

EdgeForge: GPU inference *next to the user*.

Inference is centralized in three regions; users in São Paulo, Mumbai, and Lagos pay 220 to 340ms of round-trip tax per token. Hyperscalers won't fix it (edge GPU economics are worse than dense regions). Whoever bundles silicon, registry, and routing first wins the developer.

Author S. Ize-Iyamu

Audience Edge platform and Inference PMs

Length 3 pages

Status Concept

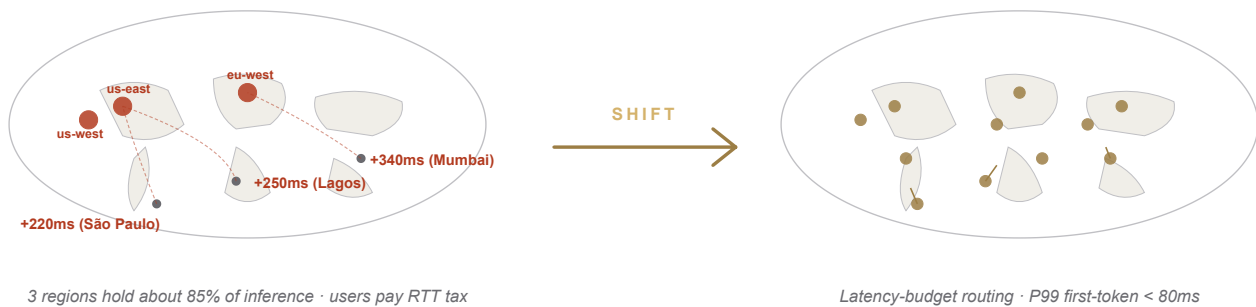
The Problem

The majority of public inference traffic terminates in a handful of US/EU regions (us-east-1, us-west-2, eu-west-1 are the dominant points-of-presence). For a developer in São Paulo or Lagos, the model can be fast and the network slow, and TTFT (time-to-first-token) becomes a regional tax: **220 to 340ms RTT before token #1**. Hyperscalers will not fix this: edge GPUs run 30 to 50% lower utilization than dense regions, and the economics structurally favor centralization. CDNs sit on the right topology but historically lacked GPU access. The opening: bundle silicon, model registry, and a routing layer where the developer never picks a region.

FIGURE 1 · TOPOLOGY SHIFT

TODAY · 3-REGION CENTRAL

EDGEFORGE · ROUTED MESH



Today (left): 3-region centralization forces RTT taxes of 220 to 340ms for distant users. EdgeForge (right): the developer never picks a region; the request routes across edge / regional / centralized GPU pools by latency budget.

Why this matters now

Three forces converge: **NVIDIA L40S / Hopper / Blackwell variants make edge GPUs viable** (the silicon density that broke edge economics in 2022 has flipped), **CDN POP density passed 300+ for tier-1 providers** (Cloudflare, Fastly, Akamai), and **developer pain on TTFT is acute and measurable**: every public LLM API now publishes per-region first-token latencies, and the gap is the leaderboard.

Sizing the opportunity

Bottom-up: **~\$70B / yr global LLM inference market by 2027** (blended estimate from a16z LLMflation analysis Nov 2024 + MarketsandMarkets AI Inference 2025 forecast), of which **about 25% is latency-sensitive** (chat, agents, RAG, voice) = **~\$17.5B / yr addressable**. Edge-bundled provider takes **about 12 to 18% on routed requests** (CDN-style margins, not hyperscaler) = **~\$2.1 to 3.2B / yr**. North-star: paid inference requests / week. The single thing we sell: P99 first-token < 80ms *without the developer choosing a region*.

Directional sizing: a16z LLMflation and MarketsandMarkets AI Inference forecast + 5 inference-platform interviews. Concept-brief ballpark.

LATENCY-SENSITIVE INFERENCE

~\$17.5B / yr

25% of \$70B by 2027

PLATFORM TAM

~\$2.1 to 3.2B / yr

12 to 18% on routed requests

Strategic insight

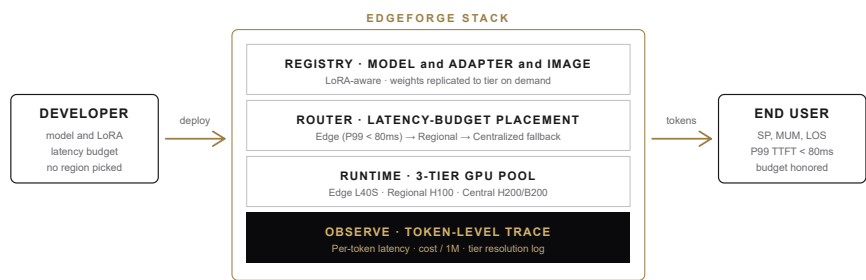
The developer doesn't want to choose a region; the developer wants a **latency budget honored**. Region-pinning is an artifact of cloud-vendor org charts, not user intent. The bundle is the moat: silicon partnership (NVIDIA reference design), CDN topology (300+ POPs), and the registry surface (model and adapter and image) live behind a single deploy. Each piece is replicable; the bundle is hard to copy because it requires a CDN that already has a global footprint and is willing to negotiate silicon at scale.

