GPU SMITH
COMPUTE ENGINEERING

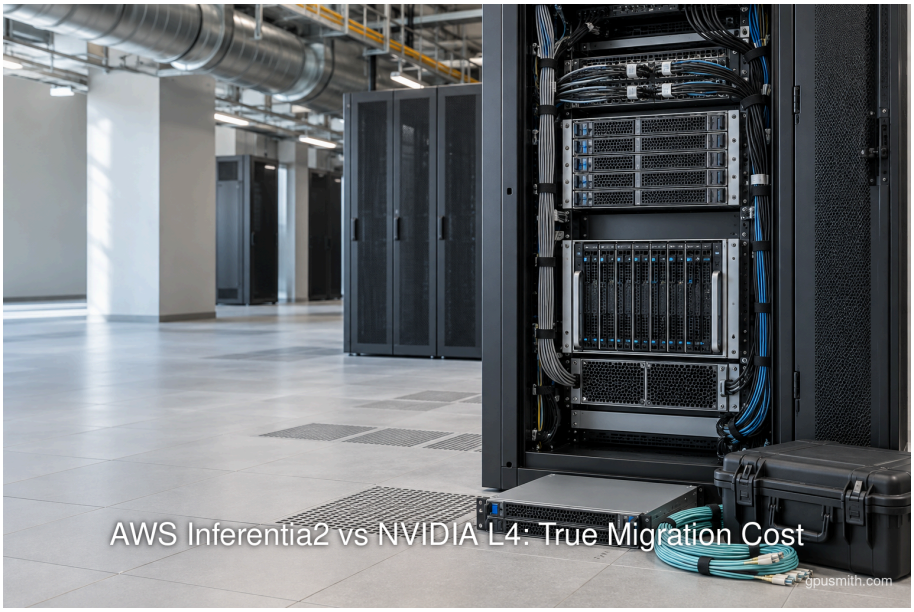
GPU SMITH / RESEARCH & PRACTICAL GUIDANCE

AWS Inferentia2 vs NVIDIA L4: True Migration Cost

> 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 · 2026-09-19 · aws inferentia2 vs nvidia l4 · inferentia2 cost · nvidia l4 · amazon ec2 inf2 · amazon ec2 g6 · aws neuron · llm inference · cost per token · gpu migration

Original article: https://gpusmith.com/articles/en/aws-inferentia2-vs-nvidia-l4-cost



In this article

Executive Summary

The useful answer to **AWS Inferentia2 vs NVIDIA L4** is not a theoretical operations-per-second comparison. It is a migration-adjusted break-even test for one frozen endpoint. For the defined case, that means one authorized **Llama-family 8B instruct checkpoint**, the same tokenizer and generation policy, 2,048 input tokens, up to 512 output tokens, a buyer-set P95 time to first token (TTFT) and inter-token latency (ITL) service-level objective (SLO), and the same measured arrival trace. The winner is the stack with the lower cost per million successful, quality-accepted, SLO-compliant output tokens after fleet rounding, idle headroom, failures, observability, storage, traffic, and commitments.

The hardware is not directly interchangeable. **Inf2** offers up to **384 GB** of shared accelerator memory (aws.amazon.com). **NVIDIA L4** is a single-slot accelerator with **24 GB GDDR6** and **300 GB/s** peak memory bandwidth (nvidia.com) (nvidia.com). AWS's EC2 specification reports **22 GiB** of accelerator memory for a full L4 on g6.xlarge, a useful warning that board memory and cloud-visible capacity should not be conflated (docs.aws.amazon.com). A published AWS example estimates about **16 GB** for Llama 3.1 8B at batch one and bfloat16, but production fit still depends on the key-value cache, runtime, sequence buckets, and concurrency (awsdocs-neuron.readthedocs-hosted.com).

The first decision is therefore feasibility, not price. Current Neuron documentation says NxD Inference entered maintenance mode with **Neuron 2.32.0** (awsdocs-neuron.readthedocs-hosted.com), while AWS's release record directs Trn1 and Inf2 users to pin **Neuron SDK 2.28** (github.com). The documented Inf2 vLLM combination is Neuron SDK 2.28.0, PyTorch 2.9, and vLLM 0.13.0 (awsdocs-neuron.readthedocs-hosted.com). That pinned bill of materials must compile the exact checkpoint and pass output, streaming, stop-token, error, and quality tests before any economics are calculated.

