GPU SMITH
COMPUTE ENGINEERING

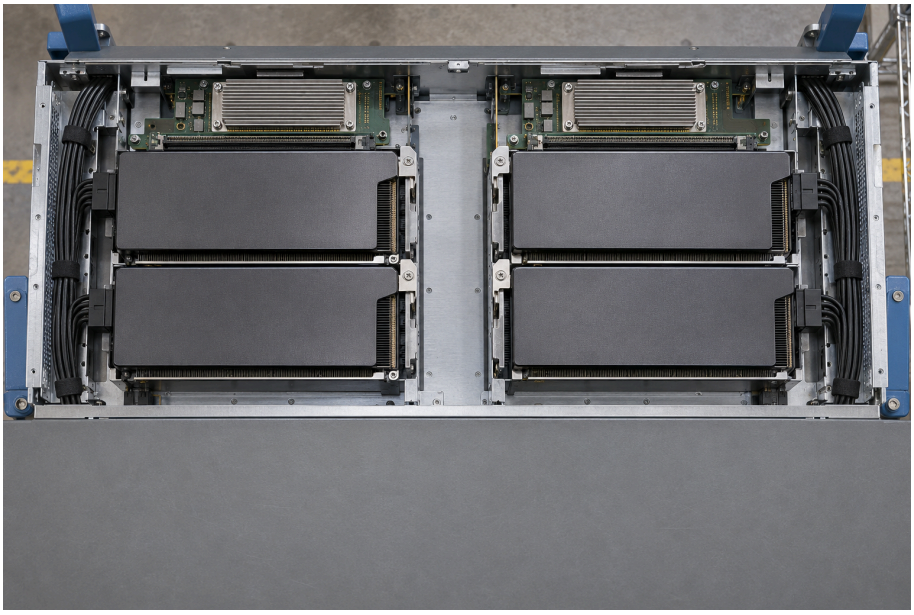
GPU SMITH / RESEARCH & PRACTICAL GUIDANCE

Multi GPU PCIe Peer Access: Purchase Evidence Matrix

> 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 · 2026-11-20 · multi gpu pcie peer access · pcie peer to peer gpu · cudaDeviceCanAccessPeer · nvidia-smi topo -p2p · tensor parallel pcie · nccl p2p · gpu server procurement · rtx pro 6000 blackwell · llm inference without nvlink · multi gpu inference

Original article: https://gpusmith.com/articles/en/multi-gpu-pcie-peer-access-purchase-evidence-matrix



In this article

1. Executive Summary
2. Introduction and Background
3. Path Terms and the Four Claims
4. Hardware and Platform Evidence Matrix
5. Software Paths and Fallbacks
6. Acceptance Evidence and Vendor Schedule
7. Data Analysis and Evidence
8. Implications and Future Directions
9. Frequently Asked Questions (FAQs)
10. Conclusion

Executive Summary

A quoted multi-GPU PCIe server that "supports peer access" bundles four separate claims: advertised hardware eligibility, runtime directional capability, observed transport and correctness, and useful application performance. NVIDIA's CUDA programming guide states that, depending on the system properties, specifically the PCIe and/or NVLink topology, devices can address each other's memory (docs.nvidia.com). Access must be enabled explicitly between two devices, and the runtime reference states that the access granted by an enable call is unidirectional (docs.nvidia.com). Once enabled, copies no longer need to be staged through the host (docs.nvidia.com). As of November 2026 the guide's footer reads "Last updated on Oct 31, 2026" (docs.nvidia.com), so buyers should record the revision they actually read.

The vendor evidence is thinner than most quotations imply. NVIDIA's RTX PRO 6000 Blackwell Server Edition page states PCIe Gen 5 support (www.nvidia.com), and a Lenovo platform guide says no NVLink bridge is applicable to that card (lenovopress.lenovo.com), but the pages read for this report did not state a directional P2P guarantee for any specific server bill of materials. Those cells stay **unknown** in the matrix below. For GeForce cards, an NVIDIA developer forum post relays engineering feedback that peer to peer is not supported on the RTX 4090 (forums.developer.nvidia.com) (a February 2023 community-hosted statement, not a current support certificate), and llama.cpp's guide says P2P requires driver support that is usually restricted to workstation and datacenter GPUs (raw.githubusercontent.com).

Platform settings decide whether a capable pair stays capable. NVIDIA documents that the IOMMU must be disabled on bare-metal Linux for PCIe P2P (docs.nvidia.com), and that ACS on PCI switches needs to be disabled (docs.nvidia.com), while virtual machines require ACS. Those requirements pull in different directions, so platform and security owners must sign the isolation posture; this report does not recommend disabling isolation globally.

Measured benefit is workload specific. A 2026 preprint on an eight-GPU RTX 5090 server reports one-way bandwidth rising from 31.5 GB/s to 56 GB/s once P2P was enabled with a modified driver (arxiv.org), while a self-reported vLLM forum experiment on two RTX 5090 cards describes itself as an exploratory point-estimate matrix, not a multi-rep confidence interval (discuss.vllm.ai). The preprint's authors state that they used a modified version of the open-source NVIDIA driver and disabled IOMMU and PCIe ACS, among other platform

changes, to enable P2P, so the gain depended on those changes. Modifying drivers or deploying GeForce cards in a data center may carry licence and warranty implications, so confirm them with the vendor and legal counsel before buying. A faster copy path is a capability result, not a latency, throughput or purchase-count result.

