

GoBubble Operating Manual

A practical, how-to guide to loading, triggering, editing, organising, recording and managing audio clips in GoBubble — plus remote web control, MIDI mapping and a developer/build appendix.

Table of Contents

1. [Introduction](#)
 2. [Quick Start \(5 minutes\)](#)
 3. [The Main Window](#)
 4. [Loading Clips](#)
 5. [Recording onto a Pad](#)
 6. [Playing Clips & Groups](#)
 7. [Selection & Keyboard Control](#)
 8. [Pages](#)
 9. [The Clip Settings Dialog](#)
 10. [Undo & Redo](#)
 11. [Global Settings](#)
 12. [Output Routing](#)
 13. [Files & the Managed Audio Library](#)
 14. [Keyboard Shortcuts Editor](#)
 15. [Layouts & Packages](#)
 16. [MIDI Control](#)
 17. [Screen Lock \(Kiosk Mode\)](#)
 18. [Remote Web Control](#)
 19. [Startup Integrity Scan](#)
 20. [How-To Recipes](#)
 21. [Troubleshooting](#)
- [Appendix A — Building from Source](#)
-

1. Introduction

GoBubble is an interactive audio clip launcher: a grid of **pads**, spread across **pages**, each holding an audio clip with its own routing, trimming, level envelope, effects, VST inserts and appearance. You can trigger clips by mouse, keyboard, MIDI, or from a phone or tablet over your local network. This manual walks through every feature as a set of tasks you can follow step by step.

2. Quick Start (5 minutes)

First Launch: Setup Wizard & License Activation

GoBubble is currently distributed as **evaluation/trial software**. A short setup wizard walks you through what's needed before you can use it, and reappears on **every launch** — not just the first — until you tick its “don't show again” box:

1. **Welcome** — a one-line overview of the wizard, plus a “**Don't show this walkthrough again**” checkbox. Leave it unticked if you'd like the walkthrough back as a refresher later.
2. **Evaluation Agreement** — read and accept the EULA. Shown only the first time; once accepted it's skipped on every later showing.
3. **License Activation** — paste the license key you were given and click **Activate**. GoBubble checks it against the license server and won't let you continue until activation succeeds. Also shown only the first time.
4. **Audio Device** — pick your input/output device (the same panel as Settings → Audio).
5. **Folders** — optionally set where **Show Files**, **Audio Files**, **Recordings** and **Processing Presets** are stored. All four can be left blank or changed later in **Settings** → **Files** (see [Section 13](#)).
6. **Tips** — a short orientation, then **Get Started** drops you into the app.

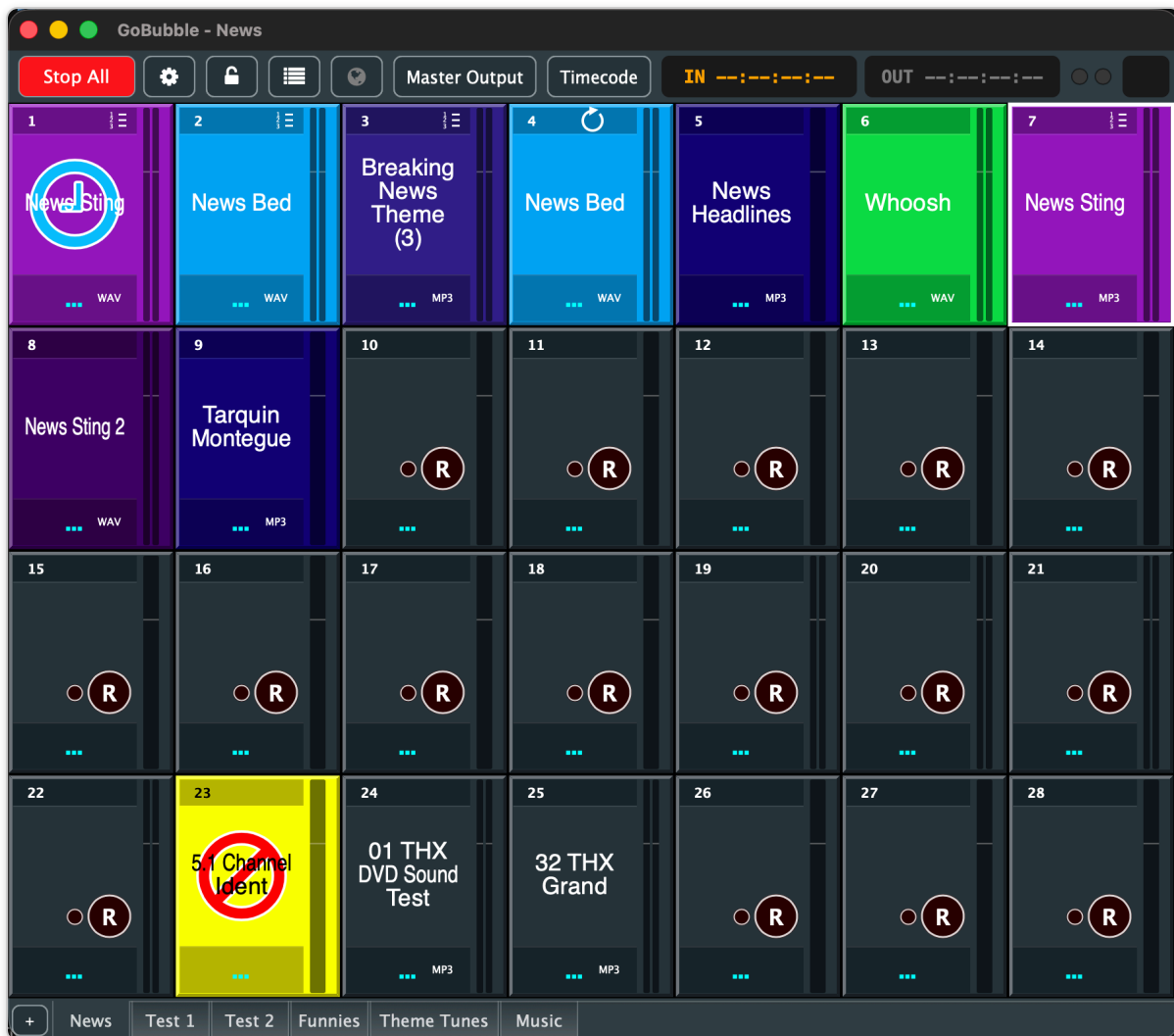
Once activated, your license key is bound to this machine, and GoBubble periodically checks in with the license server in the background to confirm it's still valid. If the server is briefly unreachable (e.g. unreliable venue wifi), GoBubble keeps working from its last known-good check-in for up to **14 days** before it insists on reconnecting. If a license is revoked or has expired, GoBubble explains why and won't let you continue until it's resolved.

The 5-Minute Walkthrough

1. **Launch GoBubble.** A startup scan checks your last layout for problems and gives you a thumbs-up if all is well (see [Section 19](#)).
2. **Add a clip.** Drag an audio file (WAV, AIFF, MP3, FLAC, OGG, M4A, WMA) from Finder onto any empty pad.
3. **Play it.** Left-click the pad. Right-click to stop. Click **Stop All** (the red button, top-left) to stop everything.
4. **Trim it.** Shift-click the pad to open its settings, go to the **Edit** tab, and drag the green **IN** and red **OUT** lines to set the playable range.
5. **Colour it.** On the **Display** tab, pick a button colour and text colour so it's easy to find on stage.
6. **Save. File** → **Save** (Cmd+S). GoBubble reopens this layout automatically next launch.

That's the core loop. Everything below expands on it.

3. The Main Window



The main window: toolbar across the top, a page of pads below, page tabs along the bottom

Toolbar (top), left to right:

- **Stop All** — red, unmistakable; stops every playing clip on every page.
- **Cog-wheel** — opens Settings.
- **Padlock** — the Lock button (open icon when unlocked, closed when locked; see [Section 17](#)).
- **List** icon — opens the **Recordings library** ([Section 5](#)).
- **Globe** icon — the **Remote** indicator (green when web clients are connected); click it to list and disconnect clients ([Section 18](#)).
- **MIDI monitor** — shows the most recently received MIDI note or CC.
- **CPU** and **Memory** LED meters — the app's live load.

Undo and Redo have **no toolbar buttons** — use **Cmd+Z** / **Cmd+Shift+Z** or the **Edit** menu.

Clip grid: the pads, up to 12 × 12 per page (size set in Settings → General/Grid).

Page tabs (bottom): switch pages, add pages, rename and delete them ([Section 8](#)).

What a pad shows: its **ID number** (top-left), **group membership** (top-right, e.g. “G1” or “G1,3”), the **clip name** (centre, wrapping to as many lines as the font allows), the **output routing** (bottom-left), a **timer** (bottom-centre, always cyan), a **loop icon** (bottom-right) when looping, a **behaviour badge** when the clip has behaviour commands, and live **per-channel meters** down the right edge (two for stereo, up to sixteen for multichannel). An **empty** pad shows a round red **R** record button instead ([Section 5](#)).

4. Loading Clips

Ways to load audio onto a pad:

- **Drag-and-drop:** drop one or more files onto a pad. Multiple files fill that pad and the ones after it in order. If a target pad already has a clip you're asked to **Replace** it, **Skip** to the next pad, or **Cancel** the rest of the drop. Dropping an *unsupported* file shows a clear error rather than failing silently.
- **Load File:** in the clip settings dialog, use the **Load File** button (available above the tabs).
- **From a recording:** load any past take from the Recordings library ([Section 5](#)).

Supported formats: WAV, AIFF, MP3, FLAC, OGG, M4A, WMA. **Multichannel** files with up to 16 channels load intact (5.1 and beyond); every channel gets its own meter.

The pad's name is taken from the file name, with the extension removed.

Copy-on-add and the managed library

When you have set an **Audio Files location** (Settings → Files), adding a file to a pad **copies it into your managed audio library** and loads the copy, so your originals are never altered and your project stays self-contained. Until a new layout is saved for the first time, added audio is staged under an *Untitled* folder; on **Save** it's filed under a folder named for the layout. See [Section 13](#).

Deleting a clip's file from disk

To remove a clip's underlying file permanently, open the clip's settings and use **Delete File** (next to Clear). It asks for confirmation, then deletes the file and clears the pad. **Clear** (or Cmd+Delete) only empties the pad — it leaves the file on disk.

