VIABANDWIDTH BUYER GUIDE

Buying GPU Services

A practical guide to comparing public rates, quotes and infrastructure claims

A field manual for turning GPU labels and headline rates into a defensible, workload-matched purchase decision.

872	722	4,511	0
candidate domains reviewed	retained active candidates	scoped price representations	required follow-ups open

Independent market audit · first-party provider evidence · private directory protected

Contents

01 Start with the workload

02 Choose the right supply model

03 Make prices comparable

04 Interrogate the quote

05 Run a proof of value

06 Score the shortlist

07 Procurement worksheet

08 Red flags and reference

Page numbering is shown in the footer; section openings are deliberately paced as an editorial report.

01

Start with the workload

The cheapest GPU is the wrong question until the workload, data path and service boundary are defined.

Workload question	Why it changes the buy
Training, fine-tuning, inference, rendering, VDI or research?	Changes model choice, interconnect, persistence, scheduler and utilization pattern.
Single GPU, multi-GPU or multi-node?	Determines whether node topology and east-west fabric dominate.
Burst, steady or deadline-driven?	Changes the value of on-demand, reservation, spot and dedicated capacity.
Data size and sensitivity?	Changes storage, egress, sovereignty, private connectivity and security requirements.
Framework and image constraints?	Changes driver/CUDA compatibility, managed platform value and migration effort.
Human support tolerance?	Changes the acceptable balance between self-service price and managed delivery.

Write the requirement as a testable sentence

Template

We need [workload] on [GPU/model class] with [GPU count/topology], [CPU/RAM/storage/network], in [region], for [duration/utilization], with [security/support] and a successful run defined as [time, throughput, quality and cost].

Model selection is a systems decision

Memory capacity, memory bandwidth, supported precision, interconnect and software compatibility often matter more than the model name alone. A lower-priced accelerator that requires smaller batches, more data movement or engineering changes may cost more per completed job.

Need	Prioritize	Do not assume
Large-model training	VRAM, multi-GPU fabric, checkpoint throughput, stable reserved capacity	Every eight-GPU node has the same topology
Fine-tuning	Memory fit, storage locality, scheduling flexibility	Peak benchmark throughput predicts time-to-train
Online inference	Latency, autoscaling, cold starts, regional proximity, serving stack	Per-GPU-hour is the relevant unit

02

Choose the right supply model

Direct cloud, marketplace capacity, dedicated servers, colocation and managed platforms solve different problems.

Model	Best when	Watch closely
On-demand GPU cloud	You need fast access, elasticity and short commitments	Availability, egress, storage and instance-level resource bundle
Reserved / committed cloud	Utilization is predictable and capacity assurance matters	Prepayment, cancellation, region lock and overcommit risk
Dedicated server / cluster	You need stable bare metal, topology or sustained use	Provisioning lead time, remote hands, replacement SLA and contract term
Marketplace / broker	You value broad inventory and flexible price discovery	Actual seller, host-level SLA, variable networking and data handling
Managed platform	Engineering time, deployment and operations matter more than raw infrastructure	Platform fee, compute pass-through, portability and service boundary
Colocation / own hardware	You already own GPUs or can support a longer deployment horizon	Capex, power density, cooling, transit, spares and operations
Hardware lease / rental	You need equipment control without outright purchase	Who hosts it, maintenance, insurance, return terms and residual risk

Identity check
Ask who signs the contract, who operates the hardware, where the hardware sits, who controls the network and who handles a failed GPU. Brand, seller, operator and facility can be four different entities.

Use the operating model to set the due-diligence depth

- Direct provider: verify the service location, operational responsibility and capacity commitment.
- Reseller/integrator: verify the upstream dependency, support handoff and pass-through terms.
- Marketplace: verify the individual host/seller, not only the platform.
- Colocation: verify power/cooling envelope, rack design, cross-connects, remote hands and spare strategy.
- Managed platform: separate platform pricing from underlying compute and data movement.

Make prices comparable

Preserve the native offer first; normalize only inside a matched cohort.

