GPU HEALTH INTELLIGENCE · TECHNICAL FINDINGS

Introducing SOIL: Supercomputer Observability and Intelligence Layer

Fault Prediction on a Production H100 SXM5 Cluster

5-day lead time · 8 × H100 SXM5 · commodity DCGM telemetry only

I. EXECUTIVE SUMMARY

Supercomputer Observability and Intelligence Layer (SOIL) is Oru'el's observability and intelligence platform for data centers. This report presents SOIL's pilot results and more insights into how these results were captured.

SOIL had access to an eight-GPU H100 SXM5 production cluster for 2 weeks. Over these two weeks, SOIL had predicted two major hardware failures with a Root Cause Analysis (RCA), approximately five days before they manifested as operator-visible incidents across NVIDIA DCGM.

The two predictions flagged different failure mechanisms on different subsystems; they were not a single shared anomaly. Both were validated against incident logs and observed hardware behaviour at the time of fault.

GPU	PREDICTED FAULT MODE	LEAD TIME	OUTCOME
GPU A	Compute-pipeline decoupling, originating fault node in a fabric-wide cascade.	**~5 days**	Hard fault at predicted mechanism; triggered cascade across 5 secondary GPUs.
GPU B	High-bandwidth memory subsystem degradation, independent of GPU A.	**~5 days**	Clock-floor collapse and memory-allocation failures under load, as predicted.

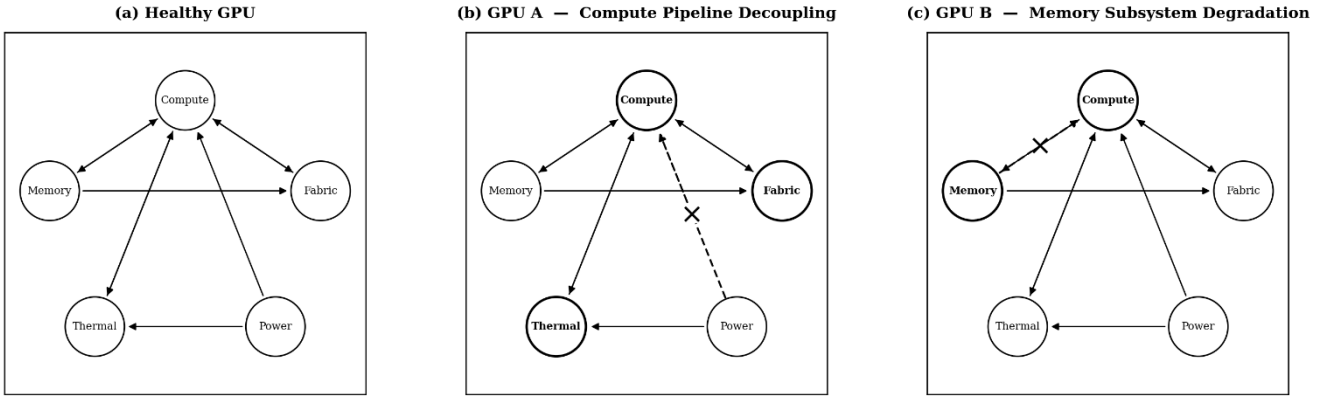

Figure 1. *Subsystem causal graph under three conditions. SOIL monitors directed edges between compute, memory, fabric, thermal, and power subsystems. A healthy GPU (a) presents all edges intact. GPU A (b) shows a persistent breakdown of the clock-to-compute edge, corroborated by thermal and fabric stress. GPU B (c) shows an isolated breakdown of the memory-to-compute handoff with all other edges nominal.*

II. FINDINGS

2.1 GPU A: Originating fault in a cascade event

SOIL flagged GPU A as a primary degradation node by detecting a persistent decoupling between the GPU's compute-clock behavior and its actual compute throughput. In a healthy GPU, clock changes drive proportional changes in compute activity. On GPU A, this relationship collapsed and remained collapsed across multiple rolling analysis windows, indicating the compute pipeline was no longer tracking its own control inputs, a structural, not transient, anomaly.

