The Economics of Open-Weight Inference

How open-weight demand can support the useful life of NVIDIA GPU families

Ornn Data

7 September 2026

Abstract

GPUs are commonly depreciated on the assumption that each new NVIDIA generation renders the previous one obsolete. In this paper, we provide a counter to this thesis by examining the effect of open-weight demand on the economic usefulness of older GPU families. Closed-model access runs through subscription allowances that the provider is able to reset, so the posted token rate card respresents the marginal price of additional usage. At standardized Artificial Analysis Intelligence Index thresholds, the cheapest qualifying open-weight model costs $0.09 per benchmark task compared with $0.43 for the cheapest closed option, a reduction of about 79%. Self-hosting on rented hardware lowers this to $0.12 to $0.35 per million output tokens at full utilization and reverses the hardware ranking: on gpt-oss-120b, a sparse model with 5.1 billion active parameters, the A100 is more monetizable than the H100 at spot, three-year, and five-year term prices. Ornn's own rental data shows the market reflecting this utility. The five-year A100 rental price maintains 80 percent of its one-month term price (vs 44 to 60% for the Hopper and Blackwell families) for a contract ending when the Ampere family is more than eleven years old. We show that today's compute-intensive workloads: long-running agents, batch evaluation, and reinforcement learning tolerate latency and are hardware agnostic, which incentivizes price elastic demand to route to any cost-efficient hardware. See NVIDIA's acquisition of Hugging Face on 3 September 2026 (NVIDIA, 2026c). These findings challenge forecasts that newer hardware eliminates the earning capacity of older GPUs. Instead, they suggest that older NVIDIA generations retain a multi-year earning life so long as they serve suitable workloads competitively and operators remain free to deploy those workloads on them.

Keywords: open-weight models; inference cost; cost per task; usage allowances; GPU rental; forward curves; economic obsolescence.
JEL classification: D24, L86, O33.

1 Introduction

Until recently, large language model (LLM) inference was priced and analyzed almost exclusively in terms of cost per million tokens (Anthropic, 2026e; OpenAI, 2026b). This framing was adequate when access consisted largely of metered, short-context queries, because the posted per-token rate coincided with the marginal price paid by the user. However, closed-model providers now distribute consumer and professional access through subscription plans. These plans grant a capacity-dependent allowance rather than a fixed token limit, and providers reset that allowance through time windows (Section 2). Subscribers who exhaust the allowance can purchase additional usage at metered rates, decoupling the subscription price from the realized cost of work.

Alongside this change in pricing, the composition of inference demand has also changed, with agentic rollouts, batch evaluation, and reinforcement learning creating demand for sustained execution (Modal Labs, 2026). These workloads execute over hours, tolerate latency, and, when the underlying model is open-weight, are portable across serving providers and hardware generations (Section 4). As a result, the revenue a GPU commands depends on which of these workloads it can serve at minimum cost per completed task, rather than on the posted price of a token alone.

These developments raise two related questions: where do open-weight models offer lower costs, and how does the freedom to deploy them affect the market for older GPUs? This paper connects three pieces of evidence. It compares hosted benchmark costs across eleven open-weight and eight closed models using the Artificial Analysis Intelligence Index (Artificial Analysis, 2026a); calculates token-production costs from published throughput and Ornn's rental index (MLCommons, 2024, 2026; GPUStack, 2026; Ornn, 2026b); and examines rental pricing and occupancy across GPU families. The organizing idea is that hardware retains a role when it can serve useful work competitively.

The argument proceeds in three steps. First, open-weight models offer lower hosted benchmark costs at several capability thresholds in the sample. Comparing the cheapest qualifying option on each side identifies where that advantage lies. Self-hosting then adds a choice between paying a provider's token rate and purchasing the hardware time needed to produce the output (Section 3).

Second, long-running agents, batch evaluation, and parts of reinforcement learning offer flexibility over when and where work runs. Open weights let independent operators choose among compatible providers and hardware generations while meeting their quality and service requirements. The deployment decision therefore extends beyond the model developer to the operators buying compute (Section 4).

Third, this freedom to move workloads connects inference costs to rental earnings. An older GPU that completes suitable work cheaply can attract price-sensitive demand. When that demand is large enough relative to available capacity, it supports rental earnings. The A100's cost advantage on the sparse-model workload shows why the arrival of a newer generation need not eliminate the older family's economic role.

Ornn's rental data give this argument a market setting. The A100 retains more of its one-month price at the five-year tenor than the Hopper and Blackwell families (Ornn, 2026a). At the same time, tracked A100 listings and rented quantity rose together over six months (Ornn, 2026c). Combining rents with throughput explains why workload choice matters: the baseline dense-model calculation favors newer hardware, while gpt-oss-120b is cheaper on the A100 than on the H100 at spot and at the five-year mark. The paper develops this connection between competitive workloads and continued demand for older capacity. Section 7 sets out the scope of the benchmark and rental evidence.

2 Closed access rationing

2.1 Access modes

To understand how users pay for inference we've laid out five modes of access in Table 1. Closed-model providers sell the same inference capacity through the first three, which differ in the basis for pricing, who determines the quantity supplied, and the cost of additional usage. The remaining two modes involve open-weight models: a hosted API run by a third party, and self-hosting on rented or owned hardware. Section 3 constructs the cost of this final option.

The first mode is the *subscription*, which provides an allowance in return for a fixed monthly fee. That allowance resets on one or more time windows. Anthropic's and OpenAI's plans follow the same general structure. Each has a $20 entry tier, $100 and $200 tiers that carry multiples of the entry allowance, and team seats. At both providers, plans reset on a rolling five-hour window (OpenAI's Pro tiers were temporarily exempt at the time of writing; Mendes, 2026), and paid plans also carry weekly limits (Anthropic, 2026d; OpenAI, 2026d,e). Neither provider states the allowance in a unit the subscriber can see. Anthropic says usage "depends on the length and complexity of your conversations, the model you choose, and the features you use, so there's no fixed message count" (Anthropic, 2026d). OpenAI publishes ranges of "local messages" per five-hour window for Codex on each plan. These published ranges span an order of magnitude. For GPT-6 Astra, released on 3 September 2026, the corresponding ranges are 5–45 (Plus) to 100–900 (Pro 20x) local messages per five hours, less than half the GPT-5.6 Sol ranges on the same plans (OpenAI, 2026c,l). It adds that "additional weekly limits may apply," and that "GPT-5.6 usage averages 5-30 credits per message," with the charge per message depending on context length, reasoning effort, and tool use (OpenAI, 2026c).

The second mode, *usage credits*, allows a subscriber to prepay for usage beyond that allowance. Anthropic provides "consumption-based pricing at standard API rates" (Anthropic, 2026c), while OpenAI states that on Plus and Pro "your plan's included usage is used first" and credits are drawn once plan limits are reached, priced per million input, cached input, and output tokens on a rate card (OpenAI, 2026f,c).

Alternatively, users can access the third mode, the *metered API*, which bills per token and carries no allowance. Table 1 gives the current rates for the Claude 5, GPT-5.6, and GPT-6 families; two of them are promotional or recently changed (Anthropic, 2026e; OpenAI, 2026b). These are also the rates at which Anthropic's usage credits are billed.

The fourth mode, the *hosted open-weight API*, is a metered API run by a third party for a model whose weights are public. Anyone can host the model, so no single provider controls supply (Section 4.1). The final option is self-hosting, for which there is no per-token list price. Its cost is the GPU-hour rent on Ornn's spot index divided by the throughput the operator achieves (Ornn, 2026b).

2.2 Usage controls and marginal pricing

Providers can limit the volume and timing of inference supplied under a subscription without changing its monthly price. They do so by adjusting allowances, imposing shorter reset windows, or changing peak-period restrictions. Anthropic has linked allowance changes to new compute capacity, while OpenAI has restored five-hour windows to smooth demand without changing subscription prices (Anthropic, 2026h; Mendes, 2026). From the provider's perspective, these controls offer a way to manage compute consumption without raising the price. From the subscriber's perspective, however, an unchanged fee need not buy an unchanged amount of work. The GPT-6 Astra release illustrates this: at the same $20 fee, the published Plus range falls from 10–100 local messages per five hours on Sol to 5–45 on Astra, because each Astra message draws

Table 1. Access modes for closed and open-weight inference, list terms as of 1 September 2026 (GPT-6 Astra terms as of 5 September 2026).

Mode	Price basis	Who sets quantity	Marginal price beyond allowance	Example terms
Subscription	Fixed monthly fee	Provider: allowance per rolling five-hour window and per week, in a unit the provider does not publish	Metered API rate, via usage credits (Anthropic) or credits at the rate card (OpenAI)	Claude Pro $20; Max $100, $200 (5x, 20x Pro). ChatGPT Plus $20; Pro $100, $200 (5x, 20x Plus); Business seats $25, $125. Codex on GPT-5.6 Sol: 10–100 (Plus) to 200–2,000 (Pro 20x) local messages per five hours; on GPT-6 Astra: 5–45 (Plus) to 100–900 (Pro 20x)
Usage credits	Metered, prepaid	Subscriber (Anthropic: user-set monthly cap, $2,000 daily redemption; OpenAI: credits valid 12 months)	Standard API rate (Anthropic); rate card per million tokens (OpenAI)	Billed at standard Claude API rates. GPT-5.6 Sol: 100 / 10 / 500 and GPT-6 Astra: 250 / 25 / 1,250 credits per million input / cached input / output tokens
Metered API	Per million tokens	Subscriber, up to tier rate limits stated as maxima, not guaranteed minimums	Same rate; 0.5x at batch, 2x at fast (Section 2.2)	Input/output rates: Fable 5.1 $10/$50; Opus 5 $5/$25; Sonnet 5 $2/$10; gpt-6-astra $10/$50 ($20/$75 above 272K context); gpt-5.6-sol $4/$20; gpt-5.6-terra $2/$12; gpt-5.6-luna $0.20/$1.20
Hosted open-weight API	Per million tokens	Operator, across competing providers	Same rate, or another provider's rate	gpt-oss-120b $0.15/$0.60 (Groq, Together); $0.03–$0.35 in, $0.17–$0.95 out across 21 listings
Self-hosted open-weight	GPU-hour rent or ownership	Operator: throughput, utilization, operations	Rent of an additional GPU	A100 $1.00, H100 $2.83, H200 $4.51, B200 $6.40 per GPU-hour (Ornn spot index, 1 Sep 2026)

Sources: Anthropic (2026c,d,e,j); OpenAI (2026b,c,d,e,f); Groq (2026); Together AI (2026); OpenRouter (2026a); Ornn daily index value fixed 1 September 2026 20:00 UTC (Ornn, 2026b). GPT-6 Astra terms retrieved 5 September 2026 (OpenAI, 2026l). Class: all plan terms and token prices are provider data; the four spot rents are observed Ornn index values. Prices in USD, excluding tax; providers state that plans and prices may change at their discretion.

2.5 times the credits (OpenAI, 2026c). Because allowances also depend on model choice, context length, and tool use, they cannot be translated into a fixed token quantity (Anthropic, 2026g; OpenAI, 2026c). The implication is that posted prices alone are an incomplete measure of the economics of inference: they do not reveal how much usage is available, when it can be consumed, or what it costs to complete a task.

This does not make the token rate irrelevant. Once a workload exceeds the included allowance, users can wait for a reset, reduce usage, or purchase additional access where available. For those who purchase it, the applicable metered rate determines the marginal charge. Anthropic bills usage credits at standard API rates, while OpenAI uses a Codex credit rate card covering input, cached input, and output tokens (Anthropic, 2026c; OpenAI, 2026c). A subscription therefore buys a bounded allowance rather than a fixed quantity of tokens. The analysis does not assign that allowance an API-equivalent value or divide the fee by an assumed

token count, because either calculation would impose an unobserved consumption pattern on an unpublished unit.

