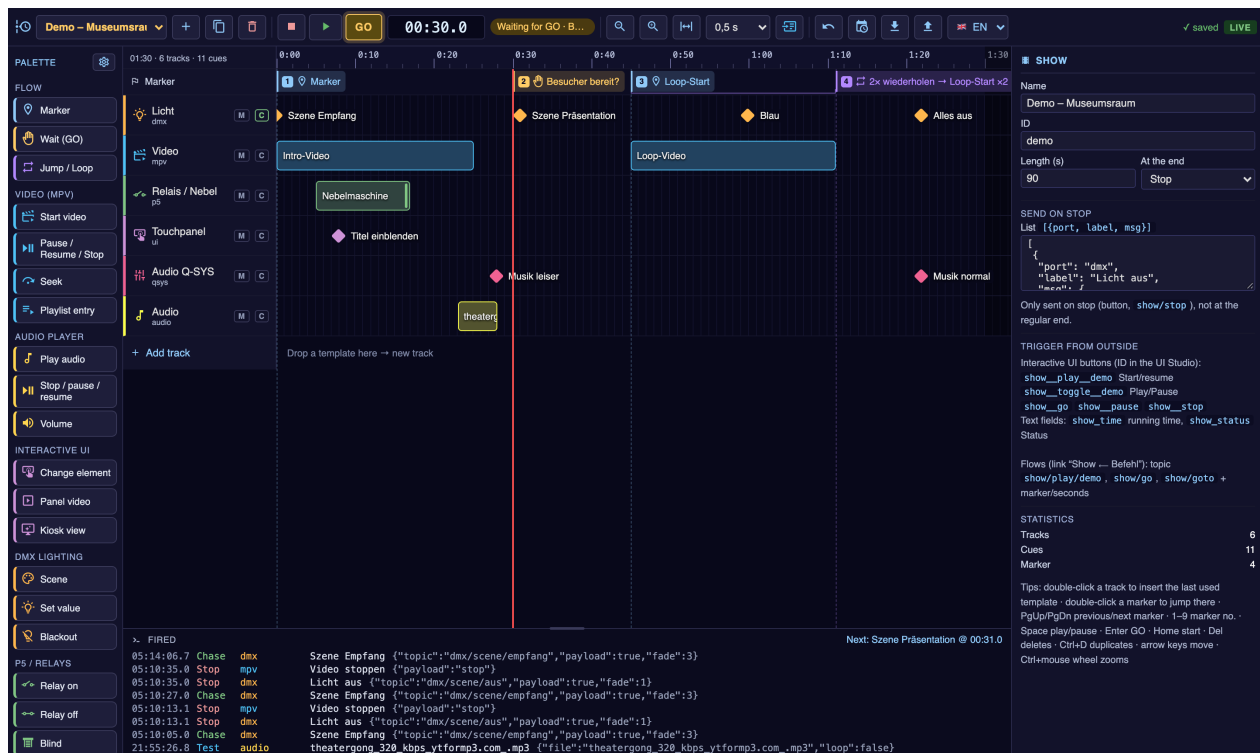


//nanoblue

nanoblue® Show Studio

Manual – creating and playing shows



As of September 2026 · Screenshots: device 876475 with the demo show “Museumsraum” (show content in German)

Contents

Contents.....	2
1 Overview	3
Terms	3
Opening the Show Studio	3
2 The user interface	4
Header	4
Palette.....	5
Timeline	6
Log.....	6
3 Step by step: your first show.....	7
4 Editing cues	8
With the mouse	8
Example: video.....	9
Example: audio	9
5 Markers: wait, jump and loop.....	10
Jumping to a marker	11
6 Tracks	12
7 Show settings	13
8 Playing and controlling from outside	13
In the Show Studio	13
From the touch panel (UI Studio)	13
From push buttons, sensors and other flows	14
Schedule.....	14
9 Managing the palette and custom templates.....	15
Creating a custom template.....	16
10 Language	17
11 Keyboard shortcuts	17
12 Tips and troubleshooting	17

1 Overview

The Show Studio is the editor for the nanobblue Show Controller. A **show** is a timed sequence: actions are placed on a timeline and sent to the connected devices at fixed points in time. The show runs on the nanobblue player itself; the editor is only used to edit and control it.

Terms

Term	Meaning
Show	A complete sequence with length, tracks, cues and markers. A device can store several shows; one is active at a time.
Track	A row on the timeline, usually per device or discipline (lighting, video, relays ...). Each track has a port.
Cue	An action on a track: point in time, optional duration, and the message that is sent.
Marker	A mark in the marker row: jump target, wait point (GO) or jump/loop.
Port	The target of a message, e.g. dmx, p5, mpv, audio, ui, qsys. It decides which driver receives the command.
Template	A ready-made building block in the palette that turns a few inputs into the right message.
Play position	The red vertical line – where the show currently is.

