

# Frequently Asked Questions

Straight answers on dMix 128 networking, control software, and the IO44 / DX08B / DX16B splitters — compiled from Violet Audio's own published documentation.

8

QUESTIONS ANSWERED

3

TOPIC AREAS

2026

LAST REVIEWED

## REGARDING DMIX

### Networking, clocking & console behaviour

How the two network port types divide work, setting up an AES67 stream, clocking options, solo/monitoring routing, and scene automation.

- Q1** Control Port vs. dNet — how do they work together?
- Q2** The Stream menu — configuring a multicast stream
- Q3** Clocking — internal or external (PTP)?
- Q4** Solo / Monitoring — Solo 1 vs 2, routing, Mix vs Last
- Q5** Scene automation & Selective Loading

## REGARDING SOFTWARE

### Beta software expanding the platform

Two new beta products that turn dMix 128 into a broadcast console and a spatial-audio production platform, on top of the same hardware.

- Q6** What is RadioX12, and how does it relate to dMix 128?
- Q8** What is Spatia 2D, and how does it relate to dMix 128?

## REGARDING IO44 / DX08B / DX16B

### Passive splitters & the Ethercon question

Channel counts, split counts, and — the one that trips people up — which of the three units actually carries signal both ways over Ethercon.

- Q7** Signal paths on the splitters, and is Ethercon bidirectional?

## ABOUT THIS FAQ

Compiled from [violetaudio.com/documentation/](https://violetaudio.com/documentation/) — the dMix 128 Reference Manual, Networking & Recovery Guide, dMix 128 FAQ, Integration Documents, and the IO44 / DX08B / DX16B product pages. Always confirm against the latest published version, since some source pages were last revised as recently as June–August 2026.

## WHAT'S COVERED

- Q1 — Control Port vs. dNet Audio Network ports
- Q2 — The Stream menu / multicast AES67 setup
- Q3 — Internal DSP clock vs. PTPv2 follower/master
- Q4 — Solo 1/2, Solo Routing, Mix vs Last, PFL/AFL
- Q5 — Scene Isolation (selective scene recall)
- Q6 — RadioX12 — virtual broadcast console for dMix 128 (beta)
- Q7 — IO44/DX08B/DX16B signal paths, Ethercon direction
- Q8 — Spatia 2D — object-based spatial audio for dMix 128 (beta)

# Control Port vs. dNet — How Are They Meant to Work Together?

The dMix 128 has two different kinds of network port, designed to be used at the same time, for different jobs.

## ● dNet Network CONTROL Port (1×)

Carries **control traffic only** — settings, routing, gain, metering, and remote-control protocols such as OSC, REST API, WebSocket and MIDI. No audio travels over this port.

## ● dNet Audio Network Ports (2×)

Carry **AoIP audio** (AES67 streams, up to 320 network channels) — and, unlike the Control port, can carry control traffic **at the same time** as audio. Patch these into your AES67/Dante/Q-SYS/Ravenna network, and use them to link multiple dMix 128 units or a redundant ring-topology switch.

✓ **So yes — the instinct is right:** use the Control port for management/OSC-style control of the mixer, and the dNet ports for AoIP audio — they'll happily carry control alongside that audio too, so a controller connected on the audio network isn't blocked.

# The Stream Menu — What's It For, and How Do I Configure a Multicast Stream?

Lives under the network/AES67 device pages — this is how the dMix 128 sends and receives AES67 multicast audio.

## TYPICAL USES

- Link multiple dMix 128 units together
- Exchange audio with Dante gear (once switched into AES67 mode in Dante Controller), Q-SYS, Ravenna, or other AES67/ST2110-30 equipment
- Feed a recording/broadcast computer beyond what the CoreAudio/ASIO driver connection covers
- Send a stereo or multichannel mix to a broadcast truck, feed several AES67 recorders/consoles from one set of buses without individual point-to-point cabling, or cascade dMix units

## HOW TO CONFIGURE IT

### STEP 1

#### Confirm the switch

The dMix 128's AES67 ports must sit on a switch that passes multicast/IGMP/PTP cleanly — managed gigabit, IGMP snooping + querier enabled, Energy-Efficient Ethernet/802.3az **disabled**.

### STEP 2

#### Check the Multicast Base

Default is `239.200.0.0`, with the "AAA" octet (200) configurable. Leave it unless you have a specific reason — it must stay inside the 239.x/16 range AES67 receivers expect.

### STEP 3

#### Set rate & channel count

Choose the sample rate (48 kHz or 96 kHz) and channel count for the stream itself.

