Lecture HSE · May 23, 2026

AIOps / LLMOps for AI Products

Executive decisions, economics and FinOps for the production AI loop

Alexander Polomodov

Speaker Context

Technical Director & Fellow, T-Technologies

- Architecture and engineering practices
- AI adoption at development scale
- Focus: managed production AI

Introduction Follow-up

The platform exists. Now manage it

Executives choose behavior, launch path and control

Vocabulary is known

Gateway, context, evals and guardrails.

Focus on decisions

SLA, autonomy and quality cost.

Money anchor

Support bot tests TCO/FinOps.

Navigation What we do not repeat

Basic components are assumed

AIOps / LLMOps starts after the platform appears

- Skip the platform map
- Choose service by scenario
- Count solved-task cost

- Focus: AI product manageability



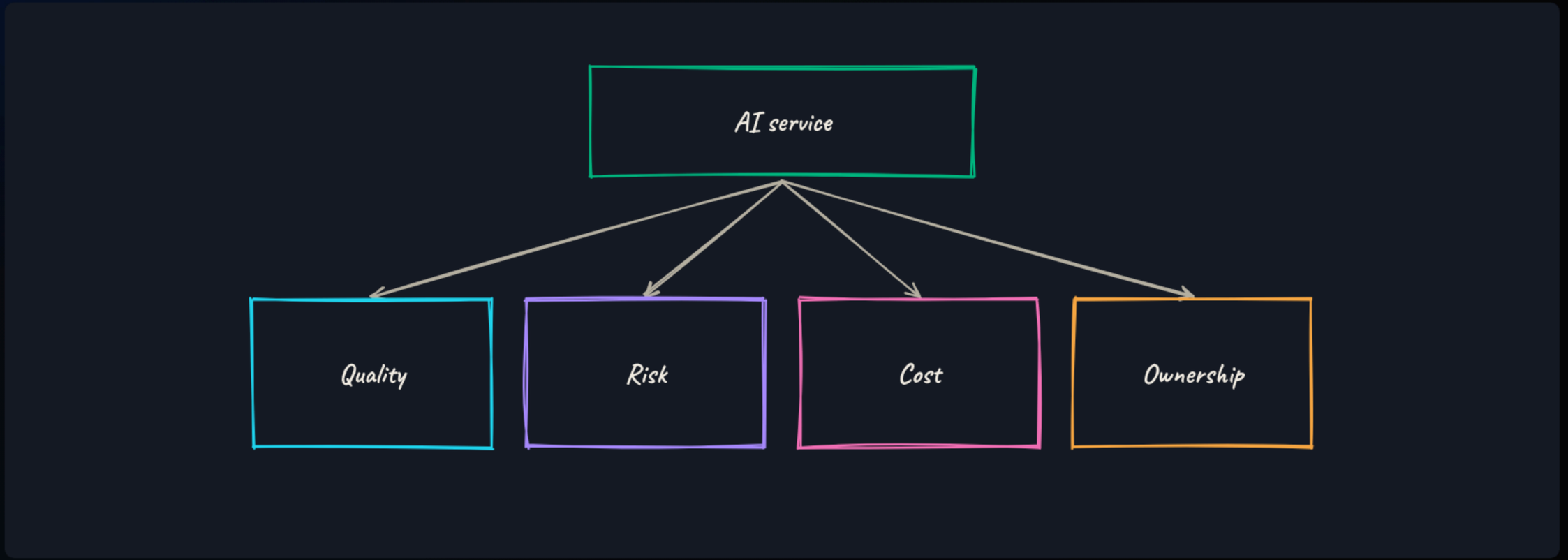
Block 1

01

From basics to governance

Do not repeat the AI platform map; translate it into executive decisions

After launch, AI becomes a service



The manager chooses the frame



AIOps is a decision loop

Block 1 · Takeaway Operating model





Block 2

02

Manager decisions

Scenario, autonomy, error cost, SLA, quality budget and ownership

Inventory

Scaling needs decisions first

Otherwise AI grows faster than control

Value and scenario

Value

Value, baseline and SLA.