5. Recording onto a Pad

Every **empty** pad shows a round red **R** button in its bottom-right corner.

To record a take:

1. Choose your input device in **Settings** → **Audio**. (GoBubble opens your audio inputs on launch; macOS asks for microphone permission the first time.)
2. Click the **R** button on an empty pad to start capturing your inputs. The disc turns **bright red** while recording (it's a dark, dim red at rest).
3. Click it again to stop. The take is instantly playable on that pad, and the pad is given a unique random colour so recordings are easy to tell apart.

Recordings are saved as **WAV** files in a **Recordings** subfolder of your audio library (Settings → Files) or, if none is set, in *Documents/GoBubble Recordings*.

The Recordings library

Click the **list** icon in the toolbar to open a window of all your past recordings. For each take you can:

- **Preview** — audition it through the outputs (click again to stop).
 - **Load** — place it on any empty pad (pick the pad from the popup).
 - **Delete** — remove it permanently (with confirmation).
-

6. Playing Clips & Groups

- **Left-click** a pad to trigger its clip (this also moves the selection cursor there).
- **Right-click** a pad to stop it.
- **Spacebar** toggles the *selected* pad: stops it if sounding, triggers it if not.
- **Stop All** (toolbar) or **Escape** stops every playing clip on every page.

Groups (exclusive playback)

Assign a clip to one or more groups (**G1–G16**) in its settings. Triggering a clip in a group automatically **stops the other playing clips that share a group with it** — ideal for “only one backing track at a time.” A clip can belong to several groups at once; ungrouped clips are unaffected. Behaviour timelines can also **Play Group** (launch a whole scene) and **Stop Group** ([Section 9.3](#)).

7. Selection & Keyboard Control

A **white border** marks the *selected* (cued) pad. The **arrow keys** move it; clicking any pad moves it there. With a pad selected:

Default key	Action
Space	Play / stop the selected pad
Cmd+L	Toggle looping on the selected pad
Cmd+Delete	Clear the selected pad back to defaults
Cmd+Z / Cmd+Shift+Z	Undo / redo
Escape	Stop all clips
Cmd+S / Cmd+O	Save / load layout
Cmd+,	Open settings

All of these can be remapped in **Settings** → **Shortcuts** ([Section 14](#)). Arrow-key navigation is fixed.

Moving and copying clips

- **Cmd-drag** a clip onto another pad to **move** it (including across pages — drag onto a page tab first to switch).
- **Shift-drag** a clip onto another pad to **copy** it.

Pad numbers are always positional (page 1 starts at 1, each page continues on from the last), so moving, copying, or adding/removing a page renumbers pads around it. Any behaviour command that targets a pad by number is automatically retargeted to follow that same physical pad through the renumbering, so a Play/Stop/Cue command set up before a rearrange still fires at the pad you originally meant.

8. Pages

Use the **page tabs** at the bottom to organise clips into sets (songs, scenes, acts).

- **Add a page:** click +.
 - **Rename a page:** double-click a tab, or right-click → **Rename Page**.
 - **Delete a page:** right-click a tab → **Delete Page**. This removes the page *and all its clips*; you're asked to confirm if the page has content, and the **last remaining page can't be deleted**. A page delete is undoable.
 - **Switch pages while dragging a clip:** hover a clip over a page tab to switch, then drop it on the destination pad.
-

9. The Clip Settings Dialog

Shift-click a pad to open its settings. The title bar shows the pad number and clip name. The footer has **Play/Pause**, **Stop** and **Loop** buttons for auditioning (see below), plus **<** and **>** arrows that step to the previous/next pad on the page — staying on whichever tab you're viewing. **Load File**, **+Silence**, **Clear** and **Delete File** sit on the left of the header strip above the tabs; the **group selector** and the **MIDI trigger note** picker (with its **Learn** button) sit on the right — all available from any tab. The tabs themselves are **Edit**, **Display**, **Behaviour** and **Processing**.

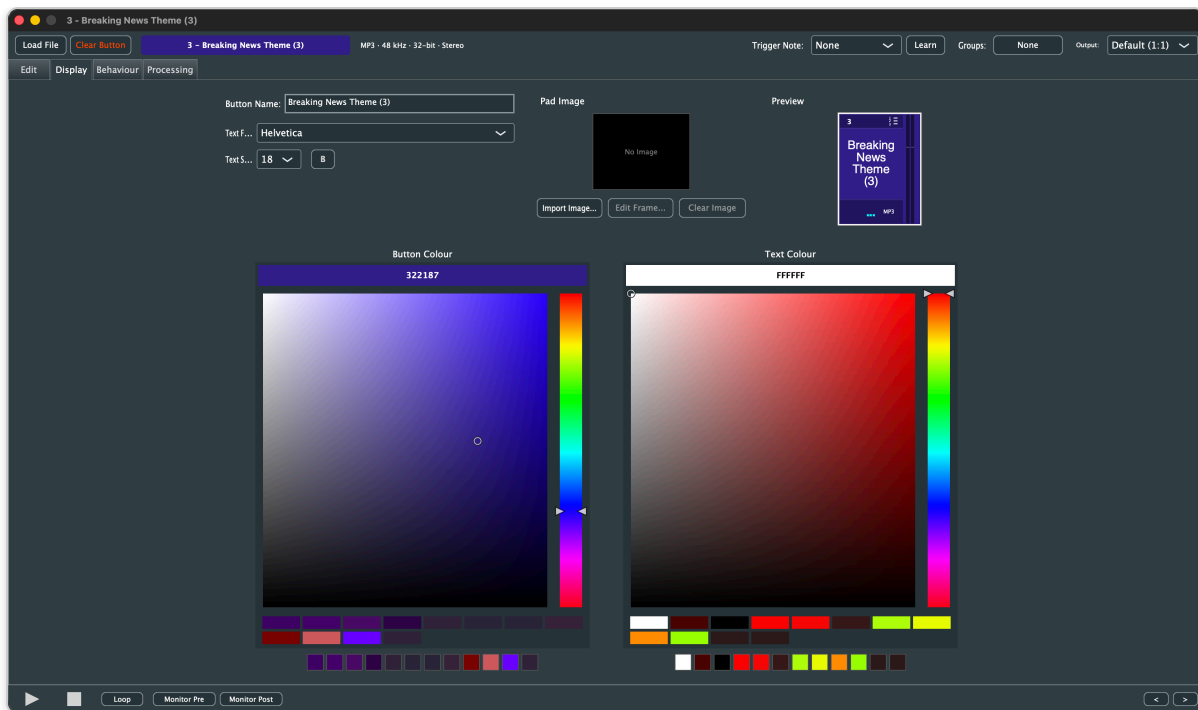
Auditioning: Play, Pause, Range Preview & Loop

The footer's **Play** button doubles as **Pause**: click it while the clip (or a Monitor Pre/Post preview — see [Section 9.4](#)) is already playing and it stops in place; click again and it resumes from exactly where it left off, rather than restarting from IN.

If you've dragged out a range selection on the Edit tab's waveform, **Play** previews just that range and stops automatically at its end instead of playing the whole clip; with nothing selected, Play covers the clip's whole IN–OUT span as usual. **Stop** is always a full reset — back to IN, or to the range's start if one is armed — and, like the toolbar's **Stop All**, it also stops this clip anywhere else it's playing (e.g. via a behaviour chain), not just here.

Loop repeats whatever Play is currently previewing — the selected range, or the whole clip if nothing's selected — purely for editing/auditioning. It's entirely separate from the clip's own real **Loop** setting on the [Processing tab](#), and a real trigger (grid click, MIDI, or a behaviour command) always plays the clip normally, ignoring anything armed here.

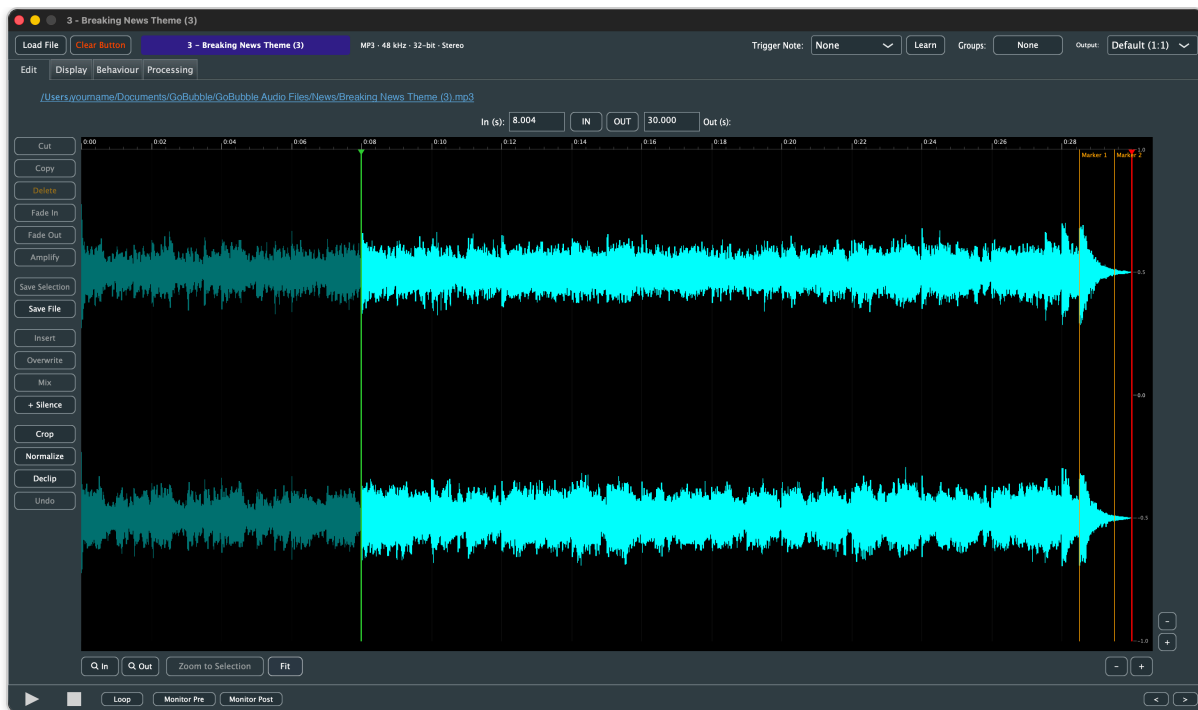
9.1 Display Tab



The Display tab: button name, font, pad image, and side-by-side button/text colour pickers with history strips

- **Button Name:** edit the pad's name (press Return to apply).
- **Button Colour / Text Colour:** two independent colour pickers side by side, each with a **history strip** of the last 12 colours used — click a swatch to reuse it. Both histories are shared app-wide and persist across clips and sessions. The text colour applies to the pad's name, ID, group, output label and loop icon; the timer stays cyan.
- **Text Font / Text Size / Bold:** typeface, size (8–72 pt) and a **B** toggle for bold on the pad name. Long names wrap onto multiple lines.

