



Buying Used Data Center GPUs: Verification Checklist

Plan and build private AI compute with GPU Smith. We connect workload requirements with GPU hardware, networking, deployment and operating decisions so your infrastructure fits the work it must perform.

Published by GPU Smith · 2026-10-29 · buying used h100 gpu · refurbished a100 gpu checklist · used data center gpu risks · how to test used gpu dcgmi diag · nvidia row remapping failure used gpu · cuda 13 pascal volta support

Original article: <https://gpusmith.com/articles/en/buying-used-refurbished-data-center-gpus-checklist>



In this article

1. The short answer
 2. When used GPUs make sense
 3. Provenance and paperwork
 4. Software support cliffs
 5. Memory health: ECC, retired pages and row remapping
 6. Reading the InfoROM data correctly
 7. DCGM diagnostics as acceptance
 8. Link and fabric checks
 9. Thermal, power and burn-in plan
 10. Silent corruption screening
 11. Contract terms and red flags
 12. Printable checklist
 13. Sources
-

The short answer

Buy a used A100, H100 or similar data center GPU only after four checks pass, before payment is final: board serials match `nvidia-smi` and the invoice; the InfoROM memory error records show no failure or pending flags and enough spare-row headroom; every card passes DCGM diagnostics through level 4 in your chassis; and the architecture is still supported by the CUDA toolkit and driver branch you plan to run. Thermal history, workload intensity and tampering cannot be proven, so handle them in the contract.

Command and field names below are quoted from NVIDIA documentation as of October 2026. NVIDIA warns that "the output of NVSMI is not guaranteed to be backwards compatible" ([NVIDIA nvidia-smi documentation](#)), so check `--help-query-*` on your driver before scripting.

When used GPUs make sense

Used hardware makes sense when your workload fits an older architecture's memory and software support and you can absorb an acceptance window: inference, fine-tuning, development and CI clusters.

Price per GPU is not the whole cost: add test labor and a shorter remaining service life (see [GPU useful life and collateral](#)). We quote no used-market prices; they move quickly. Compare dated written quotes from three sellers against a new-unit quote.

Field data shows why a fleet-retired card needs checking. Meta reported that during a 54-day snapshot of Llama 3 405B pre-training, run on up to 16K H100 GPUs, there were 419 unexpected job interruptions. GPU issues caused 58.7% of them: 148 (30.1%) were "Faulty GPU" and 72 (17.2%) were "GPU HBM3 Memory" ([Llama Team, AI @ Meta, 2024](#)). Those were in-service cards, but it shows where to look first: GPU memory.

Provenance and paperwork

Match serials three ways

`nvidia-smi -q` reports a **Serial Number** that "matches the serial number physically printed on each board. It is a globally unique immutable alphanumeric value." It also reports a **GPU UUID**, which "does not correspond to any physical label on the board" ([NVIDIA nvidia-smi documentation](#)). Require label, `nvidia-smi` and invoice serials to agree. Also record **Board Part Number**, **GPU Part Number**, **FRU Part Number**, **VBIO Version** and the InfoROM **Image Version**, which "uniquely describes the exact version of the infoROM flashed on the board" (same source).

Save a full XML dump per card with `nvidia-smi -q -x`: it is your baseline and your dispute evidence.

Warranty and RMA eligibility

We found no NVIDIA document setting a general rule for warranty transfer on resale; coverage depends on the original seller and support contract. OEM processes exist: Dell's ownership-transfer form can be submitted by the new or previous owner. Dell may ask for proof of purchase and says incomplete details "may result in a denial of the ownership transfer" ([Dell KB 000125103, modified 2026-06-10](#)). If warranty is part of what you are paying for, get the OEM's written confirmation of transfer before payment, not the seller's word.

On RMA, NVIDIA's memory-error documentation states that "the NVIDIA Field Diagnostic tool determines whether a GPU qualifies for RMA" ([NVIDIA, RMA Policy](#)). DCGM diagnostics are explicitly not designed to "replace the field diagnosis tools" or to "facilitate any RMA process" ([NVIDIA DCGM Diagnostics](#)). A DCGM result is an acceptance signal, not an RMA determination.

Export and resale constraints

Export controls on advanced accelerators can apply to secondary-market sales. Outside the US, a resale can be a reexport or an in-country transfer, which the EAR defines as "a change in end use or end user of an item within the same foreign country" ([15 CFR 734.16, eCFR current to 2026-10-27](#)). Treat this as a question for counsel (see our [export restrictions](#) overview) and get a written compliance representation from the seller.

