ON ROUTING INFERENCE, MODEL SPRAWL, AND THE DISCIPLINE OF THE SMALLER DEFAULT

The Architecture of Sovereignty

The default in AI architecture today is to send every workload to the most expensive model available. That default has stopped being defensible on cost, on control, and on the timeline over which the system one depends on continues to exist. The discipline that replaces it is not exotic, and it is not anti-cloud.

SUBJECT	FRAME	LENGTH
Local and small models	Frontier default vs routed default	~1,800 words

The default in AI architecture today is to send every workload to the most expensive model available, and the default has stopped making sense.

In late December 2025, a software engineer writing as Cyrus published a short essay on his personal blog under the title “Local AI Needs to Be the Norm,” and the unusual thing was not the argument but the reception. The post climbed to the front page of Hacker News and stayed there for most of a day, gathering more than seventeen hundred upvotes and a comment thread that mostly agreed with itself — a pattern that does not describe how Hacker News normally handles an architectural opinion. The argument itself was almost prosaic. Most application features, Cyrus wrote, do not need a model that can write Shakespeare, explain quantum mechanics, and pass the bar exam; they need a model that can do one of five things reliably — summarize, classify, extract, rewrite, or normalize — and we are, in the meantime, building applications that stop work-

ing the moment the server crashes or a credit card expires. What the essay was really arguing for, beneath the engineering specifics, was a kind of sovereignty over the systems one's software depends on.

The reception mattered because the essay named a default that architects had stopped examining. The current convention is to route every AI workload along a single axis: hit a frontier cloud API and absorb whatever bill, latency, and exposure come back. That single-axis routing is the architectural mistake worth naming. AI workloads have at least three axes worth routing on — locality (where the inference happens), size (how large the model is), and specialization (how narrow it is) — and the prudent architecture defaults along all three at once, reaching first for the smallest, most local, most specialized model that has a real chance of doing the job, and escalating only when a specific failure of the cheaper tier demands it.

§

The case for routing on size used to be theoretical, and it is now arithmetic. On February 12, 2026, MiniMax released M2.5, a 230-billion-parameter Mixture-of-Experts model that activates only ten billion parameters per forward pass. On SWE-Bench Verified — the benchmark the industry has settled on for measuring real coding capability against real GitHub issues — M2.5 scored 80.2%, against Claude Opus 4.6's 80.8%. The pricing tells the rest of the story: M2.5 lists at $0.30 per million input tokens and $1.20 per million output, against Opus 4.6's $5.00 and $25.00 — roughly seventeen times cheaper on input and twenty-one times cheaper on output, for six-tenths of a percentage point on the benchmark the frontier providers cite when they want to claim the frontier.

MiniMax has become its own existence proof. The company's engineering organization now reports that thirty percent of internal tasks are autonomously completed by M2.5 and eighty percent of newly committed code is M2.5-generated. A frontier-adjacent model, served at roughly five percent of frontier cost, has absorbed most of the work inside the company that built it.

The industry's own pricing structure tells the same story more bluntly. At NVIDIA's GTC 2026 keynote in San Jose, Jensen Huang laid out a five-tier framework for inference pricing, mapping each tier to a specific model: a free tier running Qwen 3, a $3 tier on Kimi K2.5, a $6 tier on GPT MoE, a $45 tier on GPT MoE with 400K context, and a $150 Ultra tier above that. "Tokens are the new commodity," Huang said, and like all commodities, once they mature they segment by price. The vendor selling frontier compute was pricing routine use of the frontier out of the routine. The frontier remains genuinely frontier for a narrow band of work, but the band has narrowed; everywhere else, the cost of buying breadth one no longer needs has stopped being defensible.

> *The frontier remains genuinely frontier for a narrow band of work, but the band has narrowed; everywhere else, the cost of buying breadth one no longer needs has stopped being defensible.*

§

While the middle of the curve has compressed, the floor underneath it has been quietly rising. At WWDC 2025, Apple announced its Foundation Models framework, which shipped with iOS 26 that September and exposed to every third-party developer a roughly three-billion-parameter on-device language model with two-bit quantization. Apple's own framing of the framework was unusually candid for a company that prefers superlatives: the model, the developer documentation noted, is optimized for summarization, extraction, and classification, and is not suitable for world knowledge or advanced reasoning. That is the company naming its specialization niche on the record — and it happens to be the same niche Cyrus identified as covering most application features.

The browser tier has followed. On May 5, 2026, Chrome 148 reached stable channel with the Prompt API enabled by default, after eighteen months behind feature flags. The API runs Gemini Nano locally in the browser, with no API key, no cloud round-trip, and no per-token meter. A web application that wants to classify an input or rewrite a draft can now do so on the user's device, the same way it accesses the camera or the filesystem.