Public rates provide context, not a purchase decision. AWS published **inf2.xlarge at $0.7582 per hour** in us-east-1 on April 20, 2023 (aws.amazon.com). A separate April 2026 AWS workload analysis used approximately **$0.805 per hour** for g6.xlarge On-Demand (aws.amazon.com). Those snapshots are neither contemporaneous nor a substitute for an exact regional Price List API pull. Migration proceeds only if measured monthly recurring savings are positive and `migration cost / monthly recurring savings` falls inside the buyer's planning horizon. Otherwise the correct result is **defer** or **reject**, not a forced port.

Introduction and Background

This report examines a production endpoint migration from an **NVIDIA L4-backed Amazon EC2 G6 instance** to **AWS Inferentia2 on Inf2**. The scope is deliberately narrower than a generic accelerator review. It freezes one model revision, tokenizer, prompt template, precision, sequence policy, serving interface, Region, availability target, and traffic trace. It then asks whether an Inf2 implementation that produces equivalent accepted outputs can repay the engineering and dual-run expense.

That framing matters because hourly price is only one input. Neuron compilation, unsupported-operator remediation, artifact caching, runtime pinning, container changes, telemetry, training, shadow traffic, canary capacity, rollback retention, and model-refresh revalidation can dominate a small fleet. AWS migration guidance itself treats labor in person-days and buyer-specific day rates as inputs (docs.aws.amazon.com) and explicitly includes parallel-run cost around cutover (docs.aws.amazon.com).

GPU Smith is an **adjacent independent engineering adviser**, not either accelerator vendor. Its published method covers private AI infrastructure specification, procurement, integration, and validation against written criteria (gpusmith.com). That perspective belongs in the test design: establish acceptance criteria first, record measurements, and let the buyer's traffic and loaded costs determine the result.

AWS Inferentia2 and Inf2

Capabilities

Inferentia2 is AWS purpose-built inference silicon accessed through EC2 Inf2 and the Neuron software stack. The current NxD model reference documents a shared Llama path spanning Llama 2 through Llama 3.3 (awsdocs-neuron.readthedocs-hosted.com). It also points custom models to an onboarding path rather than promising arbitrary compatibility (awsdocs-neuron.readthedocs-hosted.com).

For the defined 8B checkpoint, the feasibility review should cover:

- **Graph support:** compare the exported graph with the published `torch-neuronx` ATen operator list (awsdocs-neuron.readthedocs-hosted.com).
- **Static dimensions:** Neuron requires statically sized tensor dimensions, so variable lengths need padding or buckets (awsdocs-neuron.readthedocs-hosted.com).
- **Parallel layout:** documented Inf2 tensor-parallel degrees are 1, 2, 4, 8, and 24, still subject to head divisibility and memory fit (awsdocs-neuron.readthedocs-hosted.com).
- **Custom remediation:** C++ custom operators are a beta path for the NeuronCore-v2 architecture used by Inf2 (awsdocs-neuron.readthedocs-hosted.com).
- **Serving semantics:** the Inf2-compatible guide allows streaming to be enabled or disabled, but equivalent API behavior must still be tested (awsdocs-neuron.readthedocs-hosted.com).

Adoption Path

The lowest-risk starting point is the documented preconfigured Deep Learning Container with the vLLM-Neuron plugin installed (awsdocs-neuron.readthedocs-hosted.com). Pin its image digest, the SDK and framework packages, compiler flags, model revision, shape buckets, tensor-parallel degree, and generation

settings. Do not translate a successful compile into a production-ready finding.

Compilation is a measurable migration activity. AWS gives an illustrative example in which a roughly 15 GB model might take about **15 minutes** to compile, while warning through the example's framing that actual time varies (awsdocs-neuron.readthedocs-hosted.com). Persistent-cache identity is based on the compiler flags and XLA graph, so a changed graph or flag set can miss the cache (awsdocs-neuron.readthedocs-hosted.com).

Strengths and Limitations

The principal **Inf2 strength** is a purpose-built, vertically integrated inference path with multi-chip memory and Neuron tooling. Its principal limitation is portability cost. The current documentation makes software-version selection a first-order design choice, not an incidental package upgrade.