Software support cliffs

- **CUDA 13.0 removed pre-Turing targets.** "Offline compilation and library support for these architectures have been removed in CUDA Toolkit 13.0 major version release," referring to Maxwell, Pascal and Volta. The 12.x series still builds for them, "but newer toolkits will be unable to target these architectures" ([CUDA 13.0 Release Notes](#)).
- **Libraries followed.** cuFFT 13.0 "Removed support for Maxwell, Pascal, and Volta GPUs, corresponding to compute capabilities earlier than Turing". cuSPARSE 13.0 "Dropped support for pre-Turing architectures (Maxwell, Volta, and Pascal)" ([CUDA Toolkit 13.4 Update 2 Release Notes](#)).
- **Drivers.** NVIDIA states that "the release 580 series will be the last to support GPUs based on the Maxwell, Pascal, and Volta architectures" ([NVIDIA Developer Forums, updated 2025-07-01](#)). Rechecked on 2026-10-28: R580 is still maintained, and 580.196.07 (notes last updated 2026-10-21) still lists V100

and P100 ([NVIDIA, 580.196.07 release notes](#)). The R615 notes (615.71.10, last updated 2026-10-19) list no Volta or Pascal products at all: both the supported-products table and the device-passthrough table stop at the Turing T4/T4G ([NVIDIA, 615.71.10 release notes](#)). CUDA 13.0 corresponds to R580, and CUDA 13.4's new features require R615 or later ([CUDA Toolkit 13.4 Update 2 Release Notes](#)).

GPU (examples)	Architecture	Compute capability	Status, October 2026
V100	Volta	pre-Turing	No CUDA 13.x offline compilation or libraries; driver stops at the R580 branch
T4	Turing	7.5	Supported by CUDA 13.x
A100, A30	Ampere	8.0	Supported
L40S, L4	Ada Lovelace	8.9	Supported
H100, H200	Hopper	9.0	Supported

Compute capabilities are from [NVIDIA's CUDA GPUs page](#). Buying a cheap V100 therefore also commits you to CUDA 12.x and the R580 legacy branch for the life of that cluster.

Memory health: ECC, retired pages and row remapping

Which mechanism you inspect depends on the generation. Row remapping is available on GA100, GA10x, Ada AD10x, GH100 and GB10x ([NVIDIA, Supported GPUs](#)). NVIDIA calls it "a replacement for the page retirement scheme used in prior generation GPUs". It supports "up to 512 remapping for the frame buffer", against a maximum of 64 page retirements on legacy GPUs ([NVIDIA, Row Remapping](#)).

Commands

```
nvidia-smi -q -d ECC,PAGE_RETIREMENT,ROW_REMAPPER
nvidia-smi --help-query-remapped-rows      # lists the CSV fields your driver exposes
nvidia-smi --help-query-retired-pages
```

All three -d types and the --query-remapped-rows= / --query-retired-pages= CSV queries are documented ([NVIDIA nvidia-smi documentation](#)).

Ampere and later: the Row Remapper section

The documented fields are **Correctable Error** and **Uncorrectable Error** (rows remapped for each cause), **Pending** ("The GPU must be reset for the remapping to go into effect"), **Remapping Failure Occurred**, and the **Bank Remap Availability Histogram**. The histogram sorts each bank into Maximum, High, Partial, Low and None. Correctable remaps do not count against it, because they "can be evicted by an uncorrectable row remapping" (same source).

NVIDIA's RMA policy sets the row-remapping failure flag on any of these three events ([NVIDIA, RMA Policy](#)):

1. "A remapping attempt for an uncorrectable memory error on a bank that already has eight uncorrectable error rows remapped."
2. "A remapping attempt for an uncorrectable memory error on a row that was already remapped", which "can occur with less than eight total remaps to the same bank."
3. "After 512 total remappings for an uncorrectable memory error have occurred."

Blackwell adds a step: "For Blackwell GPUs, a third remapping attempt for an uncorrectable memory error on a bank invokes HBM channel repair if there is a spare channel available." If the repair succeeds, "the hardware will be recovered"; otherwise "the GPU will continue remapping until Row-Remap Failure flag is set" (same source, page updated 2026-09-09). NVIDIA adds that this GPU-memory RAS repair support "is not guaranteed for all Blackwell GPUs" ([NVIDIA, Supported GPUs](#)), so the eight-row and 512-row triggers above remain the acceptance gates.