THE UNLOCK

Latency-budget API: developer uploads a model, sets P99 first-token < X ms, and the platform places each request across edge / regional / centralized tiers. Per-request pricing with a public price card and SLA per latency tier. LoRA adapters as first-class objects; token-level cost and latency tracing in the same view as the dashboard.

Architecture • Latency-routed inference

FIGURE 2 • SYSTEM ARCHITECTURE



Developer uploads model and LoRA; the router places each request into the cheapest tier that meets the latency budget; weights replicate to edge on demand; token-level traces expose where latency was paid and where cost was paid, in the same view.

WORKED EXAMPLE • AGENTIC STARTUP, 11M WEEKLY REQUESTS

Default OpenAI-region routing: P99 TTFT **510ms median user, 920ms tail**. EdgeForge with P99 < 80ms budget on a 7B fine-tuned model: **P99 TTFT 71ms; 34% of traffic served from edge L40S**, 51% regional H100, 15% central. Per-token cost 18% higher than central-only, but the latency budget is met and conversion lifts **+11% on the user-facing checkout flow**.

Sequenced GTM

PHASE	CUSTOMER WEDGE	FORCING-FUNCTION WORKLOAD	PROOF POINT
Wedge M0 to 9	Voice and chat and RAG startups with global users	TTFT < 100ms target on user-facing flows	P99 first-token < 80ms • routing-tier SLA met > 95%
Beachhead M9 to 24	Mid-market SaaS embedding LLM features in their product	Region-spanning user base • cost-per-1M trade-offs	250 paying customers • NRR > 130%
Enterprise M24+	Fortune 500 customer-experience and agent platforms	Compliance and data-residency by tier (per-region pinning)	about 40% of revenue from enterprise SLA tier

Tradeoffs we accept

- **Edge tier is cost-premium vs central.** Per-token price 15 to 25% above central-only. We don't pretend edge is free; we sell the latency budget and let the developer choose.
- **Cold-start P99 < 600ms, not zero.** Edge weight replication takes seconds on cold POPs; we keep a hot replica policy on top-N models, the long tail eats the cold start.
- **Limited model-format support at v1.** Llama / Mistral / Qwen / Phi families only at GA. Closed-source gateway models (GPT, Claude) require a partner deal we don't yet have.

Metrics that matter

LAYER	METRIC	Y1 TARGET	WHY IT MATTERS
North-star	Paid inference requests / week	> 800M / wk by Y1-end	Real volume, not signups
Quality	P99 first-token latency	< 80ms	The latency budget we sign
Tier SLA	Routing-tier SLA met	> 95%	Below this, the budget is a lie
Cold-start	P99 cold-start latency	< 600ms	Long-tail models make or break long-tail customers
Liquidity	Paying developer accounts	about 5K paying by Y1	Self-serve density unlocks enterprise references
Business	Net revenue retention	> 130%	Per-request pricing compounds with product growth

Risks & mitigations

HIGH	NVIDIA partnership terms shift unfavorably (allocation, pricing). Mitigation: design for multi-silicon at v1: abstract the runtime over CUDA and ROCm + (later) custom accelerators (Groq, Cerebras for specific workloads). Lock long-term L40S allocation contracts now while pricing is favorable; carry 90-day inventory hedge.
HIGH	Hyperscaler ships an edge-GPU bundle (AWS Outposts and Bedrock). Mitigation: hyperscalers will ship to their dense regions first; edge density takes 4+ years even with infinite capital. Compete on POP count and developer-experience velocity. EdgeForge ships latency-budget API; AWS ships region-pinned API. Different products.
MED	Edge GPU utilization too low to be margin-positive. Mitigation: 3-tier router preferentially fills edge to a target utilization (about 70%); spillover routes to regional / central. Pricing absorbs lower utilization at the edge tier. Below 50% utilization for 60 days, retire the POP.
MED	Open-source model release cadence outpaces our adapter pipeline. Mitigation: 72-hour SLA on top-N model adoption from HuggingFace release; LoRA adapter format is standardized so customer adapters port across model versions. Long-tail models served via on-demand replication to one regional tier.

30 / 60 / 90, first quarter sprint plan

30 DAYS Router + 3 POPs › Latency-budget API · placement engine v0 › 3 edge POPs (LAG, MUM, GRU) · 1 regional › 10 design-partner accounts deploying real traffic	60 DAYS Registry and LoRA and traces › Model registry · LoRA adapters first-class › Token-level latency and cost trace › 50 paying customers · cold-start P99 < 800ms	90 DAYS SLA tiers and GA › Public price card · SLA tiers (latency · region-pin) › 10 edge POPs live · 3 regional · 1 central › 500 paying accounts · NRR cohort baseline
---	--	--

DECISION ASKED Authorize a 90-day build-and-prove sprint with a nine-person team (PM, four platform engineers, ML/runtime lead, silicon partnerships, devrel, edge ops) and a budget of ~\$5.4M (incl. silicon allocation deposit). Success: 500 paying accounts, P99 first-token < 80ms, routing-tier SLA > 95%, cold-start < 600ms, NRR clearing 130% on a 30-day cohort.