- **Strength:** a documented Llama architecture and OpenAI-style vLLM route.
- **Strength:** Neuron cache artifacts can reduce repeated compilation when the cache key is unchanged.
- **Strength:** Container Insights collects metrics for Inferentia2 (docs.aws.amazon.com).
- **Limitation:** unsupported operators can require code changes or beta custom-operator work.
- **Limitation:** static compilation shapes raise the importance of request-length buckets.
- **Limitation:** quantization can reduce accuracy, requiring a buyer-specific quality gate (awsdocs-neuron.readthedocs-hosted.com).

NVIDIA L4 and EC2 G6

Capabilities

L4 is an Ada-generation PCIe accelerator. NVIDIA specifies **24 GB GDDR6**, **300 GB/s** peak memory bandwidth, PCIe Gen4 x16, and a **72 W** maximum power envelope (nvidia.com). G6 packages L4 into EC2 shapes, including fractional profiles down to **3 GB** of GPU memory (aws.amazon.com). AWS says the family reaches **100 Gbps** network bandwidth and **7.52 TB** of local NVMe storage (aws.amazon.com).

The serving path has several choices:

- **vLLM:** an official Docker image runs its OpenAI-compatible server (docs.vllm.ai).
- **TensorRT:** compiles PyTorch and ONNX models into GPU engines with mixed-precision support (docs.nvidia.com).
- **TensorRT-LLM:** documents multi-GPU and multi-node execution, in-flight batching, paged key-value caching, and quantization (docs.nvidia.com).
- **API compatibility:** `trtllm-serve` starts an OpenAI-compatible model server (nvidia.github.io).

Adoption Path

For an existing CUDA endpoint, G6 often preserves more of the current operational surface. That is a portability observation, not a performance conclusion. The benchmark must pin the Amazon Machine Image, driver, CUDA, container toolkit, serving image, engine build flags, model revision, and cache settings. AWS

notes that some Deep Learning AMI dependencies require package downgrades for compatibility (docs.aws.amazon.com).

Memory fit requires measurement because weights are not the whole allocation. vLLM exposes a per-GPU key-value cache size setting (docs.vllm.ai). The test must also account for runtime workspaces, CUDA graphs, fragmentation, batching, and the maximum 2,560-token combined request class. A model that loads at batch one has not demonstrated the target concurrency.

Strengths and Limitations

- **Strength:** broader CUDA ecosystem continuity can reduce code and operational changes.
- **Strength:** a choice among vLLM, TensorRT, and TensorRT-LLM supports different optimization depths.
- **Strength:** CloudWatch can collect NVIDIA utilization, temperature, power, and memory telemetry on Linux (docs.aws.amazon.com).
- **Limitation:** 24 GB board memory does not mean 24 GiB is visible to the EC2 guest (nvidia.com) (docs.aws.amazon.com).
- **Limitation:** engine compilation and version pinning remain required even when source code stays on CUDA.
- **Licensing check:** NVIDIA describes AI Enterprise as a license addition for L4, so it should not be silently assumed in the EC2 rate (nvidia.com).

Feature Comparison

Table 1 summarizes the portability and measurement gates. A check mark here means a documented path exists, not that the pinned model has passed.

Decision area	Inf2 / Inferentia2	G6 / NVIDIA L4	Required buyer evidence
Llama-family path	NxD documents Llama text architectures; custom integrations may need onboarding.	vLLM and TensorRT-LLM provide serving paths.	Exact checkpoint revision compiles or loads and serves the same API.
Memory	Family advertises up to 384 GB shared accelerator memory (aws.amazon.com).	L4 board has 24 GB, while EC2 reports 22 GiB for a full GPU (nvidia.com) (docs.aws.amazon.com).	Peak observed device use at every tested concurrency and length bucket.
Shapes	Compiler requires static dimensions.	Dynamic shapes are supported in TensorRT's compilation path.	Identical bucket policy or explicit normalization of padding cost.
Quantization	Neuron supports lower-precision paths, with accuracy risk documented.	Multiple CUDA serving paths support quantization.	Same accepted precision policy and task-specific quality threshold.
Caching	Persistent compile cache key includes compiler flags and XLA graph.	Model-engine and prefix or KV caches depend on the chosen server.	Cold, warm, cached, and uncached results reported separately.
Streaming	Documented Neuron vLLM route can enable streaming.	vLLM and TensorRT-LLM expose OpenAI-compatible serving routes.	Stop tokens, finish reasons, token counts, retries, and stream timing match.

