NLTlabs
NEW LOGIC TECH

NLT Memory Operator Guide

Read health, capacity, and trust on the localhost:8088 dashboard

Read NLT Memory health, capacity, and trust signals at a glance.
Operator tasks on the localhost:8088 dashboard, not the report-studio CLI.

AUDIENCE NLT Memory operators · platform engineers

Executive summary

GUI WALKTHROUGH	**3** OPERATOR FLOWS	**Live** SOURCE	**Vol 8** DEEP DIVE
Annotated screenshots	Overview · memory · integrity	localhost:8088 hub	Memory architecture

This guide walks the NLT Memory dashboard at localhost:8088, the live operator view over the running memory hub. Each section pairs operator steps with an annotated screenshot: text first, image second. NLT Memory holds persistent, deterministic memory for AI coding agents, and the export carries no raw text by default. Captures refresh via scripts/capture-platform-ui-screenshots.mjs.

Key findings

1. Operator tasks on the NLT Memory dashboard, not the report-studio CLI.
2. Hotspots call out the health, capacity, and trust controls operators read.
3. Next steps link to Vol 8 architecture and Vol 10 operations.

Who should read this
Operators reading NLT Memory health on localhost:8088. Engineers continue to Vol 8 for the MCP-bridge architecture after this how-to pass.

SECTION I

Overview

Who this guide is for and what you will produce.

This guide walks the NLT Memory dashboard at localhost:8088, the live operator view over the running memory hub. Each section pairs operator steps with an annotated screenshot: text first, image second. NLT Memory holds persistent, deterministic memory for AI coding agents, and the export carries no raw text by default. Captures refresh via scripts/capture-platform-ui-screenshots.mjs.

Methodology

This brief is generated from the live NLT document suite (nlt-report-studio v0.2.0, brand nlt-v3.2). Diagrams and tables reflect build-time workspace discovery where noted.

Audience

- Investors & board — Vol 1-2
- Architects — Vol 3-4
- Engineering DD — Vol 5-8

Related proof

portfolio.nltlabs.ai · briefs.nltlabs.ai

SECTION II

Open the memory dashboard

Open the NLT Memory dashboard and read the health-at-a-glance header.

Follow the steps below, then confirm the dashboard state matches the screenshot.

- Open http://localhost:8088/ in a desktop browser; the dashboard reads the running hub.
- Confirm the LIVE badge is green with a recent refresh time, and set the Refresh interval for auto-polling.
- Use the Project filter to scope to one brain, or leave it on All projects.
- Read the KPI strip: ENTRIES against cap, DB SCHEMA, PRIVACY tier, HIVE HUB, and TIERS rows.