Opening the Show Studio

- **Via the portal:** open the device page and click **Show Studio**.
- **Directly:** <https://core.nanobblue.cloud/device/<device ID>/ui/show.html>
- **On the local network:** <http://<player IP>:1880/ui/show.html>

A badge at the top right shows the connection: **LIVE** = connected to the device, **OFFLINE** = device currently not reachable, **SIMULATION** = no device found; shows are then only stored in the browser and commands only appear in the log.

2 The user interface

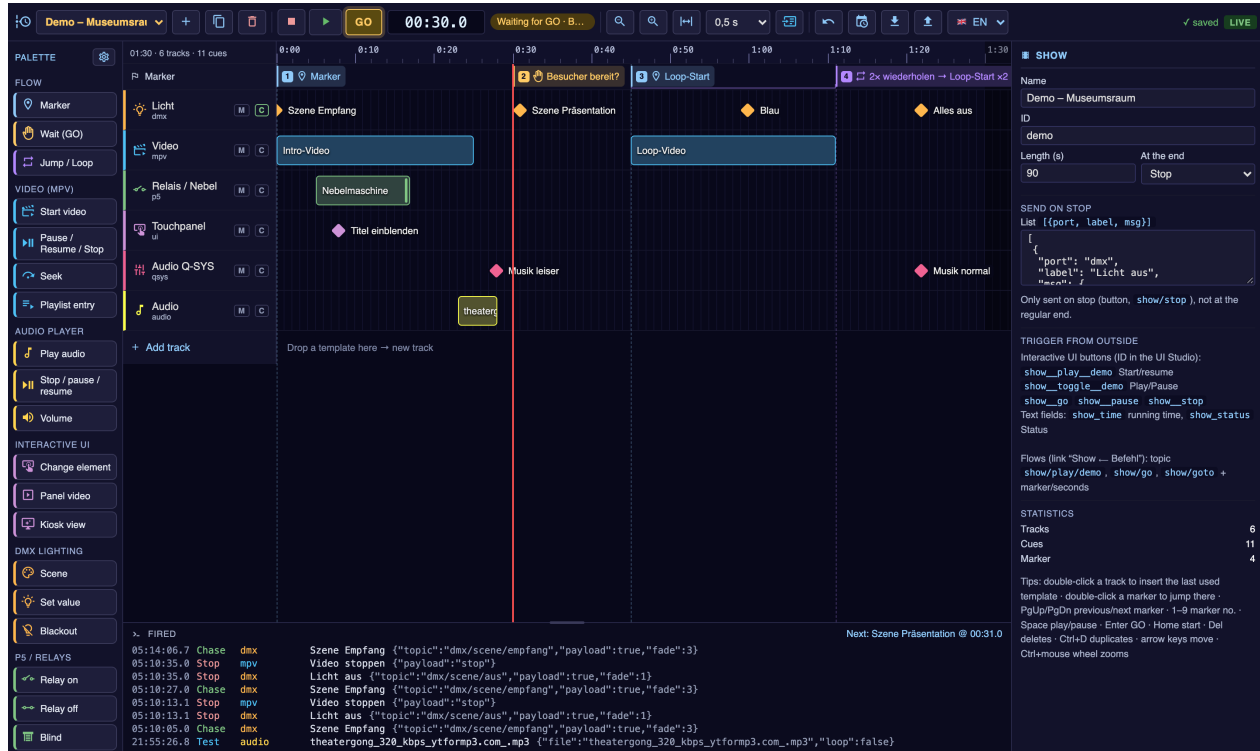


Figure 1 – Overview

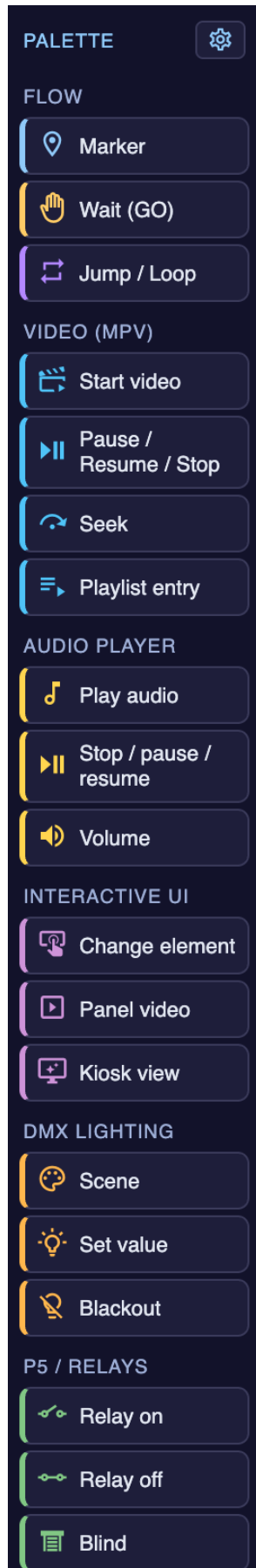
Area	Purpose
Header (top)	Show selection, transport (stop, play/pause, GO), time display, status, zoom, grid, schedule, import/export, language
Palette (left)	Marker types and templates to insert – sorted into groups
Timeline (centre)	Ruler, marker row and tracks with cues; the red line is the play position
Properties (right)	Settings of the selected element – or of the show when nothing is selected
Log (bottom)	All fired commands with time, kind, port and content