Risk and control

Risk

Allowed autonomy and human-in-the-loop.

Economics and ownership

Cost

Quality cost; product, data, risk and bill owners.

Error cost defines autonomy

Low risk	High risk
Drafts · search	Money · production
Fast model	Approval + audit

AI SLA is broader than API uptime

Service	Quality
Latency and timeout	Sources and coverage
Queues and fallback	Policies and escalation

Evaluation starts from product

The same stack can be right or wrong

- Value: process and baseline
- Risk: error and approval
- Economics: unit cost
- Operability: release and rollback

- Explain the decision, not only the stack



Block 3

03

Options and trade-offs

API-first, managed platform, rented GPU, owned self-host and hybrid

AI runtime is a portfolio

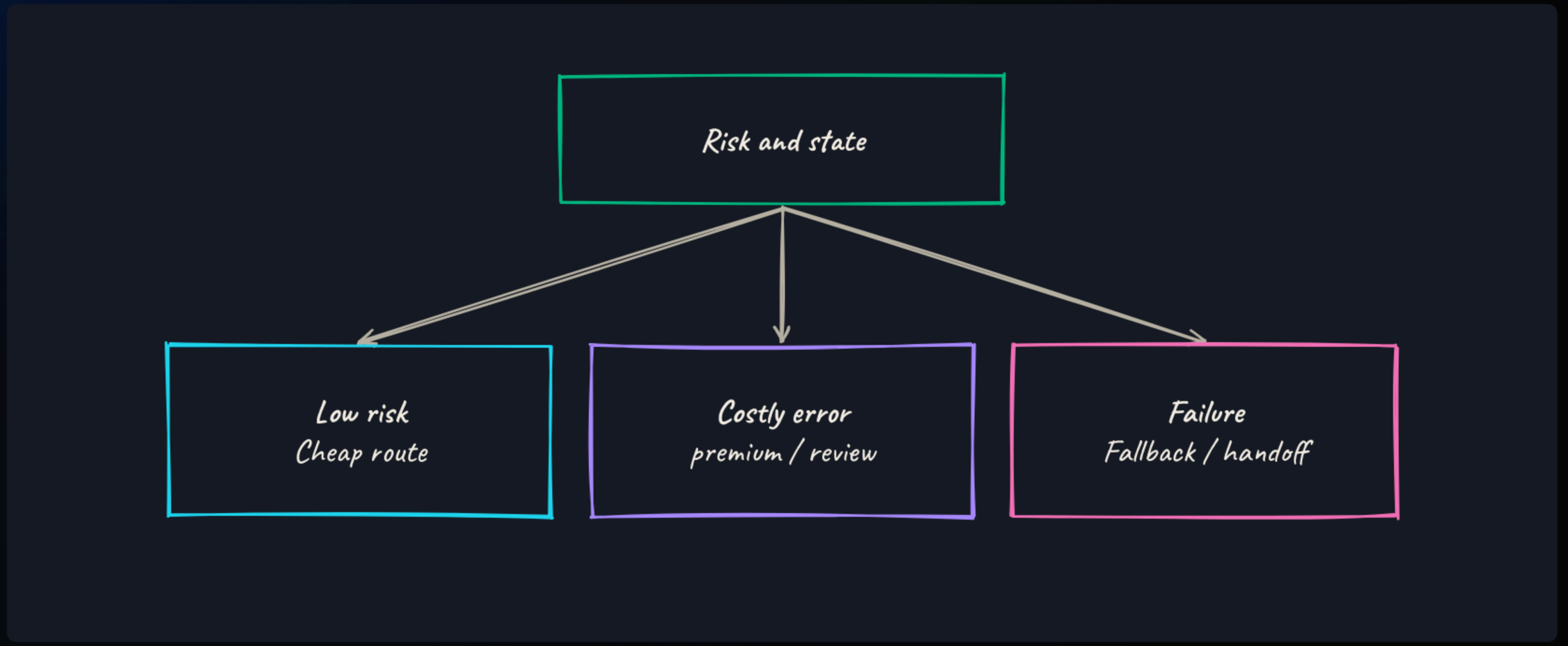
API / SaaS	Managed platform	Rent GPUs	Own GPUs
Speed Ready service	Control and lock-in	Own model OPEX	Control CAPEX + ops