There is a further distinction within metered access: the same tokens are offered at three prices, depending on the service level to which the provider commits (Table 1). At half the standard rate, Anthropic's Message Batches API and OpenAI's Batch API return results within a 24-hour window, after which requests expire; OpenAI's Flex tier, priced at batch rates for synchronous requests, "may sometimes lack sufficient resources to handle your requests" (Anthropic, 2026k; OpenAI, 2026i,h). At the standard rate availability is best-effort. Anthropic says its rate limits "represent maximum allowed usage, not guaranteed minimums," and OpenAI's usage tiers cap spend without guaranteeing capacity (Anthropic, 2026i,j; OpenAI, 2026k). At twice the standard rate both providers sell a fast mode with higher throughput per request (OpenAI, 2026j; Anthropic, 2026e). None of these listed options, however, guarantees capacity. Anthropic's Priority Tier, which targets 99.5 percent uptime for existing commitments, is "no longer available for purchase" and is not offered on its current models, and the output rate limits of its newest model are a fraction of its predecessor's at every tier (Anthropic, 2026i,j).

Despite these differences in access, the price schedule and the allowance both ration a quantity that the provider controls: the schedule through best-effort availability at the standard rate, the allowance through a level the provider resets. The relevant marginal price of sustained closed-model work is therefore the metered rate at the service level the workload requires. Comparing the cost of completing that work also requires a measure of model capability and the number of tokens needed per task, which the next section introduces.

3 Intelligence per dollar

3.1 Measuring intelligence and its price

The preceding section distinguishes the price of additional usage from the cost of completing work. Comparing deployment options also requires a measure of what each model can accomplish. The Artificial Analysis Intelligence Index, version 4.1.1, combines nine evaluations into a weighted average, emphasizing agentic and coding work; higher scores indicate stronger performance on this benchmark suite (Artificial Analysis, 2026a,c). In the manuscript's v4.1.1 comparison, scores range from 14 to 60 for open-weight models and from 30 to 66 for closed models (Table 3, Panels A and B); these are the observed sample ranges, not the bounds of the Index. Here it orders models and identifies configurations that clear a minimum score. This is a benchmark comparison: application-level performance depends on the tasks and success criteria of the deployment.

Throughout, "Index x" denotes an Intelligence Index score of x on the stated benchmark version. When used as a minimum capability threshold, it requires a score of at least x; when describing a model, it reports that model's score.

On the price side, a per-token rate determines the charge for usage, while cost per benchmark task also accounts for how much output the model produces. Artificial Analysis combines evaluation-specific average costs using the Index weights, including input, cache-hit, cache-write, reasoning, and answer tokens at the applicable hosted prices (Artificial Analysis, 2026a,c). This captures token use and reasoning effort as well as price. The comparison uses the manuscript's v4.1.1 snapshot. Section 3.5 then calculates the separate cost of producing output tokens on rented hardware.

Table 2. Open-weight models compared in this section: developer, weight release, parameters, and license (provider data, 2 September 2026).

Model	Developer	Weights released	Total / active params	License
Kimi K3	Moonshot AI	Jul 2026	2.8T / 104B	Kimi K3 License (MIT-derived, gated)
GLM-5.3	Z.ai	Aug 2026	753B / 40B	GLM-5.3 License (MIT-derived, gated)
Qwen3.8 2.4T-A95B	Alibaba	Aug 2026	2.4T / 95B	Qwen3.8-Max License (MIT-derived, gated)
GLM-5.3-Flash	Z.ai	Aug 2026	320B / 18B	MIT
DeepSeek V4 Pro 0813	DeepSeek	Aug 2026	1.6T / 49B	MIT
DeepSeek V4 Flash 0731	DeepSeek	Jul 2026	284B / 13B	MIT
MiniMax-M3	MiniMax	Jun 2026	428B / 23B	MiniMax Community License (gated)
Nemotron 3 Ultra	NVIDIA	Jun 2026	550B / 55B	OpenMDW 1.1
gpt-oss-120b	OpenAI	Aug 2025	117B / 5.1B	Apache 2.0
gpt-oss-20b	OpenAI	Aug 2025	21B / 3.6B	Apache 2.0
Llama 4 Maverick	Meta	Apr 2025	400B / 17B	Llama 4 Community License (gated)

Sources: developer model cards and licenses (Moonshot AI, 2026; Z.ai, 2026a; Qwen, 2026; DeepSeek, 2026a; MiniMax, 2026a; NVIDIA, 2026b; OpenAI, 2025, 2026a; Meta, 2025). Parameter counts are provider figures; "active" denotes parameters exercised per token. License summaries are not exhaustive. Kimi K3's US$20M and Qwen3.8's US$50M twelve-month revenue tests apply to aggregate licensee-and-affiliate revenue when the specified business conditions apply, with internal-use exceptions. Qwen's condition also covers its defined AI Work Assistant business. GLM-5.3 has a US$10B gate; Llama 4 has a 700M monthly-active-user gate. MiniMax-M3 requires commercial attribution and a one-time notice below its US$20M annual product-and-service revenue threshold, and prior authorization above it. The license texts govern each deployment.

3.2 The open-weight landscape

Table 2 lists eleven open-weight models released between April 2025 and August 2026 whose weights could be downloaded on 2 September 2026, with developer, parameter counts, and license. There are two features of this set that are important for the cost comparison that follows.

First, every model in the table is a sparse mixture-of-experts model. Active parameters help describe the arithmetic required per token, while total parameters and numerical precision determine weight-storage requirements. The balance between arithmetic, memory traffic, and communication depends on batch size, context length, expert reuse, and the serving implementation. These differences make workload choice central to hardware economics. Section 3.5 examines that relationship through a sparse-model example.

Second, six models use MIT, Apache 2.0, or NVIDIA's OpenMDW license, which permit commercial deployment subject to their terms (DeepSeek, 2026a; Z.ai, 2026a; OpenAI, 2026a; NVIDIA, 2026b). The other five use licenses with additional commercial conditions, including revenue or user thresholds (Moonshot AI, 2026; Z.ai, 2026a; Qwen, 2026; MiniMax, 2026a; Meta, 2025). These conditions differ by license: the relevant business activity, revenue base, attribution duties, and internal-use exceptions must be considered separately. Open weights therefore expand deployment choices without making every deployment unrestricted.

3.3 Cost per task side by side

With these model characteristics in view, Table 3 and Figure 1 compare the open-weight models of Table 2 with the current closed models on the Artificial Analysis Index. They show cost per task, the list prices it is computed from, and median output speed, as fetched on 1 and 2 September 2026. Unless otherwise stated, each model is evaluated at its highest reasoning setting.

The closed frontier is six Index points above the open frontier in this sample. Within the overlapping range, the most useful comparison for a cost-conscious operator is the cheapest qualifying configuration on each side, rather than an arbitrarily selected pair. Table 4 applies that rule to the configurations in Table 3. At thresholds from 53 to 57, GLM-5.3-Flash costs $0.09 per benchmark task against $0.43 for GPT-5.6 Sol at high effort, a factor of 4.8. At thresholds from 58 to 60, the corresponding costs are $0.68 for GLM-5.3 and $0.95 for Sol at maximum effort, a factor of 1.4. At Index 52, however, the closed Luna configuration is cheaper, and above 60 there is no qualifying open model in the table.

Selected pairs show larger differences: Sonnet 5 at Index 55 costs $1.72 against Flash's $0.09 at Index 57, a factor of nineteen. For a buyer choosing the lowest-cost qualifying model, however, Sol at high effort is the relevant closed alternative: it is cheaper than Sonnet and higher-scoring. The threshold comparison therefore locates the open-weight advantage where it matters for a cost-minimizing choice.

OpenAI released GPT-6 Astra on 3 September 2026, after the snapshot in Panels A and B was taken, at $10 per million input tokens and $50 per million output tokens, 2.5 times the Sol rate (OpenAI, 2026l,b). Artificial Analysis scores it only on Index v4.2, which replaced v4.1.1 in the same week and covers ten evaluations rather than nine; every model scores lower on v4.2, and Claude Fable 5.1 falls from 66 to 57 (Artificial Analysis, 2026b,c). Astra therefore has no v4.1.1 score, and Panel C of Table 3 reports it on v4.2 alongside the same-version scores of the leading Panel A and B models. Within Panel C, Astra at maximum effort scores 55, second to Fable 5.1's 57, at $2.57 per task against $6.12; at high effort it scores 53 at $1.41, above Claude Opus 5 at maximum effort (54 at $4.21) on cost and above GPT-5.6 Sol at maximum effort (51 at $1.25) on score. On the open side, Kimi K3 scores 50 at $1.58 and GLM-5.3-Flash 46 at $0.18. Among the Panel C configurations, no open model clears 51 on v4.2, and at 49 to 50 the closed Sol configuration is cheaper than the open alternatives. The v4.2 cross-section therefore does not reproduce the open-weight advantage at the top of the open range shown in Table 4. Whether that reflects Astra, the rebased Index, or both cannot be separated without a common version, and Panel C covers selected models rather than the full v4.2 leaderboard. Below Index 49 on v4.2, GLM-5.3-Flash remains an order of magnitude cheaper than any closed configuration in the panel.

Reasoning effort is an important part of this choice. Moving Sol from maximum to high effort more than halves its benchmark cost and lowers its Index score by four points. The most expensive setting is therefore unnecessary when a lower setting meets the workload's requirements. The table notes specify the price and configuration basis; the thresholds identify the cheapest options within that sample.

3.4 Cost by release cohort and frontier catch-up

The comparisons so far describe a single price snapshot. The next question is how the models meeting a score threshold differ across release cohorts. For a threshold X and release cutoff t, the reconstruction takes the lowest September 3 cost among sampled configurations released by t with a v4.1.1 score of at least X. The sample contains fifty-three configurations released between April 2025 and 2 September 2026 (Artificial Analysis, 2026b). GPT-6 Astra, released on 3 September, falls outside the window and has no v4.1.1 score; on v4.2 it scores below Claude Fable 5.1 (Table 3, Panel C), so it would not extend the closed frontier on the

Table 3. Artificial Analysis Intelligence Index v4.1.1, cost per task, list prices, and median output speed for open-weight and closed models (third-party evaluation and provider data, 1 to 2 September 2026), with a post-snapshot addendum for GPT-6 Astra on Index v4.2 (5 September 2026).