Header



Element	Function
Show selection	Choose the show to edit; next to it + (new show), copy (duplicate), bin (delete)
■ Stop	Stop the show and return to the start; sends the show's stop actions (Esc)
▶ / ⏸ Play/Pause	Play or pause (Space)
GO	Continue at a wait marker (Enter); flashes yellow while the show is waiting
00:30.0	Current position in minutes:seconds.tenths
Status	Stopped · Playing · Paused · Waiting for GO (with the marker name)
Magnifiers / arrow	Zoom out, zoom in, fit the whole show (also Ctrl + mouse wheel)
0,5 s	Grid that cues and markers snap to (off, 0.1 s ... 5 s)
Page icon	View follows the play position (blue = on)
↶	Undo (Ctrl+Z)
Calendar	Open the schedule
↓ / ↑	Export or import the show as a JSON file
DE / EN	Language of the interface
✓ saved	Save status – changes are saved automatically

Palette



The palette contains all building blocks, grouped by area.

- **Flow:** Marker, Wait (GO), Jump / Loop
- **Video, Audio player, Interactive UI, DMX lighting, P5/Relays, Audio/Q-SYS, Power/Video, General:** the templates for the devices

Inserting:

- **Drag** onto a track – the cue is placed where the mouse is released.
- **Drag onto “Drop a template here → new track”** – creates a new track for it.
- **Click** – inserts the block at the play position on a matching track.

The cog at the top opens the palette manager (chapter 9).

Timeline

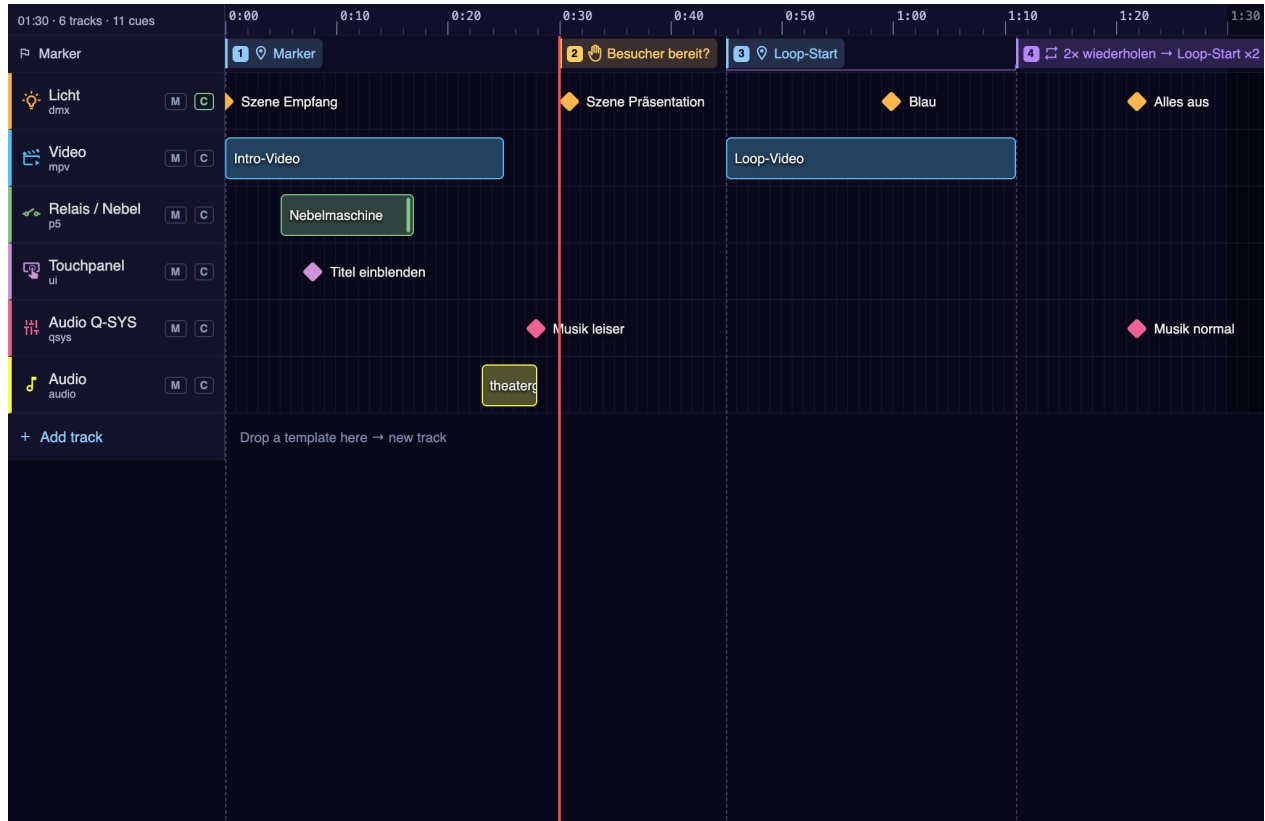


Figure 2 – Timeline with ruler, marker row and tracks

- **Ruler:** click or drag to set the play position.
- **Track header** (left): name, port, **M** = mute, **C** = chase. Clicking the name shows the track properties.
- **Point cue** (diamond): action without duration. **Bar cue:** action with duration; a bright edge on the right means a message is sent at the end.
- **Dashed lines** show the markers across all tracks.
- Double-clicking an empty spot on a track inserts the last used template.