9.2 Edit Tab



The Edit tab: full waveform view with the IN marker, editing toolbar on the left, and Fit/zoom controls along the bottom

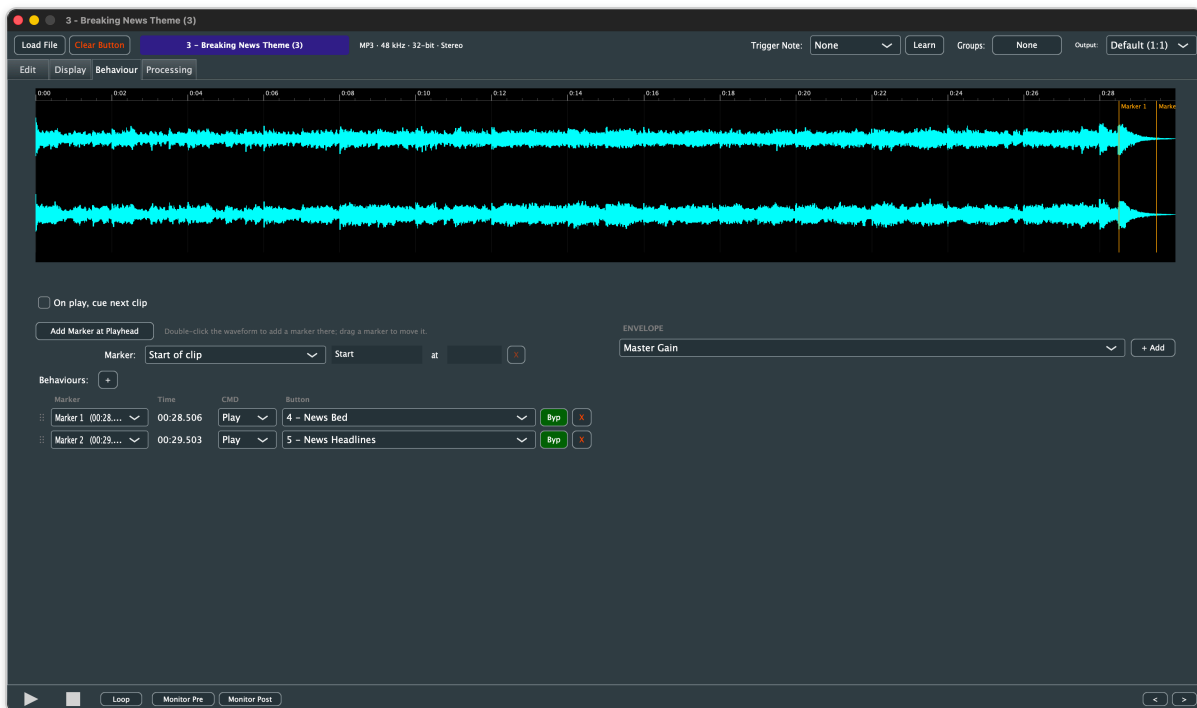
A simple waveform editor for trimming and reshaping the audio file itself — level, effects, routing and MIDI now live on the [Processing tab](#) and the header strip.

- **Format readout:** when the pad is empty this row shows the **Format/Input** pickers for recording ([Section 5](#)); once a clip is loaded it instead states the loaded file's actual format, e.g. `WAV · 48 kHz · 24-bit · Stereo`.
- **In (s) / Out (s) fields:** sit centred above the waveform, directly over the range they control, alongside the zoom cluster.
- **Waveform view:** a zoomable waveform with a graduated time scale along the top.
 - **Click anywhere** to jump the playhead there. If the clip was playing it keeps playing from that point; if it was stopped it **stays stopped** (no accidental blip).
 - **Scrub:** grab the yellow playhead (visible dimmed even when stopped) and drag — you hear the audio as you move. Release to pause, parked at that spot; the footer play button resumes from there.
 - **Spacebar** plays/stops the clip right from the waveform view (no need to click the waveform first).
 - **IN / OUT markers:** drag the green (IN) and red (OUT) lines to set the playback range. Shaded areas outside the range don't play. Named behaviour markers are also shown here for context, but — unlike on the Behaviour tab — they're **display-only**: clicking near one just moves the playhead rather than dragging it, so you can't nudge a behaviour marker by accident while editing audio.
 - **Zoom:** + / – buttons, mouse wheel, **Fit** (whole file), **Zoom to Selection** (the IN–OUT range), and the magnifier **In / Out** buttons, which zoom tight on the respective marker for precise placement. Your zoom position is remembered per clip when you reopen the tab.
- **Editing toolbar** (directly below the waveform), grouped left to right:
 - **Cut / Copy / Delete** act on the **IN–OUT range** (“the selection”). Cut and Delete are disabled until IN/OUT narrows below the whole file. **Save Selection...** exports the IN–OUT range to a new file

without touching the loaded clip.

- **Insert / Overwrite / Mix** paste the clipboard at the **playhead** (not the IN–OUT selection): **Insert** splices it in and pushes everything after the playhead later, so the clip gets longer; **Overwrite** replaces existing audio for the clipboard's own duration; **Mix** sums the clipboard onto the existing audio instead of replacing it — useful for layering a sound effect under existing audio (if the result peaks over 0 dB, run **Normalize** afterwards to bring it down).
- **Undo** reverts the most recent Cut/Delete/Insert/Overwrite/Mix. None of these show a confirmation dialog — Undo covers them.
- **Bottom row — whole-file processors:**
 - **Normalize:** sets the clip volume so its loudest peak (within IN/OUT) plays at a target level (default 0 dBFS; the target % is set in Settings → General). Non-destructive — it only moves a hidden gain trim.
 - **Crop:** writes a new file containing only the audio *between the IN and OUT points*, names it after the original with a `_copy` suffix, and loads that copy onto the pad. Non-destructive — your original file is untouched. A confirmation dialog appears first; re-cropping a `_copy` won't stack suffixes.
 - **Declip:** redraws hard-clipped (flat-topped) sections of the waveform into a smooth curve, writing a new `_declip` file and loading it — for salvaging a recording that clipped.

9.3 Behaviour Tab



The Behaviour tab: waveform with named markers, the behaviours table listing Play/Stop commands per marker, and the Envelope panel

A larger waveform sits above two stacked sections: **Envelope** and **Behaviour**.

Envelope — automate any Processing-tab parameter (Compressor, all four EQ bands plus High/Low-Pass, Delay, Gate/Expander, De-hisser, Pan, Reverb, Volume) over the clip's timeline:

- **+ Add** picks a parameter from the dropdown and adds it as a track.
- Click a track's name to make it **active** — double-click the waveform to add a breakpoint to it, drag existing points to reshape the curve. Only one track is active at a time, but any number can be visible together.
- Each track has **Byp** (bypass — keep the automation but ignore it during playback), **Vis** (show/hide its curve on the waveform without losing it), **Copy/Paste** (rescales proportionally into whatever range the destination parameter has, so you can copy a Pan curve onto an EQ frequency and it still makes sense), and **X** (delete the track entirely), plus a live value readout next to its name.

Behaviour — unchanged: place **named markers** at specific times on the waveform; each marker is listed with an **editable timecode** — click a marker's time to type an exact value. A behaviour is a table of steps, each with a **Marker/Time** trigger, a **CMD** and a target:

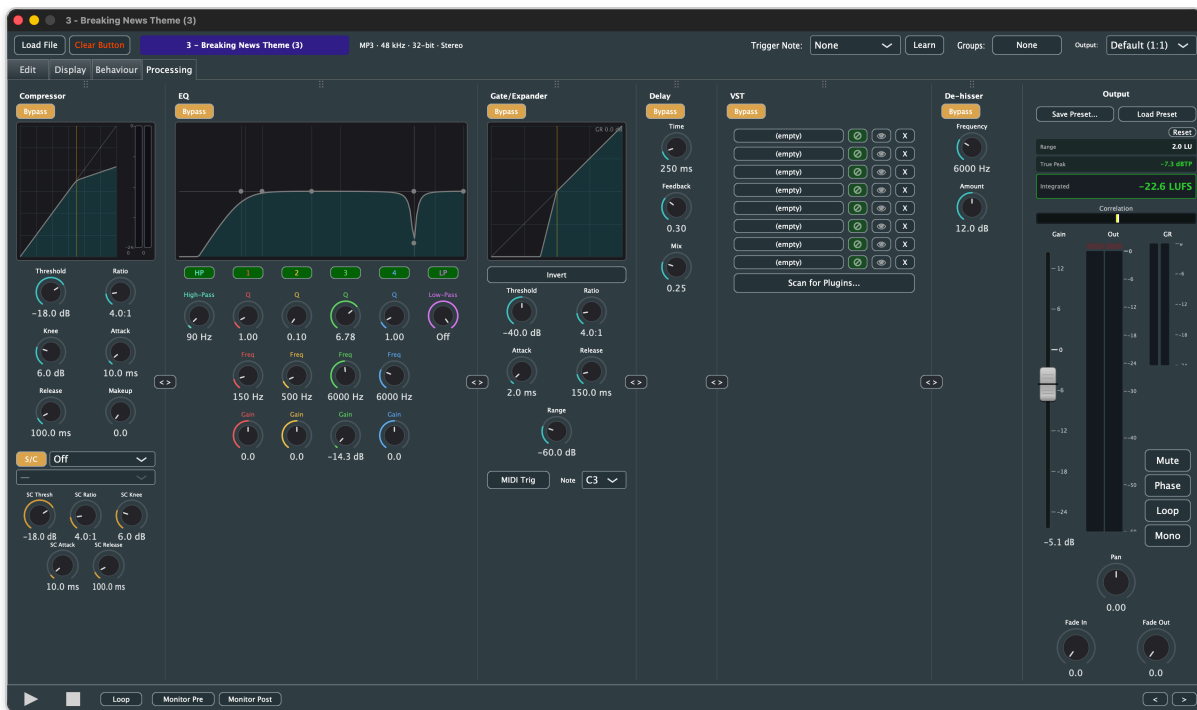
- **Play / Stop / Cue** — act on a single target **Button** (Cue arms it for the spacebar).
- **Play Group / Stop Group** — act on a whole **Group**: Play Group launches every clip in the group at once (a scene); Stop Group halts them. The clip firing the marker is exempt, so it plays through.