Field	Record exactly	Common failure
GPU	Model, variant, memory/profile, quantity	Treating H100 variants or MIG slices as equivalent
Sold object	GPU, VM, server, node, pod, endpoint or output unit	Dividing a node price into a synthetic GPU rate
Resources	CPU, RAM, storage, network, software/platform	Comparing component-only with all-in configurations
Money	Amount and native currency; tax status	Assuming every '$' is USD or tax-inclusive
Time	Second/minute/hour/day/month and minimum duration	Using 730 hours without matching commitment and utilization
Commitment	On-demand, spot, prepaid, reservation, contract length	Comparing a three-year rate with on-demand
Place	Region/facility and data-residency boundary	Ignoring regional price and availability differences
Evidence	URL, date, excerpt and card/table boundary	Joining a price from one row to a GPU from another

A safe normalization sequence

1. Freeze the native quote or rate card without arithmetic.
2. Reject offers that do not identify the sold object or GPU allocation well enough for the intended comparison.
3. Create cohorts by delivery model, model/variant, GPU count, region and commitment.
4. Add unavoidable extras: base instance, CPU/RAM, storage, network, support, orchestration, tax and egress.
5. Apply workload utilization and runtime assumptions explicitly, with a low/base/high case.
6. Convert currency only with a dated FX source and keep the original amount beside the conversion.
7. Compare cost per completed workload, not only cost per wall-clock hour.

The useful formula

Total workload cost = compute + required base resources + storage + data movement + platform/support + tax + engineering/operations impact. Divide by successful completed runs or useful output—not by headline GPU-hours alone.

Illustrative comparison worksheet

Offer	Native price	GPU / sold object	Term / region	Required extras	Comparable?
A	_____	_____	_____	_____	Yes / No / Directional
B	_____	_____	_____	_____	Yes / No / Directional

04

Interrogate the quote

A quote is useful when its assumptions are explicit and its service boundary is enforceable.

Ask	Good answer contains
What exactly is allocated?	GPU count/variant, tenancy, node topology, CPU/RAM/storage/network and software layer
Is capacity available now?	Quantity, region, start date, provisioning lead time and reservation mechanism
What is the commercial unit?	Native unit, minimums, commitment, prepayment, renewal and cancellation
What changes the invoice?	Storage, egress, public IP, network, support, platform, tax and overage rules
Who operates each layer?	Seller, hardware operator, facility, network and escalation owner
What happens on failure?	Replacement time, credits, data persistence, checkpoint handling and support response
What is measured?	Availability definition, maintenance exclusions and customer responsibilities
How do we exit?	Data export, image portability, deletion, notice, fees and termination assistance

Quote language to pin down

- ‘Starting from’: starting for which model, quantity, region, term and customer class?
- ‘Reserved’: capacity reservation, price reservation, or both?
- ‘Dedicated’: dedicated GPU, dedicated VM, dedicated server, or dedicated cluster?
- ‘Available’: orderable now, subject to approval, preorder, or roadmap?
- ‘GPU-hour’: physical GPU, partition/profile, allocated device or metered component?
- ‘No egress’: all destinations, a fair-use cap, same-region only, or selected products?

Do not confuse missing evidence with a favorable term

If ownership, stock, tax or a fee is not stated, record it as unknown and ask. Unknown is not zero.

05

Run a proof of value

Benchmark the complete path that will go into production, including data and failure behavior.

8. Use a representative model, dataset, image and framework version.
9. Measure queue/provisioning time separately from runtime.
10. Record throughput, latency, utilization, memory headroom and scaling efficiency.
11. Include checkpoint, storage and data-transfer time.
12. Trigger a restart or failure scenario and observe persistence and support response.
13. Reconcile the measured usage with the invoice or usage export.
14. Repeat enough runs to detect variability, throttling and noisy-neighbor effects.
15. Document the exact configuration so the winning result can be contracted and reproduced.