Model (reasoning setting)	Index	Cost per task, $	List price in / out, $ per M tokens	Median output tok/s
Panel A: open-weight				
Kimi K3 (max)	60	0.84	3.00 / 15.00	38
GLM-5.3 (max)	60	0.68	1.40 / 4.40	72
Qwen3.8 2.4T-A95B	58	0.81	2.00 / 6.00	41
GLM-5.3-Flash	57	0.09	0.15 / 0.50‖	43
DeepSeek V4 Pro 0813 (max)	53	0.27	1.32 / 3.96¶	54
DeepSeek V4 Flash 0731 (max)	52	0.11	0.44 / 1.32¶	108
MiniMax-M3	45	0.14	0.30 / 1.20	117
Nemotron 3 Ultra	38	0.39	0.50 / 2.20†	93
gpt-oss-120b (high)	24	0.07	0.15 / 0.60‡	169
gpt-oss-20b (high)	15	0.02	0.06 / 0.19‡	111
Llama 4 Maverick	14	0.03	0.20 / 0.80†	84
Panel B: closed				
Claude Fable 5.1 (max)	66	3.69	10.00 / 50.00	66
Claude Opus 5 (max)	63	2.34	5.00 / 25.00	54
Claude Opus 5 (high)	61	1.23	5.00 / 25.00	48
GPT-5.6 Sol (max)	61	0.95	4.00 / 20.00§	76
GPT-5.6 Sol (high)	57	0.43	4.00 / 20.00§	76
GPT-5.6 Terra (max)	57	0.53	2.00 / 12.00	105
Claude Sonnet 5 (max)	55	1.72	2.00 / 10.00	73
GPT-5.6 Luna (max)	52	0.05	0.20 / 1.20	128
Gemini 3.1 Pro Preview	48	0.33	2.00 / 12.00	103
Claude Haiku 4.5	30	0.22	1.00 / 5.00	92
Panel C: post-snapshot addendum, Index v4.2 (not comparable with Panels A and B)				
GPT-6 Astra (max)	55	2.57	10.00 / 50.00	64
GPT-6 Astra (xhigh)	54	1.85	10.00 / 50.00	62
GPT-6 Astra (high)	53	1.41	10.00 / 50.00	61
GPT-6 Astra (medium)	52	1.16	10.00 / 50.00	59
Claude Fable 5.1 (max)	57	6.12	10.00 / 50.00	69
Claude Opus 5 (max)	54	4.21	5.00 / 25.00	57
GPT-5.6 Sol (max)	51	1.25	4.00 / 20.00§	90
Kimi K3 (max)	50	1.58	3.00 / 15.00	41
GLM-5.3 (max)	49	1.26	1.40 / 4.40	83
GLM-5.3-Flash	46	0.18	0.15 / 0.50‖	46

Sources: Artificial Analysis leaderboard and model pages (Artificial Analysis, 2026a,b). Cost per task is AA's weighted benchmark-cost measure, including its cache treatment; the displayed uncached input/output rates alone do not reproduce it. Speed is the provider median. Fable uses adaptive reasoning, maximum effort, and default fallback. Displayed rates are first-party API prices (Together AI, 2026; Z.ai, 2026b; Alibaba Cloud, 2026; DeepSeek, 2026b; OpenRouter, 2026b; MiniMax, 2026b; OpenAI, 2026b; Anthropic, 2026e; Artificial Analysis, 2026b), except: † lowest DeepInfra listing where no first-party API exists (DeepInfra, 2026); ‡ provider median used by AA (Artificial Analysis, 2026b; Groq, 2026; Together AI, 2026); § promotional through at least 21 November 2026 (OpenAI, 2026b); ¶ DeepSeek peak rate, halved off-peak (DeepSeek, 2026b); ‖ Z.ai list price, halved on its endpoint through 9 September 2026 (Z.ai, 2026b). Sonnet 5's temporary rate became permanent on 1 September 2026 (Anthropic, 2026e). These are the manuscript's reported September 1–2 observations, not a refreshed leaderboard. Panel C: GPT-6 Astra (released 3 September 2026) is scored only on Index v4.2 (ten evaluations); all Panel C values were fetched 5 September 2026 (Artificial Analysis, 2026b; OpenAI, 2026l). v4.2 scores are lower than v4.1.1 scores for every model and its costs cover a different evaluation set, so Panel C is not comparable with Panels A and B, does not enter Table 4, and is a selection rather than the full v4.2 leaderboard.

Table 4. Cheapest qualifying hosted configurations in Table 3, by minimum Index score (calculated).

Minimum Index	Open model, $ per task	Closed model, $ per task	Closed / open cost
52	GLM-5.3-Flash, 0.09	Luna max, 0.05	0.56
53–57	GLM-5.3-Flash, 0.09	Sol high, 0.43	4.78
58–60	GLM-5.3 max, 0.68	Sol max, 0.95	1.40
61	No qualifying model	Sol max, 0.95	—

Calculated from the v4.1.1 scores and costs printed in Panels A and B of Table 3; ratios use the unrounded quotient of the displayed costs. GPT-6 Astra (Panel C) is scored only on v4.2 and does not enter. Each range covers integer thresholds with the same minimizing configurations. The sample is not an exhaustive set of models or reasoning settings. Clearing a common Index threshold does not establish equal application-level success.

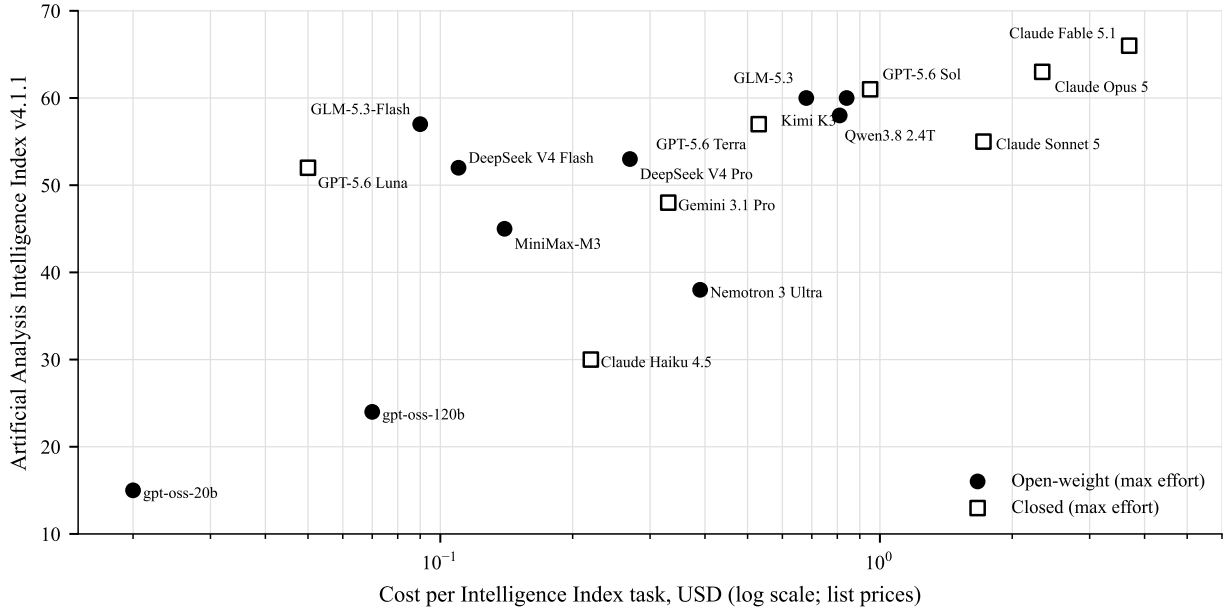

Figure 1. Artificial Analysis Intelligence Index against cost per Index task for the highest-reasoning configurations in Table 3, USD on a log scale. The cheapest-qualifying comparison in Table 4 also includes the lower-effort configurations printed in Table 3. Panel C of that table (GPT-6 Astra, Index v4.2) is excluded because its scores are on a different Index version.

Sources: Artificial Analysis (2026a,b) (third-party evaluation, fetched 1 to 2 September 2026). Open-weight models are filled circles; closed models are open squares.

current version either. Holding the benchmark and price date fixed isolates differences across these cohorts. Unlike a historical price series such as that studied by Epoch AI (2025), it values each cohort at the same price date.

Ten configurations have no reported cost and enter only the capability reconstruction.[1] The cost series uses the remaining forty-three. Its minima describe the sampled configurations and reasoning settings; the data-availability statement sets out the materials required for reproduction.

The cost differences across cohorts are substantial. At Index 30, the two models in Table 5 differ in cost by a factor of twelve and were released eleven months apart. At Index 40, 50, and 57, the corresponding factors

[1] Claude Opus 4.5 and 4.6, GPT-5.2, Gemini 3 Pro Preview, Kimi K2 Thinking, GLM-5, DeepSeek V3.2, MiniMax-M2.1 and M2.5, and o3.

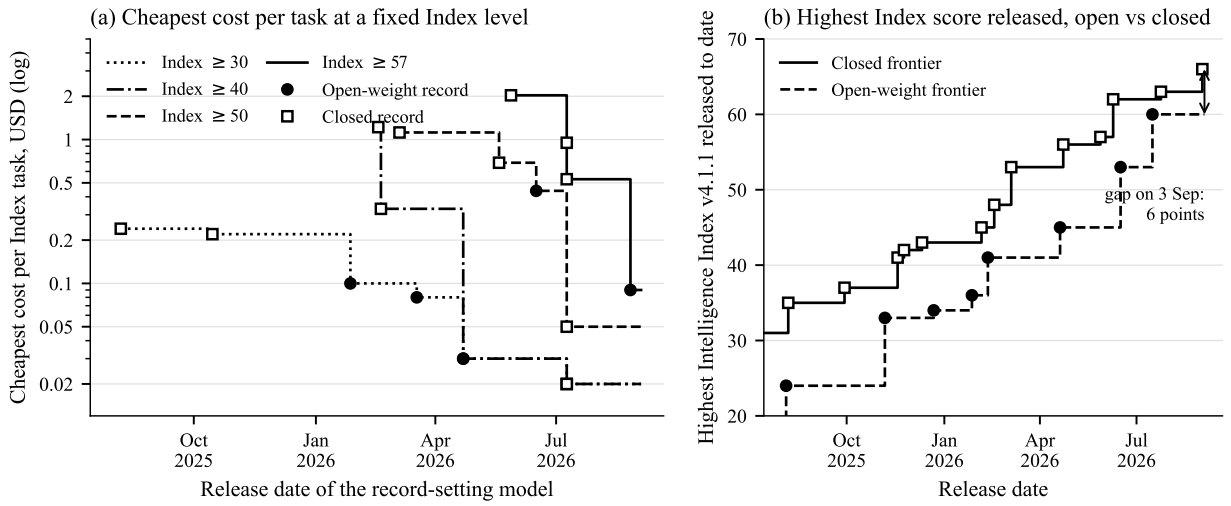

Figure 2. Reported release-cohort reconstruction. (a) Lowest September 3 cost per Index task among sampled configurations released by each cutoff and meeting four score thresholds, USD on a log scale. (b) Highest v4.1.1 score among sampled configurations released by each cutoff. This is not a series of prices paid at those dates.

Sources: Artificial Analysis (2026b) (third-party evaluation and provider release dates, fetched 3 September 2026); Ornn Data reconstruction. Cost per task is at the price posted on 3 September 2026; the ten configurations without one enter panel (b) only.

Table 5. Reported sample-minimum costs by release cohort at fixed Index thresholds; v4.1.1 scores and September 3 prices.

Threshold	First qualifying sampled model (release date, $ per task)	Cheapest sampled model (release date, $ per task)	Factor	Months apart
≥ 30	GPT-5 high (7 Aug 2025, 0.24)	GPT-5.6 Luna high (9 Jul 2026, 0.02)	12	11.0
≥ 40	Claude Sonnet 4.6 (17 Feb 2026, 1.22)	GPT-5.6 Luna high (9 Jul 2026, 0.02)	61	4.7
≥ 50	GPT-5.4 xhigh (5 Mar 2026, 1.12)	GPT-5.6 Luna max (9 Jul 2026, 0.05)	22	4.1
≥ 57	Claude Opus 4.8 (28 May 2026, 2.03)	GLM-5.3-Flash (26 Aug 2026, 0.09)	23	3.0

Source: reported Ornn Data reconstruction from Artificial Analysis (2026b). "Factor" divides the two displayed costs; "months apart" measures the interval between release dates. Both costs use the September 3 snapshot, not historical prices. Ten configurations without costs are excluded; omitted models or reasoning settings may change the minima. The complete source extract is required for independent reproduction.

are 61, 22, and 23 over shorter release intervals. Figure 2(a) shows how newer sampled models provide lower-cost ways to meet these score thresholds at the common price date.

Open-weight models also matched much of the preceding closed frontier. In Figure 2(b), the closed frontier rises from 35 in August 2025 to 66 in September 2026, while the open frontier rises from 24 to 60 by July 2026. The gap ranges from two to seventeen points and ends at six. Eleven of fourteen closed-frontier scores were subsequently matched by an open model, after 1.6 to 6.7 months and a median of 3.9 months among those eleven matches. Three scores remain unmatched at the observation cutoff. The result describes substantial catch-up within the sample alongside a continuing closed-model lead.

These comparisons establish why model choice belongs at the center of inference economics. A workload with a fixed capability requirement can have markedly different benchmark costs depending on the model selected. Open weights add another decision: which hardware should serve that model? The next subsection examines the cost of that choice.