Hybrid assigns different use cases to different paths

Assistant and agent are different products

Assistant	Agent
Reads and explains	Takes actions
ACL + sources	Permissions + approval

You need routing rules



Choice is a portfolio decision

Block 3 · Takeaway

Trade-off

- API-first validates the launch
- Rented GPU gives transitional control
- Owned self-host needs stable load
- Hybrid fits different scenarios

- Runtime cannot be chosen without scenarios and financial horizon



Block 4

04

Money case

Support bot: load, tokens, GPUs, team and the management decision

Calculate the support bot

Block 4 · Money Case

Support bot

200,000 monthly dialogs · 10 messages · 5-second SLA

What we calculate

Load, tokens, peak, GPU, CAPEX/OPEX, team.

What we compare

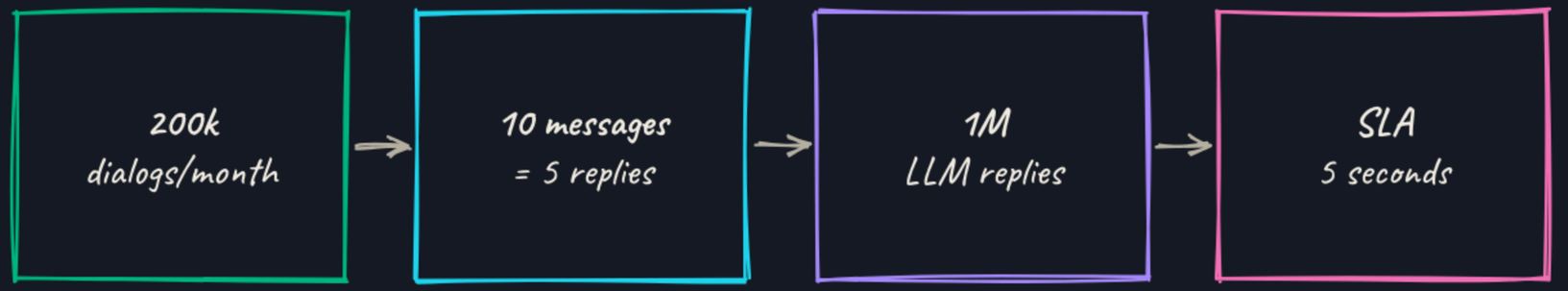
API-first, rented GPU, own self-host, hybrid.

What we decide

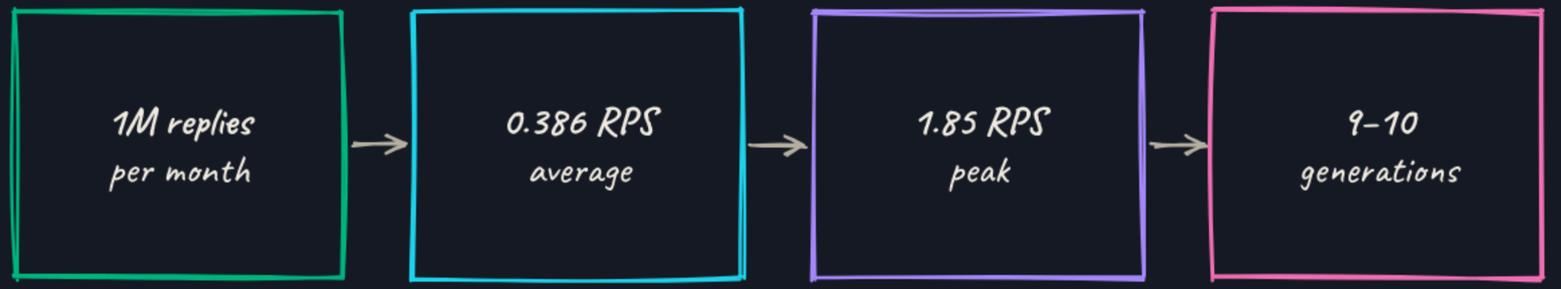
Year-one path and review metrics.