Log

```
> FIRED
05:14:06.7 Chase dmx      Szene Empfang {"topic":"dmx/scene/empfang","payload":true,"fade":3}
05:10:35.0 Stop  mpv      Video stoppen {"payload":"stop"}
05:10:35.0 Stop  dmx      Licht aus {"topic":"dmx/scene/aus","payload":true,"fade":1}
05:10:27.0 Chase dmx      Szene Empfang {"topic":"dmx/scene/empfang","payload":true,"fade":3}
05:10:13.1 Stop  mpv      Video stoppen {"payload":"stop"}
05:10:13.1 Stop  dmx      Licht aus {"topic":"dmx/scene/aus","payload":true,"fade":1}
05:10:05.0 Chase dmx      Szene Empfang {"topic":"dmx/scene/empfang","payload":true,"fade":3}
21:55:26.8 Test  audio     theatergong_320_kbps_ytformp3.com_.mp3 {"file":"theatergong_320_kbps_ytformp3.com_.mp3","loop":false}
```

Each line is a sent command: time, kind (**Start**, **End**, **Chase**, **Test**, **Stop**), port, cue name and message. On the right you see what comes next. Drag the small line at the top edge to change the height (double-click = default height); clicking “Fired” collapses the log.

3 Step by step: your first show

Example: a room is prepared for visitors. The light comes on, an intro video plays, a fog machine runs for 12 seconds, then the show waits for the guide, and the presentation follows.

1. **Create a show.** Click + in the header and enter a name, e.g. "Exhibition room". The new show has one track "Light" and is 60 seconds long.
2. **Set the length.** With nothing selected, the properties on the right show the show itself. Enter e.g. 90 at **Length (s)** and choose "Stop", "From the start (loop)" or "Hold" at **At the end**.
3. **Switch the light on.** Drag **DMX lighting → Scene** from the palette to the very start of the "Light" track. Enter the scene ID on the right (e.g. "empfang") and set the fade time.
4. **Start the video.** Drag **Video (mpv) → Start video** onto the empty area below the tracks – a "Video" track is created. Enter the file path on the player on the right.
5. **Fog with a duration.** Insert **P5 / Relays → Relay on**, enter device and channel. Drag the right edge of the bar to 12 seconds. "Switch off at cue end" is ticked – after 12 s the relay switches off by itself.
6. **Wait for the guide.** Drag **Flow → Wait (GO)** onto the marker row, e.g. at 30 s, and give it a name ("Visitors ready?").
7. **Presentation.** After the wait marker, add another lighting scene (e.g. "praesentation").
8. **Test individually.** Click a cue and press **Test start** on the right – the command is sent immediately and appears in the log.
9. **Clean up on stop.** In the show properties under **Send on stop** enter e.g. "light off" and "stop video" (see chapter 7).
10. **Play.** Press **Play**. At the wait marker **GO** flashes – GO continues the show.

Saving is automatic. Every change is saved on the device after half a second ("✓ saved"). A running show picks up changes immediately, without interruption.

4 Editing cues

CUE

Label
Nebelmaschine

Start (s) 5 Duration (s) 12

Track
Relais / Nebel (p5)

☒ Active

ACTION

Template
P5 / Relays · Relay on

Device
rel1

Channel
1

☒ Switch off at cue end

MESSAGE TO THE DRIVER

Port
- as track (p5) -

Message at start (from template)

```
{
  "payload": {
    "cmd": "on",
    "dev": "rel1",
    "ch": 1
  }
}
```

Message at end (at start + duration)

```
{
  "payload": {
    "cmd": "off",
    "dev": "rel1",
    "ch": 1
  }
}
```

Editing the message yourself turns the cue into a "Free message".

▶ Test start ▶ Test end

→ Play position 📄 Duplicate ✕ Delete

Click a cue – its properties appear on the right:

- **Label:** name on the timeline. As long as you have not changed it yourself, it follows the inputs.
- **Start / Duration** in seconds. Duration 0 = point cue.
- **Track** and **Active** (inactive cues are skipped).
- **Template** and its fields, e.g. device, channel, "Switch off at cue end".
- **Message to the driver:** port and the messages at start and at the end – generated from the template.
- **Test start / Test end** send immediately.
- → **Play position, Duplicate, Delete.**

Editing the message directly turns the cue into a "Free message" – for special cases no template covers.

With the mouse

- **Move:** drag the cue – also onto another track.
- **Change duration:** drag the right edge of a bar.
- **Fine-tune:** arrow keys ← → move by one grid step.
- **Duplicate:** Ctrl+D (Mac: ⌘+D) – the copy is placed right behind it.