3.5 Self-hosted compute cost

The preceding comparisons use hosted API prices. Open weights also allow operators to rent hardware and serve the model themselves, raising a further question: at what utilization does this become cheaper? Unlike a hosted API, self-hosting has no posted per-token price, so its compute cost must be constructed from GPU rent, throughput, and utilization. Equation (1) converts hourly rent into a cost per million output tokens, allowing direct comparisons across GPU families and with hosted API prices:

$$\text{Compute cost per million tokens} = \frac{\text{GPU-hour price}}{\text{Millions of tokens per GPU-hour} \times (1-h) \times u}, \tag{1}$$

where h is the share of capacity reserved for bursts, failures, and maintenance, and u is productive utilization of the remaining capacity. The calculation combines index rents with published throughput, using an estimate for dense-model A100. The base case sets $u = 0.5$ and $h = 0.15$; the full-utilization case sets $u = 1$ and $h = 0$. These operating assumptions translate the throughput inputs into a common cost framework.

Table 6 reports inputs for two workloads: dense Llama-2-70B and sparse gpt-oss-120b. They provide a comparison of hardware costs for these models, with gpt-oss-120b at Index 24 in Table 3. Where available, throughput comes from closed-division MLPerf Inference results, which specify the model and accuracy target (MLCommons, 2025).

For Llama-2-70B, H100, H200, and B200 throughput comes from NVIDIA's MLPerf v4.1 submissions (MLCommons, 2024). A100 throughput is estimated at 0.323 times the H100 result using NVIDIA's same-model TensorRT-LLM comparison (NVIDIA, 2024). Section 3.6 and Appendix A examine sensitivity to that ratio.

For gpt-oss-120b, A100 and H100 throughput comes from GPUStack baseline runs on single 80 GB devices (GPUStack, 2026). These use the vLLM versions shown in the table and unbounded concurrency, with p99 times to first token of approximately 59.7 and 54.0 seconds. H200 and B200 throughput comes from Red Hat's MLPerf v6.0 submissions (MLCommons, 2026). The table notes distinguish the serving setups; cross-source comparisons are indicative.

Table 7 applies equation (1) to those inputs at the spot index; Section 5.4 repeats the calculation at the forward marks.

The calculation produces three useful results. First, the baseline dense-model comparison favors the H100. Llama-2-70B costs \$0.75 per million output tokens on the A100 against \$0.68 on the H100, using the mixed-precision throughput ratio described above.

Second, the ordering reverses for gpt-oss-120b. The A100's base-case cost is \$0.29 per million output tokens, less than half the H100's \$0.64. In the published runs, A100 throughput is 78 percent of H100 throughput, compared with the 32 percent ratio used for the dense model. The smaller performance gap, combined with the lower rent, gives the older GPU a clear cost advantage in this calculation. Hardware generation alone is therefore a poor guide to token-production cost.

Third, the A100 reaches the hosted output-price break-even at substantially lower utilization. With 15 percent headroom, it matches the \$0.60 gpt-oss-120b output median at 24 percent productive utilization, against 53 percent for H100. The cheaper \$0.17 listing is harder to beat: A100 requires approximately 84 percent utilization, while H100 would require 188 percent and therefore cannot reach that break-even. These output-price thresholds show why both hardware choice and the available hosted alternative matter. Section 3.6 adds operating costs to the comparison.

Table 6. Per-GPU output throughput and hourly rents used in the self-hosted cost calculation (benchmark, third-party, estimated, and observed data).

GPU	Source	Offline tok/s	Server tok/s	Offline M tok per GPU-hr	Spot rent $/GPU-hr
Panel A: Llama-2-70B (dense)					
A100 SXM4	0.323× H100 (TensorRT-LLM v0.12.0 same-model ratio)	990	872	3.56	1.00
H100 SXM	MLPerf v4.1, NVIDIA	3,066	2,701	11.04	2.83
H200	MLPerf v4.1, NVIDIA	3,913	3,654	14.09	4.51
B200	MLPerf v4.1, NVIDIA	11,264	10,756	40.55	6.40
Panel B: gpt-oss-120b (mixture of experts, 5.1B active)					
A100 SXM4 80 GB	GPUStack, vLLM 0.11.0, one device	2,276	—	8.19	1.00
H100 SXM	GPUStack, vLLM 0.10.2, one device	2,901	—	10.44	2.83
H200	MLPerf v6.0, Red Hat	3,585	3,013	12.91	4.51
B200	MLPerf v6.0, Red Hat	11,634	8,949	41.88	6.40

Sources: MLCommons (2024) (NVIDIA DGX-H100 8x, H200-SXM-141GB 8x, B200-SXM-180GB 1x; llama2-70b-99.9), MLCommons (2026) (Red Hat 8xH200 and 8xB200, LLM-D on OpenShift, MXFP4; gpt-oss-120b), NVIDIA (2024) (Llama-2-70B, TP8, 128 in / 2,048 out; A100 FP16 5,666 tok/s and H100 FP8 17,535 tok/s per eight-GPU system), GPUStack (2026) (ShareGPT, 1,000 prompts, unbounded concurrency, single device). Per-GPU MLPerf figures are Ornn Data derivations from system-level results (eight-GPU results divided by eight); the A100 row of Panel A is an Ornn Data estimate; the GPUStack rows are third-party measurements of an offline-like kind. None of the derived, estimated, or third-party figures is an MLPerf result, and none has been verified by MLCommons. Panel B MLPerf rows use a different submitter and serving stack from Panel A and are not comparable across panels. Offline is maximum throughput with no latency bound; Server is throughput under the benchmark's latency constraints. Spot rent is the Ornn daily index value for 1 September 2026 20:00 UTC (Ornn, 2026b).

3.6 Sensitivities and scope

Utilization, overhead, and the serving configuration determine how the compute-cost advantage translates into a deployment budget. The following scenarios examine those drivers. Additional costs include any host, network, storage, operations, and labor expenses not already bundled in rent; owned systems also require electricity, facilities, and capital recovery.

Utilization. The first source of variation is productive utilization, whose effect is shown in Figure 3. Below the break-even utilizations given above, a hosted gpt-oss-120b endpoint at the median list price is cheaper than rented hardware, before any non-compute cost is added.

Overhead. Adding 25 to 60 percent to compute cost gives A100 base-case costs of $0.94–$1.20 per million output tokens for the dense model and $0.36–$0.46 for gpt-oss-120b. The sparse-model advantage survives against the $0.60 hosted output median across this range. Against the $0.17 listing, even a 25 percent markup raises the required utilization above 100 percent. The choice of hosted comparator therefore has a direct effect on the self-hosting decision.

Throughput and service requirements. Precision can reverse the dense-model ranking. The baseline A100/H100 ratio of 0.323 compares A100 FP16 with H100 FP8; varying it from 0.29 to 0.35 gives A100 costs of $0.69–$0.83. Rescaling with the vendor's same-precision ratios of 0.42–0.68 instead gives $0.36–

Table 7. Compute-only cost per million output tokens at the 1 September 2026 spot index (USD; calculated).

Hardware	Full use, $u = 1$, $h = 0$	Base case, $u = 0.5$, $h = 0.15$
Panel A: Llama-2-70B (dense)		
A100 SXM4 (est.)	0.28	0.75
H100 SXM	0.26	0.68
H200	0.32	0.81
B200	0.16	0.39
Panel B: gpt-oss-120b (mixture of experts, 5.1B active)		
A100 SXM4 (third-party)	0.12	0.29
H100 SXM (third-party)	0.27	0.64
H200	0.35	0.98
B200	0.15	0.47
Hosted gpt-oss-120b, output list price	0.60 (median across providers; Groq, Together); 0.17 lowest OpenRouter listing	

Source: Ornn Data calculation from Table 6 and equation (1), using September 1 spot rents (Ornn, 2026b). Full use takes Offline throughput with no headroom. The base case takes Server throughput where available and otherwise rescales the GPUStack baseline; the latter does not establish a Server service level. Input processing is included in the benchmark workload, while the denominator counts output tokens; a hosted bill would also include input charges. Dense A100 is estimated, and the GPUStack rows are third-party measurements, not MLPerf results. Costs exclude the items discussed in Section 3.6.

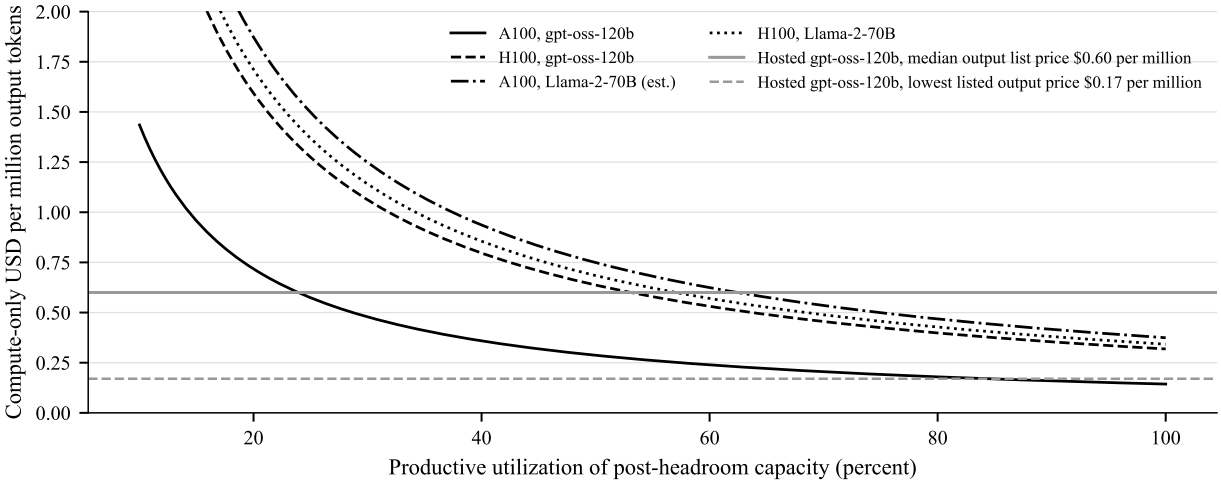

Figure 3. Compute-only cost per million output tokens as a function of productive utilization for A100 and H100, using 1 September 2026 Ornn spot rents and 15 percent reserved headroom, for gpt-oss-120b (third-party single-device throughput) and Llama-2-70B (MLPerf v4.1 server throughput; A100 estimated). The upper horizontal line is the hosted gpt-oss-120b median output list price ($0.60 per million tokens); the lower is the lowest OpenRouter listing ($0.17).

Sources: GPUStack (2026); MLCommons (2024); NVIDIA (2024); Ornn (2026b); Groq (2026); Together AI (2026); OpenRouter (2026a); Ornn Data calculation. USD per million output tokens; compute only.

$0.58, below the H100's $0.68 (NVIDIA, 2024). These scenarios show why numerical format and serving configuration belong alongside rent in the hardware decision.

Electricity. GPU-attributed energy accounts for 1.6 to 4.2 percent of spot rent at nameplate GPU power, power-usage effectiveness of 1.3, and an assumed electricity price of $0.08 per kilowatt-hour. The share is highest for the A100, at about $0.04 per GPU-hour (Appendix A; NVIDIA, 2026a). The remainder contributes toward host and network power, facilities, maintenance, capital, and other costs. An owner's operating decision also depends on the value of the rack and power allocation and the alternative of selling the equipment.

Choosing closed access. Closed access remains part of the cost-minimizing choice. It supplies the cheapest qualifying option at some thresholds in Table 4, and its frontier exceeds the open sample's Index 60. GPT-6 Astra extends that case at the top of the range: on Index v4.2 it clears scores no open model in Panel C reaches, at a lower cost per task than the other closed models at those scores, although at a token rate 2.5 times Sol's and with smaller subscription allowances (Section 2). It can also offer a better fit where usage is low or hosted caching, reliability, and latency meet the workload's needs more efficiently. Model capability and the economics of deployment should be assessed together.