The recommended posture uses five dispositions: **supported-and-proven**, **supported-but-unproven**, **observed-community-only**, **unsupported**, and **unknown**. Each needs a named support owner, dated evidence, a host-staged control run, and an engine log showing which transport actually ran. If a service-level objective is met by independent replicas, the purchase may not need peer access at all.

Introduction and Background

Enterprise AI platform leads, integrators and procurement engineers who buy two to eight PCIe GPUs for private inference face a narrow but expensive question: does the quoted configuration support the direct GPU-to-GPU path that the tensor-parallel or pipeline-parallel workload needs, and if so, does that path change how many GPUs must be purchased? The answer depends on the exact GPU board, server, PCIe switch or riser, firmware, driver, operating system, virtualization layer and serving engine. No generation-wide yes or no cell survives that list.

GPU Smith is an independent private AI infrastructure engineering consultancy that advises on GPU procurement, workload sizing, networking, inference optimization, power and cooling, and air-gapped deployments. It is not a chip manufacturer, server vendor or cloud provider, and it sells none of the products discussed here. No hardware tests were performed in preparing this report. Every measurement quoted below belongs to a named third-party source with its own hardware assumptions.

The reusable pattern in hardware-selection literature is explicit assumptions plus an explanation of which architectural constraint changes the conclusion. Tim Dettmers' widely read hardware guide is a dated decision aid: its version history lists 2023-01-30 as its latest entry (timdettmers.com). The NeurIPS 2024 SpecExec paper cites it as one basis for defining a popular consumer hardware configuration (papers.neurips.cc), and a January 2024 CNAS policy report cites it for the price comparison between gaming and data center GPUs (s3.us-east-1.amazonaws.com). These are citations of a hardware-selection reference, not endorsements of a measured P2P result, and they are not current support certification. The guide's general statement that PCIe lanes are seldom a bottleneck (timdettmers.com) is the kind of dated generalization a 2026 purchase should replace with a configuration-specific test.

Server-class and consumer-class parts sit under different support statements. NVIDIA's RTX PRO reference architecture describes systems in two, four and eight GPU configurations (docs.nvidia.com), while an October 2026 preprint on commodity-GPU mixture-of-experts inference states that consumer GPUs connect via standard PCIe buses and lack P2P support, and designs around host-mediated transfers (arxiv.org). The buyer's job is to attach the correct statement to the correct line item. The sections below define path terms, assemble a vendor-evidence matrix, map engines to fallbacks, set an acceptance schedule, and quantify what published numbers can and cannot support.

Path Terms and the Four Claims

Path vocabulary

Procurement documents mix terms that describe different mechanisms. These definitions keep a contract from accepting evidence for one path as evidence for another.

- **GPU-to-GPU peer copy and load/store.** One device reads or writes another device's memory over PCIe or NVLink; enabled peers share a unified virtual address space.
- **GPU-to-NIC RDMA.** A different path between a GPU and a network adapter. NVIDIA's GPUDirect RDMA documentation requires that the two devices share the same upstream PCI Express root complex (docs.nvidia.com). A passing GPU-NIC test says nothing about GPU-to-GPU access.
- **NVLink.** A direct GPU interconnect, shown in the nvidia-smi legend as a bonded set of NVLinks. Some PCIe cards use an optional bridge, covered in Table 1.
- **Host staging.** Data moves from one GPU into host memory and out to the other. Direct peer copies avoid this staging, which makes it the control path a buyer should always measure.
- **CUDA IPC and virtual memory management.** Processes exchange allocation handles, and the virtual memory management API requires access control to be selected explicitly with cuMemSetAccess.
- **Engine collectives.** NCCL, a custom all-reduce kernel or an engine's own copy path. NCCL favors peer-to-peer transport when CUDA reports that GPUs can access each other's memory (docs.nvidia.com), but other engines make their own decisions.

The four claims

The first claim is **advertised eligibility**: a datasheet, quote or signed support letter says the configuration qualifies. The second is **runtime directional capability**: `cudaDeviceCanAccessPeer` queries whether a device may directly access a peer device (docs.nvidia.com). Because the enable call grants access that is unidirectional (docs.nvidia.com), the capability ledger needs one row per ordered pair. The runtime reference also notes device-wide and system-wide limitations per system configuration (docs.nvidia.com), and the programming guide's per-device peer-connection limit on non-NVSwitch systems is one more item to confirm for eight-GPU boxes.

The third claim is **observed transport and correctness**. NCCL's logging documentation shows peer transport as `via P2P/CUMEM` (docs.nvidia.com) and shared-memory staging as `via SHM/direct/direct` (docs.nvidia.com). vLLM's utilities state that a real P2P access must be performed because the framework-level capability query can report availability when access is not actually possible (docs.vllm.ai). A positive query is an input to acceptance, not the acceptance itself. The fourth claim is **useful application performance**: repeated matched runs of the actual workload show a latency or capacity change that affects the number of GPUs or groups purchased.

What does not establish support

- **An x16 slot.** The RTX PRO 6000 Blackwell Server Edition datasheet lists a PCI Express 5.0 x16 graphics bus (www.akamai.com); the same text is silent on peer access.