Case parameters set the decision scale

Case · Inputs 200,000 dialogs



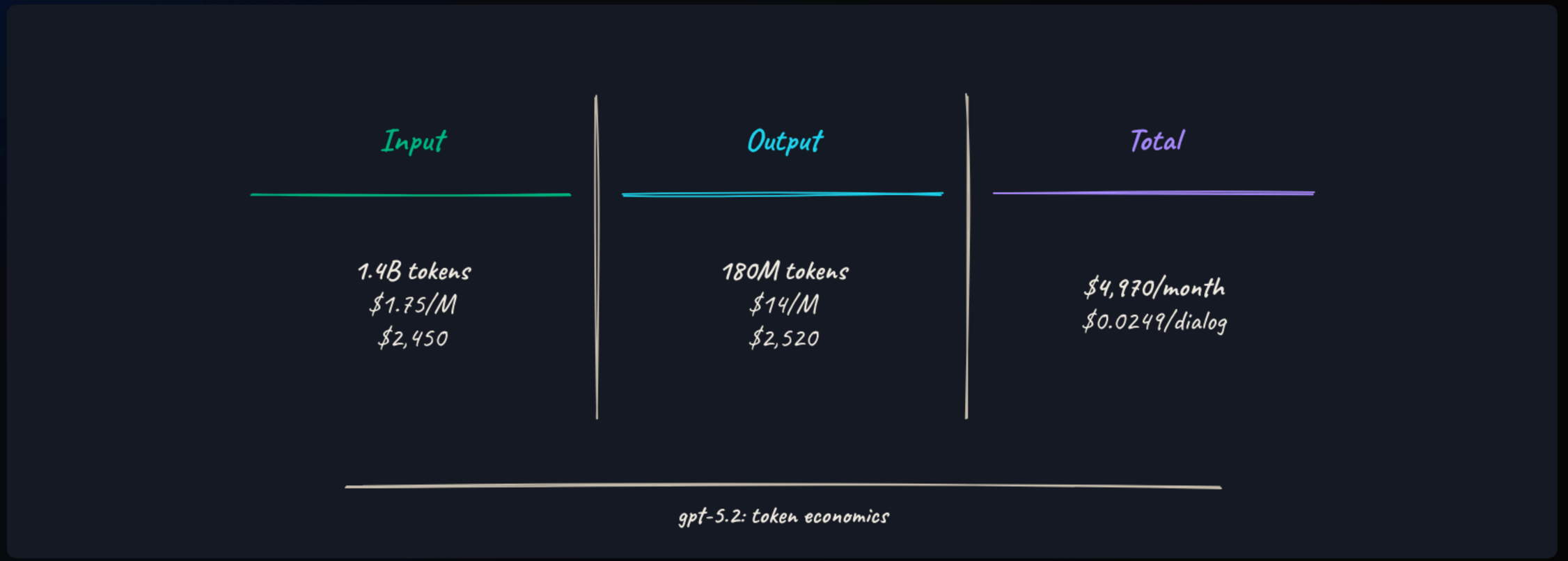
Translate parameters into RPS and parallelism