Decision area	Inf2 / Inferentia2	G6 / NVIDIA L4	Required buyer evidence
Observability	Neuron metrics can flow through Container Insights.	NVIDIA GPU telemetry can flow through CloudWatch.	Same request IDs, SLI definitions, retention, dashboards, and alert tests.
Autoscaling	Same EC2 or Kubernetes policy can be applied, with compile and load time measured.	Same policy, with driver and model load time measured.	Scale-out time, minimum capacity, cooldown, queue depth, and dropped requests.
Rollback	Retain prior image, compiled artifacts, and routing path.	Retain prior image and engine artifacts.	Timed rollback rehearsal and capacity reservation.

The matrix makes the central tradeoff visible. Inf2 may expose more purpose-built capacity per selected shape, while L4 may preserve more existing CUDA assets. Neither statement decides cost. The buyer must translate every difference into measured throughput, required instance count, engineering hours, and retained capacity.

Freeze the Comparability Manifest

The manifest should be immutable and machine-readable:

- **Model:** repository, full commit hash, weight-file hashes, license record.
- **Tokenizer:** tokenizer assets, chat template, special tokens, padding side, maximum length.
- **Generation:** temperature, top-p, top-k, seed policy, maximum output, stop strings, stop-token IDs.
- **Precision:** weight, activation, and KV-cache formats; calibration dataset if applicable.
- **Serving:** engine, compiler, flags, API schema, batching, scheduler, tensor parallelism.
- **Runtime:** container digest, SDK, framework, driver, firmware, kernel, AMI.
- **Infrastructure:** instance type, Region, Availability Zones, storage, network path, autoscaling floor.
- **Acceptance:** task score, output-policy checks, P95 TTFT and ITL, error and retry ceilings.

Hugging Face supports pinning a Hub revision to a full commit hash (huggingface.co) and downloading a complete repository snapshot at that revision (huggingface.co). Generation defaults can come from `generation_config.json` (huggingface.co), and incorrect chat control tokens can materially degrade model performance (huggingface.co). Those are benchmark inputs, not clerical metadata.

Performance and Benchmarks

Quality Gate Before Cost

The comparison stops if either stack fails quality. Exact string equality is useful for deterministic fixtures, but it is not a universal cross-platform standard. PyTorch does not guarantee complete reproducibility across releases (docs.pytorch.org), and enabling deterministic algorithms alone does not make an entire application reproducible (docs.pytorch.org).

Use three layers:

- **Deterministic fixtures:** fixed prompts, greedy decoding where applicable, token counts, stop behavior, schema validity, and expected policy-response or tool-call structure.
- **Task acceptance:** buyer-owned scoring for classification, retrieval grounding, structured extraction, code tests, or human review.
- **Distribution checks:** output length, invalid response rate, retry rate, and drift across the real prompt sample.

Quantization research cautions that perplexity does not map cleanly to downstream benchmark performance (arxiv.org). Prompt adaptation must also be held constant across systems (crfm.stanford.edu). A faster result produced with a looser prompt or weaker acceptance threshold is not equivalent.

Replay the Same Traffic

Synthetic constant-rate traffic can hide burst response and scheduling effects. The BurstGPT study found different concurrency patterns across services and model types (arxiv.org), and its authors caution that KV-cache and scheduling optimizations do not necessarily generalize across workloads (arxiv.org). A 2026 peer-reviewed serving study likewise reports that production-derived workloads benchmark systems more accurately than naive generation (usenix.org).

Table 2 is the benchmark record to populate, not a table of invented results.