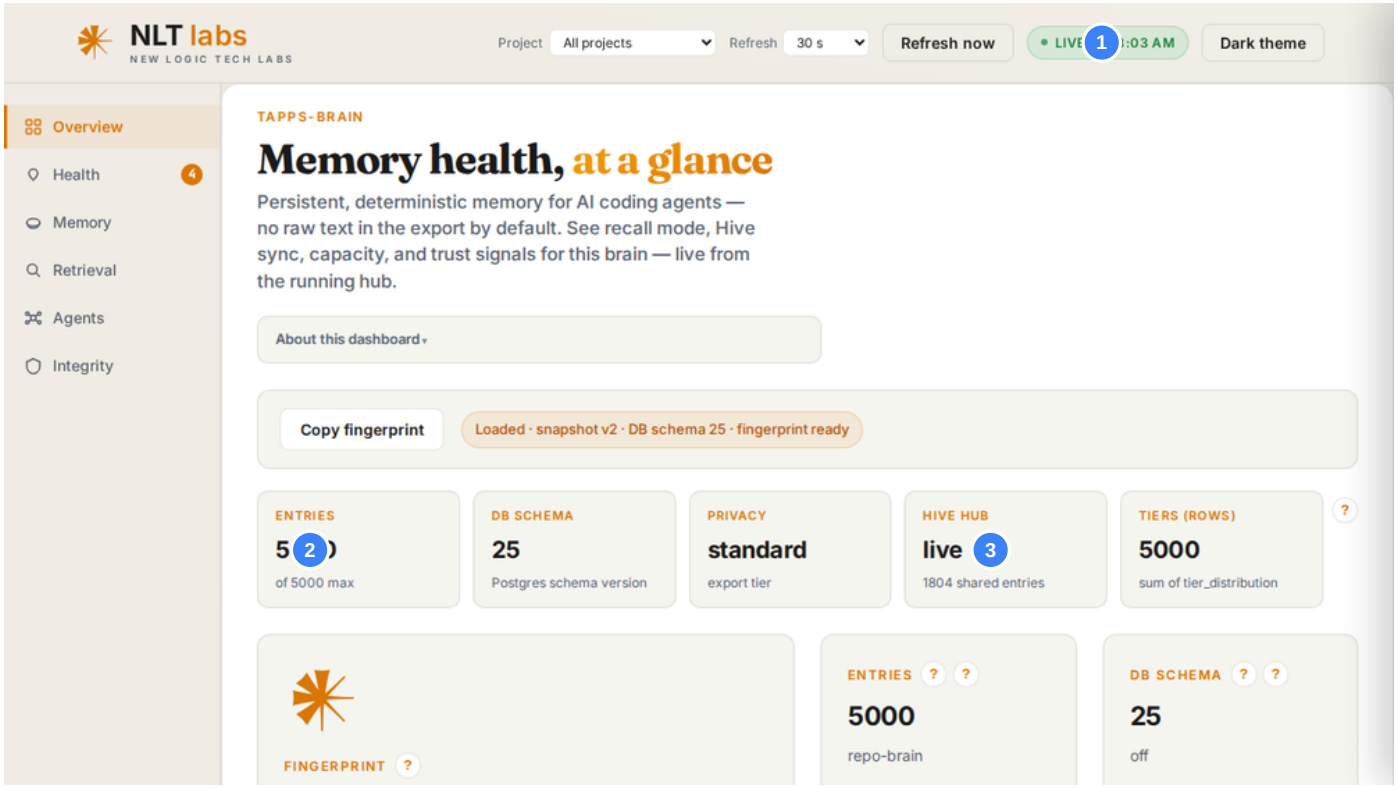

Figure 1 — NLT Memory overview: live health badge, fingerprint, and the capacity KPI strip.

- 1 — LIVE badge confirms the dashboard is reading the running hub, with last-refresh time.
- 2 — ENTRIES shows current rows against the cap (5000 of 5000 max here).
- 3 — HIVE HUB shows federation status and shared-entry count across the hive.

SECTION III

Read capacity and tier mix

Open the Memory tab to read capacity headroom and tier mix.

Work the steps in order; each maps to a numbered hotspot on the screenshot.

- Select Memory in the left navigation.
- Re-read ENTRIES against the cap to gauge headroom; auto-GC fires near 80 percent capacity.
- Scan the Pulse panel for tier distribution across architectural, pattern, procedural, and context rows.
- Use Copy fingerprint to capture the snapshot identity before sharing or comparing exports.