- **A topology listing.** The legend defines PHB as a connection traversing PCIe and a host bridge, typically the CPU (docs.nvidia.com). A community guide notes that PHB is not a dead end for P2P (raw.githubusercontent.com), so the label alone decides nothing either way.
- **A status matrix entry.** `nvidia-smi topo -p2p` shows the P2P status between all GPUs, given a capability (docs.nvidia.com). A forum thread lists a status code reading "CNS = Chipset not supported" (forums.developer.nvidia.com), which shows the matrix can report an unsupported platform even where each card is individually capable. The manual page read for this report does not define every status code, so contracts should cite the driver release notes for the exact version.
- **Aggregate VRAM.** TensorRT-LLM describes tensor parallelism as splitting the model weights across multiple GPUs (nvidia.github.io). Capacity is shared as shards, and the sources reviewed do not describe pooled memory as a feature of peer access.

Hardware and Platform Evidence Matrix

Table 1 below records what vendor pages say about interconnect and peer access for representative PCIe GPUs as of November 2026. Where a page does not address P2P over PCIe, the cell reads **unknown**, meaning no statement was located, not that support is doubtful, and the buyer must obtain a statement for the exact server. NVIDIA's CUDA guide treats device-to-device memory access as dependent on system properties, specifically the PCIe and/or NVLink topology, so PCIe P2P on data-center and workstation boards is a topology- and platform-dependent question rather than one answered by the GPU model alone.

Configuration	Interconnect stated by the vendor page	Peer-access statement located	Disposition for a quote
RTX PRO 6000 Blackwell Server Edition	PCIe Gen 5 (www.nvidia.com); a Lenovo guide says no NVLink bridge is applicable (lenovopress.lenovo.com)	None found on the product page or datasheet text	**Unknown** until the OEM states it for the exact server
RTX PRO 6000 Blackwell Workstation Edition	PCIe Gen 5 (www.nvidia.com); 96 GB GDDR7 with ECC on the family page (www.nvidia.com)	None found	**Unknown**
RTX 6000 Ada Generation	48 GB GDDR6 with ECC (www.nvidia.com)	None found on the page	**Unknown**
L40S	PCIe Gen4 x16 at 64 GB/s, labelled bidirectional (www.nvidia.com)	None found	**Unknown**
H200 NVL	2- or 4-way NVLink bridge (www.nvidia.com); page marks specifications preliminary (www.nvidia.com)	NVLink bridge is the stated GPU-to-GPU path	Bridged pairs: supported-but-unproven; PCIe-only pairs unknown
H100 NVL (PCIe)	PCIe-based card with NVLink bridge (www.nvidia.com)	NVLink bridge is the stated GPU-to-GPU path	Bridged pairs: supported-but-unproven

Configuration	Interconnect stated by the vendor page	Peer-access statement located	Disposition for a quote
A100 PCIe	NVLink bridge for up to two GPUs (www.nvidia.com)	Bridge for pairs only	Pairs: supported-but-unproven; other pairs unknown
GeForce RTX 4090	Not stated on a vendor product page read for this report	A February 2023 forum post relays Engineering feedback that peer to peer is not supported on the 4090 (forums.developer.nvidia.com)	**Unsupported** per that dated statement; recheck the current driver release

Three cautions apply. Product pages are summaries, and an acceptance decision needs the server vendor's statement. The H200 page labels its own table preliminary (www.nvidia.com), so a quote relying on it should name the revision. And one forum participant observes that NVIDIA does not publicly document the internal workings of the peer-capability API call (forums.developer.nvidia.com). That is a user's characterization, not a vendor statement, but it matches the gap seen here: no per-SKU support table was located, so the buyer needs the statement in writing for the quoted bill of materials.

Server and platform pages

Platform documentation supplies topology but not support. NVIDIA's reference architecture recommends a balanced PCIe topology with one Gen5 x16 link per GPU. It states that a PCIe switch may not be needed for low-cost inference servers (docs.nvidia.com), while an NVIDIA developer blog calls PCIe switches critical for maximizing inter-GPU communication bandwidth (developer.nvidia.com). The two statements are compatible only once the buyer states which workload the server is for.

OEM pages describe layout rather than a guarantee. A Supermicro brief explains that a dual-root setup connects each CPU to several GPUs through a PLX switch (supermicro.com), and lists single-root use when peer-to-peer communication (GPU to GPU) performance is not critical (supermicro.com). A 2026 Supermicro datasheet lists up to eight RTX PRO 6000 Blackwell GPUs (www.supermicro.com). Dell's PowerEdge XE7745 page lists eight PCIe Gen 5 x16 double-wide slots at up to 600 W (www.dell.com), and ASUS describes built-in PCIe Gen 6 switches on an RTX PRO server model (servers.asus.com). In the page text read for this report none of these OEM pages stated a directional GPU-to-GPU guarantee for a named GPU, and NVIDIA's datasheet says only that the GPU is available in NVIDIA-Certified servers (www.akamai.com). A certification listing is not a P2P statement.

The practical consequence is a request, not an inference: ask the integrator to certify in the quote that each ordered GPU pair supports direct access on the delivered firmware, driver and operating system, and to name who owns the fix if it does not.

Software Paths and Fallbacks

Engines decide what the hardware capability is used for. *Table 2* below consolidates documented behavior for engines commonly used in private inference. It is a test-planning aid, not a performance ranking.