Prompt / output bucket	Concurrency or arrival slice	P50 / P95 / P99 TTFT	P50 / P95 / P99 ITL	Successful output tokens/s	Quality / errors
2,048 input / up to 128 output	Trace decile and peak burst	Measure cold and warm	Measure streamed gaps	Accepted tokens divided by wall time	Acceptance score, failures, retries
2,048 input / 129 to 256 output	Same replay timestamps	Measure cold and warm	Measure streamed gaps	Accepted tokens divided by wall time	Acceptance score, failures, retries
2,048 input / 257 to 512 output	Same replay timestamps	Measure cold and warm	Measure streamed gaps	Accepted tokens divided by wall time	Acceptance score, failures, retries
Full daily mix	Buyer-set concurrency ladder	Fleet percentiles	Fleet percentiles	SLO-compliant accepted output only	Weighted production distribution

TTFT is the interval from submission to the first received token (docs.nvidia.com); ITL is the average interval between consecutive output tokens (docs.nvidia.com). Interpret the table by the SLO, not by the largest raw tokens-per-second cell. A stack that has more aggregate throughput but violates P95 TTFT at the peak trace needs more instances or does not qualify.

Warm up both stacks because first-use library loading can distort the initial GPU timing (docs.pytorch.org). Synchronize CPU and CUDA when timing GPU work (docs.pytorch.org). Report cached and uncached runs separately because prefix-cache state changes apparent performance (huggingface.co). Finally, calculate fleet percentiles from aggregatable histograms, not averages of replica P95 values, which Prometheus calls statistically nonsensical (prometheus.io).

Data Analysis and Evidence

Normalize to Successful Output

The primary unit is:

```
effective cost per million successful output tokens = total metered serving cost / successful SLO-compliant accepted output tokens * 1,000,000
```

"Successful" is essential. NVIDIA's benchmarking definition of requests per second uses successfully completed requests (docs.nvidia.com), and TGI exposes a distinct successful-request counter (huggingface.co). Tokens from failed, retried, rejected, quality-failing, or SLO-violating requests remain costs but do not enter the successful-output denominator.

The full recurring numerator includes:

- **Compute:** integer fleet count at measured SLO-compliant throughput.
- **Idle headroom:** capacity required for the buyer's availability and burst policy.
- **Storage:** EBS model artifacts, compiler or engine caches, logs, and rollback copies. EBS capacity is charged on provisioned GB per month (aws.amazon.com).
- **Network:** ingress and egress path, inter-zone routing, and load balancer processing. AWS documented **$0.01 per GB in each direction** for inter-zone Network Load Balancer traffic in April 2026 (aws.amazon.com).
- **Observability:** metric, log, trace, dashboard, alarm, and retention charges. CloudWatch lists **$0.50 per ingested GB** for OpenTelemetry metrics on the fetched page (aws.amazon.com).
- **Software:** any separately purchased support or enterprise license.
- **Failures:** metered work for unsuccessful requests and retries.

Required fleet size is:

```
ceil(peak SLO-compliant demand / measured per-instance SLO-compliant throughput) + buyer-defined redundancy
```

Integer rounding is decisive at small scale. Auto Scaling desired capacity cannot fall below its configured minimum (docs.aws.amazon.com), and AWS recommends spanning Availability Zones when geographic redundancy is required (docs.aws.amazon.com). **Hypothetical Example:** a 1.2-instance arithmetic result can become two serving instances plus redundancy, not 1.2 billed instances.

Migration Ledger and Break-Even

Table 3 converts the port into auditable cost inputs.

Ledger item	Measurement	Cost treatment
Discovery and feasibility	Engineer hours for graph, operator, memory, API, and licensing review	Hours times buyer-loaded rate
Code and compilation	Porting, custom operator, bucketing, compile tuning, artifact packaging	Hours plus test instance runtime

Ledger item	Measurement	Cost treatment
Quality and load validation	Dataset preparation, scoring, trace replay, defect remediation	Hours plus both benchmark fleets
CI/CD and observability	Images, caches, dashboards, alerts, runbooks, access changes	Hours plus recurring storage and telemetry
Shadow and canary	Mirrored traffic, duplicate compute, data path, comparison service	Metered dual-stack cost
Training and on-call	Documentation, exercises, operator coverage	Buyer-loaded labor cost
Rollback retention	Prior image, artifacts, routing, warm capacity, rehearsal	Storage, capacity, and engineer time
Model refresh	Recompile and regression cadence over planning horizon	Expected events times measured refresh cost

Kubernetes defines a canary as deploying a new version alongside the existing one (kubernetes.io), which makes dual capacity a real ledger item. AWS guidance also says to calculate and document rollback duration before cutover (docs.aws.amazon.com).