The hardware tier rose alongside the platform tier. AMD's Ryzen AI Max+ 395, the "Strix Halo" architecture launched in January 2025, pairs sixteen Zen 5 cores with a forty-CU RDNA 3.5 integrated GPU and supports up to 128 GB of LPDDR5X unified memory, of which up to 96 GB can be allocated to the GPU — enough to load a seventy-billion-parameter model at BF16 entirely on the iGPU of a mini-PC. A GMKtec EVO X2 with that configuration is available from $1,499, against $3,999 for a comparably specified 128 GB Mac Studio M4 Max. The Mac retains roughly 2.5 times the memory bandwidth, and the trade is now a real consumer choice rather than a research-budget question. What used to require a research budget now ships as a permission prompt.

§

Set the MiniMax numbers and the Apple framework against a calendar, and the case stops being about cost. Anthropic's published policy commits the company to a minimum of sixty days' notice before retiring a publicly released model. The policy is the floor, and the practice has been at the floor or below it for most of the past year. Claude Sonnet 3.5 received its retirement notice on August 13, 2025, and was retired January 5, 2026 — roughly the stated minimum. Claude Sonnet 3.7 was retired on October 28, 2025, the same day the notice was published. Claude Sonnet 4 and Opus 4 received notice on April 14, 2026, and were retired on April 20 — six days later. Haiku 3 and Haiku 3.5 went on February 19. By the middle of 2026, every Claude model that had been in production at the start of the year had been retired or scheduled for retirement.

Each retirement is, from inside a customer's engineering organization, a forced migration: prompts re-tuned, evaluations rerun, harnesses revalidated, regressions triaged, schedules rearranged around a deadline the operator chose. The asymmetry is structural rather than incidental. The operator decides when the model in your production system dies, on what notice, with what successor, at what price. This is the mainframe configuration restated for inference — economies of scale, central management, opaque pri-

cing, painful migrations, and a relationship in which the customer's leverage is to leave, slowly and at cost. An application built on a remote frontier model is not buying capability; it is renting it, on terms the landlord revises.

> *An application built on a remote frontier model is not buying capability; it is renting it, on terms the landlord revises.*

That is what the sovereignty Cyrus was reaching for reduces to in operational practice. It is not a slogan about privacy, although privacy is part of it. It is a question about who controls the version of the system your software depends on next quarter, and the honest answer, for any architecture that defaults to a remote frontier API, is: not you.

§

The strongest objection to all of this is that the cloud, today, is simply cheaper per token than anything one can run on idle hardware at home or on a single rack. The objection is real. Frontier providers run their fleets at very high utilization; a consumer rig or a single on-prem server runs at near zero between requests, and per-token economics favor the operator who can amortize a GPU across thousands of simultaneous customers.

The accounting is still incomplete. Current frontier API prices are not a stable reference point — OpenAI's most recent quarterly disclosures showed inference operations running at a loss even at posted sticker prices, and Anthropic's enterprise discounting suggests a similar shape — which means the comparison is against a number the market has not yet had to defend. Utilization economics also flip for the long tail of intermittent inference, which is what most production AI work actually looks like: bursty, embedded in user sessions, idle most of the day, rather than the steady-state high-volume serving the cloud is optimized for. And the cloud line item has to carry the cost of being rewritten on the operator's schedule, which several engineering organizations rediscovered in 2025 when forced migrations consumed quarters that had been planned around feature work.

The right comparison is not cents per token on a spreadsheet but cost-plus-control over the life of the application. A line item that does not include the cost of being rewritten on six days' notice is not an honest line item.

§

The architecture that follows from all of this is not exotic, and it is not anti-cloud. It is the discipline of routing every AI workload across three axes — locality, size, specialization — and defaulting, on each one, to the smallest, most local, most specialized model that has a real chance of doing the job. Escalation up any axis becomes a deliberate decision driven by a specific failure of the cheaper tier, rather than a habit of reach.

The remote frontier still wins decisively on a narrow and important band of work: hundred-file coding agents that must hold an entire system in context, genuinely novel reasoning, the long-horizon agent loops where one error at step seven of a ten-step

plan costs more than the entire inference run. Route there when the task demands it — the mistake is not using the frontier but starting there for work the frontier was never the right tool for.

The cultural shift the Hacker News thread signaled is that the default has stopped looking obvious. The thread agreed with itself because most of the people building the applications already knew — they had paid the bills, watched the deprecations, run the local benchmarks, and noticed that the answer their architecture diagrams assumed was no longer the answer their engineering judgment produced. The remote frontier API is not the architecture. It is the escalation path. The architecture is the model on the device, the small specialized weights on the on-prem rack, and the discipline, newly affordable and no longer eccentric, of reaching for them first.

COLOPHON

Set in DM Sans and IBM Plex Mono, with Source Serif 4 for display italics and pull quotes. Typeset on A4 at 10.5pt with weasyprint.

PUBLICATION

Rivoli AI · Essays · 2026-05-16

sam@rivoli.ai