Example: video

The screenshot shows the 'CUE' configuration window for a video track. It includes fields for Label ('Intro-Video'), Start (s) (0), and Duration (s) (25). The Track is set to 'Video (mpv)' and is marked as 'Active'. Under the 'ACTION' section, the Template is 'Video (mpv) · Start video' and the File path is '/home/nanoblue/videos/intro.mp4'. The 'Loop endlessly' checkbox is unchecked. The 'MESSAGE TO THE DRIVER' section shows a Port set to '- as track (mpv) -'. The 'Message at start (from template)' is a JSON object:

```
{  "commands": [    {      "set_property",      "loop-file"    }  ]}
```

. The 'Message at end (at start + duration)' is 'empty = none'. At the bottom, there are buttons for 'Test start', 'Test end', 'Play position', 'Duplicate', and 'Delete'.

The “Start video” template loads a file on the player and plays it – optionally in an endless loop.

The bar duration is only for orientation (the length of the video). A video keeps playing until another video cue or a stop action arrives.

Do not enable chase on video tracks – otherwise a video restarts after every jump.

Example: audio

The template **Audio player → Play audio** plays an audio file through the device’s audio player – e.g. a gong, a voice-over or an ambience. You can set the file (name from the player’s media, path or URL), “Loop endlessly”, the volume and “Stop at cue end”. Further templates: **Stop / pause / resume** and **Volume**.

When jumping on the timeline, audio is not moved to the matching position – for gongs, voice-overs and effects this is not needed.

5 Markers: wait, jump and loop

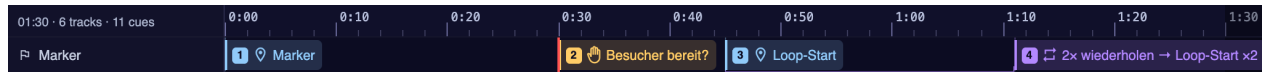


Figure 3 – Marker row: numbered markers, wait marker (GO), jump target and jump with repeats

Kind	Effect
Marker	Just a named point – jump target for loops and for the show/goto command.
Wait (GO)	The show stops at this point until GO arrives. Cues exactly at the marker fire only with GO.
Jump / Loop	Jumps to another marker (or the start). Repeats: number, 0 = endless.

WAIT (GO)

Name

Time (s)

Type

Wait (GO) ▾

The show stops here until **GO** arrives: button in the Show Studio, Interactive UI button `show__go` or topic `show/go` (e.g. from the PuKK button).

Jump: double-click the marker, PgUp/PgDn (previous/next) or key 1–9 (number on the flag).

→ Play position

📄 Duplicate

✕ Delete

Wait marker

GO can come from:

- the GO button or the Enter key in the Show Studio,
- a touch panel button with the ID `show__go`,
- a push button, NFC point or another flow via the topic `show/go`.

The status then shows “Waiting for GO” with the marker name.

JUMP / LOOP

Name

2x wiederholen

Time (s)

71

Type

Jump / Loop

Jump to

Loop-Start (00:45.0)

Repeats (0 = endless)

2

After the last repeat the show continues. An endless loop cannot be left with GO – use a wait marker or `goto` instead.

Jump: double-click the marker, PgUp/PgDn (previous/next) or key 1–9 (number on the flag).

→ Play position

📄 Duplicate

✕ Delete

Jump / Loop

“Jump to” selects the target, “Repeats” the number. After the last repeat the show continues.

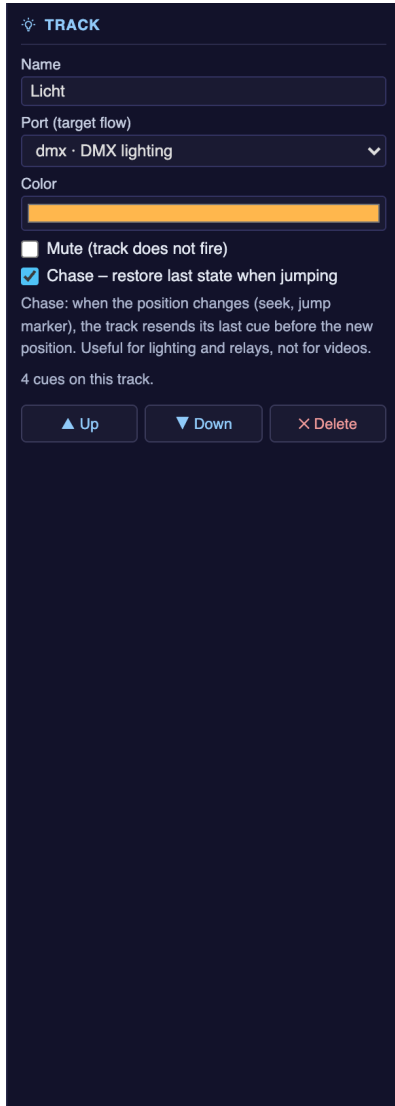
An endless loop cannot be left with GO – add a wait marker or jump from outside with [show/goto](#).