Add rows, delete any row with its own delete button, and **drag rows up/down** to reorder. **Start** and **End** act as virtual markers. An **On play, cue next clip** checkbox arms the following clip automatically (default set in Settings → General).

Silent clips (behaviour timelines without audio)

A pad doesn't need audio to be useful. In a pad's settings, **+ Silence** creates a fixed-length **silent clip** — it asks for a length in seconds and gives the pad a silent timeline. Play it like any clip and use the **Behaviour** tab to place **Play Group / Stop Group** (or per-button) commands: e.g. a 4-second silence that stops group 1 at the start and plays group 2 at 4 seconds. Silent clips are rebuilt from their stored length when a layout loads, so they travel with your project without shipping audio.

9.4 Processing Tab



The Processing tab: Compressor, EQ, Gate/Expander, Delay and VST sections feeding the Output strip on the right

Everything that shapes the clip's *sound* once it's playing: a chain of reorderable effect sections feeding a fixed **Output** strip.

- **Effect sections** (drag to reorder; each has its own **Bypass** toggle so you can compare with/without it): **Compressor**, **EQ** (high-pass, four parametric bands, low-pass — each band colour-coded on a live curve display), **Gate/Expander**, **Delay**, **VST**, and **De-hisser**. The processing order is per-clip but starts from an app-wide default (Settings → General).
- **VST inserts**: eight serial insert slots (slot 1 → 2 → ... → 8 → the rest of the chain). **Scan for Plugins...** finds installed VST3 plugins (needed once; the list is cached). Click a slot to choose a plugin; the eye icon opens its editor; the ban icon toggles bypass (lit orange — plugin stays loaded but is skipped); **X** clears the slot. Plugin choices, internal settings and bypass states are saved with layouts.
- **Save Preset / Load Preset**: save the whole Processing chain — effect sections, VST inserts and the Output strip — by name, then load it onto any other clip later. Handy for reusing a favourite chain without rebuilding it each time. Presets are stored in the **Processing Presets** folder (Settings → Files — see [Section 13](#)).
- **Output strip** (fixed, on the right — not reorderable):
 - **Gain fader** (±24 dB) with a live level meter and a **Compressor GR** (gain reduction) readout.
 - **Mute** silences the clip outright regardless of the fader's position; **⏻** inverts polarity.
 - **Pan** — a bipolar knob: its fill starts at 12 o'clock (centre) and fills toward whichever side you pan to, rather than filling from one end like a level knob.
 - **Fade In / Fade Out** knobs: ramp the level up on start and down on stop. A fade-out always runs to completion once triggered — even if the clip (or its remaining loop) is shorter than the fade time, playback continues until the fade finishes rather than cutting off early.
 - **Loop** and **Mono** toggle buttons.

- **Routing:** the clip's output routing configuration, by name ([Section 12](#)).
-

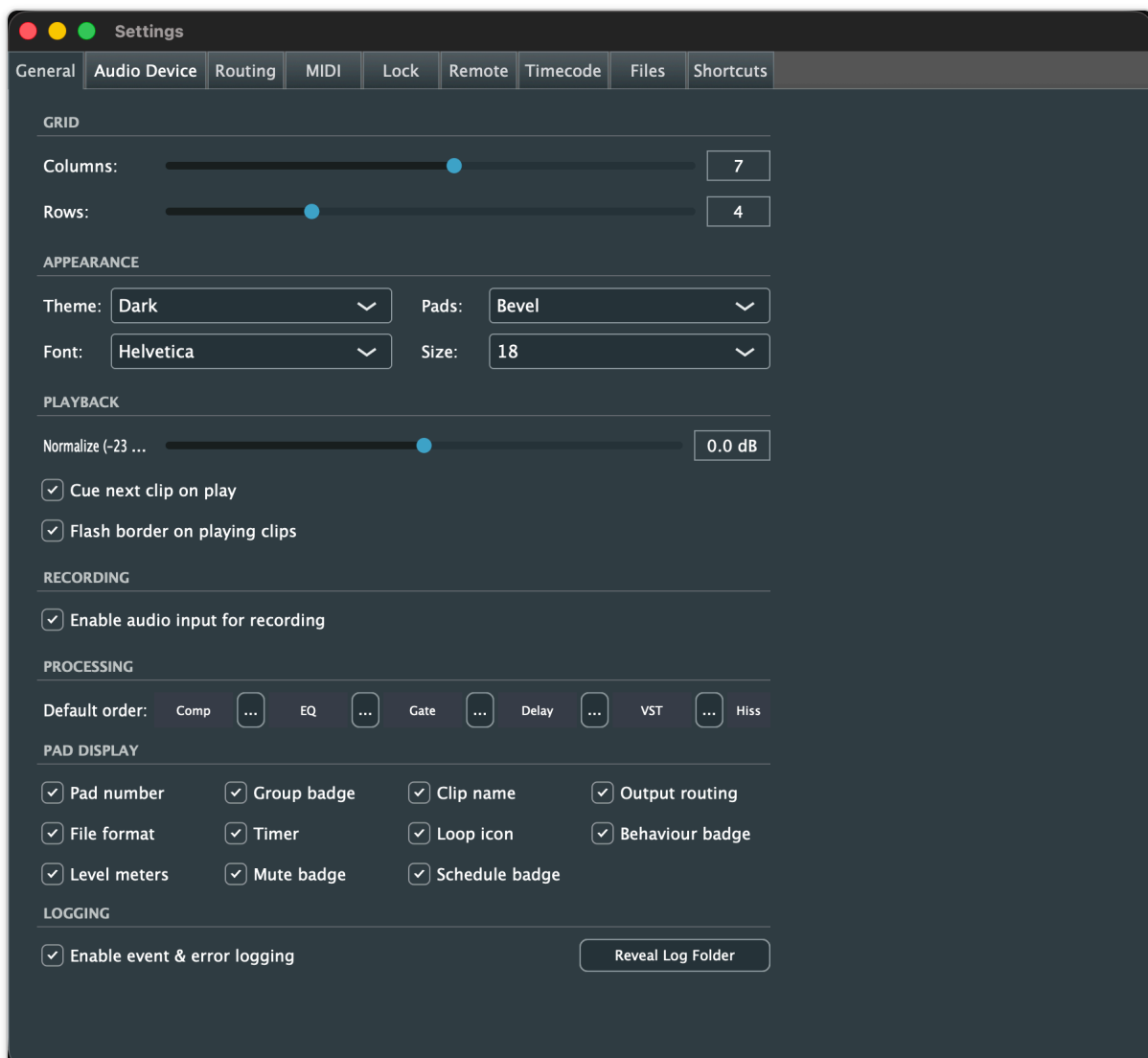
10. Undo & Redo

Destructive pad operations are undoable, up to **50 steps**: clearing a pad, file drops (a multi-file drop undoes as one step), moving/copying clips, loading a file over a clip, and deleting a page. Use **Cmd+Z** / **Cmd+Shift+Z** or the **Edit menu** (which names the action). There are no toolbar Undo/Redo buttons.

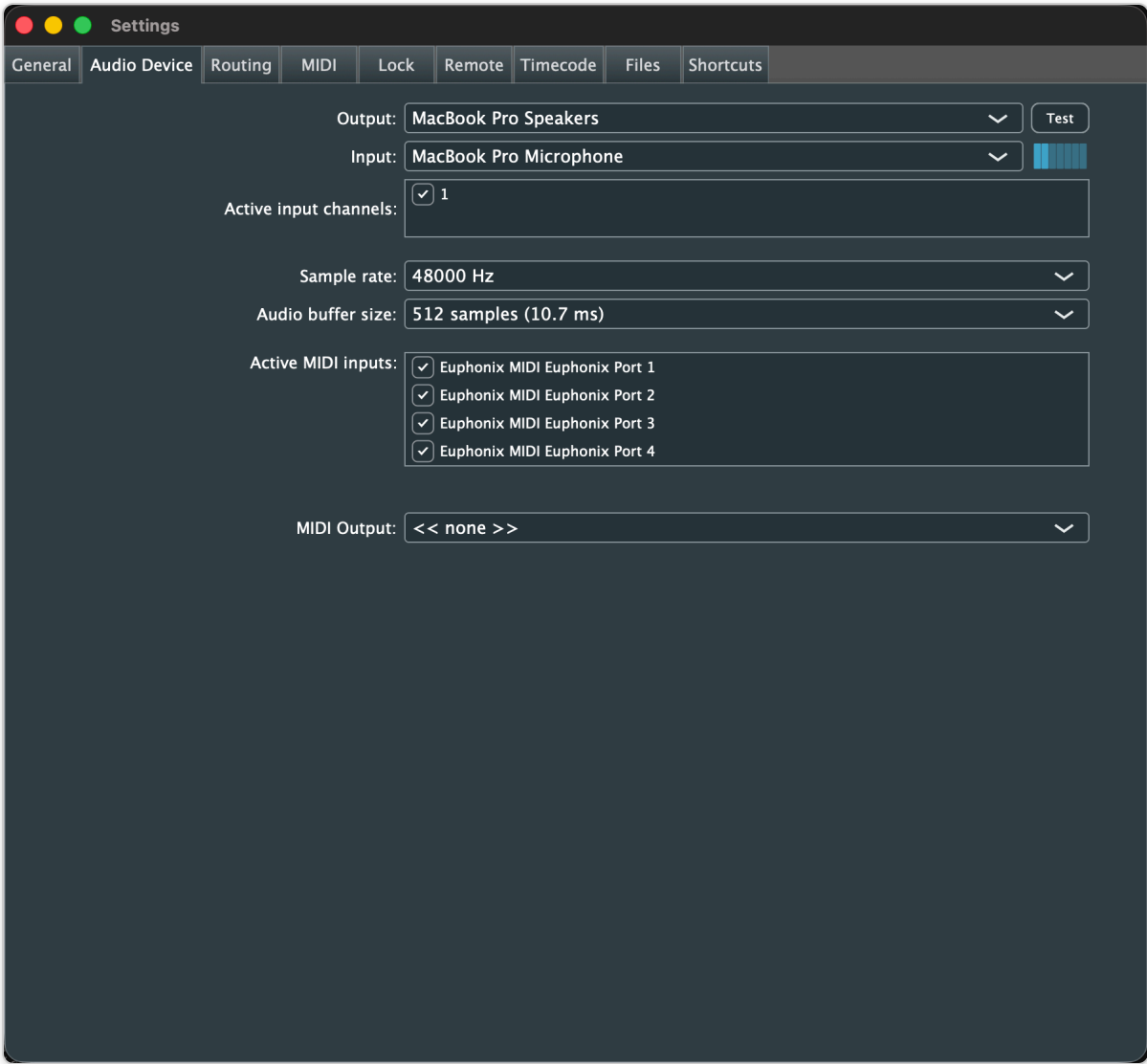
11. Global Settings

Click the cog-wheel to open Settings. The tabs are **General**, **Audio**, **Routing**, **MIDI**, **Lock**, **Remote**, **Files**, **Shortcuts**:

- **General:** default font/size for new pads, default normalize target, default “cue next clip” behaviour, flash-border option, a **Theme** selector (Dark / Grey / Midnight / Light, applied app-wide and saved), grid size and other defaults.
- **Audio:** input/output device configuration.
- **Routing:** output routing configurations ([Section 12](#)).
- **MIDI:** global MIDI action mapping ([Section 16](#)).
- **Lock:** what the screen lock blocks ([Section 17](#)).
- **Remote:** enable web control, set port and PIN, view the connect URL ([Section 18](#)).
- **Files:** where layouts and audio libraries live ([Section 13](#)).
- **Shortcuts:** keyboard shortcut editor ([Section 14](#)).

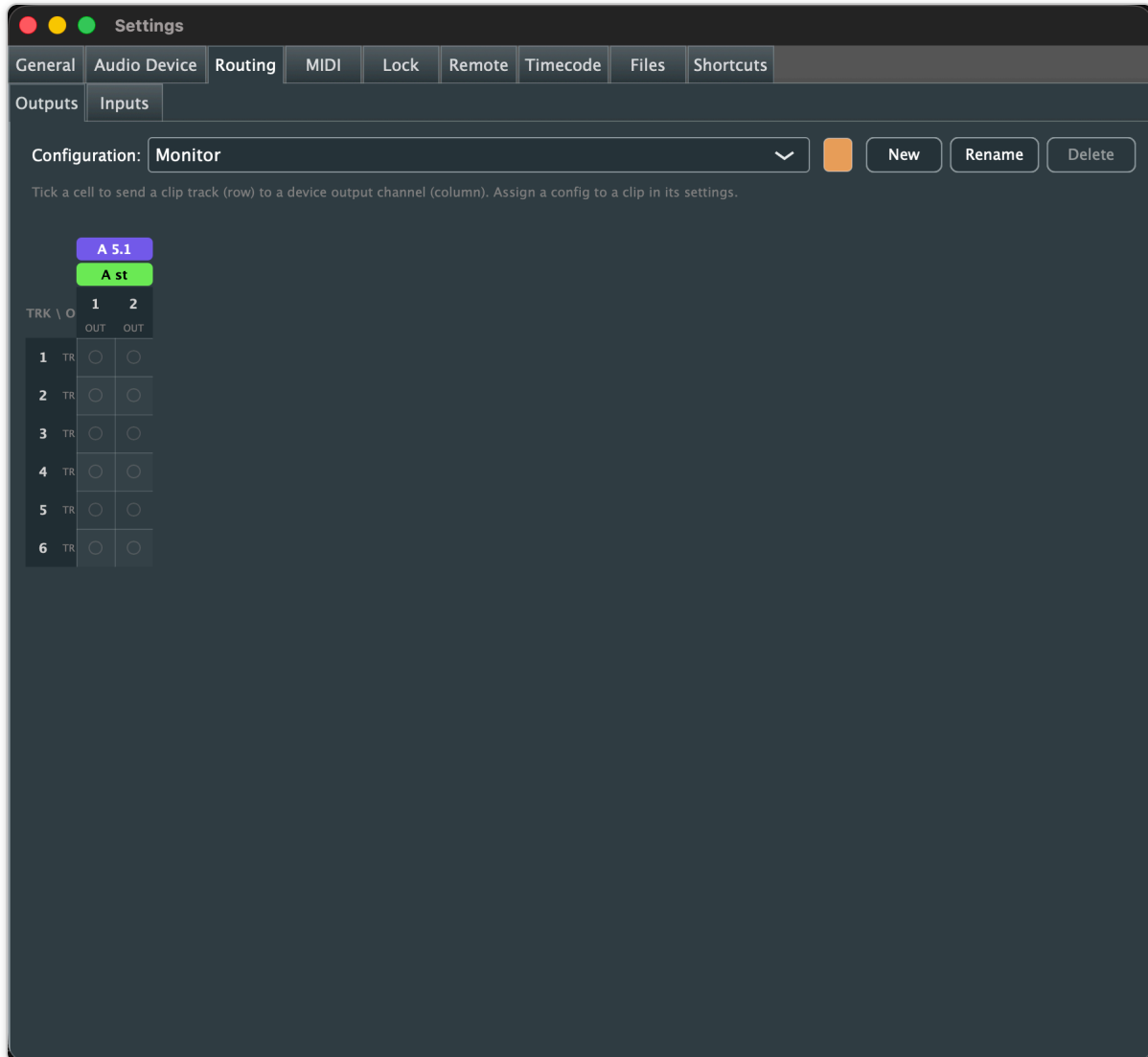


The General tab: grid size, appearance, playback, recording, default processing order and pad display options



The Audio tab: output/input device, active input channels, sample rate, buffer size and active MIDI inputs

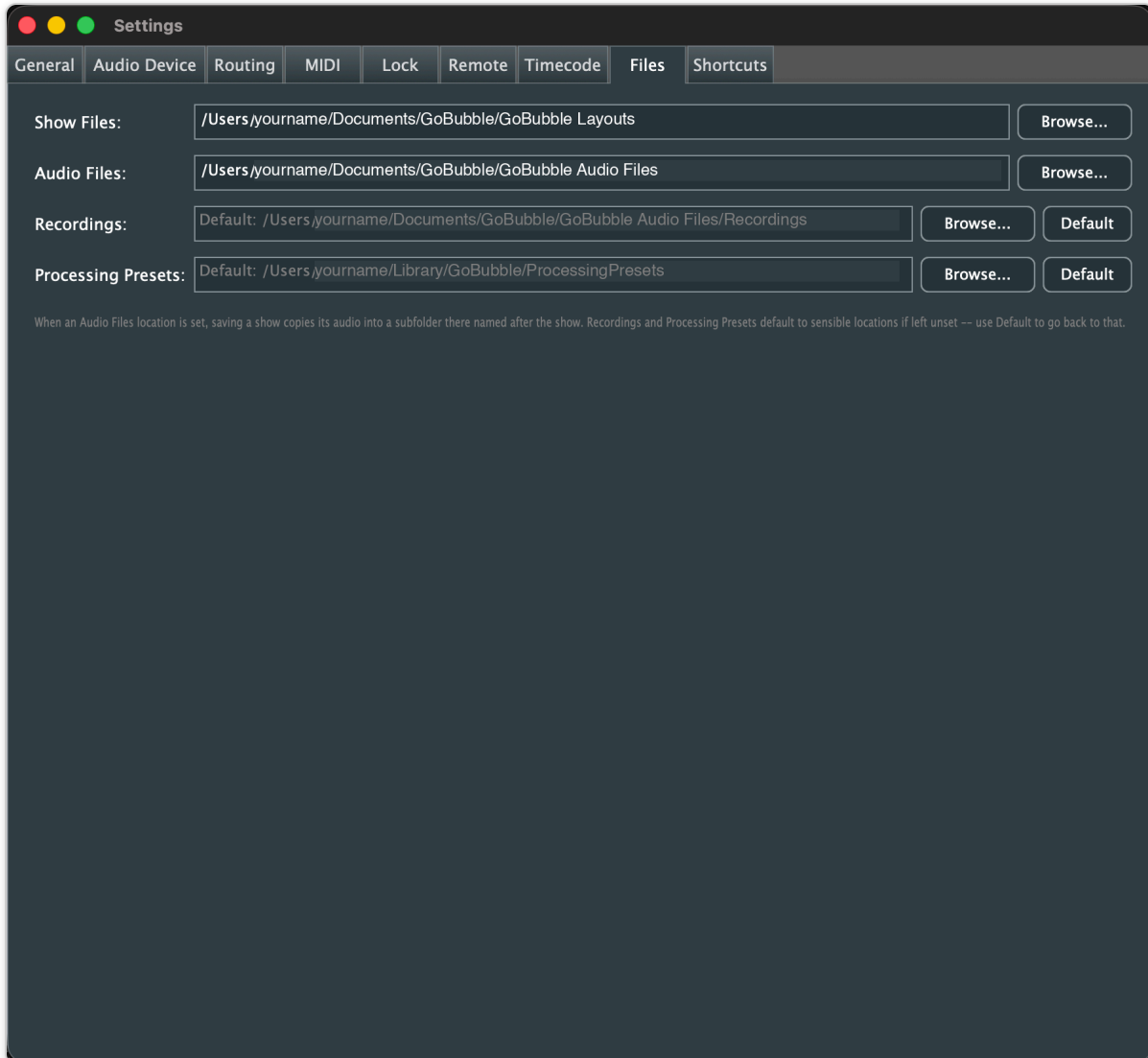
12. Output Routing



The Routing tab: a matrix patch bay mapping clip tracks (rows) to device output channels (columns), with named configuration banners above

In **Settings** → **Routing**, build named **routing configurations** on a matrix patch bay that maps a clip's **tracks** (up to 6) onto the device's **output channels** (up to 64, matching your audio device). Click a cell to patch a track to an output; name each configuration. All configurations are shown as coloured name banners above the output columns they use, with contrasting colours so overlapping ranges stay readable. A clip picks its routing by name on the [Processing tab](#). Routing configurations are saved app-wide and also travel inside layout and package files, so 5.1 and multichannel mappings survive sharing.