Engine and mode	Parallelism and path documented	Fallback or control documented	What to log for acceptance
vLLM, tensor parallel	Distributed inference is probably unnecessary if the model fits on a single GPU (docs.vllm.ai); for nodes without NVLink the docs name L40S as an example where pipeline parallelism is preferred (docs.vllm.ai)	Custom all-reduce can be disabled to fall back to NCCL (docs.vllm.ai); the source disables it when the platform lacks GPU P2P capability or the P2P test failed (docs.vllm.ai)	Chosen all-reduce implementation, NCCL transport lines, vLLM and driver versions
vLLM, data parallel	Model weights are replicated across separate instances to process independent batches of requests (docs.vllm.ai)	No peer path needed between replicas	Qualified requests per second per replica
SGLang	Data-parallel replica count is configurable (docs.sglang.io)	Custom all-reduce can be disabled to fall back to NCCL (docs.sglang.io)	Selected all-reduce, NCCL log lines
llama.cpp	Default split mode is layer, pipelined (github.com); a tensor mode is marked experimental (github.com)	An environment variable enables peer-to-peer access between GPUs (github.com); the build guide documents a caution for some motherboard and BIOS settings, naming IOMMU as an example (github.com)	Whether the P2P variable was set, split mode, build flags
TensorRT-LLM	Tensor parallelism splits weights across GPUs (nvidia.github.io)	A PCIe-targeted all-reduce optimization exists (nvidia.github.io), with automatic fallback when no benefit is expected (nvidia.github.io)	Strategy selected at run time, message sizes

Several details change how a buyer should read the table. vLLM's environment-variable documentation says its P2P check assumes drivers can report P2P status correctly (docs.vllm.ai), so a driver that over-reports capability can still flow into engine selection. vLLM also documents a FlashInfer-based backend that targets small tensor-parallel all-reduces on a single PCIe-only node (docs.vllm.ai), so "PCIe-only" and "no optimized path" are not synonyms.

llama.cpp's multi-GPU guide summarizes the trade-off as pipeline-parallel maximizing batch throughput while tensor-parallel minimizes latency (raw.githubusercontent.com). One community guide states that setting `NCCL_P2P_DISABLE` with llama.cpp's layer split does nothing (raw.githubusercontent.com), a caution against assuming that a library-level switch governs every engine. NVIDIA's NCCL documentation describes `NCCL_P2P_DISABLE` as disabling direct GPU-to-GPU communication (docs.nvidia.com) and warns that its tuning variables should generally not be set for production code (docs.nvidia.com). The accepted configuration should therefore list which variables, if any, are set, and require a retest when they change.

The decision for each cell concerns the fallback, not the headline. If a custom kernel disables itself, the engine falls back to NCCL, and NCCL can in turn use shared memory when P2P is unavailable. Each hop is legitimate behavior with different throughput, so the acceptance log must show which one ran.

Acceptance Evidence and Vendor Schedule

The directional device-pair ledger

The acceptance artifact is a ledger with one row per ordered GPU pair: twelve rows for four GPUs and fifty-six for eight. Each row should carry:

- **Hardware identity.** Exact GPU SKU and board revision, serial or UUID, PCI bus/device/function address, server model, riser or backplane, and PCIe switch model if any.
- **Firmware and platform.** BIOS and BMC versions, IOMMU state, ACS state on each bridge, Above 4G decoding and BAR sizing, NUMA mapping, and any hypervisor.
- **Software provenance.** Operating system, kernel, driver version and kernel-module provenance (stock, vendor-supplied or locally modified), CUDA, NCCL and engine versions, and container image digest.
- **Runtime query result.** The `cudaDeviceCanAccessPeer` result in the row's direction, link attributes such as the performance rank, and the date.
- **Observed transport.** The transport string from engine and NCCL logs for the workload actually used.
- **Correctness result.** Integrity checks on transferred data and on end-to-end model output.
- **Vendor statement.** Reference to the signed support letter, with scope, date and owner.

The ledger separates the query from the observation. A community maintainer frames the question as three independent questions stacked on top of each other (raw.githubusercontent.com), which maps to query, driver or kernel-module path, and engine behavior.

Acceptance tests

NVIDIA's `nvbandwidth` is a tool for bandwidth measurements on NVIDIA GPUs (github.com). It implements copy-engine copies, which use memcpy APIs, and streaming-multiprocessor copies, which use kernels (github.com). Its bidirectional tests drive traffic in both directions concurrently (github.com), and a pair option restricts device-to-device tests to one explicitly specified pair of GPUs (github.com). The README lists a default memcpy buffer of 512 MiB (raw.githubusercontent.com): note the binary MiB buffer unit against decimal GB/s in output, and fix both in the acceptance script. A pass establishes that pair on that system; it does not certify every board.

For collectives, the `nccl-tests` documentation defines bus bandwidth so that it can be compared with the hardware peak bandwidth independently of the number of ranks used (github.com). Report algorithm bandwidth and bus bandwidth separately and compare each against the host-staged control.

The recommended sequence, none of which is an installation procedure:

1. **Pair query in both directions.** Record each ordered pair's result with driver and CUDA versions.
2. **Read, write and integrity transfers.** Use the allocation types the application uses, in buffer sizes that bracket its message sizes.
3. **Bidirectional and concurrent transfers.** Run both directions at once, then every pair the application uses simultaneously, since contention is where single-pair tests mislead.
4. **Host-staged control.** Repeat with the peer path disabled by the documented means for the layer under test, and archive both runs.