Jumping to a marker

- **Double-click** the marker flag.
- **PgUp / PgDn** (Mac: Fn+↑ / Fn+↓): previous or next marker.
- **Keys 1–9**: marker number 1–9 – the number is shown on the flag.

Rehearsal tip: After jumping to a wait marker, the show is “Waiting for GO”. A single GO continues from exactly this point.

6 Tracks



The screenshot shows a dark-themed interface for editing a track. At the top, it says "TRACK" with a small icon. Below this, there are several sections: "Name" with a text input field containing "Licht"; "Port (target flow)" with a dropdown menu showing "dmx · DMX lighting"; "Color" with a horizontal color bar; and a section with two checkboxes: "Mute (track does not fire)" which is unchecked, and "Chase – restore last state when jumping" which is checked. Below the checkboxes is a paragraph explaining the Chase function: "Chase: when the position changes (seek, jump marker), the track resends its last cue before the new position. Useful for lighting and relays, not for videos." Below this is a line stating "4 cues on this track." At the bottom, there are three buttons: "▲ Up", "▼ Down", and "✕ Delete".

TRACK

Name
Licht

Port (target flow)
dmx · DMX lighting

Color
[Orange bar]

☐ Mute (track does not fire)
☒ Chase – restore last state when jumping

Chase: when the position changes (seek, jump marker), the track resends its last cue before the new position. Useful for lighting and relays, not for videos.

4 cues on this track.

▲ Up ▼ Down ✕ Delete

Clicking the track name shows the track properties:

- **Name and Color.**
- **Port (target flow):** where the cues of this track are sent.
- **Mute:** the track does not fire – handy for rehearsals without fog or sound.
- **Chase:** after a jump or seek, the track resends its last cue before the new position. Useful for lighting and relays, so the room is always in the right state.
- **Up / Down** change the order, **Delete** removes the track including its cues.

The M (mute) and C (chase) toggles are also available directly in the track header.

7 Show settings

SHOW

Name
Demo – Museumsraum

ID
demo

Length (s) 90 At the end Stop

SEND ON STOP
List [{port, label, msg}]

```
[
  {
    "port": "dmx",
    "label": "Licht aus",
    "msg": f
```

Only sent on stop (button, show/stop), not at the regular end.

TRIGGER FROM OUTSIDE
Interactive UI buttons (ID in the UI Studio):
show__play__demo Start/resume
show__toggle__demo Play/Pause
show__go show__pause show__stop
Text fields: show__time running time, show__status
Status

Flows (link "Show — Befehl"): topic
show/play/demo, show/go, show/goto + marker/seconds

STATISTICS
Tracks 6
Cues 11
Marker 4

Tips: double-click a track to insert the last used template · double-click a marker to jump there · PgUp/PgDn previous/next marker · 1–9 marker no. · Space play/pause · Enter GO · Home start · Del deletes · Ctrl+D duplicates · arrow keys move · Ctrl+mouse wheel zooms

With nothing selected, the properties show the show itself:

- **Name** (the ID stays fixed).
- **Length** in seconds.
- **At the end**: Stop, From the start (loop) or Hold.
- **Send on stop**: list of commands executed on a stop – not at the regular end. Example:

```
[{"port":"dmx","label":"Light off","msg":{"topic":"dmx/scene/aus","payload":true}}]
```

- **Trigger from outside**: the IDs and topics touch panels and flows use to control this show.
- **Statistics** and tips for operation.

8 Playing and controlling from outside

In the Show Studio

Play/pause, stop and GO in the header or via the keyboard. The editor always shows the show that is currently active on the device. If you work on a different show and press play, that show is loaded and started.

From the touch panel (UI Studio)

Button ID in the UI Studio	Effect
show__play__<show ID>	Start or resume the show
show__toggle__<show ID>	Toggle play/pause
show__go	GO at a wait marker
show__pause	Pause
show__stop	Stop (with stop actions)
Text field show__time	shows the running time (mm:ss)
Text field show__status	shows the status (currently in German, e.g. "Läuft – Museumsraum")

The play button of the running show is automatically marked as active.

From push buttons, sensors and other flows

Via the link “Show ← Befehl” in the nanoblu Editor with topics such as [show/play/<ID>](#), [show/go](#), [show/stop](#) or [show/goto](#) (payload: seconds or marker name). This lets e.g. a PuKK push button give GO.

Schedule

ACTIVE	TIME	DAYS	ACTION	SHOW
☒	10:00	Mo Tu We Th Fr Sa Su	Start	Demo – Museumsraum
☒	18:00	Mo Tu We Th Fr Sa Su	Stop	Demo – Museumsraum

+ Entry

Cancel Save

Figure 4 – Schedule: start at 10:00 on weekdays, stop daily at 18:00

The calendar icon in the header opens the schedule. Each entry has a time, weekdays (none selected = daily), an action (start, restart, stop) and a show. The schedule runs on the device – even when nobody has the editor open.

