SPROUTVEST FIELD NOTES · NO. 1 · JULY 2026

Build, Buy, or Rent: An Operator's Guide to AI Compute Decisions

Erick Watson, Managing Director, SproutVest Advisers

> *Pricing figures in this report were gathered in July 2026 and are cited to their sources in the appendix. GPU pricing moves monthly. The numbers will change, but the framework will not.*

The thesis

Most AI startups treat compute as merely a budget line item. Often they will sign up for the "*one year of free startup compute*" offer from any of the top three hyperscalers, and tout this as evidence that their business has secured its first investor! Compute is not a line item, or something to be handed off to the first hyperscaler that comes along. It is a strategic decision that determines a company's margin structure and forms the backbone of its fundraising narrative. In some cases, it can even decide whether the company survives first contact with production-level traffic.

The cloud-first default is often wrong. The build-your-own path is more often wrong. The founders who get this right are the ones who model power, utilization, and depreciation before they model tokens. That ordering may seem backwards to many product people, and I get why. Tokens are the product, and infrastructure feels like mere plumbing. But in this case, the plumbing sets the unit economics, and unit economics set everything else.

I write as one who has participated in this decision from several different perspectives. I led product for an AI valuation platform running inference against every residential property in the United States, all 105 million households, where model serving cost was a permanent line in every planning conversation. I led the buildout of a boutique one megawatt, Tier 3 datacenter in Washington State, powered by renewable hydro and wind. It was from these experiences that I learned what power actually costs when you are the one buying it. And through Chainlodge, an AI datacenter infrastructure venture, I see today's demand from the supply side, where the primary constraints are no longer racks or chips but grid access.

This report is about economics and decision structure. It is not a silicon benchmarking guide; I will not tell you whether your workload prefers an H100 or a B200. What I will give you is the cost model that makes that a conversation worth having.

Section 1: Why compute decisions silently kill AI companies

This failure is rarely dramatic. No one writes a post-mortem entitled "*we chose the wrong instance type*." What happens instead is that unit economics which worked in the demo stop working at production scale. By the time your leadership team notices, the burn rate has already told the story.

The pattern I see most often begins with a reasonable decision. A team prototypes on a hyperscaler because that is where the credits are and where spin-up is fastest. This is correct behavior at the prototype stage. The mistake is never revisiting it. The workload stabilizes, inference volume becomes predictable, and the company keeps paying on-demand rates for capacity it uses around the clock. Industry cost analyses in 2026 put hyperscaler H100 rates at three to six times what specialized GPU clouds charge for identical silicon. The same chip, the same memory, a bill that differs by a factor that would be a scandal in any other procurement category. CloudZero's May 2026 analysis found a 23x spread in per-GPU-hour pricing across providers for the H100. Two hours on the cheapest verified provider costs less than twelve minutes on the most expensive. Very few companies would tolerate that spread when buying laptops. Most tolerate it when buying compute, because nobody truly owns the number.

The second failure mode is the opposite reflex. A founder looks at the cloud bill, does the arithmetic on GPU purchase prices, and concludes that owning hardware is obviously cheaper. The raw math seems to support this: at roughly $25,000 for an H100 and $3.00 per hour to rent one, the naive break-even is about 8,300 hours, less than a year of continuous use. The naive math is wrong. It ignores the fact that a $25,000 GPU needs a home, and the home costs as much as the tenant. Power distribution, cooling, racks, networking, and spares push total infrastructure cost 30 to 50 percent above the card price. With those included, published analyses put the actual break-even at eighteen months or more of continuous, near-full utilization. Very few startups sustain that. What they get instead is a balance sheet full of depreciating silicon and an ops burden they did not plan for.

The third failure mode is GPU hoarding as fundraising theater. Committed capacity became a status symbol during the shortage years, and

some of that reflex still survives. Buying or reserving GPUs ahead of demonstrated demand ties up capital in an asset that loses value on a schedule, whether or not it earns. A reserved commitment running at thirty percent utilization does not cost 45 percent less than on-demand, whatever the rate card says. As the worked example below shows, it costs much more.

> All three failures share a root cause: nobody modeled utilization. Utilization is the multiplier on every other number in the stack, and it is the number teams are most consistently wrong about, usually in the optimistic direction.

Section 2: The real cost stack

To reason about compute paths you need to see the full stack of costs, because the paths differ mainly in which costs they make visible. Cloud pricing hides everything inside one hourly number. Ownership exposes every component and makes you pay for each one separately. Neither is cheaper by nature; they are the same stack with different accounting.

