GEN6 ENCLOSURE PLATFORM · V1.0

Hydra Command Center

Bring up, control, and qualify eight Gen6 drive bays — with zero host drivers.

Hydra Command Center is the control and validation plane for the Gen6 Hydra enclosure. It drives every bay over the enclosure's management MCU — **no host PCIe link, no vendor tools, no OS driver** — from a browser UI, a REST API, or the **hydractl** CLI. Add the optional **Host Agent** to see what the Linux host sees in-band, and run repeatable **Workloads** against both views.

THREE WAYS TO DRIVE THE ENCLOSURE

One control plane, three surfaces

The same commands are available everywhere — click a bay in the UI, call the same route from a script, or automate the whole bench from the CLI. Nothing here requires the drive to be visible to a host OS.

Web UI

A live front panel for all eight bays — click a bay to select it, double-click for full Bay Detail. Every control has a mouse-over tooltip, with a light/dark theme for the bench.

REST API

Every UI action is a documented JSON route with interactive docs — slots, protocol consoles, diagnostics, firmware, workloads, and fault injection all scriptable end to end.

hydractl CLI

A single command-line tool that mirrors the API 1:1 — drop it straight into bench scripts, CI jobs, and regression sweeps. 23 ready-to-run example scripts ship with it.

CLICK A BAY ON THE PANEL, THEN REACH IT FROM A SCRIPT

Every bay, every path, one Active Bay model

Select a bay in the UI and it becomes the Active Bay for the protocol consoles too — or skip the UI entirely and drive every one of the eight bays directly by slot number from **hydractl** or the API.

8	3	7	0
Independently controlled drive bays	Control surfaces — UI, REST API, CLI	Live pass/fail validation gates per NVMe drive	Host drivers required for out-of-band control

MONITOR & CONTROL

Every bay, live — power, protocol, and environment

Each of the eight bays reports live drive health and identity in the background, so the control surface never waits on a slow read. Every protocol console works whether or not a host is even attached.

Slot Control

Power, LEDs, resets, and dual-port/PWRDIS per bay — plus automatic detection of drives mounted through a **U.3 interposer**, with a dedicated SAS page *(experimental)*, and a one-click **Recover** for a bay stuck off the bus.

Protocol Console

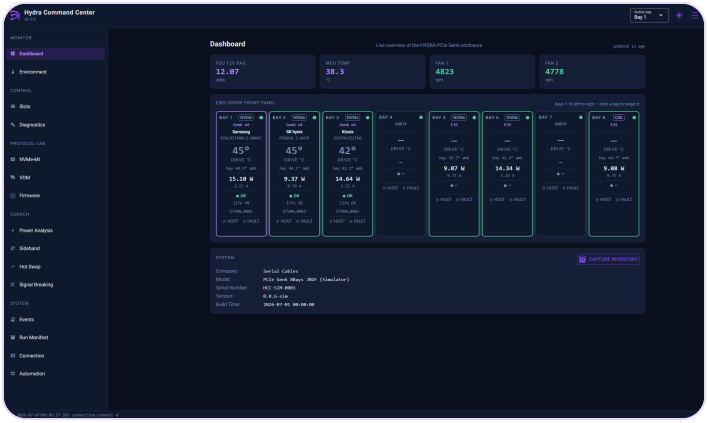
NVMe-MI, VDM, and **CXL CCI** *(Technology Preview)* consoles — typed presets, raw opcode escape hatch, live annotated frame preview, full history/replay.

Environment & Telemetry

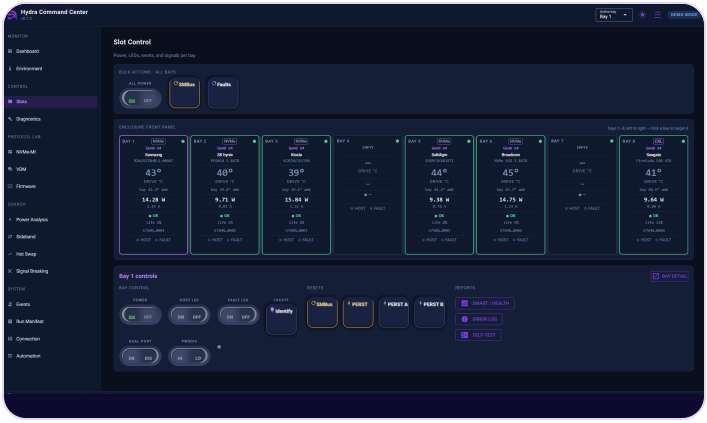
Enclosure thermal, fan, power, and clock telemetry alongside per-bay ambient and drive-reported temperature, persisted for later review.