### STEP 4

#### Set the Clock

**Auto** — PTP negotiates a master automatically — or **Forced**, if you want to nominate a specific master.

### STEP 5

#### Match the far end

Confirm its multicast prefix matches (239.x) and subscribe via SDP or SAP. A mismatched prefix is the most common reason a subscription "connects" but no audio passes.

**!** Presets/scenes do NOT store AES67 stream flows — you'll need to recreate them after a preset/scene recall.

# Can dMix Only Clock Internally, or Can I Clock It to Other Devices?

Both are possible — and here's how to set either one up.

## Internally

The dMix 128's native DSP runs at **96 kHz**, and it has built-in sample-rate converters (e.g. on MADI) so incoming/outgoing signals always match your selected rate even if the source differs.

## Externally, over the network

Clocking is handled via **IEEE 1588 PTPv2** (AES67/SMPTE-2059-2 profile). The dMix 128 can act as either a PTP follower (standard/default behaviour) or PTP master (when configured to take that role).

It does **NOT** support PTP Boundary Clock or Transparent Clock modes.

## SETUP — NETWORK / AES67 CONFIGURATION PAGE

### Auto

The network's Best Master Clock Algorithm (BMCA) elects a master automatically (default Priority 1 = 128).

### Forced

You nominate a specific master — set that device's Priority 1 to 1 to force it into the master role.



Make sure your switches are **PTP-aware** and that **EEE (Energy-Efficient Ethernet) is disabled** on every audio port — it interferes with PTP timing.

# Solo / Monitoring — Solo 1 vs Solo 2, Solo Routing, and Mix vs Last

Three separate settings that often get asked about together — each covers a different part of how solo/cue actually behaves.

---

## SOLO 1 / SOLO 2

### Two independent cue buses

Each has its own full set of settings (mode, routing, type, delay, output level). Two people can solo independently at once — e.g. the FOH engineer on Solo 1, a monitor engineer on Solo 2 — without interfering with each other. Or keep one bus set up one way (e.g. PFL for gain-staging) and the other different (e.g. AFL for checking the mix).

## SOLO ROUTING

### HP only vs. Main + HP

Decides whether the chosen solo bus feeds **only** the headphone outputs, or the headphones **and** the main output. Live, you'd almost never route solo to the mains — it would put whatever's soloed into the PA. In a studio it's common to route through the mains too, kept lower via the PFL/AFL "Dim" setting.

## MODE

### "Mix" vs "Last"

**Mix** lets you solo several channels at once and hear them summed together (traditional multi-solo). **Last** is exclusive solo — pressing Solo on a new channel automatically clears whichever channel was soloed before it, so only the most recently pressed Solo is ever active.

## ALSO WORTH KNOWING

**Solo Type** offers PFL (pre-fader — checking input source/gain), AFL (post-fader — checking a channel's actual contribution to the mix), and Auto (PFL for inputs, AFL for auxes/groups, automatically). **Solo In Place** mutes everything else to the main output when engaged. **Solo Follows Select** can auto-solo whatever channel you're currently editing.

# Scene Automation / Selective Loading — Is This Understanding Correct?

What Violet Audio documents as "Scene Isolation" — restricting what a scene recall actually touches.



The workflow understanding here is essentially right, and it matches what Violet Audio documents as "Scene Isolation".

## HOW IT ACTUALLY WORKS

1. By default, recalling a scene/snapshot replaces the **entire mixer state** for the channels included.
2. Turning isolation on lets you restrict what actually gets recalled: choose which channels are affected (all channels, filtered by channel type, or hand-picked via a Select option), then — per parameter category (EQ, dynamics, routing, etc.) — mark whether that category should be pulled in from the stored scene (automated/recalled) or left alone at its current live value (ignored/not touched).
3. Marking everything to be recalled is the same as disabling isolation — the full stored state loads, matching the all-green "Load All" behaviour.

# What Is RadioX12, and How Does It Relate to dMix 128?

A virtual broadcast console software layer that expands dMix 128 into a software-defined production platform.

• **Beta — coming Q4 2026**

• macOS now, Windows in beta prep

• Manual ships inside the app

## WHAT IT IS

RadioX12 is a **virtual broadcast console** — software that turns dMix 128 into a software-defined production platform for enterprise-class broadcast workflows, running natively on Windows and macOS.

It supports **up to three independent studios** at once, with isolated routing between them, on top of the same dMix 128 hardware.

## KEY FEATURES

- 64-channel AES67/RAVENNA audio drivers (Core Audio and ASIO)
- Axia Livewire+ interoperability, with optional Dante expansion
- Mix-minus automation, dedicated headphone mixes, talkback
- Customizable fader layouts with user permission controls
- OSC integration plus hardware/software triggers

## Requires dMix 128

RadioX12 is a software layer on top of dMix 128, not a standalone product — it draws on the console's 32 analog mic/line inputs, 24 XLR outputs, MAD1 connectivity, and its 128 mixing channels / 88 busses of onboard DSP. Standard Ethernet carries the audio, with drop-in connectivity to Pro Tools, Logic Pro, Reaper, Cubase and Nuendo.