The stack, from the silicon up: the GPU itself, at roughly $25,000 to $30,000 for an H100 PCIe and $35,000 to $40,000 for the SXM5 variant as of mid-2026, with a fully configured 8-GPU server landing between $350,000 and $480,000. Then power, which I will return to. Then cooling, which is really a power tax. Then networking, which at cluster scale is its own capital project; InfiniBand for even a modest four-node cluster runs into six figures. Then real estate, security, and compliance. Then staffing, the cost everyone forgets until the first 2 a.m. hardware failure. And multiplying across every one of these, utilization: the share of provisioned capacity doing revenue-generating work.

Power deserves its own treatment because over a multi-year horizon it is the dominant variable you can actually influence. When I built the Washington State facility, site selection was the single highest-leverage decision we made. We located in Okanogan County specifically for access to inexpensive renewable hydro and wind generation. I will be direct about scale: one to five megawatts is boutique. The hyperscale operators building today measure campuses in hundreds of megawatts, and nothing about my facility qualifies me to speak about theirs. But the cost discipline transfers exactly, because physics does not care about scale. Every kilowatt-hour has a price, every price varies enormously by geography, and every watt your equipment draws is multiplied by your facility's overhead before it reaches your meter.

That overhead has a name, power usage effectiveness, and the industry's report card on it is instructive. Uptime Institute's 2025 global survey put the average PUE at 1.54, a figure that has barely moved in six years. Facilities commissioned in the past five years average 1.48, and new high-latitude builds reach 1.3 or better. The hyperscalers operate near 1.1, with Google reporting a 2024 fleet average of 1.09 and AWS 1.15. Read those numbers as an operator: at an average facility, every watt of compute costs you roughly one and a half watts at the meter. At a hyperscaler it costs about 1.1. Part of the cloud premium you pay is real efficiency you could not replicate at small scale, and honest treatment of that belongs in any build-versus-buy analysis.

The rate you pay for those watts varies just as much. The US industrial average ran about 8.6 cents per kilowatt-hour in 2025 per EIA data and has been climbing at high single digits year over year, with data center demand itself named as a driver. State-level spreads run nearly four to one. This is why serious infrastructure decisions start with a power map, not a GPU catalog.

The worked example

Here is the stack condensed into one table. The scenario: one 8x H100 SXM5 server's worth of capacity over a three-year horizon. Stated assumptions: on-demand cloud at $3.00 per GPU-hour, the middle of the 2026 market-median band; reserved cloud at $1.65 per GPU-hour, reflecting the roughly 45 percent committed-term discount available for one-to-three-year terms, paid around the clock whether used or not; and an owned-hardware path with the server at a $400,000 midpoint amortized straight-line over three years, colocated at the North American wholesale average of $196 per kilowatt per month on a 10.2 kilowatt committed draw. Staffing, networking, and storage are excluded, and they penalize the owned path most; treat the owned column as a floor.

Effective cost per useful GPU-hour	30% utilization	60% utilization	90% utilization
On-demand cloud ($3.00/hr, pay per use)	$3.00	$3.00	$3.00
Reserved cloud ($1.65/hr, committed 24/7)	$5.50	$2.75	$1.83
Owned + colocation (~$2.24/hr capacity cost)	$7.47	$3.73	$2.49

The table repays a slow read. On-demand looks expensive and is often the cheapest option, because paying only for what you use is a powerful property when your usage is uncertain. Reserved capacity beats on-demand only above roughly 55 percent sustained utilization. The owned path wins only above roughly 75 percent on this simplified math, and realistically above 85 percent once staffing and failure risk enter. An independent sanity check: a mid-2026 infrastructure cost guide estimates owned hardware at 70 percent utilization at $2.50 to $4.00 per all-in GPU-hour, which lands directly on the band this table computes.

One honest caveat about the on-demand row: it assumes you actually release capacity when idle. Teams routinely fail to do this, which converts on-demand pricing into reserved-level commitment at on-demand rates, the worst cell in an expanded version of this table. The discipline of releasing what you are not using is worth more than most provider negotiations.

Section 3: Build, buy, or rent, and when each is right

The table above gives you the economics. The decision framework adds two more dimensions: what kind of workload you run, and how predictable your demand is.

Workloads split into three broad types. Training is bursty, tolerant of interruption at the margin if you checkpoint well, and enormously bandwidth-hungry across nodes. Fine-tuning is smaller, more frequent, and schedulable. Inference is the one that pays the bills, and it is a service-level business: latency floors, uptime obligations, and demand that follows your customers' clock rather than yours. These workloads want different homes. Training wants cheap interruptible capacity or short intense reservations. Inference wants stable, right-sized, boring capacity as close to predictable cost as you can get.