5. **Engine transport capture.** Start the real engine with logging that prints transport per channel, and save the lines.
6. **End-to-end correctness.** Compare model outputs on a fixed prompt set with a reference configuration.

A community guide advises running a bandwidth test on any new multi-GPU layout before trusting the topology matrix (raw.githubusercontent.com), and states that a passing transfer test is necessary but not sufficient (raw.githubusercontent.com). A pass is one row of evidence.

Isolation, IOMMU and ACS

The documented requirements are not uniform, which is why the security owner must sign. NVIDIA's CUDA guide states that the IOMMU must be disabled when running Linux on a bare metal system (docs.nvidia.com). For virtual machines the same guide reverses the advice: the IOMMU should be enabled and the VFIO driver used (docs.nvidia.com). NCCL's troubleshooting page says that if PCI switches have ACS enabled it needs to be disabled (docs.nvidia.com), and that virtual machines require ACS, so disabling it there is not an option (docs.nvidia.com). NVIDIA's virtual GPU documentation states that P2P transfers over PCIe are not supported (docs.nvidia.com) on that feature page.

The Linux kernel documentation explains the mechanism: with PCIe switches, ACS settings determine whether a transaction can route entirely within the PCIe hierarchy (docs.kernel.org), and one reaching the host bridge might have to hairpin back out the same root port (docs.kernel.org). Forcing ACS redirect off removes isolation between devices and may put more devices in an IOMMU group (docs.kernel.org).

The procurement reading is conservative: the schedule should state the isolation posture the vendor tested, record who approved it, and avoid a blanket instruction to turn isolation off. Shared multi-tenant or virtualized designs may need a different server or independent replicas rather than a peer path.

Support ownership and dispositions

Stock behavior and community modification must stay in separate rows. One community guide states that enabling P2P on consumer cards requires a patched kernel module (raw.githubusercontent.com), and a vLLM forum experiment on two consumer cards used a patched open NVIDIA kernel module so that BAR1 GPU P2P was available (discuss.vllm.ai). These report what skilled users achieved on their own hardware. They are not vendor-supported operation, and a contract that depends on them has no support owner. The correct label is **observed-community-only**. *Table 3* below converts the evidence into a disposition vocabulary for the quote and the acceptance schedule.

Disposition	Minimum evidence	Acceptance deliverables	Retest triggers
Supported-and-proven	Signed vendor statement naming exact SKU, server, firmware, driver and OS, plus a passing ledger	Ledger for every ordered pair, host-staged control, engine log, correctness report, test code and configuration hashes	Driver, firmware, kernel, engine, NCCL or BIOS change
Supported-but-unproven	Signed statement without the buyer's test run	Same tests run before payment release	As above, plus any workload change
Observed-community-only	A working result on similar hardware, no vendor statement	Written acceptance that no vendor owns the path; fallback plan to host-staged or replicas	Every module or engine update

Disposition	Minimum evidence	Acceptance deliverables	Retest triggers
Unsupported	Vendor statement that the pair or class is not supported	Design for host-staged operation or independent replicas	New vendor statement
Unknown	No statement located	Written request; keep out of the performance plan	On receipt of a statement

Only the first two rows justify designing a cluster around direct access. The rest are fallbacks with labelled risk, and the alternates should be decided in advance: a different server with a vendor-documented switch topology, a different GPU class, independent replicas, or a measured host-staged configuration accepted on its own latency and throughput numbers.

Matched-run workload template

A capability matters only if it changes a required service level. The workload test should repeat realistic prefill and decode requests at the buyer's concurrency and context lengths, alternate configurations (peer path on, host-staged control) in randomized order, discard a warm-up, and run enough repetitions to report a mean and an interval for each metric. Required fields:

- **Workload definition.** Model, precision, context length, concurrency, request mix, quality metric and the latency percentile named in the service-level objective.
- **Configuration identity.** Ledger row identifiers, engine flags and all environment variables.
- **Repetitions.** Independent runs per configuration and the interval statistic used.
- **Outcome.** Qualified requests per service group, not only average tokens per second.

The vLLM forum post is a useful model of disclosure. It states a limitation of one timed aggregate per cell with no CI (discuss.vllm.ai), says differences of a few percent may be run-to-run noise until multi-rep confirmation (discuss.vllm.ai), and verifies that the transport change was real rather than a silent no-op (discuss.vllm.ai). Demonstrating that the path changed and separately demonstrating that the application changed is the right order of evidence.

Data Analysis and Evidence

Link rates and direction conventions

PCIe numbers are published in different conventions, so every comparison must state direction and decimal units. A reprint of the PCI-SIG PCIe 5.0 release gives 32 GT/s and up to 128 GB/s via x16 (us.design-reuse.com), and an analyst quoted in the same reprint gives 64 GB/second for PCIe 5.0 (us.design-reuse.com); these are consistent only if 128 GB/s is the two-direction total. PCI-SIG's PCIe 6.0 release states 64 GT/s and up to 256 GB/s via x16 (www.businesswire.com) without a direction word, while its PCIe 7.0 release says 128.0 GT/s and up to 512 GB/s bi-directionally via x16 (www.businesswire.com). The PCIe 8.0 draft announcement states 256.0 GT/s and up to 1.0 TB/s bi-directionally via x16 (pcisig.com); as a draft it is not a purchasable generation.