Diagnostics & Firmware

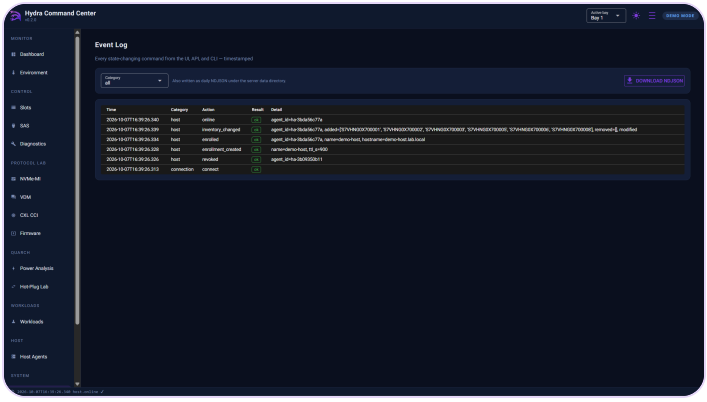
Self-test, raw I2C/SMBus access, **I3C** *(Technology Preview)* discovery and a CCC engine, error-log triage, and safety-gated out-of-band drive firmware update.



Dashboard — all eight bays at a glance: health, temperature, power, and life for every drive.



Slot Control — per-bay power, LEDs, resets, and the eight-bay front panel.



Event Log — a millisecond-timestamped audit trail of every state-changing command.

Audit trail & run manifest

Every command from the UI, API, or CLI is recorded, so you can line a pulled drive up against a host-side analyzer capture or **dmesg**. A one-click **run manifest** captures exactly which firmware and serial numbers a test campaign ran on.

NEW IN V1.0 · IN-BAND & REPEATABLE

See both sides of the link — then repeat the test

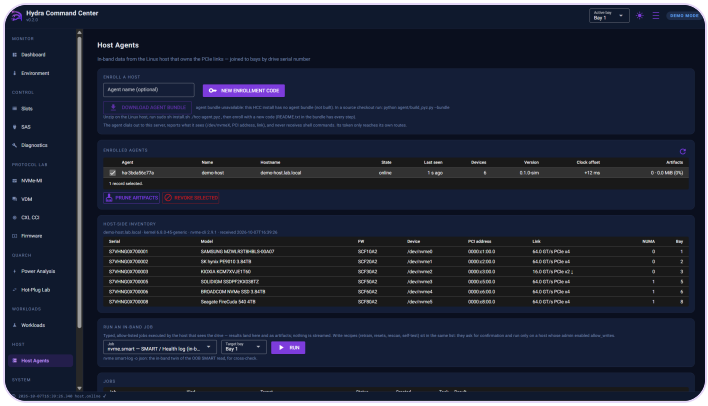
The enclosure reads each drive out-of-band. The **Host Agent** adds what the Linux host that owns the PCIe links sees, and **Workloads** turns the combined view into saved tests you can run again and compare.

Host Agent (in-band)