The separate **SRAM Threshold Exceeded** flag trips on "more than 4 UCE Unique Count events within an address bank for parity protected SRAMs" or "more than 2" for SECDED-protected SRAMs ([NVIDIA, SRAM Uncorrectable Errors](#)).

Volta and earlier: Page Retirement

The fields are **Double Bit ECC**, **Single Bit ECC** and **Pending**. NVIDIA warns that "pages that are retired but not yet blacklisted can still be allocated, and may cause further reliability issues" ([NVIDIA nvidia-smi documentation](#)). The retirement table fills at 64 pages, and "a board that generates 60 or more retired pages is eligible for an RMA". A board with 15 or more retired pages that keeps retiring at least one new page per week "can be evaluated for an RMA" earlier ([NVIDIA, Dynamic Page Retirement FAQ](#)).

ECC counters and a trap on older cards

nvidia-smi keeps **volatile** counters ("since the last driver load") and **aggregate** counters that "persist indefinitely and thus act as a lifetime counter". nvidia-smi -p 1 resets aggregate counts, but "clearing aggregate counts is not supported on Ampere+" ([NVIDIA nvidia-smi documentation](#)). On a V100, all-zero aggregate counts on a fleet-retired card may simply have been cleared. Confirm **ECC Mode** Current: Enabled, and enable persistence mode (nvidia-smi -pm 1) before testing, since otherwise Linux volatile counts reset "each time a compute app is run".

Reading the InfoROM data correctly

- **Counts are records, not hardware state.** The number of remapped rows "is the number of entries recorded in the InfoROM, not the ones remapped in hardware" ([NVIDIA, User Visible Statistics](#)). Related Xids are 63 (entry recorded successfully) and 64 (failure to record an entry) (same source).
- **Pending means the work is unfinished.** Row remapping "requires a GPU reset to take effect and will remain persistent throughout the life of the GPU" ([NVIDIA, Row Remapping](#)). Reset with nvidia-smi -r (requires root and no running apps; NVLink-connected pre-Ampere GPUs must be reset together) ([NVIDIA nvidia-smi documentation](#)). Then re-read the remapper and diff it against your snapshot.

- **Check InfoROM integrity.** `nvidia-smi --query-gpu=inforom.checksum_validation --format=csv` returns "valid", "invalid" or "N/A". `nvidia-smi` return code 14 means "infoROM is corrupted" (same source). If it is corrupted, the error history cannot be trusted.
- **GPU Recovery Action** should read None. "Drain and Reset" is "asserted only for UCE row remaps" (same source).
- **Approximate usage hours.** `--query-gpu=bbx.time_run`, added in the `nvidia-smi` v595 changes, "shows the cumulative number of seconds a GPU has been running with the driver loaded" (same source). It says nothing about load or temperature. NVIDIA's documentation (read 2026-10-28) gives no supported-GPU list for the field and does not say whether it can be reset, so treat N/A as no information and never as proof of low use.

Worked example (hypothetical numbers). Suppose a card reports `bbx.time_run = 78,000,000` seconds.

- $78,000,000 / 3,600 = 21,667$ hours.
- $21,667 / 8,760$ hours per year = 2.47 years with the driver loaded.

Against a "6 months, lightly used" claim, that gap needs an explanation. For the histogram, NVIDIA's example shows all 640 banks in the Max bucket (8 available rows) when every remapping resource is free ([NVIDIA, User Visible Statistics](#)). Our policy assumption: reject cards with any bank in Low or None, since such a bank is close to the eight-row failure trigger.

DCGM diagnostics as acceptance

DCGM diagnostics (formerly NVVS) run as `dcgmi diag -r <level>`, and "higher numbered tests include all beneath" ([NVIDIA DCGM Diagnostics](#), last updated 2026-06-17). NVIDIA measured the durations on Hopper systems:

Level	Tests added	Duration, 4 GPU	Duration, 8 GPU
r1 (Short)	software	< 2.5 seconds	< 2.5 seconds
r2 (Medium)	+ pcie (PCIe + NVLink), memory, memory_bandwidth	< 2.5 min	< 10.5 min
r3 (Long)	+ diagnostic, targeted_stress, targeted_power, nvbandwidth, nccl_tests	< 10 min	< 35 min
r4 (Extra Long)	+ memtest, pulse	< 45 min	< 2.25 hours

What each test means for a used card:

- **The deployment plugin fails on history you care about.** It fails if "any pages are pending retirement" or on "any pending row remaps or failed row remappings on the target GPU(s)".
- **diagnostic** runs large matrix multiplies for five minutes by default while checking "the correctness of data being written and read". `diagnostic.gflops_tolerance_pcmt` flags a GPU whose GFLOPS deviate from the mean across GPUs, useful for spotting one slow card in a batch.
- **targeted_power** drives the GPU "towards TDP power usage" (default target TDP - 1). It fails if the GPU cannot reach 75% of the target by default.