Metric	Target	Measured	Notes
Provisioning / queue time	_____	_____	_____
Time to first useful output	_____	_____	_____
Throughput / latency	_____	_____	_____
GPU utilization / memory	_____	_____	_____
Scaling efficiency	_____	_____	_____
End-to-end workload cost	_____	_____	_____
Failure recovery	_____	_____	_____
Invoice reconciliation	_____	_____	_____

Benchmark governance

Keep the benchmark owned by the buyer, not the vendor. Use the same acceptance criteria across offers, store raw logs, and distinguish provider tuning help from an independently reproducible result. A proof of value should end with a contractable configuration and a total-cost model, not only a screenshot of peak throughput.

Score the shortlist

Weight evidence quality and operational fit alongside price.

Criterion	Suggested weight	Score 1–5	Evidence
Workload performance / fit	20%	___	_____
All-in cost	20%	___	_____
Capacity assurance / lead time	15%	___	_____
Reliability / failure handling	10%	___	_____
Data, security and sovereignty	10%	___	_____
Network / storage path	10%	___	_____
Support / operations	10%	___	_____
Portability / exit	5%	___	_____

Gate before score

A weighted score should not rescue a failed mandatory requirement. Put security, region, model compatibility, capacity date and maximum commitment through pass/fail gates first.

Evidence quality scale

Score	Evidence quality
5	Contract, order form or first-party documentation binds the exact configuration and term.
4	First-party public page or portal binds the relevant fields; minor operational items remain.
3	Provider claim is clear but configuration, identity or lifecycle detail is incomplete.
2	Sales statement or announcement is plausible but not bound to an orderable product.
1	Third-party claim, conflicting evidence or inaccessible source; do not rely on it.

Procurement worksheet

A compact record for the final commercial and technical decision.

Requirement	Buyer entry
Workload and success metric	_____
GPU model / acceptable alternatives	_____
GPU count / topology / tenancy	_____
CPU / RAM / storage / network	_____
Region / sovereignty	_____
Start date / duration / utilization	_____
Security / compliance	_____
Support / SLA / replacement	_____
Budget / native currency / tax	_____
Exit / portability	_____

Final offer record

Field	Selected offer
Seller / contracting entity	_____
Operator / facility / upstream	_____
Exact configuration	_____
Native amount and unit	_____
Commitment / minimum / cancellation	_____
Extras and exclusions	_____
Availability and provisioning date	_____
Evidence URL / quote version / date	_____
Proof-of-value result	_____

08

Red flags and reference

Use uncertainty to ask better questions—not to manufacture certainty.

Red flag	Why it matters	Buyer action
GPU name without quantity/profile	The sold allocation is ambiguous	Require exact physical or virtual allocation
Price detached from its card/table	Configuration binding may be wrong	Request a bound quote or order summary
Future facility or preorder presented as live	Lifecycle mismatch	Require orderable date and capacity commitment
'Dedicated' without layer	Tenancy may apply only to VM or GPU	Define device, host and cluster tenancy
Marketplace brand hides host	Operational counterparty is unclear	Record seller, operator, region and SLA
Very low rate with missing extras	Headline may be a component or conditional floor	Build the all-in workload cost
No exit terms	Data/image migration can erase savings	Contract export, deletion and termination

Quick glossary

Term	Buyer meaning
GPU-hour	One hour of the published allocation; confirm whether this means a whole device, partition or component.
Instance-hour	One hour of the full VM/server configuration; do not divide unless allocation is explicit and the use case supports it.
Spot / preemptible	Interruptible capacity with price/availability risk; checkpointing and restart cost matter.
Reserved / committed	A duration or spend commitment; confirm whether capacity is guaranteed.
MIG / vGPU	A partitioned or virtualized GPU profile; compare profile and memory, not the parent GPU name alone.

Final rule. Keep the original offer intact. Every conversion, allocation assumption and workload factor should be visible, dated and reversible. That is how a price comparison becomes a buying decision.

Evidence base

Derived from ViaBandwidth's Q3 2026 finite audit: 872 candidate domains, 4,511 scoped price representations and closure at checkpoint 162. The private directory and row-level evidence are not reproduced. October 2026 rate captures are not historical Q3 averages. Earlier edition: https://www.viabandwidth.com/reports/state-of-gpu-cloud