Source discrepancy: NVIDIA's L40S page lists PCIe Gen4 x16 as 64 GB/s bidirectional (www.nvidia.com), whereas an RTX A5000 study describes approximately 32 GB/s on a sixteen-lane PCIe Gen 4 link and labels it bidirectional (arxiv.org). Both cannot describe the same quantity under the conventions above. Contracts should say "GB/s per direction" or "GB/s aggregate" explicitly, use decimal GB for link rates, and reserve GiB for buffer sizes. NVIDIA's NVLink page describes sixth-generation NVLink as 3 TB/s bidirectional (www.nvidia.com).

Published measurements and their limits

The following measurements illustrate magnitudes. Each is tied to named hardware, most are preprints, and none is a 2026 procurement benchmark for another configuration.

- **Host-staged versus P2P bandwidth.** On an eight-GPU RTX 5090 platform with a modified driver, a 2026 preprint reports one-way bandwidth improving from 31.5 GB/s to 56 GB/s (arxiv.org) and bidirectional bandwidth from 32 GB/s to 111 GB/s (arxiv.org). It also reports that communication latency decreased drastically from 14.3 microseconds (arxiv.org) to a sub-microsecond value.
- **Host-staged contention.** ThunderEP, a September 2026 preprint, states that neither of its systems supports GPUDirect P2P access between any pair of GPUs (arxiv.org) and that communication between GPUs must be staged through host memory (arxiv.org). A GPU communicating alone reaches 56.5 GB/s on PCIe Gen5 (arxiv.org), and each GPU's bandwidth falls to 35.5 GB/s (arxiv.org) when a second GPU under the same host bridge communicates.
- **NVLink versus PCIe at decode sizes.** A four-card RTX A5000 study reports that realized NVLink and PCIe bandwidths are close at its message sizes (arxiv.org), and that the fast bridge slows the collective by a factor of 1.73 at 512 KiB with the direct path off (arxiv.org). The authors say that putting all four cards on one shared fast link would require a switch fabric (arxiv.org), so these are measured points, not a full separation of effects.
- **Host-mediated design.** CoMoE states that all cross-GPU traffic must be host-mediated, bounced through host DRAM (arxiv.org), and reports performance approaching NVLink-capable A800 GPUs at only 23.4% of the hardware cost (arxiv.org) on a server with two Intel Xeon Platinum 8575C CPUs (arxiv.org). This is an author-reported result for a mixture-of-experts workload, not a statement about dense tensor-parallel models.
- **Overhead even on fast links.** TokenWeave reports that distributed inference of LLMs can incur overheads of up to 20% (arxiv.org), so communication cost is non-trivial even where a fast path exists.

Communication volume and a limited screen

Tensor parallelism generates frequent collectives. A characterization study states that each transformer layer generates two Allreduce operations (arxiv.org), the original Megatron-LM paper describes only two all-reduces in the forward path and two in the backward (arxiv.org), and a later Megatron paper notes that tensor parallelism is best within a node due to its expensive all-reduce communication (arxiv.org). The characterization study concludes that pipeline parallelism suits bandwidth-constrained environments (arxiv.org). That is why engine choice, and not only link rate, belongs in the purchase decision.

As an explicitly limited screen, serial communication time is the sum over messages of latency plus bytes divided by measured effective bandwidth:

T_comm = sum over i of (latency_i + bytes_i / effective_bandwidth_i)

This ignores overlap, contention and the collective algorithm, so it bounds one serial term and is not a model of total inference time. For all-reduce, nccl-tests multiplies algorithm bandwidth by 2(n-1)/n to obtain bus bandwidth, which is 1.0 for two GPUs, 1.5 for four and 1.75 for eight, so numbers from different group sizes are not interchangeable.

(Hypothetical Example) Assume a dense decoder with 80 layers, hidden size 8,192, 16-bit activations and a decode batch of 32 tokens. Each all-reduce moves 32 x 8,192 x 2 = 524,288 bytes, and with two all-reduces per layer a step has 160 messages. Suppose a host-staged configuration measured 15 microseconds latency and 10 GB/s effective bandwidth, and a direct path measured 5 microseconds and 25 GB/s (all four are assumed inputs, not measurements). The staged message costs 15 + 52.4 = 67.4 microseconds and the direct message 5 + 21.0 = 26.0 microseconds, giving 10.8 ms versus 4.2 ms of serial communication per step, a difference of 6.6 ms. Whether that matters depends on the measured step time and latency objective. If both configurations meet the objective, or replicas meet demand without tensor parallelism, the direct path releases no GPU. If only the direct path meets it, whole service groups can be sized from measured qualified requests per group: divide required qualified requests per second by the measured figure per group, round up, and compare both configurations on that integer. A percentage gain does not necessarily release a purchased GPU.

Implications and Future Directions

Procurement language should follow the dispositions. The RFP response should contain the ledger template, the vendor's signed statement per disposition, and the retest triggers. The vendor should state who owns a regression after a driver, BIOS or engine update; without that, a "supported" cell is an opinion.