4 Open-weight workload portability and demand

4.1 Portability across providers and hardware

For cost differences to influence hardware demand, operators must be able to choose where to run their workloads. Closed models are available only through deployments authorized by their developers. Open weights allow independent deployment and, where compatible endpoints exist, a choice of providers serving the same checkpoint. The reported September 1–2 OpenRouter snapshots list eighteen to twenty-two providers for gpt-oss-120b, GLM-5.3-Flash, and Kimi K3, with highest-to-lowest output-price ratios between 1.8 and 5.6 (OpenRouter, 2026a,c,d). The reported volume-weighted gpt-oss-120b output price was $0.39 per million tokens, against a $0.60 provider median. These figures illustrate price variation, not interchangeable service: quantization, capacity, reliability, and migration costs can limit substitution. Portability expands the customer's options without making supply unlimited or switching costless.

The same choice extends to hardware. Memory, numerical format, software, and licensing determine which deployments are feasible. Eight 80 GB GPUs provide 640 GB in aggregate, a weights-only ceiling of approximately 320 billion parameters at two bytes per parameter; cache, buffers, and runtime overhead reduce the practical limit. Quantization widens the available choices. gpt-oss-120b uses MXFP4 expert weights and has been served on a single 80 GB A100 through vLLM (OpenAI, 2025, 2026a; GPUStack, 2026). Operators can therefore consider older hardware where the model fits, the software supports it efficiently, and the license permits the intended use. The rental comparisons that follow cover NVIDIA families.

The economic distinction is therefore one of who can make the deployment decision. Open weights let independent operators weigh hardware rent against performance on their own workloads. Section 3.5 shows the value of that choice: the A100 is cheaper in the sparse-model calculation, while the baseline dense-model comparison favors H100. The right hardware depends on the work being purchased.

4.2 Throughput-bound workloads

Workload timing determines how operators use this flexibility. Interactive serving places a premium on response time and can benefit from newer hardware's bandwidth and processing capabilities (NVIDIA,

2026a). Reserving capacity for bursts also lowers productive utilization. For these deployments, the relevant comparison is cost at the required response time and reliability.

Agent rollouts, batch evaluations, and offline generation offer opportunities to trade scheduling flexibility for lower cost. An agent rollout combines model calls and tool execution; in reinforcement learning from verifiable rewards, generated trajectories are scored and used to update a policy. Jobs that can be batched, distributed, or restarted at acceptable cost are candidates for otherwise idle or preemptible capacity. Scheduling must still accommodate sequential calls, training synchronization, and deadlines. The objective is lower cost and elapsed time to a useful outcome.

Batch size and model architecture then determine which hardware benefits. Greater weight reuse increases the importance of arithmetic, while expert routing, memory traffic, and communication shape sparse-model performance. The gpt-oss-120b result shows that a workload can retain much of its throughput on older hardware while benefiting from a substantially lower rental price. This is the economic opening that portability makes available to operators.

4.3 The scale of the workload

These workload characteristics explain why some models can run competitively on older GPUs. Whether they can sustain rental earnings also depends on the scale of demand. Public measures of this demand are indirect, with the most specific evidence coming from Modal, a serverless compute platform that sells GPU and CPU capacity by the second and offers a sandbox product for running untrusted code. On 21 May 2026 it reported "surpassing $300 million in annualized revenue" after "growing fivefold since September," and that "sandboxes already drive more than a third of our revenue" (Modal Labs, 2026), which puts the sandbox product alone above $100 million of annualized revenue. The sandboxes are used for reinforcement learning environments, coding agents, and "autoresearch agents that run their own training experiments at scale," and the company attributes its growth to a shift "from frontier APIs to model ownership" as "open-weight models from DeepSeek, Qwen and others have reached production quality" (Modal Labs, 2026).

Platform usage provides another view of demand. On OpenRouter, input-plus-output token volume attributed to agentic use exceeded that attributed to human use around 1 February 2026, using its API-key classification. Open weights accounted for about one third of platform token volume by late 2025, and DeepSeek's share doubled between January and June 2026 (OpenRouter, 2026e; OpenRouter and a16z, 2026). Menlo Ventures' enterprise survey measures spending instead, placing the open-source share at 11 percent in 2025, down from 19 percent in 2024 (Menlo Ventures, 2025). These populations and units differ: lower-priced models can represent a larger share of tokens than of expenditure. Together, the sources place open-weight use within a substantial market for inference and execution.

5 Evidence from Ornn forward curves and spot index

The rental data add the hardware-market perspective. Prices and occupancy show how older capacity is used, while term marks describe the price of access over longer commitments. The A100 provides the clearest example of continued demand across these measures.

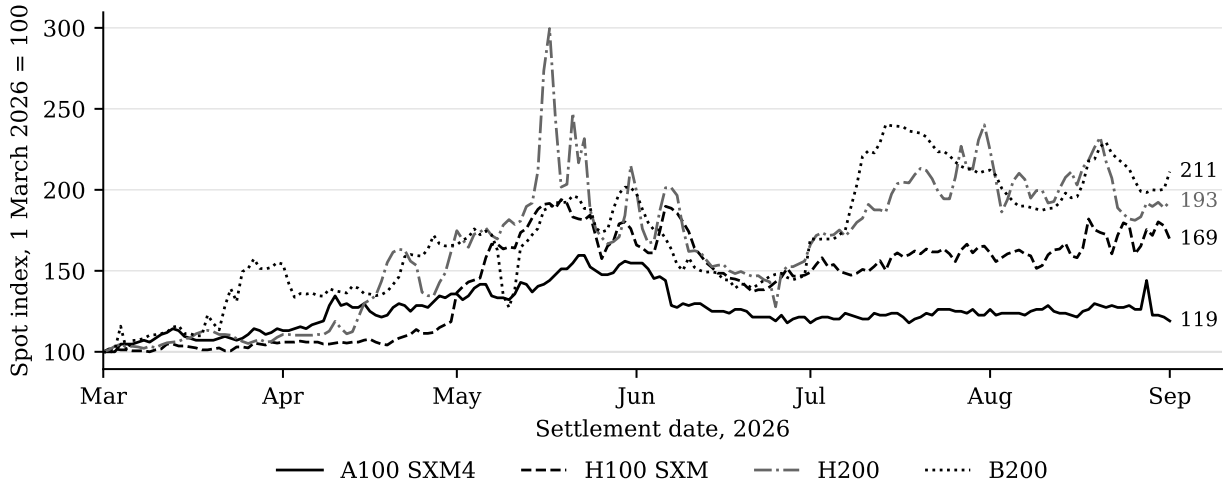

Figure 4. Ornn settled daily spot index for A100 SXM4, H100 SXM, H200, and B200, 1 March to 1 September 2026, each series normalized to its 1 March 2026 value (= 100). End-of-period values are marked at the right.
Source: Ornn (2026b), retrieved via the Ornn Data API on 1 September 2026; calculated normalization. Daily settlements at 20:00 UTC (21:00 UTC before 8 March). B300 has no public spot index.

5.1 Data and definitions

Ornn publishes a settled daily rental index, referred to here as the spot index, and term-price curves at one-month, six-month, one-year, three-year, and five-year tenors (Ornn, 2026b,a).[2] The analysis uses August 13, 2026 term marks for A100, H100, H200, and B200, September 1 marks for B300, and September 1 spot values. The spot methodology uses executed on-demand transactions; analysts assemble the term curves from active deals (Ornn, 2026b,d). Family-age comparisons use an illustrative September 1 start, with mark vintages identified in the table notes.

Two price measures organize the comparison. A *term price* is a flat GPU-hour rate over a stated contract tenor. An *implied forward* differences cumulative term costs to obtain a rate for the interval between two tenors, assuming a common start and comparable contract specifications (Appendix A). The latter describes the structure of term pricing rather than forecasting future spot rent.

The unit of analysis is access to a GPU family. *Product-family age* runs from the vendor's announcement (NVIDIA, 2020–2025), and *contract tenor* measures the duration of access. Devices supplying that access can be replaced over time. The marks therefore price a rental service; an individual device's resale value and operating life are separate quantities.

Contract terms help explain why term prices differ from spot prices. Committed utilization, earlier payment, and revenue that supports financing can justify a discount; guaranteed access and flexibility can justify a premium. These channels provide an economic interpretation of curve shape alongside expectations about future demand. Their separate contributions are not estimated here.

5.2 Occupancy, listed supply, and owner economics

Prices alone do not show how much capacity is being rented. Alongside the spot index, which is a clearing price, Ornn reports the share of tracked on-demand capacity that is rented, referred to here as *occupancy*

[2]Here "settlement" denotes the fixing of the daily index value, not settlement of a traded contract.

Table 8. A100 occupancy, listed capacity, and spot price in the Ornn on-demand index, March to September 2026.

Family	Occupancy, 1 Mar 2026	Occupancy, 1 Sep 2026	Occupancy, Mar–Sep mean	Listed capacity, Mar–Sep	Spot, Mar–Sep
A100 SXM4	74%	90%	80%	+13%	+20%

Sources: Ornn (2026c,b), retrieved through the Ornn Data API on 2 September 2026. Occupancy is rented capacity divided by listed capacity across tracked on-demand providers in the global region. Listed-capacity and spot changes compare the 1 March and 1 September settlements. Listed capacity measures tracked on-demand supply, not the installed base.

(Ornn, 2026c). Occupancy is a rental-market measure: it indicates whether listed capacity has a paying tenant, not how intensively that tenant uses the GPU, and it is distinct from the productive utilization u of equation (1). Table 8 reports A100 occupancy and listed capacity over the window in which the marks were set.

The A100 occupancy series shows strong use of tracked capacity. Occupancy averaged 80 percent over the six-month window and reached 90 percent on September 1, up from 74 percent on March 1.

This increase occurred alongside an expansion in supply. Listings rose 13 percent, and combining that increase with the occupancy change implies approximately 37 percent growth in rented capacity. More A100 capacity was available in the tracked market, and a larger share of it was rented. The newer families show different listing patterns, with Hopper capacity contracting and Blackwell capacity expanding.

Rising occupancy on an expanding base is the important result. The A100's spot strength cannot be attributed solely to fewer tracked GPUs being available for rent. The series measures the on-demand market covered by the index; the scope of that coverage is discussed in Section 7.

An owner-side illustration. At a constant rent of $1.00 per GPU-hour, occupancy of 80 percent, and GPU-attributed energy cost of $0.042 per calendar hour, the operating contribution is approximately $553 per month before all other costs. Holding those inputs fixed for five years gives approximately $33,200. This constant-rent scenario quantifies the contribution available toward acquisition, maintenance, facilities, and financing costs. It provides a starting point for an owner's budget rather than a forecast of net profit.

5.3 The marks

Before comparing long-term retention, it is useful to consider the reported spot changes. Figure 4 shows increases of 70 to 111 percent for the newer families and about a fifth for the A100 over six months. These movements provide context, but spot prices and one-month term marks are different observations. Percentage retention also depends on each family's starting level, so a flatter normalized curve need not mean a higher absolute rent or a slower lifetime decline. Table 9 reports each tenor as a percentage of the family's one-month term price, together with the implied-forward retention ratio for months 37 to 60 and family age at the end of a five-year contract beginning 1 September 2026. Figure 5 presents the same normalization, allowing the shapes of the curves to be compared independently of their price levels.

The long end of the curve adds a different perspective from current occupancy. It shows how the price of access changes with the length of commitment, including the contractual risks and benefits described in Section 5.1.

Table 9 shows that the A100's five-year term price retains four-fifths of the one-month price for a contract that would run to September 2031, when the family will be more than eleven years old; the implied forward for the final two years retains nearly three-quarters. The Hopper and Blackwell curves are steeper: they retain between 44 and 60 percent at five years and less in the implied forward. One-year retention is above 80

Table 9. Ornn forward term-price retention by contract tenor (percentage of one-month term price).

