the following code:

```
/// <summary>
/// This class is public static, so it's methods can be accessed from the workflow designer.
/// </summary>
public static class RdmTest01
{
    /// <summary>
    /// This method is public static, so it can be accessed from the workflow designer.
    /// </summary>
    /// <param name="command">RDM command to be send to root device.</param>
    public static void SendCommand(IRdmCommand command)
    {
        // Find the RDM root device based on the specified manufacturer ID and device ID
        var rootDevice = ecue.Rdm.GetRootDeviceByUid(0x453A, 0x80020101);

        // Send the command we got as method parameter e.g. from the workflow designer // to the root
        // Device we found in the previous line of code.
        rootDevice.Send(command);
    }
}
```

This example has a class called RdmTest01 which is having a method called SendCommand. This method expects one method parameter. The parameter is required to be of the type IRdmCommand and has the name command. The next line of code is known from the previous chapter. In the following line, the rootDevice is used to call the Send method for this rootDevice. Like the SendCommand method, the Send method expects a method parameter of type IRdmCommand. Give the parameter command of the SendCommand method into the Send method as parameter.

Now the questions is “Who is providing the command method parameter of the SendCommand method?” That’s where the Workflow Designer comes in. As shown below, the Set DMX Start Address command is built in the Workflow Designer:



The resulting RDM command has the exact type as expected parameter in the script method (IRdmCommand). Call the script method by using a Script block targeting the SendCommand method and class RdmTest01. Now triggering the Create EventIn of the Build RDM Command block creates the DMX Start Address RDM command. This RDM command is forwarded to the script method. The method sends the RDM command to the root-device. Now the RDM root-device has set the DMX Start Address to the received value (here 29).

It is possible to create those commands also directly in scripting. This is also demonstrated in the RDM example project located in the Examples folder of the SYMPHOLIGHT installation directory.




17 DALI Feedback Function

17.1 Introduction

The DALI standard 62386-332 supports visual feedback channels for DALI Input Device instances like DALI buttons. Visual feedback could include changing the brightness or the colour of a feedback LED.




The feedback is binary. The DALI feedback function in SYMPHOLIGHT only supports the commands `ACTIVATE_FEEDBACK` (= luminance on) and `STOP_FEEDBACK` (= luminance off). All other commands from the standard are not supported.

SYMPHOLIGHT provides an icon to test whether a DALI Button offers a Feedback channel. In the Real Device View, in the power button column click the  icon and check your DALI Button for any feedback.

Real Devices		Line #1		DALI 2			
Matched				Name	Type	Address Info	Line
<input type="checkbox"/>				<DaliInput#1>	Input Device	I 1	Line #1
<input checked="" type="checkbox"/>				Button #1	Button	I 10	Line #1
<input checked="" type="checkbox"/>				Button #2	Button	I 11	Line #1
<input checked="" type="checkbox"/>				Button #3	Button	I 12	Line #1

How to use the DALI Feedback Function:

- Install your DALI system (DALI input device with feedback functionality, SYMPL dali Node).
- Open SYMPHOLIGHT -> license is required.
- In the Setup Tab, add the connected SYMPL dali Node to the project (=online).
- Scan the DALI line in the Real Device view.
- Drag a real DALI device onto the canvas (e.g. DALI button).
- Identify and test the feedback function by clicking the  icon. The feedback channel instance on the real device turns on/off.
- In the Automation Tab, add the following blocks to the Workspace as appropriate for your project.

DALI Button Block: set the Target, set the Behavior of Feedback and connect the desired Workflow to the PortIn.

DALI Feedback Channel Line Block: set the target, set the address and instance properties, and connect the desired Workflow to the PortIn.

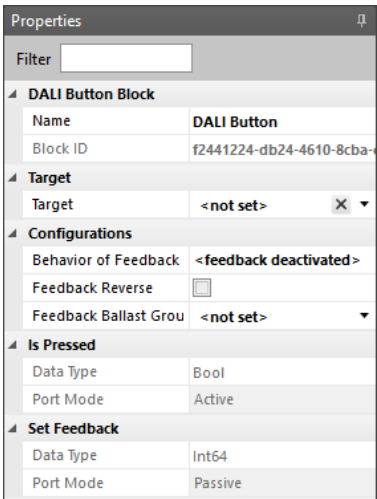
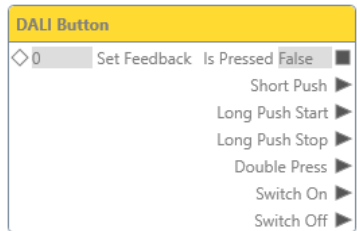
17.2 Sending Feedback

17.2.1 DALI Button Block

Regarding the Feedback function, the DALI Button Block provides a PortIn and Configurations properties. Find a complete description of the block here: „11.15.50 Button“ on page 269.