Retest after every layer change. The peer-access API page and the programming guide carry separate revision dates. Documentation, drivers and engines change on different cadences, and vLLM's custom all-reduce behavior depends on the version in which its checks live (docs.vllm.ai). The acceptance record should name versions, not generations.

Interconnect generations are moving. PCI-SIG describes a tradition of doubling IO bandwidth every three years (www.businesswire.com). A faster link narrows the bandwidth gap between host-staged and direct paths but does not remove the extra copy or the contention found on shared host bridges, as the ThunderEP topology of GPUs distributed across three PCIe host bridges shows (arxiv.org). Designs that work around absent P2P, such as CoMoE, whose transfer kernel occupies SMs because it moves data with loads and stores (arxiv.org), suggest software mitigation will keep improving, which strengthens the case for measuring rather than assuming.

Isolation and performance need a joint owner. Because bare-metal and virtualized requirements diverge, one security policy cannot be inherited across both. The platform owner, security owner and model-serving owner should sign the same page.

Unknowns remain unknown. Vendor statements were not located for several SKUs, and no hardware tests were run. A table that fills those cells from a neighbouring generation would be less trustworthy than one that says so.

Frequently Asked Questions (FAQs)

Which consumer GPUs support P2P over PCIe, and what about the RTX 4090?

No current vendor support table was located for this report. One NVIDIA developer forum post relays Engineering feedback that peer to peer is not supported on the RTX 4090 and says drivers should not report such a configuration as peer capable (forums.developer.nvidia.com); the post dates from February 2023. Community reports of working peer transfers on consumer cards rely on patched kernel modules and belong in the observed-community-only row. Treat any consumer SKU as unknown or unsupported until the buyer holds a current vendor statement.

Why does cudaDeviceCanAccessPeer return false?

The call queries whether a device may directly access a peer device (docs.nvidia.com). Documented reasons for a negative or unusable result include topology and platform settings such as bare-metal IOMMU, which CUDA and the display driver do not support for PCIe peer-to-peer transfers (docs.nvidia.com). The query is directional, so check both orderings. The manual page read for this report does not list every cause, so the vendor's release notes are the record of reference.

How is peer access checked with nvidia-smi topo -p2p?

The manual describes the option, as discussed above, as showing P2P status between all GPUs for a given capability. Record the output with the driver version and treat it as one ledger input, because vLLM performs a real access test precisely since a framework-level query can disagree with reality (docs.vllm.ai).

Can multi-GPU inference run without NVLink over PCIe, and what setup is best for local inference?

Yes, but engine behavior changes. Independent replicas need no GPU-to-GPU path (docs.vllm.ai), and llama.cpp's default layer split is pipelined (github.com). The evidence reviewed supports no single best setup. The sound order is to size the model and context to one GPU first, since distributed inference is probably unnecessary if the model fits on a single GPU (docs.vllm.ai), use replicas when throughput is the aim, and use tensor or pipeline groups only when the model requires them, with the disposition table deciding which path to rely on.

How much does PCIe bandwidth affect tensor-parallel inference?

It depends on message size, group size and the collective. Published results range from near-parity at small decode sizes in one four-card study (arxiv.org) to large one-way bandwidth gains with P2P on one eight-GPU server (arxiv.org). Neither transfers to another server. TensorRT-LLM's documentation says its PCIe all-reduce optimization improves performance for large message sizes in PCIe-based topologies (nvidia.github.io), which marks message size as a deciding variable.

Is a PCIe switch needed?

NVIDIA's reference architecture says a switch may not be needed for low-cost inference servers, while NVIDIA's developer blog calls switches critical for inter-GPU bandwidth (developer.nvidia.com). The answer follows the workload: if tensor-parallel groups span many GPUs, ask the vendor for the switch topology and the ACS configuration it ships with.

Conclusion

Multi-GPU PCIe peer access is a chain of conditions, and a purchase should be accepted only when every link has an owner and a record. A product page, an x16 slot, a topology listing or a successful capability query supports one link at most. The vendor must state support for the exact configuration, the runtime must show directional capability, the engine log must show the transport that actually ran, and the application test must show that the change matters for the service level and the number of GPUs bought.

The disposition vocabulary of supported-and-proven, supported-but-unproven, observed-community-only, unsupported and unknown keeps those conditions visible in the contract. Dated documentation, preprints with explicit hardware assumptions and a clearly labelled forum experiment all inform the plan, and none replaces the acceptance ledger. Where evidence is missing, the right entry is unknown, and the right fallback is a measured host-staged configuration or independent replicas, accepted on their own numbers.