Family	6M/1M	1Y/1M	3Y/1M	5Y/1M	Fwd 37–60/1M	Age at 5Y end (yrs)
A100 SXM4	95.0%	93.1%	84.2%	80.2%	74%	11.3
H100 SXM	95.0%	86.2%	68.2%	59.8%	47%	9.4
H200	92.4%	80.2%	51.0%	43.7%	33%	7.8
B200	99.1%	97.9%	69.1%	53.8%	31%	7.5
B300	93.8%	81.3%	60.9%	53.8%	43%	6.5

Source: Ornn (2026a) (reported marks). A100, H100, H200, and B200 marks were published August 13; B300 September 1. Retention and implied-forward ratios are calculated from those marks. Ages use an illustrative September 1, 2026 start and September 1, 2031 end, measured from family announcement dates (NVIDIA, 2020–2025); this does not change the marks' publication dates.

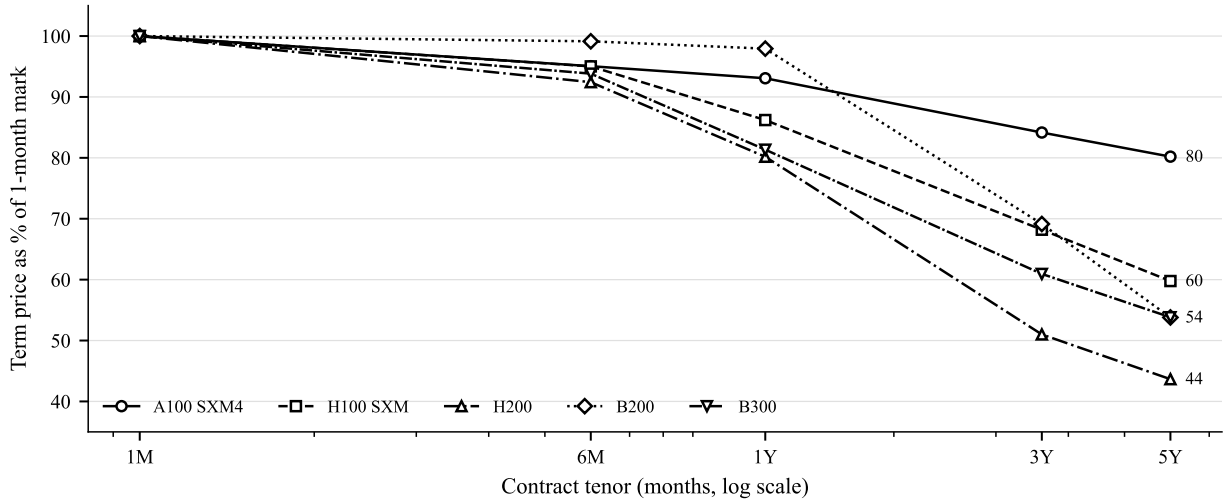

Figure 5. Ornn forward term prices as a percentage of each family's one-month term price, by contract tenor.
Source: Ornn (2026a); Ornn Data calculation. Publication dates as in Table 9.

percent for every family and orders the families differently from five-year retention, so the long end carries information the short end does not.

The oldest family has the strongest five-year percentage retention in the table. This is evidence of continued pricing for A100 access well beyond the introduction of its successors. It makes the economic role of older capacity a question of the work and rental commitments it can attract, rather than its generation alone.

5.4 Per-token cost at the forward marks

Translating the term marks into token costs makes the workload comparison explicit. Table 10 applies equation (1) to dense Llama-2-70B and sparse gpt-oss-120b, holding throughput fixed with equal utilization and headroom across families. It asks which hardware produces output most cheaply at each term-price basis. The table note records the input requirements and differences between serving setups.

The baseline dense-model calculation favors newer hardware: A100 is the most expensive family per output token at each term-price basis. Strong rental retention and low token cost are therefore different

Table 10. Relative compute-only cost per million output tokens at the three-year and five-year term prices and at the implied forward for months 37 to 60, by family and workload (A100 = 100; calculated; $u = 0.5$, $h = 0.15$).

	Llama-2-70B (dense)			gpt-oss-120b (5.1B active)		
Family	3Y term	5Y term	Fwd 37–60	3Y term	5Y term	Fwd 37–60
A100 SXM4	100 (est.)	100 (est.)	100 (est.)	100	100	100
H100 SXM	68	62	53	165	152	129
H200	64	57	46	201	181	147
B200	38	31	19	119	97	60

Source: reported Ornn Data calculation using August 13 term marks (Ornn, 2026a) and equation (1). Each cell indexes token cost to A100 at the same tenor, with equal $u = 0.5$ and $h = 0.15$. Server throughput is used where available; A100 and H100 gpt-oss-120b use GPUStack baseline throughput instead. Dense A100 throughput is estimated. The cross-family one-month price ratios are not supplied, so the table cannot be independently reproduced from Table 9. Comparisons across GPUStack and MLPerf setups are indicative only. The implied-forward formula and its assumptions are given in Appendix A.

properties. Section 3.6 shows how the dense-model result changes with the precision and throughput assumptions.

For gpt-oss-120b, the A100 retains its calculated advantage over the H100 at the three-year and five-year marks and at the implied forward. The advantage is therefore present across several rental tenors, not only at spot. B200 is cheapest at the implied forward in the cross-source comparison. The table shows how older and newer hardware can each offer an advantage, depending on the workload and the price of access.

Together, the cost and rental observations give older hardware a clear place in the analysis. The A100 is competitive on the sparse-model workload, its tracked rented capacity has grown, and its long-term price retention exceeds that of newer families. Open-weight portability provides a channel through which cost-sensitive workloads can take advantage of this capacity. The distinction between that mechanism and causal attribution is addressed in Section 7.

6 Discussion

Figure 6 brings the argument together. Open weights turn hardware selection into a decision available to independent operators. A model that delivers the required performance on an older GPU gives those operators a reason to rent it. The scale of that work and the supply of competing capacity determine how much demand reaches the older family. This is the connection between inference economics and continued rental earnings.

Three findings make this connection economically important. Open-weight models offer lower benchmark costs at several common Index thresholds. The A100 produces gpt-oss-120b output at less than half the H100's calculated base-case cost. And Ornn's data show rising rented A100 capacity alongside stronger five-year price retention than the newer families. These results put workload economics, rather than generation alone, at the center of the hardware-usefulness question.

For users of compute, this means choosing the model and deployment together. The cheapest qualifying model provides the starting point; hosted pricing, utilization, operating costs, and service requirements determine where to run it. The gpt-oss-120b comparison shows that older hardware can reach an attractive output cost at substantially lower utilization than newer hardware. The lower-priced hosted alternatives also show the value of comparing the full set of available options.

For owners and financiers, the useful question is which workloads will continue to pay for access to the fleet. The A100's five-year mark retains 80 percent of its one-month price, showing substantial term pricing

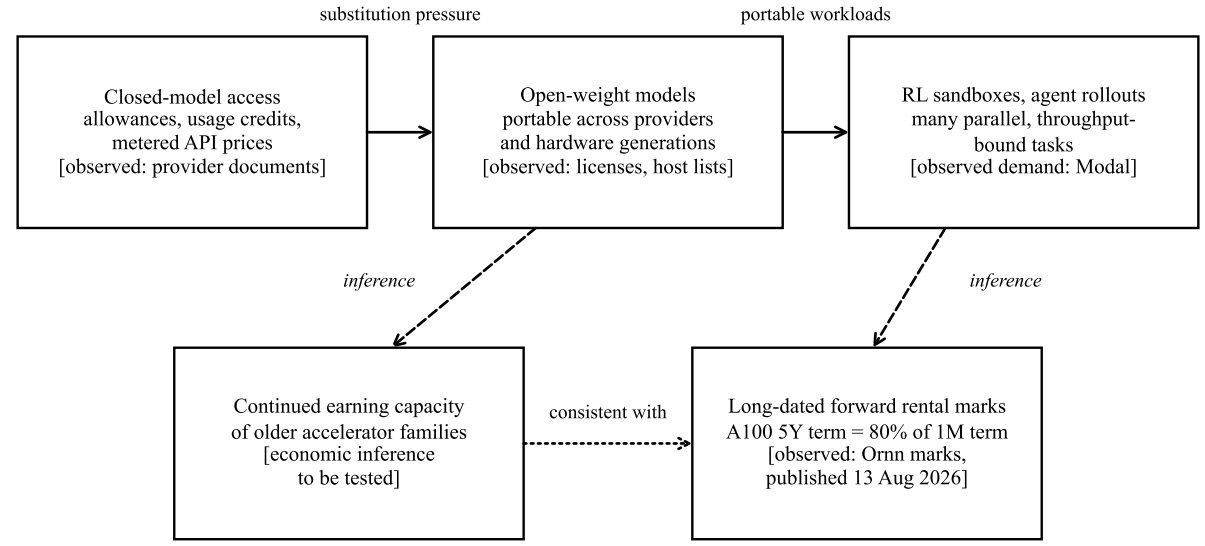

Figure 6. The economic mechanism: open weights expand deployment choices, competitive workloads create demand for compatible hardware, and rental markets price access to that capacity.

for an older family. Evaluating an individual investment then requires acquisition costs, operating cash flows, and the alternatives available to the owner. As a result, financing and contract terms are necessary in order to build a complete picture.

Older hardware does not need to win every workload to retain an economic role. Newer GPUs can offer lower costs on demanding tasks while older capacity serves work suited to its memory, software, and rental price. Availability, migration costs, and service requirements also influence that allocation. A market with several hardware generations is therefore compatible with continued technical progress.

The central conclusion is that a new generation does not, by itself, make its predecessors economically obsolete. Older GPUs retain a role when useful workloads remain competitive on them, and open weights give independent operators the freedom to place that work there. The right measure of hardware usefulness is the cost of the work it can serve and the demand for that work, not simply the age of the chip.

7 Limitations

The paper combines benchmark costs, hardware-cost calculations, and family-level rental observations. The following boundaries define how those results should be used.

Forward-curve provenance. Analysts assemble the term curves from active deals, as described in Section 5.1. Separating financing, scarcity, and expected-rent components requires the underlying contract terms. The data-availability statement lists the missing inputs needed for reproduction.

Occupancy and listed supply. The series records rental status across tracked on-demand providers, not the installed base, tenant workloads, or realized token throughput. Provider entry, customer composition, and shifts between on-demand and term contracts can affect the totals. A fixed-provider panel and workload-tagged GPU-hours would separate these effects. September 1 occupancy does not establish occupancy on the

August 13 mark date, and token shares or platform revenue elsewhere in the paper do not measure GPU-hours by generation.

Family age, not device age. Ages run from NVIDIA's family announcement, not a device's commissioning date. A rental commitment can be fulfilled by replacement devices over its term. Strong percentage retention can also follow earlier declines in the starting rent. The marks therefore do not establish individual-device survival, profitable operating life, residual value, or the appropriateness of an owner's accounting depreciation schedule.

Throughput inputs. Dense A100 throughput is estimated from a vendor ratio; alternative precision assumptions can reverse its cost ranking. The GPUStack runs use different engine versions and unbounded concurrency, and H200/B200 use another serving setup. The utilization rescaling is an accounting scenario, not a measured throughput–latency curve. Small active-parameter counts alone do not establish a bandwidth bottleneck. Extending the result to other models or reinforcement-learning workloads requires common accuracy and service targets, profiling, and end-to-end costs including learner waiting and trajectory quality.

Index and prices. A common Index threshold screens models but does not establish equal task success; a one- or two-point difference is not a validated equivalence margin. Sample minima may change with omitted models, reasoning settings, providers, or service levels. AA's costs include evaluation weights and cache treatment; no additional batch or negotiated discount is applied here. Section 3.4 values release cohorts at one price date, so its ratios are not historical price declines or recurring halving rates. Missing costs can change either end of a comparison. The catch-up median covers eleven matches and excludes three unresolved observations, whose eventual lags remain unknown.