**Target applications:** *radio, television, live production, houses of worship, educational institutions, podcast studios, and content creation facilities consolidating onto an IP-based workflow. Part of a broader wave of dMix ecosystem software currently in beta alongside V2 firmware, MCP control software, MIDI control, dMix STUDIO (offline editor) and network discovery tools — check with Violet Audio for current beta access.*

# Signal Paths on the Splitters — Is Ethercon Bidirectional?

All three units differ in channel count and split count — and only one of them is actually bidirectional over Ethercon.

All three units are entirely **PASSIVE splitters** (no power required) — they differ in channel count, number of splits, and, importantly, which one is actually bidirectional over Ethercon.

## IO44

ETHERCON BIDIRECTIONAL

4-channel, 2-way splitter/converter

**Front:** 4× locking XLR inputs.

**Rear:** 4× XLR outputs, plus 1× etherCON (4-channel).

The ONE bidirectional-by-design unit: the Ethercon/RJ45 port carries all 4 analog XLR signals over a single shielded Cat5e/Cat6 run to another device — a second IO44, or directly into a DX08B/DX16B's Ethercon port. A point-to-point XLR-over-Ethercon "snake" leg, not a networked/AoIP connection.

## DX08B

OUTPUT ONLY

8-channel, 2-way passive splitter

**Front:** 8× locking XLR inputs only (no front-panel outputs).

**Rear:** 8× XLR outputs (split 1, isolated), plus 2× etherCON, 8ch each (split 2, isolated), AES72-20 standard.

Mic/line in on a front XLR → isolated copy 1 out the rear XLR → isolated copy 2 out the rear Ethercon (plain analog audio on the Ethercon's pins — e.g. feeding an IO44). NOT a network/AoIP link — just using the connector shell/pinout to carry channels down one cable, one-way (output only).

## DX16B

OUTPUT ONLY

16-channel, 3-way passive splitter

**Front:** 16× locking XLR inputs AND 16× XLR outputs (direct/thru split).

**Rear:** 16× more XLR outputs (split 2), plus 16ch via etherCON and 16ch via 25-pin DB25 (split 3, two connector formats).

Front XLR in → front XLR out to FOH (direct), rear XLR out to monitors (split 2), rear Ethercon/DB25 out to recording/broadcast (split 3) — three isolated, galvanically separate feeds. Ethercon here carries one-way analog audio only, same role as the DB25.

✓ **In short:** Ethercon is bidirectional only on the IO44, built as a point-to-point send/return pair to another IO44 or a splitter's Ethercon port. On the DX08B and DX16B, Ethercon is just one of the splitter's fixed OUTPUT legs, functionally identical to the DB25 output — it does not carry two-way or networked signal.

# What Is Spatia 2D, and How Does It Relate to dMix 128?

Object-based spatial audio software for dMix 128 — position, move, and automate audio objects across a 2D sound field instead of fixed speaker buses.

• Beta — coming Q4 2026

• macOS now, Windows in beta prep

• Manual ships inside the app

## WHAT IT DOES

Rather than assigning audio to fixed speaker buses, Spatia 2D works directly with **audio objects** that can be freely positioned, moved, grouped, and automated throughout a two-dimensional sound field — up to **24 freely positioned loudspeaker outputs**.

Custom movement paths and timeline-based trajectory automation let an object's position, depth and width be shaped and repeated identically across multiple shows via scene/cue recall.

## KEY FEATURES

- Up to 24 freely positioned loudspeaker outputs
- Custom movement paths, timeline-based trajectory automation
- Advanced panning — object position, depth and width
- Scene and cue recall for repeatable shows
- Real-time object control

## TECHNICAL SPECIFICATIONS

**Platform** Windows and macOS

**Sync** MIDI, Timecode, and OSC synchronization

**Networking** AES67 and SMPTE ST 2110

## Requires dMix 128

Spatia 2D expands the dMix 128 platform, drawing on its powerful DSP architecture, low-latency audio engine, scalable I/O, and browser-based control environment — it is a software layer on top of the console, not a standalone product.

**Target applications:** *live productions, theatrical performances, installations, broadcast environments, and studio work. Part of the same beta wave as RadioX12, V2 firmware, MCP control software, MIDI control, dMix STUDIO (offline editor) and network discovery tools — check with Violet Audio for current beta access.*