Map demand predictability against those three categories and the paths sort themselves.

Rent on-demand when demand is unproven or spiky. This is every company at the prototype stage and many companies well beyond it. The premium you pay per hour buys you the option to be wrong about demand, and early on, you will most certainly be wrong about demand. Spot and preemptible capacity, at 60 to 90 percent below on-demand rates, belongs here too for any workload that checkpoints gracefully.

Reserve when a baseline becomes visible. Once inference traffic shows a floor that persists across months, moving that floor onto committed pricing at a 45 to 50 percent discount is close to free money, provided you commit only the floor and keep the spikes on demand. The error here would be committing to the forecast instead of the floor.

Colocate when your baseline is large, stable, and long-lived. This is the middle path most founders never evaluate, and it deserves more attention than it gets. You own the hardware and rent the building: power, cooling, security, and connectivity arrive as a monthly per-kilowatt fee at wholesale rates around $196 per kilowatt-month in North America, with committed single-rack arrangements from roughly $130 to $225. You capture most of ownership's cost advantage without building anything. The catch is operational: quoted rates historically understate the delivered bill by 20 to 60 percent once cross-connects, remote hands, and bandwidth enter, so diligence the full invoice, not the top line. The other catch is availability. Vacancy in primary North American colocation markets ended 2025 at 1.4 percent, and high-density AI capacity increasingly requires pre-leasing with delivery timelines beyond twelve months.

Build only when you have scale, a decade-long horizon, and above all, power. What I see through Chainlodge is consistent on this point: the binding constraint on new AI datacenter capacity is no longer construction timelines or even GPU supply, it is grid interconnection. Sites with secured power have become the scarce asset, which is why the industry now talks about bringing workloads to power rather than power to workloads. If your company's plan involves building, your real project is a utility relationship, and it starts years before the first rack arrives. For nearly every startup reading this, build is the wrong answer, and the fact that it is the wrong answer for you is precisely why the companies doing it at scale are interesting.

A word on hybrids, because the best answer is usually plural. The pattern that works: train in rented bursts, serve the inference floor on reserved or colocated capacity, and keep the spike on demand. Each workload lives where its behavior is cheapest. The companies that get compute right rarely have one provider; they have a portfolio of providers and an executive who owns it.

Section 4: What investors should actually diligence

I evaluate technology companies for private investors as part of my practice, and compute claims have become one of the places where diligence pays fastest. A pitch deck's infrastructure slide tells you a great deal about founder discipline if you know what to ask. Here are some questions I ask:

First: show me the utilization model behind the committed spend. Any reservation or purchase should come with an analysis like the table in Section 2, built on the company's own traffic. If the answer is a rate-card discount percentage with no utilization assumption behind it, the company has committed to a forecast it has not made. Committed spend without utilization modeling is the single most common infrastructure red flag I encounter.

Second: is the GPU inventory framed as a moat? Hardware anyone can buy is not a moat; it is capex with a depreciation schedule. During supply-constrained periods, allocation access had genuine option value, and in tight markets it can again. But silicon depreciates and generations turn over. When a deck presents owned GPUs as a durable competitive advantage rather than a cost position with a shelf life, I read it as either naivety about the asset or a bet the founder has not stress-tested.

Third: what power price does the model assume, and is it constant? US industrial electricity rates rose at high single digits over the past year, with data center demand itself as a named driver. A model that holds power flat for five years in a market where compute demand is repricing the grid is not conservative. If the company operates or plans facilities, ask where, and ask what the interconnection queue looks like; the answer separates companies with a site from companies with a blueprint.

Fourth: does the margin story survive the compute bill? For an AI product company, ask for gross margin with inference fully loaded, not blended into R&D. The difference between a software margin and an infrastructure margin is often sitting in that allocation choice, and it changes what multiple the business deserves.

None of these questions requires a deep technical background. They require the willingness to treat infrastructure claims with the same rigor as revenue claims. In my experience the second is rarer.

Section 5: The strategic layer

Everything above treats compute as a cost problem. At the fundraising table it becomes a narrative problem, and the narrative cuts both ways.

A well-modeled compute plan is one of the cheapest credibility signals available to a technical founder. When a Series A deck shows an inference cost curve with stated utilization assumptions, a reservation strategy tied to a demonstrated traffic floor, and a margin bridge from today's blended number to the target model, it communicates that the founder runs the business on numbers. Investors extrapolate. The founder who owns the utilization number probably owns the CAC number too.