Fully loaded cost. The overhead range is a scenario, not an empirical bound. Fixed expenses and items already bundled in rent require a deployment-specific budget. Hosted output-price comparisons also exclude input charges. The two self-hosted examples do not establish the performance of the higher-scoring models in the hosted comparison, and no common successful-work comparison has been run across deployment modes. The owner-side calculation holds spot rent and occupancy constant; it is neither historical earnings nor a term-implied profit forecast.

Causality. The rental observations do not identify the contribution of open-weight demand. Device retirements, dense inference, non-LLM work, operating withdrawal costs, and financing or scarcity effects remain alternative explanations. A competitive workload alone does not guarantee enough demand to absorb supply or establish a rental floor. A stronger test would compare workload-tagged capacity exposed to open-weight releases with comparable unexposed capacity, controlling for supply and service changes. The mechanism would be weakened if lower rents failed to attract additional eligible work, or if the cost advantage disappeared under common quality and service requirements.

Disclosures and reproducibility

Ornn Data publishes this paper and the rental index, occupancy series, and term marks used in Section 5. It licenses data commercially and also offers GPU rentals through Ornn Compute (Ornn, 2026d). These activities create a commercial interest in both the interpretation and adoption of its data. Independent review of the underlying records would help readers assess that relationship. There is no external funding to report for this analysis.

The spot index uses executed on-demand transactions under the cited methodology; occupancy measures the rented share of tracked listings. Term marks come from curves assembled by analysts from active deals, as described in Section 5.1. They may be revised. Statements about future demand and rental earnings are

conditional interpretations, not valuations of individual devices or conclusions about any company's financial reporting. Nothing in this paper is an offer to trade or investment, accounting, tax, or legal advice. Inclusion of a source carries no endorsement. Product and company names are the trademarks of their owners. MLPerf is a trademark of MLCommons; the derived and estimated figures reported here are not MLPerf results and have not been verified by MLCommons.

The manuscript reports Ornn price and forward retrievals on 1 September 2026 and occupancy retrievals on 2 September. The raw extracts, complete model-configuration history, calculation scripts, and referenced figure files are not included in the supplied manuscript package, so full independent reproduction is not currently possible. The archive needed for that purpose should preserve timestamped source responses, model and reasoning settings, AA version and cost fields, benchmark configurations, cross-family term-price ratios, and every transformation used to generate the results. Table 4 is reproducible directly from Table 3; other calculations can be checked from printed inputs where those inputs are complete. No missing observations have been replaced with inferred raw data. Table and figure notes distinguish reported observations, calculations, benchmark derivations, provider statements, and estimates, in the interest of transparent reporting (Taylor and Kuyatt, 1994; AEA, 2026). Live source pages can change; a fresh retrieval is not a replacement for a dated snapshot.

A Calculation notes

Compute-only cost per million output tokens. The calculation uses $c = p/(T \cdot 3600 \cdot u \cdot (1-h)/10^6)$, where p is the rent per GPU-hour, T the per-GPU throughput in output tokens per second, u the productive utilization, and h the capacity held in reserve. The corresponding break-even utilization against a hosted output price P is $u^* = p/(T \cdot 3600 \cdot (1-h) \cdot P/10^6)$.

A100 throughput on Llama-2-70B. In the absence of a comparable A100 submission in the cited MLPerf inputs, the estimate scales H100 throughput (3,066 output tokens per second offline; 2,701 server) by 0.323. This is the A100 FP16/H100 FP8 ratio in NVIDIA's TensorRT-LLM v0.12.0 table at tensor parallelism 8, 128 input tokens, and 2,048 output tokens (NVIDIA, 2024). Varying the ratio from 0.29 to 0.35 gives A100 base-case costs of \$0.69–\$0.83. Ratios of 0.42–0.68 from matched-FP16 comparisons give \$0.36–\$0.58 under the same rescaling, against H100's \$0.68. These scenarios illustrate sensitivity; importing a ratio from another benchmark does not create a matched MLPerf result. At the stated rents, equal token cost requires a throughput ratio of $1/2.83 \approx 0.353$.

gpt-oss-120b throughput. Red Hat's MLPerf v6.0 submissions give per-GPU output throughput of 3,585 offline and 3,013 server tokens per second for H200, and 11,634 and 8,949 for B200. GPUStack's single-device baseline runs give 2,276 for A100 and 2,901 for H100 on ShareGPT at unbounded concurrency, using different vLLM versions. These inputs support the printed arithmetic but not a common service-level comparison. Cross-source comparisons in Table 10 are indicative only.

Implied forward. For flat term prices F_{36} and F_{60} with a common start, the implied rate for months 37–60 is $(60F_{60} - 36F_{36})/24$. The identity also holds for within-family retention ratios: for A100, $(60 \times 0.802 - 36 \times 0.842)/24 = 0.742$. It does not identify expected spot rent. A cross-family token-cost ratio additionally requires p_j/p_{A100} and T_{A100}/T_j, whose product gives the cost ratio when utilization and headroom are equal.

The one-month cross-family price ratios needed to reproduce Table 10 are not supplied; they cannot be recovered independently from within-family retention alone.

Electricity. The calculation uses nameplate power of 400 W for the A100 SXM4, 700 W for the H100 SXM and H200, and 1,000 W for the B200. It assumes a power usage effectiveness of 1.3 and an electricity price of $0.08 per kilowatt-hour. The resulting energy costs per GPU-hour are $0.042, $0.073, $0.073, and $0.104, respectively. These represent 4.2, 2.6, 1.6, and 1.6 percent of the 1 September 2026 spot index.

Owner-side scenario. The monthly contribution before costs other than GPU-attributed energy is $(p \times o - e) \times 730$, where p is on-demand rent, o occupancy, and e energy cost per calendar hour. At $p = 1.00$, $o = 0.80$, and $e = 0.042$, this gives $553.34 per month and $33,200.40 over five years if all assumptions remain fixed. Energy is charged even during unoccupied hours in this scenario. It is not a term-contract revenue calculation or a capital-recovery estimate. Separately, $5 \times 0.802 = 4.01$ expresses five-year gross term pricing in units of one-month-priced access; an absolute price level and contract terms are needed to turn it into cash flow.

Family age. Family age is measured from NVIDIA's announcement: A100 May 2020, H100 March 2022, H200 November 2023, B200 March 2024, and B300 March 2025. Announcement and availability dates can differ by product and configuration. These dates do not establish the manufacture, commissioning, or retirement dates of devices supplying a rental contract.

References

Alibaba Cloud. 2026. "Qwen3.8-Max." Model Studio documentation, pricing by region (Singapore: US$2.00 input, US$6.00 output per million tokens). https://www.alibabacloud.com/help/en/model-studio/qwen3-8-max. Page last updated 2 September 2026. Accessed 2 September 2026.

American Economic Association. 2026. "Data and Code Availability Policy." https://www.aeaweb.org/journals/data/data-code-policy. Accessed 1 September 2026.

Anthropic (@ClaudeDevs). 2026a. Posts on X, 29 August 2026, announcing a permanent 25 percent increase in standard weekly Claude Code limits from 14 September 2026 and a 17 percent reduction relative to the temporary 50 percent increase. https://x.com/ClaudeDevs/status/2093742321473065266.

Anthropic. 2026b. "Claude Code May–August 2026 weekly limits promotion." Anthropic Help Center, article 15910845. Updated 1 September 2026; accessed 1 September 2026. https://support.claude.com/en/articles/15910845.

Anthropic. 2026c. "Manage usage credits for paid Claude plans." Anthropic Help Center, article 12429409. Accessed 1 September 2026. https://support.claude.com/en/articles/12429409-manage-usage-credits-for-paid-claude-plans.

Anthropic. 2026d. "Plans and Pricing." Accessed 1 September 2026. https://claude.com/pricing.

Anthropic. 2026e. "Pricing." Claude Developer Platform documentation. Accessed 1 September 2026. https://platform.claude.com/docs/en/about-claude/pricing.

Anthropic. 2026f. "Manage costs effectively." Claude Code documentation. Accessed 1 September 2026. https://code.claude.com/docs/en/costs.

Anthropic. 2026g. "Usage limit best practices" (2 June 2026) and "Models, usage, and limits in Claude Code" (15 April 2026). Anthropic Help Center, articles 9797557 and 14552983. Accessed 1 September 2026.

Anthropic. 2026h. "Higher usage limits for Claude and a compute deal with SpaceX." 6 May 2026. https://www.anthropic.com/news/higher-limits-spacex. Accessed 2 September 2026.

Anthropic. 2026i. “Service tiers.” Claude Developer Platform documentation. https://docs.claude.com/en/api/service-tiers. Accessed 2 September 2026.

Anthropic. 2026j. “Rate limits.” Claude Developer Platform documentation. https://docs.claude.com/en/api/rate-limits. Accessed 2 September 2026.

Anthropic. 2026k. “Batch processing.” Claude Developer Platform documentation. https://docs.claude.com/en/docs/build-with-claude/batch-processing. Accessed 2 September 2026.

Artificial Analysis. 2026a. “Artificial Analysis Intelligence Index v4.1.1” (component evaluations and weights) and “LLM Leaderboard” (Intelligence Index, cost per task, median output tokens per second). https://artificialanalysis.ai/models; https://artificialanalysis.ai/leaderboards/models; https://artificialanalysis.ai/models/open-source. Accessed 1 September 2026 (23:53 UTC−4) and 2 September 2026.

Artificial Analysis. 2026b. Per-model pages reporting Intelligence Index, cost to run the Index, cost per task, output tokens generated, price basis, output speed, and release date, for Kimi K3, GLM-5.3, GLM-5.3-Flash, Qwen3.8 2.4T A95B, DeepSeek V4 Pro, MiniMax-M3, gpt-oss-120b, gpt-oss-20b, Claude Fable 5.1, Claude Opus 5, Claude Sonnet 5, GPT-5.6 Sol, GPT-5.6 Terra, GPT-5.6 Luna, and Gemini 3.1 Pro Preview (accessed 2 September 2026), and for the fifty-three models of Section 3.4 released between April 2025 and September 2026 (accessed 3 September 2026); Intelligence Index v4.2 pages for GPT-6 Astra, Claude Fable 5.1, Claude Opus 5, GPT-5.6 Sol, Kimi K3, GLM-5.3, and GLM-5.3-Flash (accessed 5 September 2026). https://artificialanalysis.ai/models/.

Artificial Analysis. 2026c. “Intelligence Benchmarking” (version history and cache-aware cost metrics) and “gpt-oss-120b” (definition of cost per Intelligence Index task). https://artificialanalysis.ai/methodology/intelligence-benchmarking; https://artificialanalysis.ai/models/gpt-oss-120b. Methodology checked 5 September 2026. Live pages use v4.2; they document the metric but do not replace the manuscript’s v4.1.1 observations or its missing dated extracts.

DeepInfra. 2026. “Pricing.” https://deepinfra.com/pricing. Accessed 2 September 2026.

DeepSeek. 2026a. Model cards and licenses for DeepSeek-V4-Pro-0813, DeepSeek-V4-Pro (Preview), and DeepSeek-V4-Flash-0731. https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro-0813; https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro; https://huggingface.co/deepseek-ai/DeepSeek-V4-Flash-0731. Accessed 2 September 2026.

DeepSeek. 2026b. “Models & Pricing.” DeepSeek API documentation (peak and off-peak rates for deepseek-v4-pro and deepseek-v4-flash). https://api-docs.deepseek.com/quick_start/pricing/. Accessed 2 September 2026.

Epoch AI. 2025. “LLM inference prices have fallen rapidly but unequally across tasks.” Data insight, by Ben Cottier, Ben Snodin, David Owen, and Tom Adamczewski, 12 March 2025. https://epoch.ai/data-insights/llm-inference-price-trends. Accessed 3 September 2026.