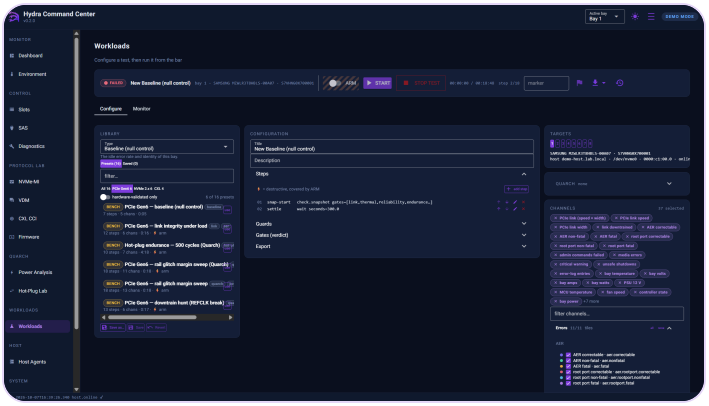
A single-file, standard-library Python agent that **dials out** to HCC — nothing listens on the host. It reports **/dev/nvmeX**, the trained link, kernel AER counters, and fio throughput, and joins them to each bay by **drive serial number**. It only runs typed jobs with a fixed command line, never a shell, and its token can never touch slot power, resets, I2C, or firmware.

Workloads

A saved test: which bay (and which Quarch module), which steps, which channels, which gates to read at start and end. **Errors are counted and shown, never auto-failed**; **STOP TEST** is never gated; destructive steps need an **ARM**. A finished run exports a signable **Qualification Report (HTML)** — findings, device/setup, results by phase, health before/after, sign-off — plus a live **Statistics** panel (IOPS/W, J/GB, P99). Sixteen presets ship across PCIe Gen6, NVMe 2.x, and CXL.



Host Agents — enrollment, host-side inventory joined to bays by serial, and in-band jobs.



Workloads — run-control bar, preset library, configuration, and live channels.

Catch what the enclosure alone can't

Three **agreement gates** — **host-identity**, **host-link**, **host-smart** — compare the out-of-band report with the in-band one. A link the enclosure says is x4 Gen5 and the host trained at x2 is the classic finding. Hot-plug **enumeration timing** measures how long a drive takes to come back.

Reports you can diff

Every run produces a verdict (**passed, review, failed, or incomplete**) and exports as a JSON manifest, aligned CSV, NDJSON, and a self-contained HTML report — with counters shown, not judged.

Maturity — the Host Agent and Workloads are fully supported. More than half the shipped Workload presets (9 of 16) are now proven end to end on hardware; the rest carry a **BENCH** marker in the library — informational, not a lock.

QUALIFY & PROVE

Pass/fail validation, traceability, and fault injection

Every gate reads the drive live and rolls up to a fail-closed verdict — an unread or unresolved bay is never a green pass.

GATE	WHAT IT CHECKS
Link	PCIe link training vs. the drive's max (optional expected Gen/width)
Thermal	Current hottest temperature vs. limit
Reliability	Critical-warning flags, media errors, spare vs. threshold
Endurance	Percentage of rated life used vs. limit
Firmware	Active firmware revision vs. expected
Traceability	Resolved model + serial vs. expected DUT manifest
Self-Test	Latest Device Self-Test result

Note — the seven gates above apply to NVMe bays. A bay auto-detected as a **CXL Type-3 device** runs its own three-gate set (health, events, traceability) instead — automatically. When a Host Agent sees the drive, three more host agreement gates join the rollup.

Serial-keyed device library

Every drive identity is folded into a durable, serial-keyed library for FA/RMA traceability — identity only, never health, never silently merged.

Run manifest & inventory

Capture exactly what hardware ran a test campaign, tied to the same identity the validation gates use.

Event-log audit trail

Every firmware stage, destructive action, and fault-injection event is recorded for later review.

AUTOMATION-FIRST · FAIL-CLOSED GATING · NO HOST DRIVER · SERIAL-KEYED TRACEABILITY · 23 EXAMPLE SCRIPTS

OPTIONAL QUARCH HARDWARE

Hardware fault injection, built in

With an optional Quarch drive-control/breaker module, the **Hot-Plug Lab** unifies signal routing, targeted signal breaking, and timed hot-swap cycling with a hardware **glitch engine** (nanosecond-class, module-timed) — plus a **Power Analysis** page for per-rail and sideband scope capture. Works on Windows with the vendor USB driver; Linux needs a one-time udev rule.

SERIAL CABLES · GEN6 HYDRA ENCLOSURE

Bring up and qualify Gen6 drives, driver-free

One control plane for power, protocol, environment, and pass/fail qualification — across all eight bays, out-of-band, from day one of bring-up.