The reverse signal is just as strong. A company raising partly to cover a cloud bill it cannot explain is asking investors to fund a variable it has not measured. And margin structure is destiny at pricing time: whether your gross margin reads as software or as infrastructure determines which comparables you get priced against, and the compute decisions in this report are frequently the difference.

The China and APAC dimension of this topic, where power economics, latency geography, and regulatory surface all differ enough to change the answers, is a report of its own, and it will be the next one in this series.

The takeaway

Compute is a strategic decision wearing a line item's clothing. The stack is the same whether you rent it or own it; the paths differ only in which costs they show you and which they hide. Utilization is the multiplier on everything, and it is the number your team is most likely wrong about today.

Three questions to ask of your own compute plan this week. What is our actual sustained utilization, measured, not assumed? At that utilization, which row of the Section 2 table are we in, and which row are we paying for? And if our traffic doubled next quarter, do we know where the capacity would come from and what it would cost?

If any of those questions is uncomfortable, that is worth a conversation.

Erick Watson advises AI, blockchain, and data platform founders and funds through SproutVest Advisers, serving as fractional CPO and conducting technical due diligence for private investors.

erick@sproutvest.com · cal.com/sproutvest · linkedin.com/in/erick-watson

APPENDIX

A. The worked example in full

Scenario: capacity equivalent to one 8x NVIDIA H100 SXM5 server, three-year horizon, 8,760 hours per year, 26,280 hours per GPU over the term.

Owned-path capacity cost derivation:

- Hardware: $400,000 (midpoint of the $350,000 to $480,000 configured-server range) / (8 GPUs x 26,280 hours) = $1.90 per GPU-hour
- Colocation: 10.2 kW committed draw x $196 per kW-month x 36 months = $71,970 / (8 x 26,280) = $0.34 per GPU-hour
- Capacity cost at 100 percent utilization: approximately $2.24 per GPU-hour
- Excluded: staffing, networking beyond the server, storage, spares, and failure risk. Published estimates suggest these add 10 to 20 percent or more.

Crossover points at a $3.00 on-demand baseline:

- Reserved ($1.65 committed) beats on-demand above 55 percent sustained utilization
- Owned + colocation beats on-demand above 75 percent on capacity cost alone, realistically above 85 percent with overheads

Effective $/useful GPU-hour	30%	45%	60%	75%	90%
On-demand ($3.00)	3.00	3.00	3.00	3.00	3.00
Reserved ($1.65 committed)	5.50	3.67	2.75	2.20	1.83
Owned + colo ($2.24 capacity)	7.47	4.98	3.73	2.99	2.49

B. Glossary

- **Utilization rate:** the share of provisioned GPU-hours doing productive work. The multiplier on every committed cost.
- **PUE (power usage effectiveness):** total facility power divided by IT equipment power. 1.0 is theoretical perfection; the 2025 industry average is 1.54.
- **On-demand vs. reserved:** pay-per-use hourly pricing vs. discounted pricing in exchange for a one-to-three-year commitment, billed regardless of use.
- **Spot / preemptible:** deeply discounted capacity the provider can reclaim with little notice. Suitable for interruption-tolerant, checkpointed workloads.
- **Colocation:** owning hardware housed in a third-party facility that supplies power, cooling, security, and connectivity, typically priced per committed kilowatt per month.
- **Cross-connect:** a physical link from your equipment to a carrier or cloud on-ramp within a facility, billed monthly and a common source of invoice surprise.

C. Methodology and sources

Figures were gathered from published sources in July 2026 and cross-checked between vendor materials and neutral trackers where both existed; where a vendor source and a neutral source conflicted, the neutral source was used. All computations in the worked example are original to this report from the cited inputs. GPU pricing is volatile; on-demand H100 rates fell 64 to 75 percent from 2023 peaks before rising roughly 14 percent in the year to July 2026. Readers should re-verify rates at decision time.

Sources: GetDeploying H100 pricing tracker (July 2026); CloudZero H100 cost analysis (May 2026); Spheron and SynpixCloud provider rate reviews (April to May 2026); IntuitionLabs H100 rental market review (November 2025); Haink AI Infrastructure Cost Guide (Q2 2026); CBRE North American colocation data as reported by Encor Advisors and Data Center Knowledge (H1 to H2 2025); QuoteColo AI colocation pricing guides (2026); EIA Electricity Monthly Update and industrial retail price series (2025 to 2026); Uptime Institute Global Data Center Survey 2025; company PUE disclosures (Google, AWS, Microsoft, 2024 reporting).