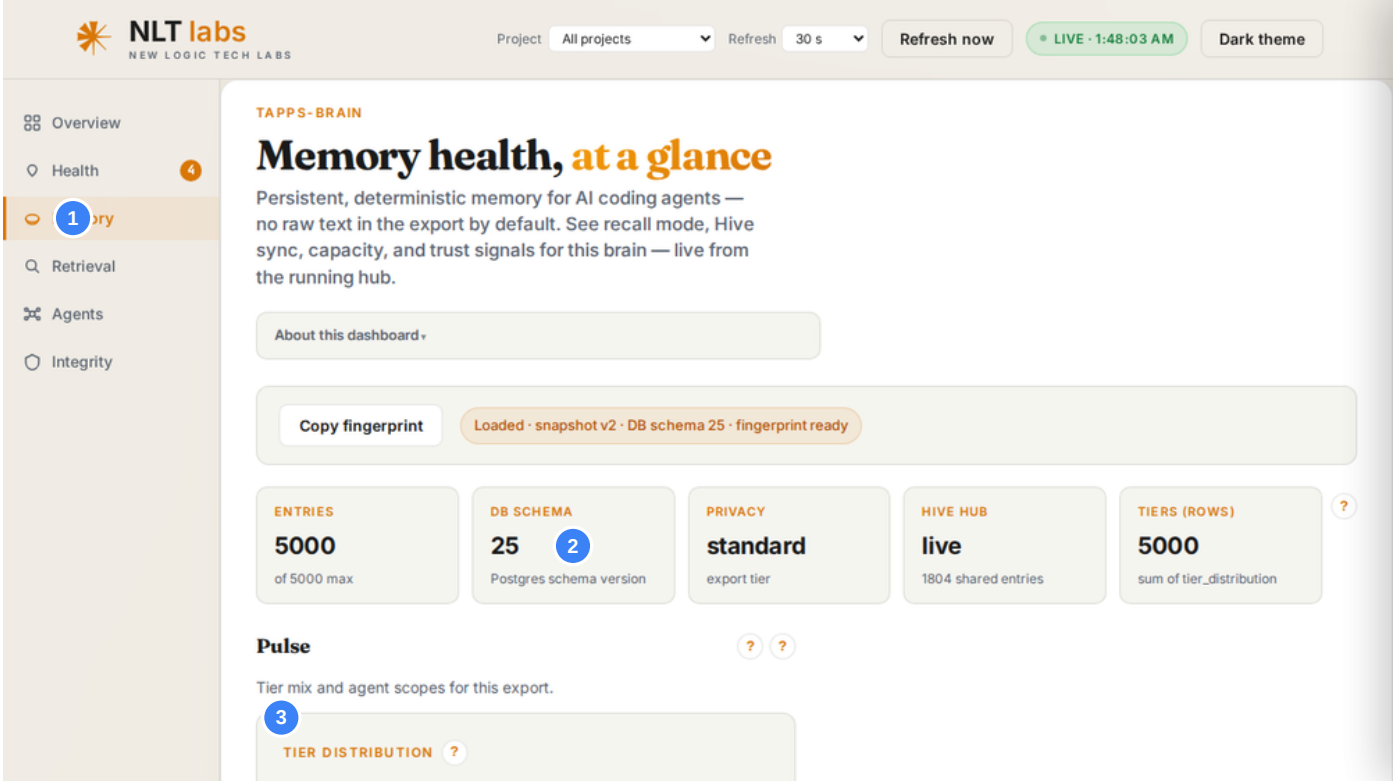

Figure 2 — Memory tab: capacity KPIs and the Pulse tier-distribution panel.

- 1 — Memory tab opens capacity, tier mix, and agent scopes for this export.
- 2 — DB SCHEMA is the Postgres schema version the snapshot was exported from.
- 3 — Pulse and Tier distribution show how entries split across the four memory tiers.

SECTION IV

Verify integrity and trust

Open the Integrity tab to verify trust signals before relying on the export.

Read each trust signal below and resolve any warning before sharing the snapshot.

- Select Integrity in the left navigation.
- Check VERIFIED, NO HASH, and KEY MISMATCH; a high KEY MISMATCH after a key rotation means re-sign, not tampering.
- Read RELATIONS for the knowledge-graph edge count and confirm the graph is populated.
- Heed the snapshot-schema warning and re-export for full fidelity when the snapshot trails the live schema.