13. Files & the Managed Audio Library



The Files tab: Show Files, Audio Files, Recordings and Processing Presets locations, the latter two with Default buttons

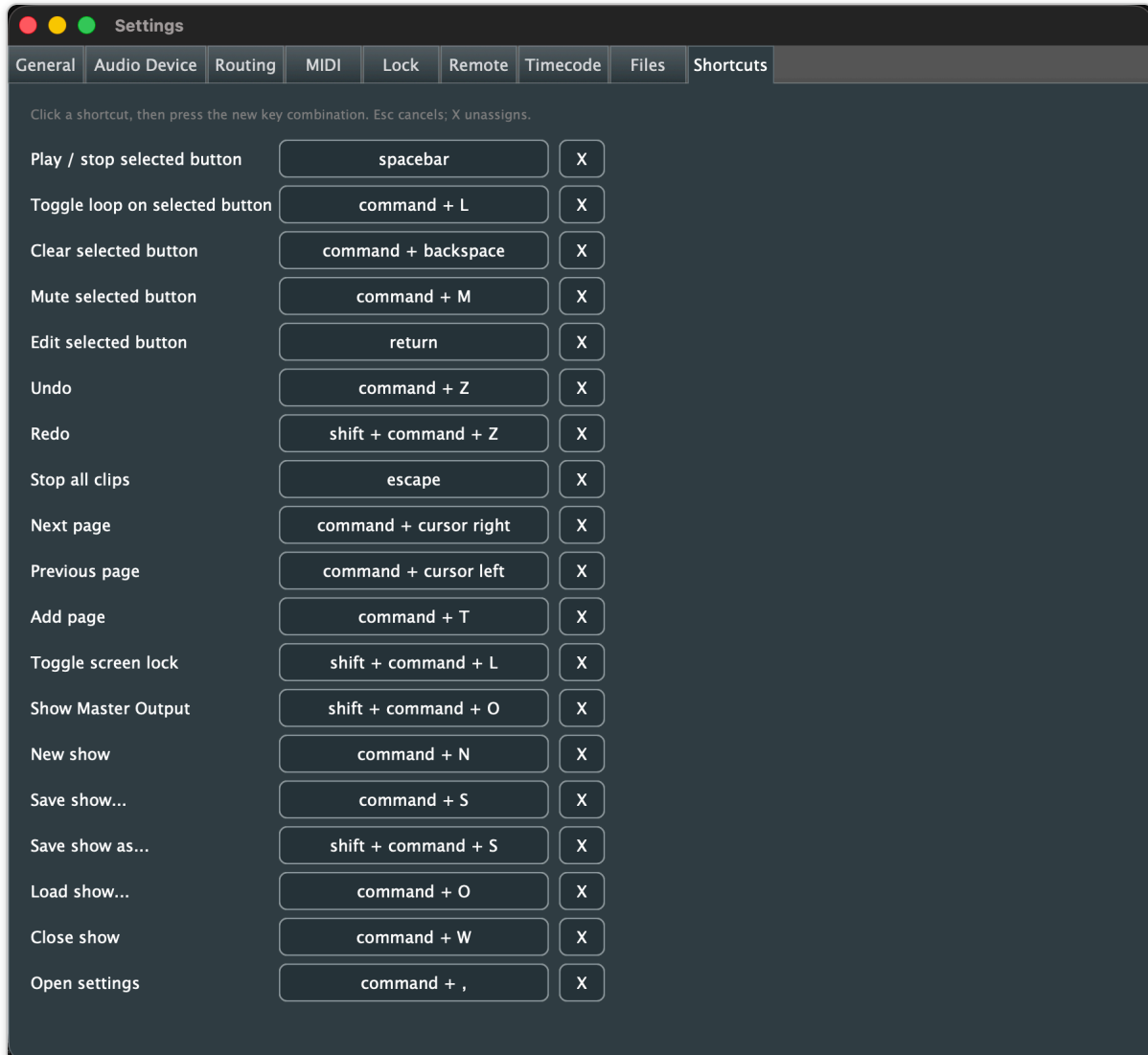
In **Settings** → **Files** you can set:

- **Show Files location:** the default folder for saving and opening layouts.
- **Audio Files location:** a managed audio library. When set, adding audio to a pad copies it here (see [copy-on-add](#)), and saving a layout files that layout's audio under a subfolder named after the layout — keeping projects self-contained.
- **Recordings location:** where takes made with a pad's record button ([Section 5](#)) are saved.
- **Processing Presets location:** where presets saved from the Processing tab's **Save Preset** ([Section 9.4](#)) are stored.

Show Files and Audio Files have no computed default — leaving either blank just means GoBubble asks you to choose a location every time you save/open. Recordings and Processing Presets always have

somewhere sensible to fall back to (a subfolder of your Audio Files location if set, otherwise a folder in your user Library or Documents), so those two rows show their current default as greyed placeholder text and offer a **Default** button to clear an override and go back to it. All four locations can also be set from the first-launch wizard's **Folders** page ([Section 2](#)) — the two panels stay in sync.

14. Keyboard Shortcuts Editor



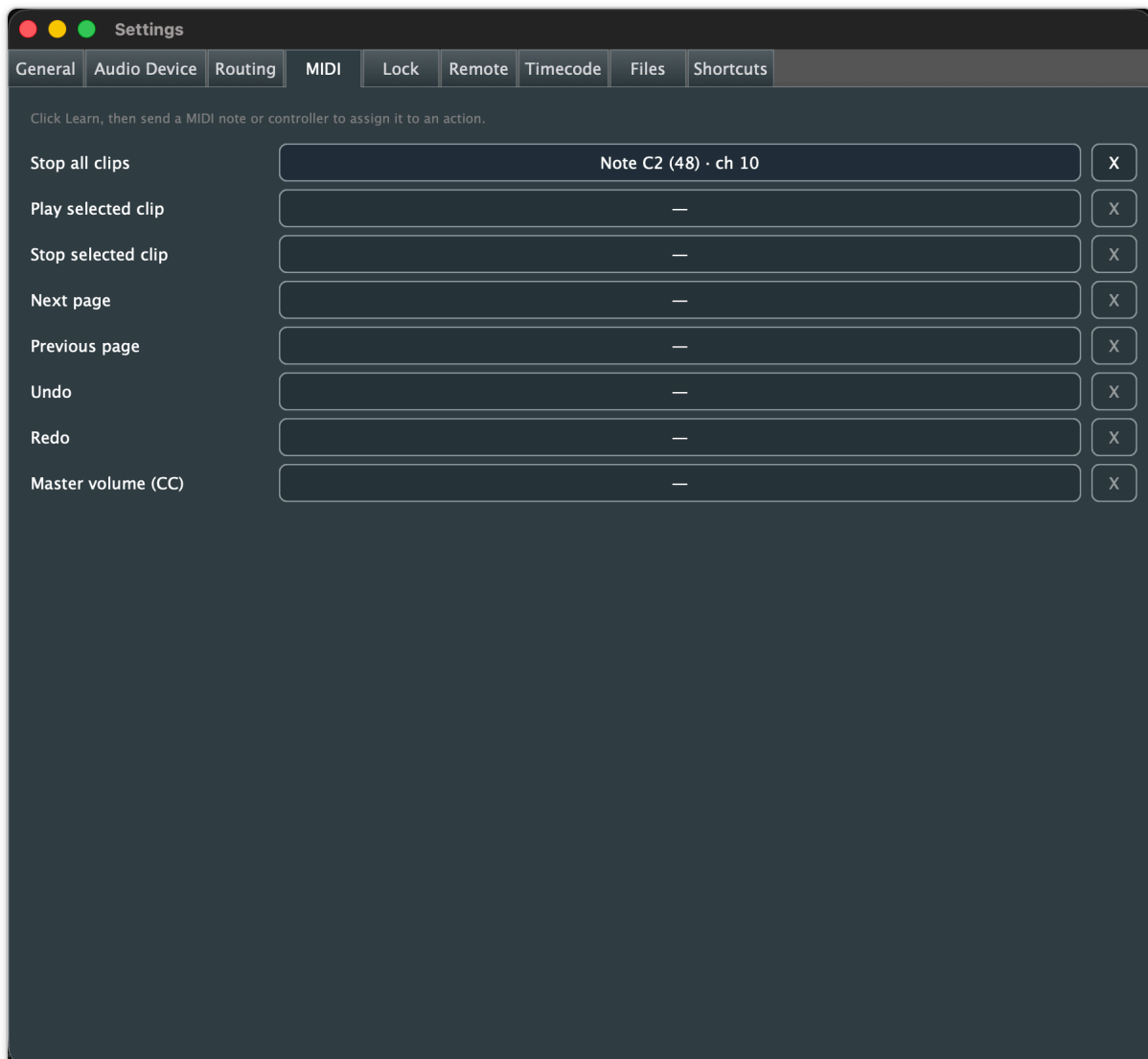
The Shortcuts tab: every app command listed with its current key combination and an X to unassign

In **Settings** → **Shortcuts**, every app command is listed with its current key. Click the shortcut, press the new key combination, and it's assigned (Esc cancels; **X** unassigns). Assigning a key another action was using unassigns that other action. Custom shortcuts persist across restarts.