The equations are straightforward:

- **Monthly recurring delta:** fully loaded L4 fleet cost minus fully loaded Inf2 fleet cost.
- **Migration cost:** measured engineering hours times buyer-loaded rate, plus testing, shadow, dual-run, tooling, training, and rollback-retention cost.
- **Payback months:** migration cost divided by a positive monthly recurring delta.
- **No payback:** monthly recurring delta is zero or negative.

Commitments require special treatment. AWS advertises Compute Savings Plans at up to **66% off** On-Demand, but that is a maximum, not a realized endpoint discount (docs.aws.amazon.com). Their terms cannot be changed after purchase (docs.aws.amazon.com). FOCUS requires purchased discounts to be amortized (focus.finops.org) and warns that unused amounts reduce potential savings (focus.finops.org). Use the buyer's effective covered rate and unused commitment allocation, not a headline percentage.

Sensitivity Grid

Run at least five axes:

- **Demand:** current trace, low case, growth case, and peak-event case.
- **Utilization:** observed scheduling efficiency and required idle headroom.
- **Rate:** On-Demand, the buyer's actual covered rate, and a renewal scenario.
- **SLO:** tighter and looser TTFT and ITL thresholds, with fleet recalculated.
- **Port effort:** low, expected, and high measured-hour cases.
- **Refresh cadence:** checkpoint, tokenizer, compiler, or serving-engine changes per year.

Do not allocate shared costs arbitrarily. FOCUS explicitly addresses resources whose costs are split across workloads or consumers (focus.finops.org). State the allocation rule for load balancers, Kubernetes control planes, monitoring, storage, and engineering platforms.

Implications and Future Directions

The 2026 decision is less about whether purpose-built silicon can be inexpensive and more about whether the buyer can maintain a reproducible, supported Inf2 route for its model lifecycle. A port that looks attractive for one frozen checkpoint can lose its payback if quarterly model changes repeatedly invalidate compile artifacts or quality evidence. Conversely, a stable endpoint with sustained traffic and high measured Inf2 utilization can amortize one-time work over more accepted tokens.

Use three decision states:

- **Go:** Inf2 passes feasibility and quality, its integer fleet meets the same availability and SLO, recurring savings are positive, and payback fits the planning horizon.
- **Defer:** the port is feasible, but evidence is incomplete, the refresh is near, or payback is beyond the horizon.
- **Reject:** the exact model or API semantics fail, quality is unacceptable, the required fleet costs at least as much, or operational risk exceeds the buyer's threshold.

Production rollout should begin with shadow evaluation, then a small canary, then staged traffic increases. Kubernetes topology-spread controls can place replicas across fault domains (kubernetes.io). The user-facing SLI should measure service behavior from the user's perspective (opentelemetry.io), so rollback triggers should include P95 TTFT, ITL, errors, quality, and cost drift, not only accelerator utilization.

Retest after any checkpoint or tokenizer revision, precision change, compiler or driver update, serving-engine change, new sequence bucket, material traffic-shape shift, or SLO change. NIST describes post-deployment monitoring as validating that AI systems continue to operate reliably in real-world conditions (nist.gov).

GPU Smith's stated position that the engineering recommendation is the product supports a dispassionate outcome, including a decision not to migrate (gpusmith.com). Its role here is method and validation, not placement in the Inf2-versus-L4 vendor matrix.

Frequently Asked Questions (FAQs)

What is the AWS Inferentia2 vs NVIDIA L4 cost difference?

There is no defensible universal difference. Pull both exact SKUs for the same Region, operating system, tenancy, and date. The AWS Price List Query API applies AND matching across specified filters (docs.aws.amazon.com). Then divide fully loaded fleet cost by successful, accepted, SLO-compliant output tokens. EC2 On-Demand Linux usage has a **60-second minimum** and no long-term commitment (aws.amazon.com).

Are there transferable Inferentia2 LLM inference benchmarks?

Published results are context only unless they use the same checkpoint, tokenizer, precision, 2,048-token prompt class, output distribution, traffic trace, concurrency, cache state, SLO, and quality rule. MLPerf's use of a standard load generator illustrates why request pattern must be controlled (mlcommons.org). The decision result must come from the buyer-run trace.