9 Managing the palette and custom templates

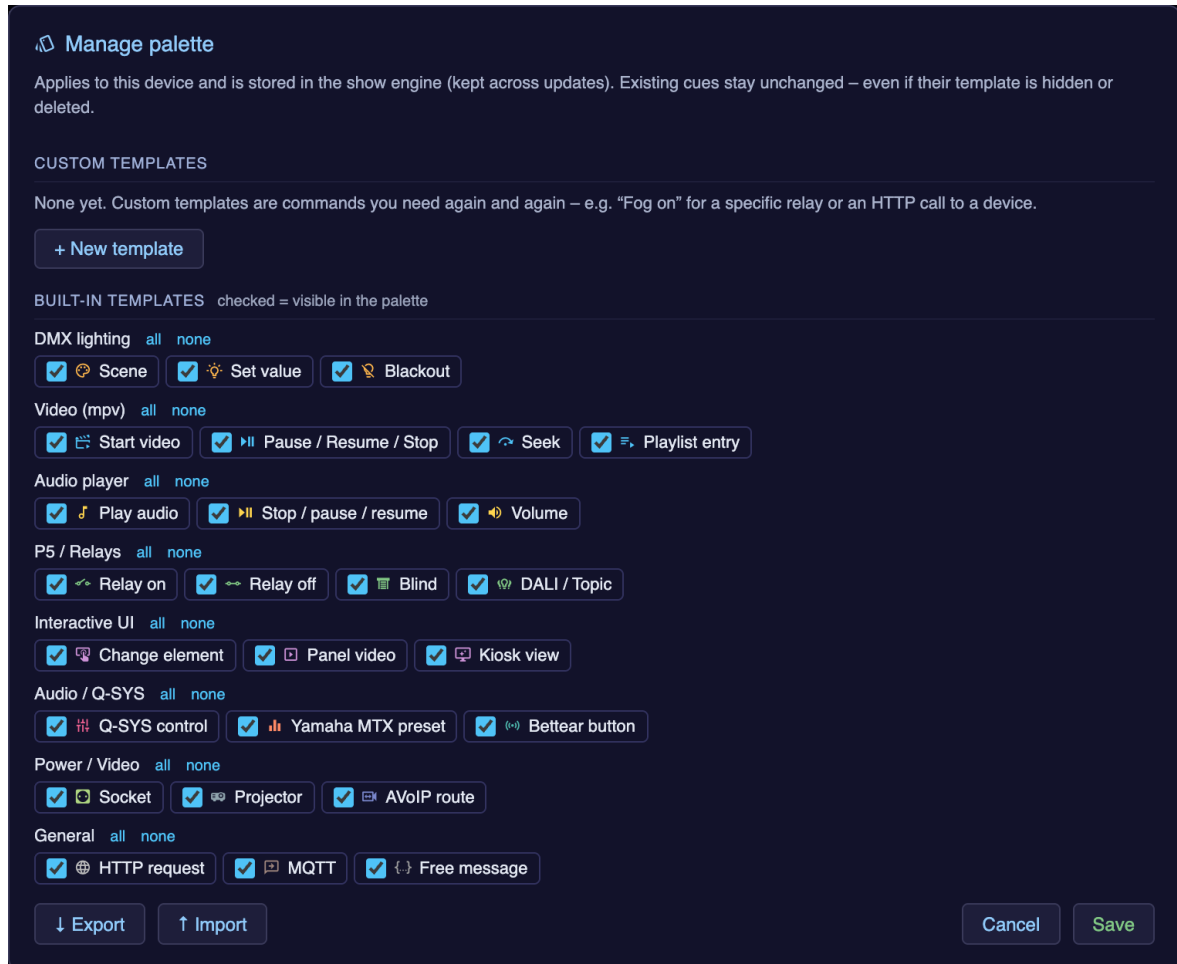


Figure 5 – Manage palette

- **Show/hide built-in templates:** tick or untick, per group with “all”/“none”. This way each device only shows the templates it really needs.
 - **Custom templates:** “+ New template” – for commands you need again and again.
 - **Export / Import:** save the palette as a file and transfer it to other devices. It is applied only after “Save”.
- The palette applies per device. Existing cues stay unchanged, even if their template is hidden or deleted.

Creating a custom template

New template

Name in the palette:

Group: Port (target flow):

Icon: Default duration (s):

Label on the timeline (optional):

Message at start (JSON):

```
{
  "payload": {
    "cmd": "on",
    "dev": "{{device:rel1}}",
    "ch": "{{channel:1}}"
  }
}
```

Message at end (optional, at start + duration):

```
{
  "payload": {
    "cmd": "off",
    "dev": "{{device:rel1}}",
    "ch": "{{channel:1}}"
  }
}
```

Placeholders become input fields in the cue:
{{name}} or with a default {{name:value}} .
Always write them in quotes. If a placeholder stands alone, the value is converted: "{{channel:1}}" → number 1, "{{on:true}}" → true (checkbox).
Example P5 relay:
Start: {"payload":{"cmd":"on","dev":"{{device:rel1}}","ch":"{{channel:3}}"}
End: {"payload":{"cmd":"off","dev":"{{device:rel1}}","ch":"{{channel:3}}"}
Input fields:
device = rel1 channel = 1 · number
Example with defaults → port p5
Label: Fog Ch1
Start: {"payload":{"cmd":"on","dev":"rel1","ch":1}}
End: {"payload":{"cmd":"off","dev":"rel1","ch":1}}

Figure 6 – Custom template “Fog on” with placeholders

1. **Name, group, port, icon** and optionally a default duration.
2. **Message at start** as JSON. Values that should be adjustable in the cue become placeholders: `"{{channel:1}}"` – name “channel”, default 1.
3. **Optionally a message at the end** – it is sent at start + duration.
4. **Optionally a label** with placeholders, e.g. `Fog Ch{{channel}}`.
5. **Check the preview** (right: detected input fields and the resulting message), then “Apply” and “Save”.