To use the feedback

- define the Behavior of Feedback



Category	Property	Description
Target		Define the target to bind the Block to a DALI Button.

Configurations

Configure the feedback parameter and the type of execution.

Behavior of Feedback

Define how the Block should get feedback:

<feedback deactivated>: The Feedback function is disabled and the Feedback channel is off.

By DALI Ballast Group: The feedback is defined by the luminance status or the scene value of the reference ballast of a DALI Ballast Group. Set the DALI Group at "Feedback Ballast Group". Target luminance values > 0 activate the feedback channel.

By Button Switch On/Off Event: Only for a DALI Button configured as a switch. On a "Switch On" command, the feedback is activated. On a "Switch Off" command, the feedback is stopped.

By Feedback In Port: For using the "Set Feedback" PortIn. Incoming values != 0 activate the feedback channel. Incoming value= 0 deactivates feedback.

Feedback Reverse

Enable to inverse the Feedback commands: on an "activate feedback" command, the feedback is stopped. On a "stop feedback" command, the feedback is activated.

Feedback Ballast Group

Only for "Behavior of Feedback - By DALI Ballast Group".
Define a DALI ballast group that has an associated (i.e. target based) DALI Group Block in the Workflow and whose reference ballast and its luminance or scene number is to determine the feedback status..



Is Pressed

An actively sent bool value whether the bound target is being pressed or not.

Set Feedback

Only for "Behavior of Feedback - By Feedback In Port".

A passively received Int64 value to send a value to the Feedback channel of the DALI button.

When the incoming value is != 0, the feedback is on (= active).

When the incoming value is =0, the feedback is off (= stopped).

Note that there are no parameter validity checks.

- All configurations are executed on the next incoming command.
- For the SYMPHO DALI2 Control Panels (e.g. SYMPHO DALI2 Panel 4-Key): each button owns two Feedback channel instances (two LEDs of different color). Only one color is addressed by the DALI Button Block. To access the second color / LED, see the User Instructions of the product for the LED's instance number. Then use the DALI Feedback Channel Line Block with the LED's address and instance number.

17.2.2 DALI Feedback Channel Line Block

Use this Block to control DALI feedback channels of a line in the direct way by input address and instance number. The Block sends DALI commands to the associated line, device and instance number(s). Only for DALI v2 Feedback functionality supporting devices.

You can use multiple blocks of the same target.

To use the feedback

- define the Target Property.
- define the short address and / or instance number to select button inputs from the line.



DALI Feedback Channel Line	
Line #1	
◇ 0	Set Feedback
◇ 255	Address
◇ 255	Instance No

Properties	
Filter	
DALI Feedback Channel Line Block	
Name	DALI Feedback Channel Line
Block ID	103d6967-becc-4af8-9fab-5
Target	
Target	<not set> X ▾
Address	
Data Type	Int64
Input Type	Property
Value	255
Instance No	
Data Type	Int64
Input Type	Property
Value	255
Set Feedback	
Data Type	Int64
Port Mode	Passive

Category	Property	Description
Target		Define the target to bind the Block to a DALI line.
Address		An Int64 value for the short address(es) of the line. To filter devices of the DALI Line by short address. Set the Input Type. When defined as Property, configure a value.
	Value	Define a short address value between 0 and 63. Select multiple devices by comma separation. Leave empty to get all devices on the Line. 255 (initial value) - Feedback disabled