How should NVIDIA L4 LLM inference cost be calculated?

Use the same successful-token denominator, include integer G6 fleet count, redundancy, storage, monitoring, traffic, licenses, failed attempts, and idle headroom, then apply only commitments the buyer actually holds. Do not price a full 24 GB board when the chosen EC2 profile exposes a different capacity.

How does AWS Inf2 vs NVIDIA GPU instances affect portability?

Inf2 introduces Neuron compilation, supported-operator checks, shape bucketing, Neuron artifacts, and version-specific deployment work. G6 keeps a CUDA path but still requires pinned drivers, engine builds, images, and telemetry. Compare the actual change ledger, not the brand of accelerator.

How is cost per token Inferentia2 vs L4 computed?

Add all metered serving costs for the measurement window, then divide by output tokens from requests that passed quality and the SLO. Multiply by one million. Keep failed and retried work in the numerator, exclude their output from the denominator, and publish the traffic and availability assumptions.

What does it cost to migrate CUDA models to AWS Neuron?

Measure hours for discovery, code changes, operator remediation, compilation tuning, quality validation, CI/CD, observability, training, and rollback. Add benchmark, shadow, and dual-run infrastructure. Multiply labor hours by the buyer's loaded rate. There is no authoritative universal **AWS Neuron migration cost**.

Does Inferentia2 price performance guarantee a faster payback?

No. Vendor price-performance claims do not include the buyer's porting cost, fleet rounding, SLO headroom, quality failures, commitments, availability target, or model-refresh cadence. Only a positive measured monthly recurring delta can create payback.

Conclusion

For a steady Llama-family 8B endpoint, **AWS Inferentia2 vs NVIDIA L4** should be decided as an engineering investment, not a chip benchmark. First freeze the model, tokenizer, generation policy, precision, software bill of materials, request class, real traffic trace, SLO, and availability target. Then prove that both stacks produce acceptable outputs and equivalent API behavior.

Run the same cold-start, warmup, concurrency, cache, failure, and autoscaling tests. Convert measured SLO-compliant throughput into an integer fleet with buyer-defined redundancy. Price compute, storage, network, observability, licenses, idle capacity, failures, and actual commitments. Separately record engineer hours, test fleets, canary and shadow capacity, training, rollback, and expected revalidation.

Inf2 is a **go** only when it passes feasibility and quality, reduces the fully loaded monthly run rate, and repays the full migration ledger within the planning horizon. It is a **defer** when the result depends on missing measurements or an imminent stack change, and a **reject** when savings are non-positive or equivalence fails. That decision rule is more durable than any advertised throughput ratio because it can be rerun for the next checkpoint, SDK, Region, rate, or traffic distribution.

Source links

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

1. https://gpusmith.com/articles/en/llm-cost-per-million-tokens-benchmark
2. https://aws.amazon.com/ec2/instance-types/inf2/
3. https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/l4/PB-11316-001_v01.pdf
4. https://docs.aws.amazon.com/ec2/latest/instancetypes/ac.html
5. https://awsdocs-neuron.readthedocs-hosted.com/en/v2.28.0/libraries/nxd-inference/developer_guides/feature-guide.html
6. https://awsdocs-neuron.readthedocs-hosted.com/en/latest/release-notes/components/nxd-inference.html
7. https://github.com/aws-neuron/aws-neuron-sdk/releases
8. https://awsdocs-neuron.readthedocs-hosted.com/en/v2.28.0/libraries/nxd-inference/developer_guides/vllm-user-guide-v1.html
9. https://aws.amazon.com/blogs/machine-learning/aws-inferentia2-builds-on-aws-inferentia1-by-delivering-4x-higher-throughput-and-10x-lower-latency/
10. https://aws.amazon.com/blogs/machine-learning/cost-effective-multilingual-audio-transcription-at-scale-with-parakeet-tdt-and-aws-batch/
11. https://docs.aws.amazon.com/prescriptive-guidance/latest/application-portfolio-assessment-guide/understanding-complete-assessment-data-requirements.html
12. https://docs.aws.amazon.com/prescriptive-guidance/latest/application-portfolio-assessment-guide/detailed-business-case.html
13. https://gpusmith.com/
14. https://awsdocs-neuron.readthedocs-hosted.com/en/v2.31.1/libraries/nxd-inference/developer_guides/model-reference.html
15. https://awsdocs-neuron.readthedocs-hosted.com/en/v2.31.1/frameworks/torch/torch-neuronx/pytorch-neuron-supported-operators.html
16. https://awsdocs-neuron.readthedocs-hosted.com/en/latest/compiler/error-codes/EMOD025.html
17. https://awsdocs-neuron.readthedocs-hosted.com/en/v2.30.0/neuron-customops/index.html
18. https://awsdocs-neuron.readthedocs-hosted.com/en/v2.27.0/about-neuron/arch/neuron-features/neuron-caching.html
19. https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/Container-Insights-metrics-enhanced-EKS.html
20. https://gpusmith.com/articles/en/aws-g6f-fractional-vs-whole-l4
21. https://aws.amazon.com/ec2/instance-types/g6/
22. https://docs.vllm.ai/en/latest/deployment/docker/
23. https://docs.nvidia.com/deeplearning/tensorrt/latest/
24. https://docs.nvidia.com/tensorrt/