- **memtest** is "similar to memtest86". By default, Test7 and Test10 alternate for 10 minutes. NVIDIA documents a one-hour run of all eleven patterns with `memtest.test_duration = 3600`. Use that for HBM you have not seen before.
- **pulse** creates current spikes "to ensure that the power supply is fully functional". A pulse failure can come from your PSU or chassis rather than the GPU, so retest a failing card in a known-good slot.

DCGM is "not designed to provide comprehensive hardware diagnostics" (same source). A pass is necessary, not sufficient; our [acceptance testing runbook](#) covers cluster-level pass, fail and quarantine rules.

Worked example: time budget for four used 8-GPU nodes (assumptions labeled).

1. Snapshot and r1: about 15 minutes per node, mostly manual (assumption).
2. r4: under 2.25 hours per 8-GPU node (NVIDIA figure, measured on Hopper; older GPUs may differ).
3. Extended memtest: 1 hour in NVIDIA's example; budget up to 8 hours if you run it one GPU at a time (assumption).
4. Sustained burn-in: 48 hours (our policy assumption, below).
5. Run the nodes in parallel. Wall clock is about $0.25 + 2.25 + 8 + 48 \approx 58.5$ hours, or roughly 2.5 days, before a reset and re-snapshot. Your return window must comfortably exceed this.

Link and fabric checks

From `nvidia-smi -q`, compare **GPU Link information** Current against Max. Current "may be reduced when the GPU is not in use", so read it under load. Also record **Replays Since Reset** and **Replay Number Rollovers**. A rollover "occurs after 4 consecutive replays and results in retraining the link" ([NVIDIA nvidia-smi documentation](#)). In DCGM, `targeted_stress` fails if PCIe replays exceed `max_pcie_replays` (160 by default) ([NVIDIA DCGM Diagnostics](#)).

For NVLink systems, use `nvidia-smi nvlink` ("Use `nvidia-smi nvlink -h` for more information") and check the fabric **Health** fields, such as Bandwidth "<Degraded/Full>" ([NVIDIA nvidia-smi documentation](#)). DCGM's `pcie`, `nvbandwidth` and `nccl_tests` plugins exercise the links at r2 and r3. On used HGX systems, a link fault may be the baseboard, not the module; retest in a known-good baseboard.

Thermal, power and burn-in plan

During every stress phase, watch **Clocks Event Reasons** in `nvidia-smi -q`. If all reasons are "Not Active", clocks are "running as high as possible". Watch **HW Thermal Slowdown**, **SW Thermal Slowdown** and **HW Power Brake**; a card that throttles where its batch-mates do not may have degraded thermal interface material. Meta's interruption table includes a "GPU Thermal Interface + Sensor" category with 6 interruptions (1.4%) ([Llama Team, AI @ Meta, 2024](#)).

Suggested plan (our policy assumption, not an NVIDIA requirement):

1. Day 0: full `nvidia-smi -q -x` snapshot per GPU, `dcgmi diag -r 1`, then `-r 4`.
2. 48 hours of your real workload, or a sustained GEMM load at production power limits, with 10-second telemetry on power, temperatures, clocks event reasons and Xid logs.

3. Reset the GPUs, take a second snapshot, and diff it against Day 0.
4. Reject any card with new uncorrectable remaps, Pending or Remapping Failure Occurred set, Xid 48, 64, 94 or 95, or thermal throttling at ambient conditions where its peers do not throttle.

For Xid interpretation, use our [ECC and Xid triage](#) decision tree.

Silent corruption screening

Silent data corruption (SDC) is wrong output with no error raised, so ECC and diagnostics miss it. The best fleet evidence concerns CPUs: Meta's tests across hundreds of thousands of machines over more than 18 months found hundreds of CPUs producing SDCs (Dixit et al., 2021). For GPUs, Meta's Llama 3 table attributes 6 of 419 unexpected interruptions (1.4%) to "Silent Data Corruption" in the GPU category (Llama Team, AI @ Meta, 2024).