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/cuda/cuda-programming-guide/03-advanced/multi-gpu-systems.html
2. https://docs.nvidia.com/cuda/cuda-runtime-api/group__CUDART__PEER.html
3. https://www.nvidia.com/en-us/data-center/rtx-pro-6000-blackwell-server-edition
4. https://lenovopress.lenovo.com/lp2181-lenovo-hybrid-ai-285-platform-guide
5. https://forums.developer.nvidia.com/t/standard-nvidia-cuda-tests-fail-with-dual-rtx-4090-linux-box/233202/16
6. https://raw.githubusercontent.com/ggml-org/llama.cpp/HEAD/docs/multi-gpu.md
7. https://docs.nvidia.com/deeplearning/nccl/user-guide/docs/troubleshooting/gpu_troubleshooting.html
8. https://arxiv.org/pdf/2608.27370
9. https://discuss.vllm.ai/t/dual-rtx-5090-tp-2-shm-vs-patched-bar1-p2p-cumem-single-pass-2-7-mean-point-estimate/2870
10. https://timdettmers.com/2023/01/30/which-gpu-for-deep-learning
11. https://papers.neurips.cc/paper_files/paper/2024/file/1d91d5689e251d27993a3c2182dddcf7-Paper-Conference.pdf
12. https://s3.us-east-1.amazonaws.com/files.cnas.org/documents/CNAS-Report-Tech-Secure-Chips-Jan-24-finalb.pdf
13. https://docs.nvidia.com/enterprise-reference-architectures/rtx-pro-ai-factory/latest/components.html
14. https://arxiv.org/html/2610.09424v1
15. https://docs.nvidia.com/cuda/gpudirect-rdma/index.html
16. https://docs.nvidia.com/deeplearning/nccl/user-guide/docs/troubleshooting/logging.html
17. https://docs.vllm.ai/en/latest/api/vllm/distributed/device_communicators/all_reduce_utils

18. https://www.akamai.com/site/en/documents/corporate/nvidia-rtx-pro-6000-blackwell-server-edition-datasheet.pdf
19. https://docs.nvidia.com/deploy/nvidia-smi/index.html
20. https://raw.githubusercontent.com/noonghunna/club-3090/master/docs/PCIE_P2P.md
21. https://forums.developer.nvidia.com/t/what-is-the-meaning-of-cns-chipset-not-supported-error-in-nvidia-smi/68055
22. https://nvidia.github.io/TensorRT-LLM/latest/features/parallel-strategy.html
23. https://www.nvidia.com/en-us/products/workstations/professional-desktop-gpus/rtx-pro-6000
24. https://www.nvidia.com/en-us/products/workstations/professional-desktop-gpus/rtx-pro-6000-family
25. https://www.nvidia.com/en-us/products/workstations/rtx-6000
26. https://www.nvidia.com/en-us/data-center/l40s
27. https://www.nvidia.com/en-us/data-center/h200
28. https://www.nvidia.com/en-us/data-center/h100
29. https://www.nvidia.com/en-us/data-center/a100
30. https://forums.developer.nvidia.com/t/p2p-access-ada-gpus-with-pcie-switch/330994
31. https://developer.nvidia.com/blog?p=99991
32. https://supermicro.com/products/brief/product-brief-GPU-Servers-Root-Explained.pdf
33. https://www.supermicro.com/datasheet/datasheet_Supermicro_RTX_PRO_BSE.pdf
34. https://www.dell.com/en-us/shop/ipovw/poweredge-xe7745
35. https://servers.asus.com/products/detail/overview/ESC8000A-E13X
36. https://docs.vllm.ai/en/latest/serving/parallelism_scaling.html
37. https://docs.vllm.ai/en/latest/configuration/engine_args.html
38. https://docs.vllm.ai/en/v0.10.2/api/vllm/distributed/device_communicators/custom_all_reduce.html
39. https://docs.vllm.ai/en/latest/serving/data_parallel_deployment.html
40. https://docs.sglang.io/advanced_features/server_arguments.html
41. https://github.com/ggml-org/llama.cpp/blob/master/tools/server/README.md
42. https://github.com/ggml-org/llama.cpp/blob/master/docs/build.md
43. https://nvidia.github.io/TensorRT-LLM/legacy/advanced/lowprecision-pcie-allreduce.html
44. https://docs.vllm.ai/en/latest/configuration/env_vars.html
45. https://docs.vllm.ai/en/latest/api/vllm/distributed/device_communicators/flashinfer_pcie_ipc_all_reduce
46. https://docs.nvidia.com/deeplearning/nccl/user-guide/docs/env.html
47. https://github.com/NVIDIA/nvbandwidth
48. https://raw.githubusercontent.com/NVIDIA/nvbandwidth/main/README.md
49. https://github.com/NVIDIA/nccl-tests/blob/master/doc/PERFORMANCE.md
50. https://docs.nvidia.com/ai-enterprise/release-9/latest/infra-software/vgpu/features/p2p.html
51. https://docs.kernel.org/driver-api/pci/p2pdma.html
52. https://docs.kernel.org/admin-guide/kernel-parameters.html
53. https://us.design-reuse.com/news/46222/32gt-s-pci-express-5-0-specification.html

54. https://www.businesswire.com/news/home/20220111005011/en/PCI-SIG%C2%AE-Releases-PCIe%C2%AE-6.0-Specification-Delivering-Record-Performance-to-Power-Big-Data-Applications
55. https://www.businesswire.com/news/home/20250611299049/en/PCI-SIG-Releases-PCIe-7.0-Specification-to-Support-the-Bandwidth-Demands-of-Artificial-Intelligence-at-128.0-GTs-Transfer-Rates
56. https://pcisig.com/blog/pcier-80-specification-draft-05-now-available-track-2560-gts-transfer-speeds-2028
57. https://arxiv.org/pdf/2607.11368
58. https://www.nvidia.com/en-us/data-center/nvlink
59. https://arxiv.org/html/2609.40093
60. https://arxiv.org/html/2505.11329v5
61. https://arxiv.org/html/2507.14392v1
62. https://arxiv.org/pdf/1909.08053
63. https://arxiv.org/pdf/2104.04473

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.