GPUStack. 2026. “Optimizing GPT-OSS-120B Throughput on NVIDIA A100 GPUs” and the companion H100 page, GPUStack Performance Lab (vLLM v0.11.0 on one A100 SXM 80GB; vLLM v0.10.2 on one H100 SXM; ShareGPT, 1,000 prompts). https://docs.gpustack.ai/latest/performance-lab/gpt-oss-120b/a100/; https://docs.gpustack.ai/latest/performance-lab/gpt-oss-120b/h100/. Accessed 2 September 2026.

Groq. 2026. “Supported Models.” GroqCloud documentation. Accessed 1 September 2026. https://console.groq.com/docs/models.

Mendes, Marcus. 2026. “OpenAI restores 5-hour Codex and Work limits for ChatGPT Plus users.” 9to5Mac, 24 August 2026. https://9to5mac.com/2026/08/24/openai-restores-5-hour-codex-and-work-limits-for-chatgpt-plus-users/. Accessed 2 September 2026. Quotes the X post of Thibault Sottiaux (OpenAI).

Menlo Ventures. 2025. “2025: The State of Generative AI in the Enterprise.” December 2025. https://menlovc.com/perspective/2025-the-state-of-generative-ai-in-the-enterprise/. Accessed 2 September 2026.

Meta. 2025. “Llama 4 Model Card” and “Llama 4 Community License Agreement.” 5 April 2025. https://github.com/meta-llama/llama-models/blob/a9c89c47/models/llama4/MODEL_CARD.md; https://raw.githubusercontent.com/meta-llama/llama-models/main/models/llama4/LICENSE. Accessed 2 September 2026.

MiniMax. 2026a. “MiniMax-M3” model card and MiniMax Community License. https://huggingface.co/MiniMaxAI/MiniMax-M3. Accessed 2 September 2026.

MiniMax. 2026b. “Pricing (pay-as-you-go).” MiniMax Open Platform documentation. https://platform.minimax.io/docs/guides/pricing-paygo. Accessed 2 September 2026.

MLCommons. 2024. *MLPerf Inference v4.1 results*, closed division: NVIDIA DGX-H100 8x H100 SXM 80GB, H200-SXM-141GB 8x, and B200-SXM-180GB 1x, llama2-70b-99.9, Offline and Server performance summaries. https://github.com/mlcommons/inference_results_v4.1. Accessed 1 September 2026.

MLCommons. 2025. *MLPerf Inference Rules*, closed division requirements for model, accuracy target, and scenario definitions. https://github.com/mlcommons/inference_policies/blob/master/inference_rules.adoc. Accessed 1 September 2026.

MLCommons. 2026. *MLPerf Inference v6.0 results*, closed division: Red Hat 8xB200-LLM-D-Openshift and 8xH200-LLM-D-Openshift, gpt-oss-120b, Offline and Server performance summaries. https://github.com/mlcommons/inference_results_v6.0, commit 4d3916a. Accessed 31 August and 1 September 2026.

Modal Labs. 2026. “Modal’s Series C: Raising $355M at a $4.65B valuation.” Modal blog, 21 May 2026. https://modal.com/blog/modal-series-c.

Moonshot AI. 2026. “Kimi-K3” model card and Kimi K3 License. https://huggingface.co/moonshotai/Kimi-K3. Accessed 2 September 2026.

NVIDIA. 2020–2025. Newsroom announcements: “NVIDIA’s New Ampere Data Center GPU in Full Production” (14 May 2020, https://nvidianews.nvidia.com/news/nvidias-new-ampere-data-center-gpu-in-full-production); Hopper architecture and H100 (GTC, 22 March 2022); H200 (SC23, 13 November 2023); Blackwell platform and B200 (GTC, 18 March 2024); Blackwell Ultra and B300 (GTC, 18 March 2025). Dates identify family announcements, not the acquisition cost or commissioning date of a rental device. https://nvidianews.nvidia.com.

NVIDIA. 2024. “TensorRT-LLM v0.12.0 Performance Overview” (Llama-2-70B output tokens per second on A100-SXM4-80GB, H100 80GB HBM3, and H200 141GB HBM3 at matched tensor parallelism and sequence lengths). Release published 29 August 2024. https://raw.githubusercontent.com/NVIDIA/TensorRT-LLM/v0.12.0/docs/source/performance/perf-overview.md. Accessed 2 September 2026.

NVIDIA. 2026a. Product specification pages for A100 Tensor Core GPU (SXM4, 400 W), H100 Tensor Core GPU (SXM, up to 700 W), H200 Tensor Core GPU (SXM, up to 700 W), and HGX B200 (up to 1,000 W per GPU). https://www.nvidia.com/en-us/data-center/. Accessed 1 September 2026.

NVIDIA. 2026b. “NVIDIA-Nemotron-3-Ultra-550B-A55B-NVFP4” model card (release date 4 June 2026; OpenMDW License Agreement v1.1). https://huggingface.co/nvidia/NVIDIA-Nemotron-3-Ultra-550B-A55B-NVFP4. Accessed 2 September 2026.

NVIDIA. 2026c. “NVIDIA to Acquire Hugging Face.” NVIDIA Blog, 3 September 2026. https://blogs.nvidia.com/blog/nvidia-to-acquire-hugging-face/. Accessed 3 September 2026.

OpenAI. 2025. *gpt-oss-120b and gpt-oss-20b Model Card*. August 2025. https://cdn.openai.com/pdf/419b6906-9da6-406c-a19d-1bb078ac7637/oai_gpt-oss_model_card.pdf. Accessed 1 September 2026.

OpenAI. 2026a. gpt-oss repository README and LICENSE (Apache License 2.0). https://github.com/openai/gpt-oss. Accessed 31 August 2026.

OpenAI. 2026b. “Pricing.” OpenAI API documentation. Accessed 1 September 2026; GPT-6 Astra rates accessed 5 September 2026. https://developers.openai.com/api/docs/pricing.

OpenAI. 2026c. “Pricing.” ChatGPT Learn: Codex and ChatGPT Work usage limits by plan and credit rate card. https://learn.chatgpt.com/docs/pricing. Accessed 2 September 2026; GPT-6 Astra message ranges and credit rates accessed 5 September 2026.

OpenAI. 2026d. “About ChatGPT Pro tiers.” OpenAI Help Center, article 9793128. https://help.openai.com/en/articles/9793128. Accessed 2 September 2026.

OpenAI. 2026e. “Business Pricing.” https://openai.com/business/pricing/. Accessed 2 September 2026.

OpenAI. 2026f. “Using Credits for Flexible Usage in ChatGPT (Personal plans).” OpenAI Help Center, article 12642688. https://help.openai.com/en/articles/12642688-using-credits-for-flexible-usage-in-chatgpt-personal-plans. Accessed 2 September 2026.

OpenAI. 2026g. “ChatGPT Rate Card (Business, Enterprise/Edu credit-based pricing).” OpenAI Help Center, article 11481834. https://help.openai.com/en/articles/11481834. Accessed 2 September 2026.

OpenAI. 2026h. “Flex processing.” OpenAI API documentation. https://developers.openai.com/api/docs/guides/flex-processing. Accessed 2 September 2026.

OpenAI. 2026i. “Batch API.” OpenAI API documentation. https://developers.openai.com/api/docs/guides/batch. Accessed 2 September 2026.

OpenAI. 2026j. “Fast mode.” OpenAI API documentation. https://developers.openai.com/api/docs/guides/fast-mode. Accessed 2 September 2026.

OpenAI. 2026k. “Rate limits.” OpenAI API documentation. https://developers.openai.com/api/docs/guides/rate-limits. Accessed 2 September 2026.

OpenAI. 2026l. “GPT-6 Astra: A new generation of intelligence” (3 September 2026) and “GPT-6 Astra” model page (model ID gpt-6-astra; 1M-token context; reasoning effort low, medium, high, xhigh, and max; US$10 input, US$1 cached input, US$50 output per million tokens, US$20/US$75 above 272K context; Batch and Flex at 0.5x, Fast mode at 2x). https://openai.com/index/gpt-6-astra/; https://developers.openai.com/api/docs/models/gpt-6-astra. Accessed 5 September 2026.

OpenRouter. 2026a. “OpenAI: gpt-oss-120b,” provider table with listed prices and traffic shares. Accessed 1 September 2026. https://openrouter.ai/openai/gpt-oss-120b.

OpenRouter. 2026b. “DeepSeek: DeepSeek V4 Flash 0731,” provider table with listed prices and throughput. https://openrouter.ai/deepseek/deepseek-v4-flash-0731. Accessed 2 September 2026.

OpenRouter. 2026c. “Z.ai: GLM 5.3 Flash,” provider endpoint table with listed prices. https://openrouter.ai/api/v1/models/z-ai/glm-5.3-flash/endpoints. Accessed 2 September 2026.

OpenRouter. 2026d. “MoonshotAI: Kimi K3,” provider table with listed prices. https://openrouter.ai/moonshotai/kimi-k3. Accessed 2 September 2026.

OpenRouter. 2026e. “DeepSeek V4 Is Earning Agentic Token Share.” OpenRouter blog, 30 June 2026. https://openrouter.ai/blog/insights/deepseek-v4-adoption/. Accessed 2 September 2026.

OpenRouter and Andreessen Horowitz. 2026. “State of AI: An Empirical 100 Trillion Token Study with OpenRouter.” arXiv 2601.10088, January 2026. https://arxiv.org/pdf/2601.10088; https://openrouter.ai/state-of-ai. Accessed 2 September 2026.

Ornn Data LLC. 2026a. GPU forward curve marks, retrieved through the Ornn Data API (https://data.ornn.com/docs) on 1 September 2026. Marks for A100 SXM4, H100 SXM, H200, and B200 published 13 August 2026; B300 published 1 September 2026. Reported here as retention ratios relative to each family’s one-month term price.

Ornn Data LLC. 2026b. Settled daily GPU rental index (global region), snapshot for settlement 1 September 2026 20:00 UTC and daily history 1 March to 1 September 2026 for A100 SXM4, H100 SXM, H200, and B200, retrieved through the Ornn Data API on 1 September 2026. Methodology: https://data.ornn.com/methodology.

Ornn Data LLC. 2026c. GPU market-utilization and listed-capacity series (global region), including daily occupancy, demand and supply decomposition, and aggregate GPU-load measures, retrieved through the Ornn Data API on 2 September 2026. Occupancy is the share of tracked on-demand capacity that is rented. Documentation: https://data.ornn.com/docs/analytics; https://data.ornn.com/docs/api-reference/analytics/get-market-utilization.

Ornn Data LLC. 2026d. “GPU Price Index FAQ” and “OCPI Methodology.” Descriptions of term-curve construction, executed on-demand index inputs, and Ornn Compute rental services. https://data.ornn.com/faq; https://data.ornn.com/methodology. Checked 5 September 2026; these public descriptions do not supply the underlying deal records.

Qwen Team, Alibaba. 2026. “Qwen3.8-2.4T-A95B” model card and Qwen3.8-Max License. https://huggingface.co/Qwen/Qwen3.8-2.4T-A95B. Accessed 2 September 2026.

Taylor, Barry N., and Chris E. Kuyatt. 1994. *Guidelines for Evaluating and Expressing the Uncertainty of NIST Measurement Results*. NIST Technical Note 1297. https://www.nist.gov/pml/nist-technical-note-1297.

Together AI. 2026. “Pricing.” Accessed 1 September 2026. https://www.together.ai/pricing.

Z.ai. 2026a. “GLM-5.3” and “GLM-5.3-Flash” model cards and licenses. https://huggingface.co/zai-org/GLM-5.3; https://huggingface.co/zai-org/GLM-5.3-Flash; https://github.com/zai-org/GLM-5. Accessed 2 September 2026.

Z.ai. 2026b. “Pricing.” Z.ai API documentation. https://docs.z.ai/guides/overview/pricing. Accessed 2 September 2026.