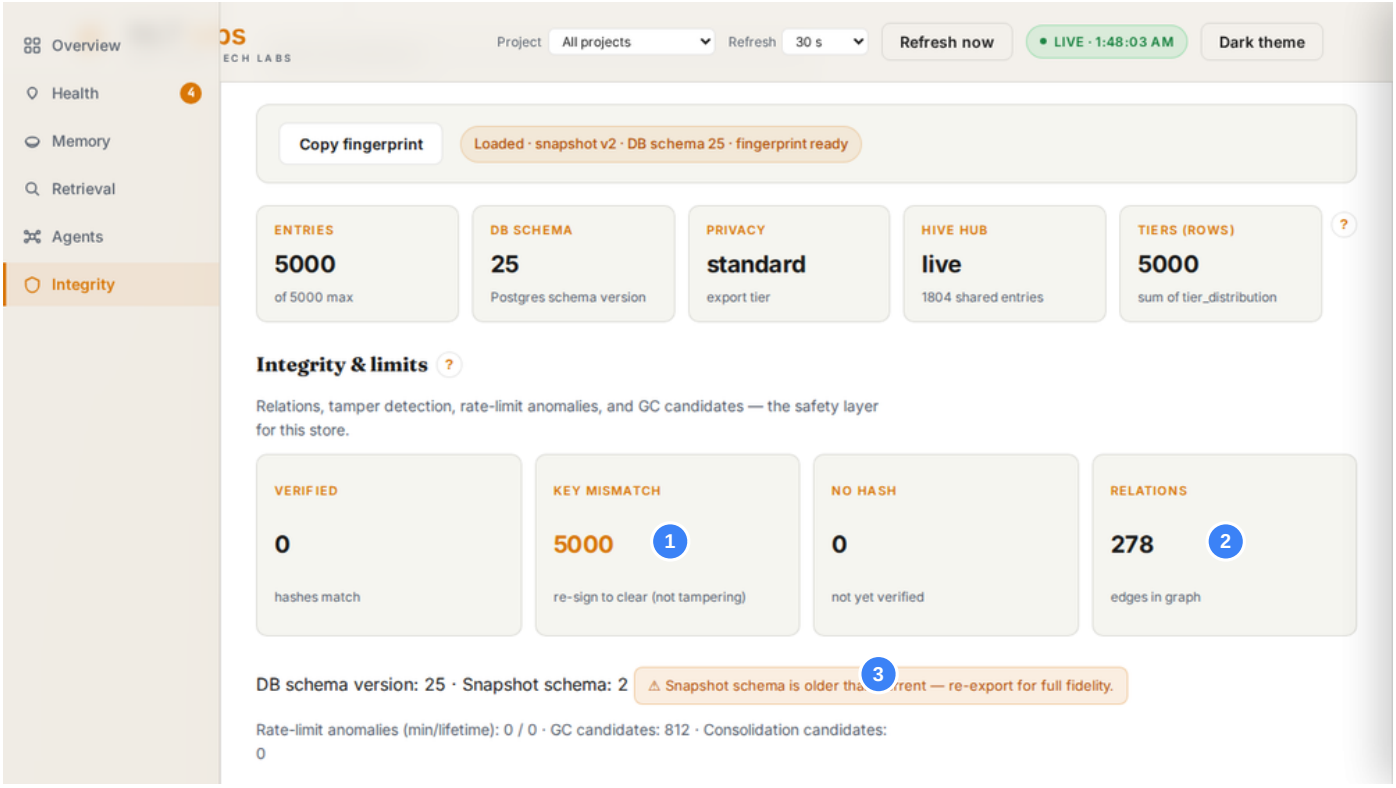

Figure 3 — Integrity and limits: verified vs. key-mismatch counts, relations, and the schema-drift warning.

- 1 — KEY MISMATCH counts entries needing re-sign after a key rotation, not tampering.
- 2 — RELATIONS counts edges in the knowledge graph connecting memory entries.
- 3 — Snapshot-schema warning: re-export when the snapshot trails the live DB schema.

SECTION V

Next steps

Where to go after your first workflow.

- Deep dive: NLT Memory architecture is Vol 8 of the suite (docs/NLT_DOCS_SUITE.md).
- Operations: Vol 10 covers federation, GC, and capacity playbooks.
- Refresh UI shots: node scripts/capture-platform-ui-screenshots.mjs (needs localhost:8088 live).
- Memory CLI: uv run tapps-mcp memory search|save|health for scripted access.

Engineer path
Engineers integrating NLT Memory should read Vol 8 for the MCP-bridge architecture and use uv run NLT Quality Pipeline memory for scripted save, search, and health checks.