Placeholder rules: Always write them in quotes. If a placeholder stands alone, the value is converted: number → number field, true/false → checkbox. Inside a text (e.g. `dmx/{{target}}/dim`) it is inserted as text.

10 Language

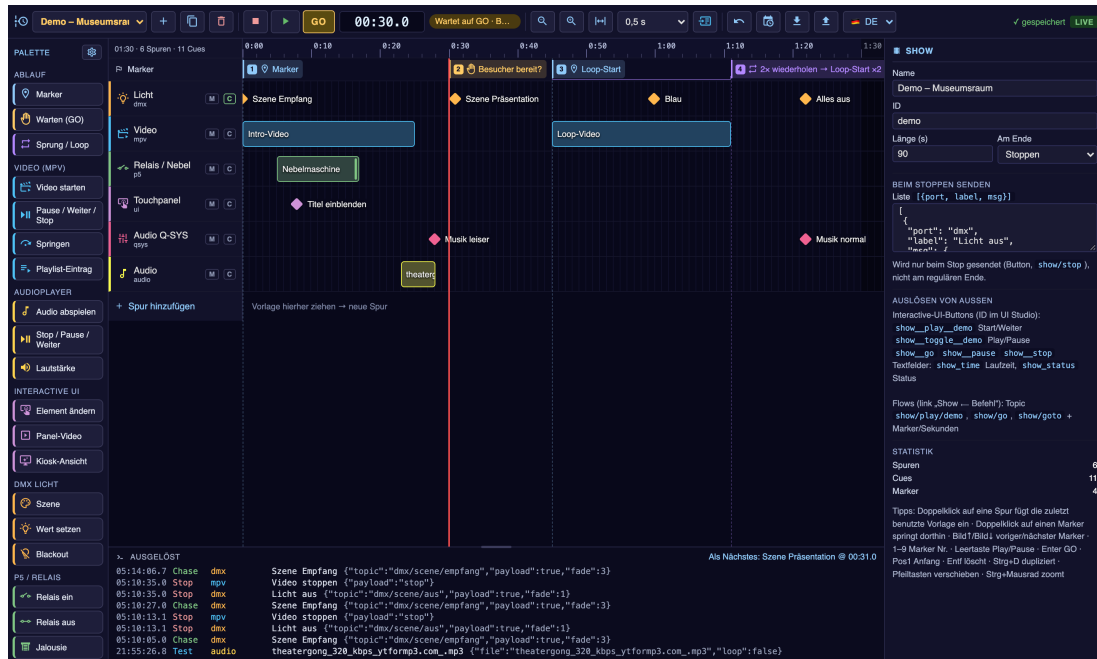


Figure 7 – Show Studio in German

The DE/EN selector in the header switches the interface. The choice is remembered in the browser and also applies to the UI Studio. Your own content (names of shows, cues and templates) stays as you entered it.

11 Keyboard shortcuts

Key	Function
Space	Play / Pause
Enter	GO
Esc	Clear selection – with nothing selected: stop
Home	Go to the start
PgUp / PgDn (Mac: Fn+↑ / Fn+↓)	Previous / next marker
1 – 9	Go to marker no. 1–9
← / →	Move the selected cue or marker by one grid step
Del / Backspace	Delete selection
Ctrl+D (Mac: ⌘+D)	Duplicate selection
Ctrl+Z (Mac: ⌘+Z)	Undo
Ctrl+S (Mac: ⌘+S)	Save now
+ / – · Ctrl + mouse wheel	Zoom
Double-click track	Insert the last used template
Double-click marker	Jump to the marker
Double-click marker row	Add a new marker

12 Tips and troubleshooting

Observation	Cause / solution
A cue appears in the log, but nothing happens on the device.	The output “Show → ...” in the “Show Controller” flow (nanoblock Editor) is not yet connected to the driver flow (add a link-in and connect it). Video (mpv), audio player and touch panel work without further connection.

The running video stops when the demo show plays.	The demo uses example paths. Change the paths of the video cues to real files on the player or mute the video track.
After a jump the light is wrong.	Enable chase for the lighting track.
“OFFLINE” at the top right.	Device not reachable – check the connection. Continue editing after it reconnects.
“SIMULATION” at the top right.	No device found (e.g. file opened locally). Shows are then only stored in the browser.
The show is stopped after the device restarts.	Intended – no show starts by itself after a restart. Use the schedule for automatic operation.