Instance No	<p>An Int64 value for the instance number(s) (=input instance of max. 32 instances inside the input device) of the line.</p> <p>To filter instances of the DALI Line by instance number. Set the Input Type. When defined as Property, configure a value.</p>
Value	<p>Define an instance number value between 0 and 31. Select multiple instances by comma separation. Leave empty to get all instances of a device. 255 (initial value) - Feedback disabled</p>
Set Feedback	<p>A passively received Int64 value to send a value to the Feedback channel of the DALI button.</p> <p>When the incoming value is != 0, the feedback is on (= active).</p> <p>When the incoming value is =0, the feedback is off (= stopped).</p> <p>Note that there are no parameter validity checks.</p> <ul style="list-style-type: none">— Only use this Block for feedback channels with very special behavior that are not matched to a button instance.— On sending feedback activate/stop command, there are no checks for valid parameters.— You can use multiple blocks of the same target.— All configurations are implemented on the next incoming command.— For the SYMPHO DALI2 Control Panels (e.g. SYMPHO DALI2 Panel 4-Key): each button owns two Feedback channel instances (two LEDs of different color). Only one color is addressed by the DALI Button Block. To access the second color / LED, see the User Instructions of the product for the LED's instance number. Then use the DALI Feedback Channel Line Block with the LED's address and instance number. <p>Example: You want to use the second Feedback LED for Button #1 (Address Info: I 1 0) of a SYMPHO DALI2 Panel 4-Key</p>



Real Devices						
		Line #1		DALI 2		
Matched			Name	Type	Address Info	Line
<input type="checkbox"/>	⚠	⌚	<DaliInput#1>	Input Device	I 1	Line #1
<input checked="" type="checkbox"/>	⚠	🔘	Button #1	Button	I 10	Line #1
<input checked="" type="checkbox"/>	⚠	🔘	Button #2	Button	I 11	Line #1
<input checked="" type="checkbox"/>	⚠	🔘	Button #3	Button	I 12	Line #1

The instance number of the second LED is 4, according to the User Instructions. Then you need to configure the DALI Feedback Channel Line Block like this:

DALI Feedback Channel Line

Line #1

0

Set Feedback

1

Address

4

Instance No

Target

TargetLine #1

Address

Data TypeInt64

Input TypeProperty

Value1

Instance No

Data TypeInt64

Input TypeProperty

Value4



↑
content

18 SymphoCity

For project control, scheduling and maintenance, the SymphoCity app offers extended remote control options. Install, configure and run the application on a server - a SYMPL Fusion or SYMPL Core S(p) does not work as a hosting instance. However, a connection to the SYMPL Fusion or SYMPL Core S(p) is required in order to control the device from within SymphoCity. For further information and requirements, visit <https://eu.traxon-ecue.com/symphocity/>.



19 Glossary

Art-Net Art-Net is a DMX over Ethernet protocol. Art-Net has been invented by Artistic Licence and has now been published into the public domain.
Link: www.art-net.org.uk

Block A function element in the Workflow Designer. A Block consists of data inputs and outputs and/or Trigger inputs and outputs. The internal functions of a Block execute an action, calculate a result or send data.

Cue A control element in SYMPHOLIGHT. Cues in a Sequence can be addressed as time markers. You can skip to Cues to control the execution of a Sequence.

DALI Digital Addressable Lighting Interface (DALI) is a standard for control lighting in buildings. It was established as a successor for 0-10 V lighting control systems, and as an open standard alternative to Digital Signal Interface (DSI), on which it is based. The DALI standard, which is specified in the IEC 60929 standard for fluorescent lamp ballasts, encompasses the communications protocol and electrical interface for lighting control networks.

DMX DMX512 (Digital Multiplex) is a standard for digital communication networks to control stage lighting and effects such as fog machines and moving lights. DMX512 employs EIA-485 differential signalling at its physical layer, in conjunction with a variable-size, packet based communication protocol at 250kBit/s. It is unidirectional and does not include automatic error checking and correction. DMX is the most used connection type in lighting control.

e:bus The e:bus is a 2-wire, polarity-independent bus system for communication between lighting components and engines. Due to the free topology design and capability to wear data communication and power supply voltage at the same time, it simplifies cabling and installation costs. e:bus has a maximum cable length of up to 400 m.

e:net An Ethernet-based e:cue protocol used as the backbone communication standard between most e:cue engines and interfaces.



e:pix e:pix is a protocol based on DMX, developed by e:cue, for an improved communication between the VMC and Traxon e:pix-ready LED products.

Effect An algorithm creating content in a Layer. In general, Effects provide colors, intensities, media or text content that is placed in a Layer.

Engine An engine is a standalone-capable device that controls fixtures, receives signals and reacts on input from external devices like terminals or switches.

Fixture General expression for a lighting system, usually equipped with LEDs, but also laser driven. Like LED liners, head washers or video matrix panels.

HCL Human centric lighting (HCL) is the art of creating lighting that mimics the natural daylight that drives our bodily functions by changing the color temperature and intensity to stimulate natural light colors. It enhances human performance, comfort, health and well-being. Human Centric Lighting solutions essentially contribute to levels of well-being. It balances visual, emotional and biological needs of humans in lighting application.