25. https://nvidia.github.io/TensorRT-LLM/quick-start-guide.html
26. https://docs.aws.amazon.com/dlami/latest/devguide/aws-deep-learning-ami-gpubasesinglecuda-al2023-2026-09-18.html
27. https://gpusmith.com/articles/en/llm-inference-hardware-sizing-guide
28. https://docs.vllm.ai/en/latest/api/vllm/config/cache/
29. https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/CloudWatch-Agent-NVIDIA-GPU.html
30. https://www.nvidia.com/en-eu/data-center/l4/
31. https://huggingface.co/docs/huggingface_hub/main/guides/download
32. https://huggingface.co/docs/transformers/main_classes/text_generation
33. https://huggingface.co/docs/transformers/main//chat_templating
34. https://docs.pytorch.org/docs/stable/notes/randomness.html
35. https://docs.pytorch.org/docs/main/generated/torch.use_deterministic_algorithms.html
36. https://arxiv.org/abs/2402.16775
37. https://crfm.stanford.edu/2022/11/17/helm.html
38. https://arxiv.org/abs/2401.17644
39. https://www.usenix.org/conference/nsdi26/presentation/xiang-servegen
40. https://docs.nvidia.com/nim/benchmarking/llm/latest/metrics.html
41. https://docs.pytorch.org/tutorials/recipes/recipes/benchmark
42. https://huggingface.co/docs/text-generation-inference/conceptual/chunking
43. https://prometheus.io/docs/practices/histograms/
44. https://huggingface.co/docs/text-generation-inference/en/reference/metrics
45. https://aws.amazon.com/ebs/pricing/
46. https://aws.amazon.com/blogs/networking-and-content-delivery/optimizing-data-transfer-costs-when-using-aws-network-load-balancer/
47. https://aws.amazon.com/cloudwatch/pricing/
48. https://docs.aws.amazon.com/autoscaling/ec2/userguide/asg-capacity-limits.html
49. https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-add-az-console.html
50. https://kubernetes.io/docs/tutorials/stateless-application/canary-deployment/
51. https://docs.aws.amazon.com/wellarchitected/latest/migration-lens/assess-rel.html
52. https://docs.aws.amazon.com/savingsplans/latest/userguide/plan-types.html
53. https://focus.finops.org/docs/specification/v1-3/attributes/discount-handling/
54. https://focus.finops.org/docs/use-cases/v1-4/identify-resources-with-shared-cost-allocation/
55. https://gpusmith.com/articles/en/good-gpu-utilization-percentage
56. https://kubernetes.io/docs/setup/best-practices/multiple-zones/
57. https://opentelemetry.io/docs/concepts/observability-primer/
58. https://www.nist.gov/publications/challenges-monitoring-deployed-ai-systems-center-ai-standards-and-innovation
59. https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/using-price-list-query-api.html
60. https://aws.amazon.com/ec2/pricing/on-demand/

61. https://mlcommons.org/benchmarks/inference-datacenter/?trk=article-ssr-frontend-pulse_little-text-block

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