A practical screen: run a fixed-seed deterministic workload (a large GEMM with a known checksum) many times on every GPU under thermal load and compare outputs bit for bit across GPUs and repeats. A card that disagrees is your suspect. This cannot prove the absence of SDC, but it raises the odds of catching a bad unit early.

Contract terms and red flags

Term	What to ask for
Serial schedule	Invoice lists every board serial; the label and <code>nvidia-smi</code> must match it
Return window	Longer than your full acceptance and burn-in (about 2.5 days in the example above) plus shipping
DOA definition	Objective: Remapping Failure Occurred, Pending that does not clear after reset, SRAM threshold exceeded, InfoROM checksum invalid, DCGM r3/r4 fail, serial mismatch
Payment	Escrow or staged payment released on acceptance
Firmware	Seller states whether VBIOS or InfoROM were reflashed; record versions on receipt
Warranty	Written OEM transfer confirmation if warranty is priced in
Compliance	Seller representation on export and title chain

Red flags: no `nvidia-smi -q` dumps before shipment, ECC disabled on arrival, "tested" with no logs, and all-zero aggregate ECC counts on a pre-Ampere fleet-retired card.

Printable checklist

#	Check	Command or source	Go if
1	Serial match	<code>nvidia-smi -q</code> : Serial Number	Label = <code>nvidia-smi</code> = invoice

#	Check	Command or source	Go if
2	Architecture support	CUDA 13.0 notes; driver branch	Turing or newer, or you accept CUDA 12.x/R580
3	ECC enabled	-d ECC: ECC Mode	Current: Enabled
4	Row remapper (Ampere+)	-d ROW_REMAPPER	Pending: No; Remapping Failure Occurred: No; no banks Low/None
5	Page retirement (pre-Ampere)	-d PAGE_RETIREMENT	Pending: No; well below 60 retired pages
6	SRAM	-d ECC	Threshold flag not set
7	InfoROM integrity	--query-gpu=inforom.checksum_validation	valid
8	Recovery action	nvidia-smi -q	None
9	Usage hours (if exposed)	--query-gpu=bbx.time_run	Consistent with seller's claim
10	Link	GPU Link information; replays	Current = Max under load; replays stable
11	DCGM	dcgmi diag -r 4	Pass on every GPU
12	Burn-in	48 h load + re-snapshot	No new UCE remaps, no Xid 48/64/94/95, no isolated throttling
13	SDC screen	Deterministic repeat runs	Bit-identical outputs
14	Paperwork	Warranty transfer, export, title	In writing before final payment

GPU Smith helps teams design acceptance plans for new and secondary-market GPU hardware ([contact](#)).

Sources

1. nvidia-smi documentation. NVIDIA. <https://docs.nvidia.com/deploy/nvidia-smi/index.html>
2. DCGM Diagnostics. NVIDIA. Last updated 2026-06-17. <https://docs.nvidia.com/datacenter/dcgmlatest/user-guide/dcgml-diagnostics.html>
3. NVIDIA GPU Memory Error Management: Supported GPUs. NVIDIA. Updated 2026-09-09. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/latest/supported-gpus.html>
4. NVIDIA GPU Memory Error Management: Row Remapping. NVIDIA. Updated 2026-09-09. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/latest/row-remapping.html>
5. NVIDIA GPU Memory Error Management: User Visible Statistics. NVIDIA. Updated 2026-09-09. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/latest/user-visible-statistics.html>
6. NVIDIA GPU Memory Error Management: RMA Policy. NVIDIA. Updated 2026-09-09. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/rma-policy-thresholds-for-row-remapping.html>