Input Device (DALI) The basic DALI hardware device providing data, usually consisting of multiple (max. 32) instances such as a Generic Input and sensors.

Interface A device that converts interfaces or protocols, e.g. from e:net to DMX, from e:net to DALI, or from e:net to relay contacts. In general, a device that connects other devices to SYMPHOLIGHT.

KNX KNX is a standardised (EN 50090, ISO/IEC 14543), OSI-based network communications protocol for intelligent building automation. KNX is the successor to, and convergence of, three previous standards: the European Home Systems Protocol (EHS), BatiBUS, and the European Installation Bus (EIB or Instabus). The KNX standard is administered by the KNX Association.



Layer	A track in a Sequence. All Layers in a Sequence contain the final content. Layers in SYMPHOLIGHT are organized in hierarchical order: content of Layers below another Layer are only visible on the fixture if the upper Layer has transparent pixels or an opacity < 100%.
Layout	The configuration and placement of fixtures and devices in a SYMPHOLIGHT project. The Layout is created in the Setup tab of SYMPHOLIGHT, providing input and output channels to fixtures, interfaces or system functions.
Method	A function of an object. Derived from class-oriented programming, executing a function or an action.
MQTT	MQTT is a lightweight publish/subscribe messaging protocol for the Internet of Things (IoT) that is ideal for connecting remote devices with a small code footprint and minimal network bandwidth. It is an open protocol for building management systems.
Object	Any element in SYMPHOLIGHT, a Sequence, an Effect, a button in the Action Pad. Objects in SYMPHOLIGHT can be accessed via Scripting or with the Workflow Designer, to influence a Property or to call a function of this object.
Property	An attribute of an object. Properties can be e.g. colors and gradients, text, orientation, size, address / ID.
RDM	Remote Device Management, a protocol based on DMX512 with bidirectional communication capability between a lighting controller and RDM-capable lighting fixtures or devices. In this way, information from fixtures can be given back to the controller, a feature that standard DMX does not support.
sACN	Streaming ACN (sACN) is a DMX over Ethernet protocol. It is a standard created by ESTA (Entertainment Services and Technology Association). Link: www.esta.org
Sequence	The complete content that creates a "project". A Sequence contains Layers. Layers contain Effects or media, Cues and Triggers, Scenes and other content.



SYMPHOLIGHT instance A hardware device running the SYMPHOLIGHT software.

SYMPL Fusion Base Device e:cue's SYMPL Fusion is a controller that can be used as a simple standalone controller, expandable in a network with other e:cue devices, or as a decentralized control unit. Its modular slot card concept allows the freedom to compose your individual demands, functional lighting, dynamic lighting control or if needed the combination of both. The SYMPL Fusion Base Device is the controller device that houses the optional modules.

TCP/IP The Transmission Control Protocol/Internet Protocol (TCP/IP), are standard Internet communications protocols that allow computers in a network to communicate over long distances. TCP is the component that collects and reassembles the packets of data, while IP is responsible for making sure the packets are sent to the right destination.

Terminal Any device that transfers user interaction or commands into a lighting system, e.g. Glass Touches.

Triggers

- a) Triggers in a Cue are control objects, they cause an event in the Workflow Designer which can be caught and lead to an action.
- b) An output of a Block in the Workflow Designer. In case of an event, e.g. a time and date, receiving an external signal, or pressing a DALI button, the output sends a Trigger to the following Block, causing the execution of an action or to calculate a result.

UDP The User Datagram Protocol (UDP) is a communication protocol and standardized method for transferring data between computers in a network for especially time-sensitive transmissions. It sends packets (messages) directly to a target computer, without establishing a connection first, indicating the order of said packets, or checking whether they arrived as intended. UDP is faster but less reliable than TCP.

Universe Means DMX universe. A DMX universe are all devices connected to one DMX line, so a maximum of 512 addresses.



20 Notes





Downloads and more information at:
www.ecue.com and www.traxon-ecue.com

Traxon Technologies Europe GmbH
Sales Operations
Karl-Schurz-Str. 38
33100 Paderborn, Germany
+49 5251 54648-0
support@ecue.com



ecue

TRAXON | ecue
MEMBER OF PROSPERITY GROUP

Please check for the latest updates and changes on our website.

© 2025 TRAXON TECHNOLOGIES. ALL RIGHTS RESERVED.

www.traxon-ecue.com