15. Layouts & Packages

- **New / Open / Save / Save As / Close Layout:** in the **File** menu (Save/Open also on Cmd+S / Cmd+O). A layout stores everything: grid size, pages and names, all routing configurations, and per pad — the file, name, colours, text formatting (incl. bold), volume, loop, mono, routing, IN/OUT points, level envelope, group membership, MIDI note, markers, behaviours, silent-clip length, and the full VST chain with each plugin's settings and bypass state.
 - **Packages (.gob): Save Package** bundles the layout *and copies of all its audio* into a single portable `.gob` file. **Open Package** unpacks it — you choose where the layout and audio are saved — so a project can move intact to another machine.
 - **Recent Layouts:** the File menu lists recent layouts; missing entries are reported and removed.
 - **Startup:** GoBubble automatically reopens the last loaded layout (and runs the integrity scan on it — [Section 19](#)).
 - **Error handling:** failed saves/opens (missing folder, unwritable location, corrupt or non-GoBubble file, failed audio copy) report a clear message rather than failing silently.
-

16. MIDI Control



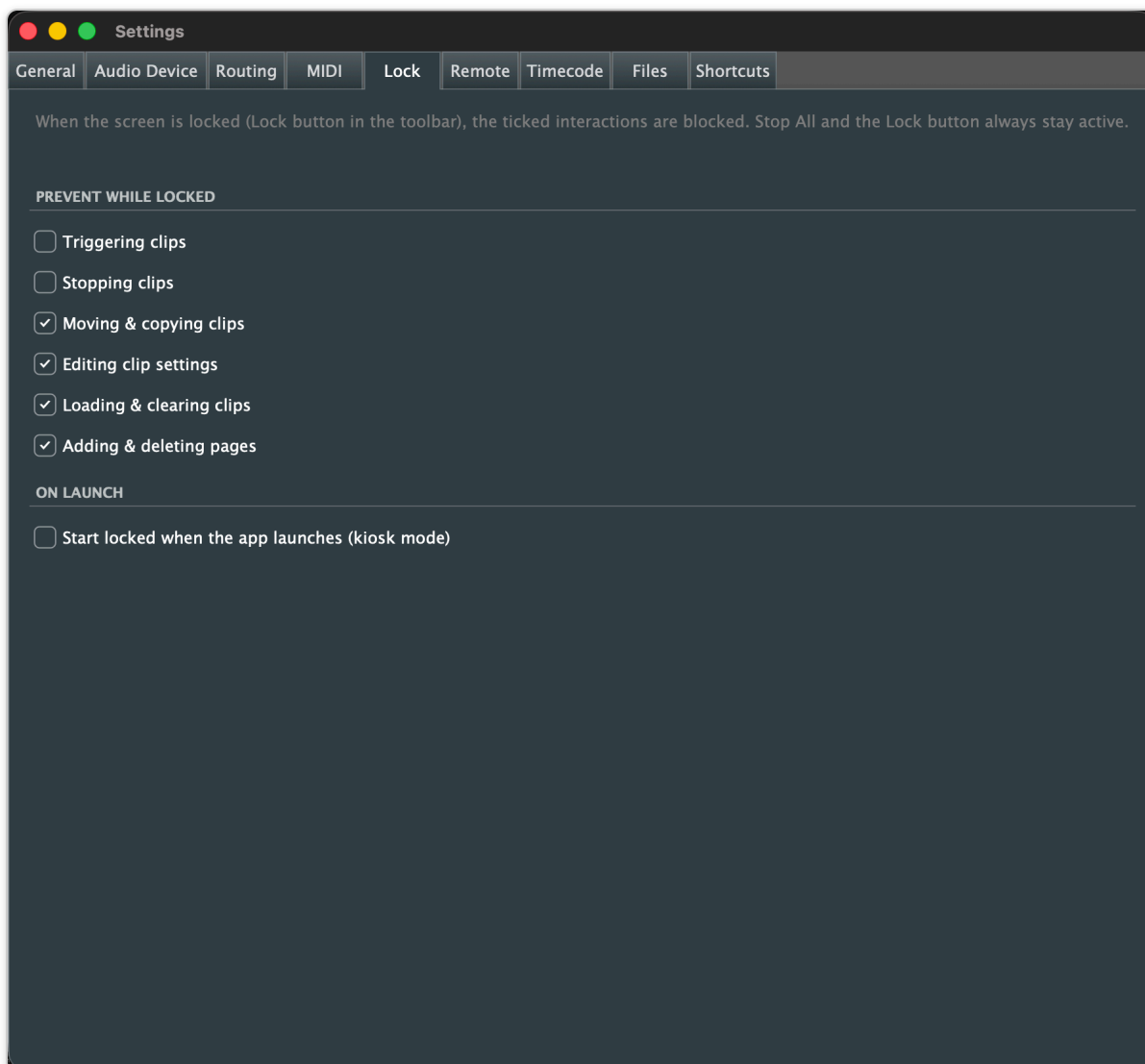
The MIDI tab: global note/CC mappings for actions like Stop all clips, Undo/Redo and Master volume

- GoBubble listens to all enabled MIDI input devices.
- **Per-clip Trigger Note:** assign one in the clip settings header (next to the group selector); receiving that note plays the clip. If the same note is also bound to a global action (below), a warning explains that the global action will fire instead — global mappings always take priority over per-clip triggers.
- The toolbar's **MIDI monitor** shows the most recent note or CC received — useful for finding out what a control sends.
- **Global MIDI mapping (Settings → MIDI):** map a MIDI note or controller to an app action. Click **Learn**, send the note or CC, and it's assigned; **X** clears it. Available actions:
 - **Stop all clips**
 - **Play selected clip / Stop selected clip**
 - **Next page / Previous page**

- **Undo / Redo**
- **Master volume (CC)** — a *continuous* control: bind it to a fader/knob and its CC value drives the overall output level in real time.

Mapped messages take priority over per-clip triggers, and a message assigned to one action is automatically removed from any other. Continuous controls (like Master volume) can only be learned from a CC, not a note. Mappings are saved app-wide.

17. Screen Lock (Kiosk Mode)



The Lock tab: checklist of interactions blocked while locked, plus the kiosk-mode start-locked option

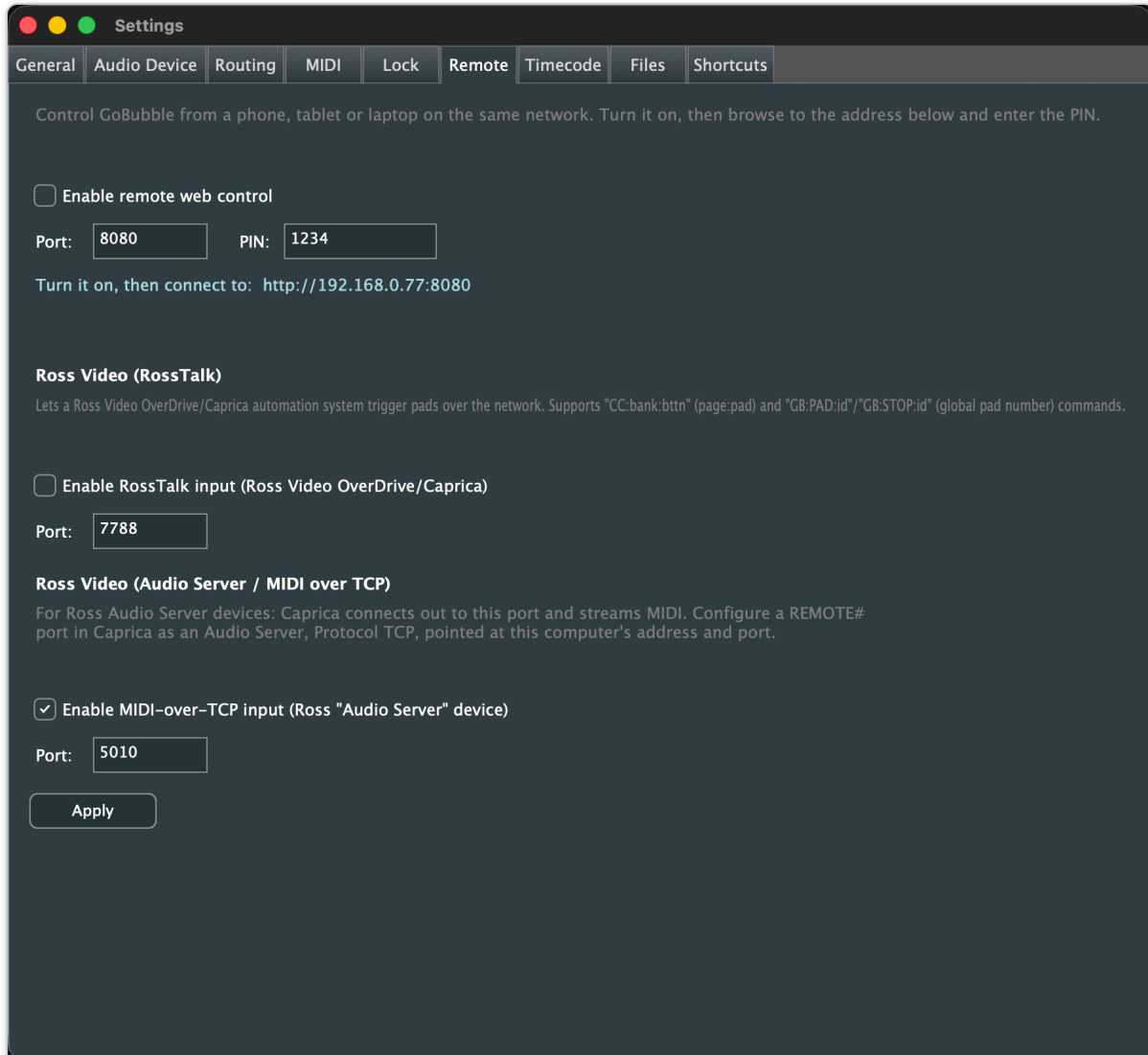
The toolbar **Lock** button freezes the interface so a live layout can't be disturbed by accident. When engaged, the button turns orange and reads **Locked**; click it again to unlock.

Choose exactly what the lock blocks in **Settings** → **Lock** — a checklist:

- **Triggering clips** (off by default, so you can still perform)
- **Stopping clips**
- **Moving & copying clips**
- **Editing clip settings**
- **Loading & clearing clips**
- **Adding & deleting pages**

The **Lock** button and **Stop All** always stay active, so you can never lock yourself out or lose the panic button. Your choices are saved app-wide; the lock itself starts *off* each launch.

18. Remote Web Control



The Remote tab: enable web control, port and PIN fields, and the connect URL

Control GoBubble from a phone, tablet or laptop on the **same local network** — no app to install, just a web browser. Remote access is **off by default** for security.

To turn it on:

1. Open **Settings** → **Remote**.
2. Tick **Enable remote access**.
3. Set a **Port** (default **8080**; any free port above 1024 works, e.g. 1212) and a **PIN** (default **1234** — change it).
4. Click **Apply**. The panel shows a **connect URL** like `http://192.168.1.42:8080`.
5. On the other device's browser, open that URL and enter the PIN.

What the remote page gives you: a live, responsive grid mirroring your pads across pages. From it you can **trigger** and **stop** clips, **Stop All**, and **switch pages**. The page updates in real time (via Server-Sent Events) so pad names, colours and playing state stay in sync.