7. NVIDIA GPU Memory Error Management: SRAM Uncorrectable Errors. NVIDIA. Updated 2026-09-09. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/sram-uncorrectable-errors.html>
8. Dynamic Page Retirement: FAQ. NVIDIA. Updated 2026-09-09. <https://docs.nvidia.com/deploy/dynamic-page-retirement/faq.html>
9. CUDA Toolkit 13.0 Release Notes. NVIDIA. <https://docs.nvidia.com/cuda/archive/13.0.0/cuda-toolkit-release-notes/index.html>
10. CUDA Toolkit 13.4 Update 2 Release Notes. NVIDIA. <https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>
11. CUDA GPUs: Compute Capability. NVIDIA. <https://developer.nvidia.com/cuda/gpus>
12. Unix graphics feature deprecation schedule. NVIDIA Developer Forums. Updated 2025-07-01. <https://forums.developer.nvidia.com/t/unix-graphics-feature-deprecation-schedule/60588>
13. Data Center GPU Driver 580.196.07 Release Notes. NVIDIA. 2026-10-21. <https://docs.nvidia.com/datacenter/tesla/tesla-release-notes-580-196-07/index.html>
14. Data Center GPU Driver 615.71.10 Release Notes. NVIDIA. Last updated 2026-10-19. <https://docs.nvidia.com/datacenter/tesla/tesla-release-notes-615-71-10/index.html>
15. Ownership Transfer and Dell Product Registration Guides (KB 000125103). Dell. Modified 2026-06-10. <https://www.dell.com/support/kbdoc/en-us/000125103>
16. The Llama 3 Herd of Models. Llama Team, AI @ Meta. arXiv 2407.21783, 2024. <https://arxiv.org/abs/2407.21783>
17. Silent Data Corruptions at Scale. H. D. Dixit et al. arXiv 2102.11245, 2021-02-22. <https://arxiv.org/abs/2102.11245>
18. 15 CFR 734.16, Transfer (in-country). eCFR, current to 2026-10-27. <https://www.ecfr.gov/current/title-15/subtitle-B/chapter-VII/subchapter-C/part-734/section-734.16>

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. <https://docs.nvidia.com/deploy/nvidia-smi/index.html>
2. <https://arxiv.org/abs/2407.21783>
3. <https://www.dell.com/support/kbdoc/en-us/000125103>
4. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/rma-policy-thresholds-for-row-remapping.html>
5. <https://docs.nvidia.com/datacenter/dcgm/latest/user-guide/dcgm-diagnostics.html>
6. <https://www.ecfr.gov/current/title-15/subtitle-B/chapter-VII/subchapter-C/part-734/section-734.16>
7. <https://docs.nvidia.com/cuda/archive/13.0.0/cuda-toolkit-release-notes/index.html>
8. <https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>
9. <https://forums.developer.nvidia.com/t/unix-graphics-feature-deprecation-schedule/60588>
10. <https://docs.nvidia.com/datacenter/tesla/tesla-release-notes-580-196-07/index.html>
11. <https://docs.nvidia.com/datacenter/tesla/tesla-release-notes-615-71-10/index.html>
12. <https://developer.nvidia.com/cuda/gpus>
13. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/latest/supported-gpus.html>

14. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/latest/row-remapping.html>
15. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/sram-uncorrectable-errors.html>
16. <https://docs.nvidia.com/deploy/dynamic-page-retirement/faq.html>
17. <https://docs.nvidia.com/deploy/a100-gpu-mem-error-mgmt/latest/user-visible-statistics.html>
18. <https://arxiv.org/abs/2102.11245>

THE PUBLISHER

About GPU Smith

GPU Smith provides independent engineering and research for private AI infrastructure. We help technical buyers and operators reason about GPU workload sizing, hardware selection, procurement, deployment and inference operations, from the compute system to the networking, power and cooling requirements around it.

Start with the workload

Useful infrastructure decisions begin with the models, data, throughput, latency and operating constraints a team actually has. GPU Smith helps connect those requirements with system design and deployment choices. The goal is a privately controlled AI environment whose capacity and operational demands are understood before a purchasing decision.

Hardware and supplier research

Our public reference library covers GPUs, complete systems and networking components. The server-vendor directory and manufacturing research help teams investigate suppliers, compare options and follow sources behind technical claims. We distinguish manufacturer specifications from measured benchmarks and advertised capabilities from tested configurations. Reference pages are research resources, not a live inventory listing or a binding equipment quote.

Deployment and operations

GPU Smith's engineering scope includes infrastructure integration, networking, power, cooling and the practical operation of inference workloads. Our published articles explain the tradeoffs behind deployment, procurement and ongoing operations so teams can ask better questions and document decisions.

Work with GPU Smith

Explore the [hardware library](#), [GPU references](#), [networking references](#) and [vendor research](#). [Contact GPU Smith](#) with your workloads and deployment constraints to discuss a project and confirm current scope, pricing and availability.

Inclusion of a manufacturer or product in our research does not imply a partnership, certification or endorsement.

Website: <https://gpusmith.com>

This article is educational. Verify consequential medical, legal, financial, regulatory and technical decisions with appropriate professionals and the cited primary sources. Company services are described separately from the article's research findings.