



Mynah user guide

Mynah is a **lighting-desk command line for Analog Way LivePremier (Aquilon) switchers.**

```
Recall Screen 1 Memory 5
Store Master 12 If Screen 1 + 3 Category Source + Position
Recall Screen 1 Thru 4 Memory 7 Program
Take
```

Type the command, press Enter. **The grammar rules are grandMA3's** — verb first, any unambiguous abbreviation, `Thru` / `+` / `-` ranges, an `If` clause that masks a store. **The vocabulary is the switcher's own:** Screen, Aux, Layer, Master, Memory, Preview, Program. Nothing here calls anything a fixture or a cue.

A LivePremier has a lot of memories — 1,000 screen, 500 master, 50 layer — and the vendor UI reaches them by pointing at things. Anyone who has operated a lighting desk knows the alternative: **say what you want, in one line, without your eyes leaving the screen.**

This guide is the operator's overview. **New to it? [docs/GUIDE.md](#) is the programming guide** — every command from one word upwards, with worked examples you can paste into the simulator. The full grammar is in [SYNTAX.md](#), and the device paths behind it, with the firmware they were verified on, are in [PATHS.md](#).

Built with AI assistance, directed and reviewed by a human author.

Read this first: the hosted copy cannot drive a switcher

A LivePremier serves its web UI over plain http on port 80, with 443 closed. There is no secure WebSocket to connect to, and browsers block an insecure socket from a secure page as mixed content.

Where Mynah runs	Can it drive a switcher?
<code>localhost</code> from a checkout	yes
self-hosted over http on the show network	yes
the desktop app	yes
the hosted https:// copy	no — blocked before the socket leaves the browser

This is not a limitation anyone can code around. **The hosted copy runs a simulator instead**, and says so in a banner you cannot miss.

The desktop app is the simple answer — macOS, Windows and Linux, about 5 MB, talking to the switcher directly with no browser sandbox in the way.

The two defaults that decide what reaches air

An unqualified `Recall` goes to **Preview**. An unqualified `Store` takes from **Program** — because that is the look you just made live, and the device's own default agrees.

So getting to air always costs an explicit word: `Program` , or `Take` . A recall never reaches program by accident.

An empty memory is reported as empty

The device **accepts a recall of a slot that holds nothing, and then does nothing** — no error, no response at all.

Mynah watches for that silence and says `empty memory` rather than `done` , **because a green tick on a command that changed nothing is worse than no feedback**. If you see it, the slot is empty; the command was not wrong.

What it drives

Memories — recall, store, delete, label, take, and scope.

Bank	Slots	Scope
Screen	1-1000	Screen S1-S24, Aux A1-A96, × Preview/Program
Master	1-500	record mask: screens, auxes, layers, categories
Layer	1-50	Screen × Preview/Program × layer (Native, 1-128)
Multiviewer	1-50	output

Live layer control — source, size, position and opacity, on any layer of any screen:

```
Set Screen 3 Layer 2 Source 1 Size 50% Position 33% 50%
```

Percentages resolve against the screen's real canvas, and **position is the layer's centre**, because that is the device's own anchor.

How it connects

The browser cannot open a raw TCP socket, so the documented control protocol on port 10606 is out of reach from a page. **Mynah uses the Web RCS WebSocket instead** — the same socket the vendor's own UI speaks.

That is better than a bridge in three ways: no install, **no five-client cap**, and it pushes state without a subscription list to get wrong.

It also means **Mynah sees what a human does in the vendor UI**, and follows the Web RCS's own screen selection — so selecting a screen in either place selects it in both.

The simulator is a rehearsal, not a demo

The hosted copy's simulator is a model of a LivePremier built into the page: four screens with layers and three preset buffers each, the four memory banks, and a wireframe of what every screen is putting out.

It reproduces the device's awkward behaviour on purpose, because those are the things worth meeting for the first time somewhere other than a show:

- a recall of an empty memory is answered with **silence**, and reported as an empty memory rather than as success;
- **preview and program are names for whichever buffer is pending or live, the mapping differs per screen**, and a Take swaps it.

That second one is the one that catches people on real hardware.

If something is wrong

Symptom	Cause
The hosted copy will not connect to my switcher	It cannot, and no setting will change that. Use the desktop app or self-host over http.
A command reported <code>empty memory</code>	The slot is empty. The device answers silence; Mynah refuses to call that success.
Nothing reached air	An unqualified Recall goes to preview. Add <code>Program</code> , or send a <code>Take</code> .
A layer moved to the wrong place	Position is the layer's centre , and percentages resolve against the screen's canvas.
Preview and program seem swapped on one screen	They are per-screen names for buffers, and a Take swaps them.

Mynah · v1.3.3 · guide updated 2026-08-22 · stoatworks-labs.com/software/mynah/

Latest version of this guide: stoatworks-labs.com/software/mynah/guide/ · Source and issues:

<https://github.com/stoatworks-labs/mynah>