Monitoring & security:

- The toolbar **globe** icon turns **green** when one or more clients are connected. Click it to see the connected clients and **disconnect** (kill) any of them.
- Every request needs the PIN; a wrong PIN is rejected (HTTP 401).
- Traffic is plain HTTP on your LAN — keep it to trusted networks. (HTTPS/TLS and bad-PIN lockout are noted as future enhancements.)

To turn it off: untick **Enable remote access** in Settings → Remote and click Apply, or just quit GoBubble.

19. Startup Integrity Scan

Whenever a layout opens (at launch, from the File menu, from Recent, or from a package), GoBubble scans it for problems:

- **Missing audio files** — for each, you're offered to **Locate** the file, **Recover**, **Skip**, or other available options.
- **Files not yet copied** into your managed audio library (when an Audio Files location is set) are flagged so your project can be made self-contained.
- **Plugin / configuration issues** are reported.

If everything checks out, you get a short **thumbs-up** confirmation that **auto-closes after 3 seconds**, so a clean startup doesn't slow you down.

20. How-To Recipes

Short, task-focused walkthroughs that combine features.

Build a one-song page with exclusive backing tracks 1. Add each backing track to its own pad (drag-and-drop). 2. Shift-click each pad → set the same **Group** (e.g. G1). 3. Now triggering any one stops the others — only one plays at a time.

Make a trimmed copy without touching the original 1. Shift-click the pad → **Edit**. 2. Drag the **IN/OUT** lines (or type exact values) around the part you want. 3. Click **Crop** → confirm. A `_copy` file is created and loaded; the original is untouched.

Fire a lighting/scene cue with no audio 1. Shift-click an empty pad → + **Silence** → enter a length (e.g. 4 s). 2. Go to **Behaviour**, add a marker at 0 s with **Stop Group** G1, and one at 4 s with **Play Group** G2. 3. Playing this silent pad now sequences your groups on a timeline.

Record a quick take and place it 1. Set the input in **Settings** → **Audio**. 2. Click the red **R** on an empty pad; click again to stop. 3. Later, open the **Recordings library** (list icon) to Preview, Load onto another pad, or Delete.

Control the show from your phone 1. **Settings** → **Remote** → Enable, set a PIN, Apply. 2. On your phone's browser open the shown URL and enter the PIN. 3. Watch the toolbar **globe** turn green; tap pads to trigger. Disconnect a device by clicking the globe on the desktop.

Drive master volume from a MIDI fader 1. **Settings** → **MIDI** → find **Master volume (CC)** → **Learn**. 2. Move the fader/knob you want to use; it binds to that CC. 3. The fader now rides the overall output level live.

Lock the rig for a performer 1. **Settings** → **Lock** → tick everything except *Triggering clips*. 2. Click the toolbar **padlock**. Performers can trigger clips but can't move, edit, load or delete anything. Stop All still works.

Move a project to another machine 1. **File** → **Save Package** → creates a `.gob` with the layout + all audio. 2. Copy the `.gob` across; **File** → **Open Package** on the other machine and choose where to unpack.

21. Troubleshooting

- **App bounces in the Dock but won't open:** this was caused by a duplicate audio-format registration tripping a debug assertion and is fixed in current builds. If you build from source, make sure you're on the latest `Source/`.
 - **A pad shows the red R button even though it "has" a clip:** the clip's file is missing (e.g. moved or deleted on disk). Use the startup scan's **Locate** option, or reload the file. GoBubble also tries extension fallbacks when resolving a moved file.
 - **Clicking the waveform used to blip the audio:** clicking to place the playhead no longer starts playback when the clip is stopped.
 - **Remote page won't load:** confirm remote is **enabled** and **Applied**, both devices are on the same network, the **port** isn't blocked by a firewall, and you're using the exact URL and PIN shown in Settings → Remote.
 - **Plugin scan is slow:** scanning runs on the main thread for compatibility; large collections take a while. If interrupted, re-run — previously found plugins are remembered.
 - **Audio glitch while editing:** editing a level envelope or swapping a plugin never blocks the audio engine; a brief change while doing so is by design (the engine skips one block rather than stall).
 - **Compile errors after editing source:** open the project in Projucer, re-save the Xcode exporter, then clean and rebuild in Xcode.
-

Appendix A — Building from Source (Developers)

GoBubble is a JUCE 7 application built with **Projucer + Xcode** on macOS (not Make or CMake).

1. Open `GoBubble/GoBubble.jucer` in **Projucer** and let it locate your JUCE modules.
2. Save the project (**File** → **Save Project**) to regenerate the Xcode exporter under `GoBubble/Builds/MacOSX/`.
3. Open `GoBubble/Builds/MacOSX/GoBubble.xcodeproj` in **Xcode** and build the **GoBubble - App** scheme.

Command-line build (from `GoBubble/Builds/MacOSX`):

```
DEVELOPER_DIR=/Applications/Xcode.app/Contents/Developer \  
xcodebuild -project GoBubble.xcodeproj -configuration Debug -scheme "GoBubble - App"  
build
```

When adding a new `.cpp` / `.h` source file, register it in **both** `GoBubble.jucer` (via Projucer) and the generated Xcode project, then re-save from Projucer so the two stay in sync.

Versioning

The version shown in **GoBubble** → **About GoBubble** and baked into the app bundle comes from the `version` attribute on the `GoBubble.jucer` file's root `JUCERPROJECT` element. To cut a new version: edit that attribute, re-save from Projucer (regenerates the Xcode/Linux/VS exporters with the new version), then rebuild.

Tests

GoBubble ships a JUCE `UnitTest` harness. Run it with:

```
GoBubble.app/Contents/MacOS/GoBubble --run-tests
```

(313 assertions at time of writing — MIDI mapping incl. continuous control, silence flag, remembered zoom, marker/behaviour-command indexing, Processing-parameter envelopes, Monitor Pre/Post preview engine, preview-range/edit-loop playback, Processing-tab presets, destructive normalize, and more.)

Cross-platform builds (Linux / Windows)

The project ships **Linux Makefile** and **Visual Studio 2022** exporters alongside the Xcode one:

1. Open `GoBubble.jucer` in Projucer, point it at your JUCE modules, and save — this generates `Builds/LinuxMakefile/` or `Builds/VisualStudio2022/`.

2. **Linux:** `cd Builds/LinuxMakefile && make CONFIG=Release` (needs the usual JUCE dev packages: ALSA, X11, FreeType, WebKitGTK, etc.).
3. **Windows:** open the generated `.sln` in Visual Studio and build (the Psapi library, used for the memory meter, is already linked).

Platform-specific code is confined to guarded blocks: the memory meter has macOS (`mach`), Windows (`GetProcessMemoryInfo`) and Linux (`/proc/self/statm`) implementations; the native menu bar branches on `JUCE_MAC` ; extra OS audio codecs (AAC/M4A/ALAC) register only on Apple platforms. Everything else is portable JUCE. *Note: the Linux/Windows targets are wired but not yet built/verified on those platforms.*

Architecture at a glance

- **MainComponent** — owns the pages, pad grid, audio device manager and mixing, layout/package persistence (XML), routing configurations, undo/redo stacks, top-bar meters, the embedded remote server, the startup scan, and marker-command dispatch (`fireMarkersInRegion` / `handleClipPlayingChanged` , driven by `ClipAudioSource` 's audio-thread region callbacks).
- **ClipComponent** — one pad: rendering, mouse/drag handling, per-channel meters, text formatting, and the per-pad record button.
- **ClipAudioSource** — one clip's audio: transport, IN/OUT range, mono handling, the reorderable effect chain (Compressor/EQ/Gate/Delay/VST/De-hisser) and VST insert slots, fade in/out, per-channel metering, group membership, routing, silence length, destructive editing (Crop/Declip/Cut/Copy/Insert/Overwrite/Mix/Delete/Save Selection, with a file-based undo stack), and marker/behaviour-command storage. Marker crossings are detected sample-accurately on the audio thread (per block) and queued through a lock-free FIFO for the message thread to dispatch — not polled — so a command can't be missed or double-fired even for markers within a few milliseconds of a clip's end.
- **RemoteServer** — the embedded HTTP server (control page, PIN auth, Server-Sent Events for live state, POST commands, connection tracking/kill).
- **WaveformDisplay** — waveform, time scale, markers, playhead scrubbing. Named markers are draggable on the Behaviour tab but display-only on the Edit tab, where they're shown for context but managed elsewhere.
- **MidiMapping** — global note/CC → action map, including continuous (master-volume) controls.
- **ConfigScanDialog** — the startup integrity scan and its recovery options.
- **FirstRunWizardContent** / **FirstRunWizardWindow** — the setup wizard shown on every launch until dismissed (welcome, EULA, license activation, audio device, folders, tips); the page sequence adapts once activation has ever succeeded, skipping the EULA/license pages on repeat showings.
- **LicenseClient** — talks to the license server (a Supabase project, schema in `Server/supabase_schema.sql`) to activate a key and periodically check in; `MainComponent` caches the last-known-good result so the app keeps working through short outages (see [Section 2](#)).
- **Settings panels** — `GeneralSettingsPanel` , `AudioEditPanel` (Edit tab), `BehaviourPanel` , `ProcessingPanel` , `SettingsComponent` , and the Routing/MIDI/Lock/Remote/Files/Shortcuts panels.

Real-time audio safety

The audio callback (`ClipAudioSource::getNextAudioBlock`) must never block. Shared state edited from the GUI (level envelope, VST chain) is guarded by `CriticalSection` locks the audio thread takes with `tryEnter` : if the GUI holds the lock, the audio thread skips that block's envelope ramp or plugin processing rather than stall. Cross-thread scalar state uses `std::atomic` .

Persistence

Layouts are XML. App-wide preferences live in `~/Library/GoBubble/GeneralSettings.xml` , including the evaluation license state (key, cached expiry/validity, last successful server check-in), whether first-run setup has ever completed, and whether the wizard has been dismissed for good. Packages (`.gob`) are zip archives containing `layout.xml` plus an `Audio/` folder of de-duplicated copies. Routing masks are stored as 64-bit strings so wide multichannel setups round-trip correctly. Remote settings (`remoteEnabled` , `remotePort` , `remotePin`) persist in `GeneralSettings.xml` , with remote **off** by default.