Secondary corroborating evidence: sustained operation within roughly 9°C of thermal throttle (the highest sustained thermal stress in the fleet) and a fabric error-counter population approximately 35 % above the fleet floor. SOIL combined these into a single explanation: the GPU was accumulating silicon-level stress that its own governor could no longer mask. The same reflected across the degradation patterns which were modeled using our proprietary Physics AI models.

Outcome: During the training load, GPU A produced a hardware-layer register read failure and an irrecoverable kernel assertion, and propagated a fabric-level cascade that temporarily destabilized five secondary GPUs in the cluster. We believe SOIL correctly identified GPU A as the *origin*, not merely one of many affected GPUs, a distinction that threshold-based monitoring (which would have flagged all five equally) cannot make.

2.2 GPU B: Independent memory subsystem degradation

GPU B was flagged separately, and for an entirely different reason. SOIL detected that memory activity on this GPU had stopped driving compute activity in the normal causal direction. Memory was working; compute was working; but the handoff between them, the mechanism by which memory bandwidth is converted into useful compute was progressively breaking down. This signature appeared with high statistical significance across five independent analysis windows, ruling out noise or a single bad sample.

Outcome: Under load, GPU B's compute clock collapsed to a sustained floor well below the fleet median even when power was not limiting, and memory-bound workers failed to allocate. Both behaviors are consistent with a degraded memory controller, and both matched the prediction's causal mechanism exactly.

Crucially, GPU B showed *none* of the stress markers observed on GPU A, no thermal-headroom issue and the lowest fabric error population in the fleet. A standard health score would have cleared this GPU. SOIL caught it because the causal relationships between its telemetry signals, not the values themselves, had already begun to change. This is the core thesis: raw telemetry viewed in isolation misses the physics of what is actually happening inside the hardware.

III. HOW SOIL REACHED THESE CONCLUSIONS

SOIL does not score GPUs against threshold rules or learned anomaly distributions. It models each GPU as a system of interacting physical subsystems: compute, memory, fabric, thermal, and power. It continuously tests whether the causal relationships between those subsystems remain intact. A healthy GPU has a stable causal signature; a degrading GPU's causal signatures drifts in a characteristic way long before its scalar metrics cross any alert threshold. These degradations are better captured using a physics-inspired approach.

In this pilot, SOIL isolated *which* causal edge was breaking, on *which* GPU, and inferred the *mechanism* from the identity of that edge. GPU A's failure was a compute-pipeline problem; GPU B's was a memory-subsystem problem. The two predictions used the same engine but produced mechanistically distinct, independently verifiable diagnoses, which is why both held up against the ground-truth incident data which reflected 5 days later.

IV. POTENTIAL BEYOND THIS PILOT

— **Lead-time scaling.** Lead time in this pilot was five days using the single lowest-fidelity telemetry source. Incorporating fabric, host, power, and cooling signals already exposed by standard data centre

instrumentation shifts the prediction surface from structural drift to micro-precursor events, moving the time delta from *days-ahead* to **minutes-ahead with sub-GPU localization** (HBM controller vs. SM vs. fabric link).

— **Mechanistic outputs.** Because SOIL outputs a causal mechanism rather than a scalar score, every prediction is actionable at the maintenance-ticket level: replace, throttle, migrate, or re-schedule. This is the difference between an alert and a decision. SOIL fastens the decision-making process, at scale.

— **Zero-label transfer.** The two faults here were distinguishable from each other and from the rest of the fleet without labelled failure history. The same engine transfers to new clusters and new GPU generations with no retraining on customer data required.

— **Cascade root-cause.** Correctly identifying the originating node of a cascade event (GPU A) rather than alerting on all five affected GPUs equally, is a capability threshold-based observability stacks structurally cannot deliver. The commercial wedge sits here: fewer unnecessary GPU replacements, faster root-cause on cascade incidents, and actionable pre-failure windows long enough to migrate workloads rather than abort them.

END OF REPORT