Concurrency ≈ peak RPS × 5 seconds

GPT-5.2 API gives a clear starting OPEX

Case · Step 2 API-first



API-first buys speed

Case · API-first Manager view

The risk is silent unit-economics growth

+ Why launch is convenient

- Near-zero CAPEX
- Faster production launch
- Thinner surrounding platform

! What must be controlled

- Output, cache, retries
- Provider limits and degradation

Size self-host from SLA and peaks

Qwen-14B · INT4

~90 tokens/s
180 tokens: 2.0s
~2 requests/GPU
6–8 GPUs

Qwen-32B · INT4

~45 tokens/s
180 tokens: 4.0s
~1 request/GPU
≈12 GPUs

Peak ≈10 replies, 5 s SLA; with batching, fewer GPUs are needed

GPU rental trades CAPEX for OPEX



Rate: cloud.ru price list, February 2026; network and reserve extra

Self-host changes the problem

Rental	Own GPUs
Test the hypothesis OPEX and MLOps	Steady load CAPEX and delivery lead time

Strategy comparison shows the year-one decision

	API	Rented GPUs	Own GPUs
CAPEX	Low	Low	High
OPEX	~$5k LLM/month	1.17–2.33M RUB/month	Infrastructure + team
Launch	Fast	Medium/fast	Slower
Control	Lower	High	Maximum

For this workload: API-first

The decision must include a review point

Case · Decision 12 months



Strict on-prem requirements change the choice

The case decision

Managerial answer

- Launch API-first unless owned perimeter is mandatory.
- Track unit economics immediately.
- Set review triggers.
- Self-host must pass calculation, SLA and operations readiness.

- The first decision is speed with control, not eternal architecture



Block 5

05

Manageability and FinOps

Dashboard, cost allocation, unit economics review and a 120-day plan

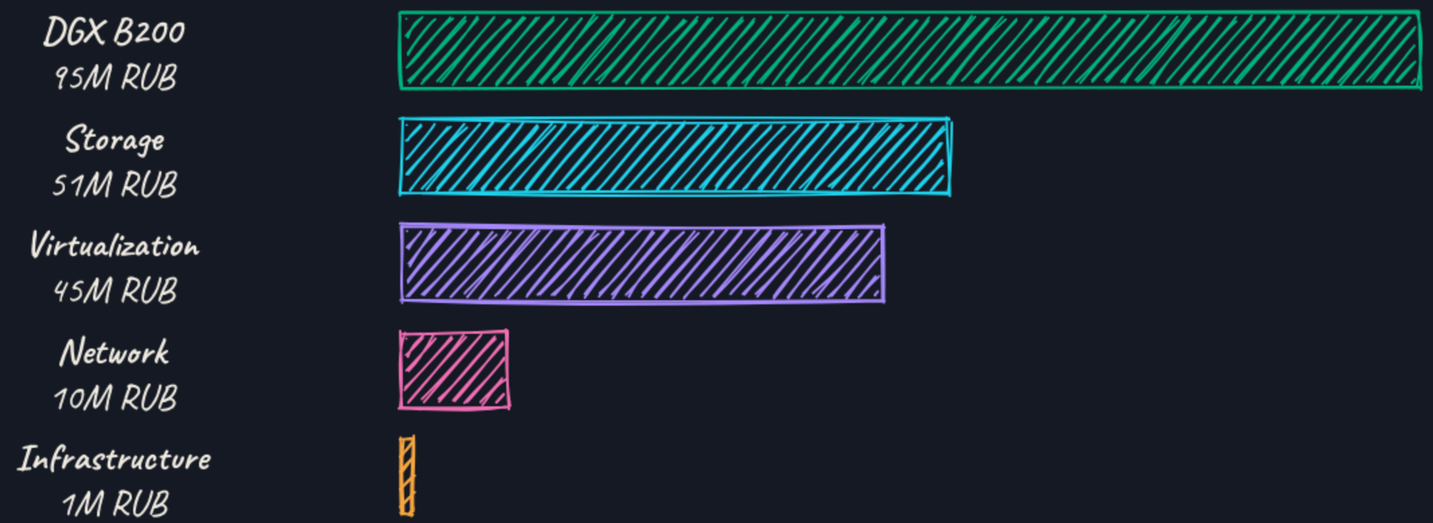
Control

FinOps starts with bill ownership



Owned hardware makes the bill platform-wide

FinOps · Reality check CAPEX

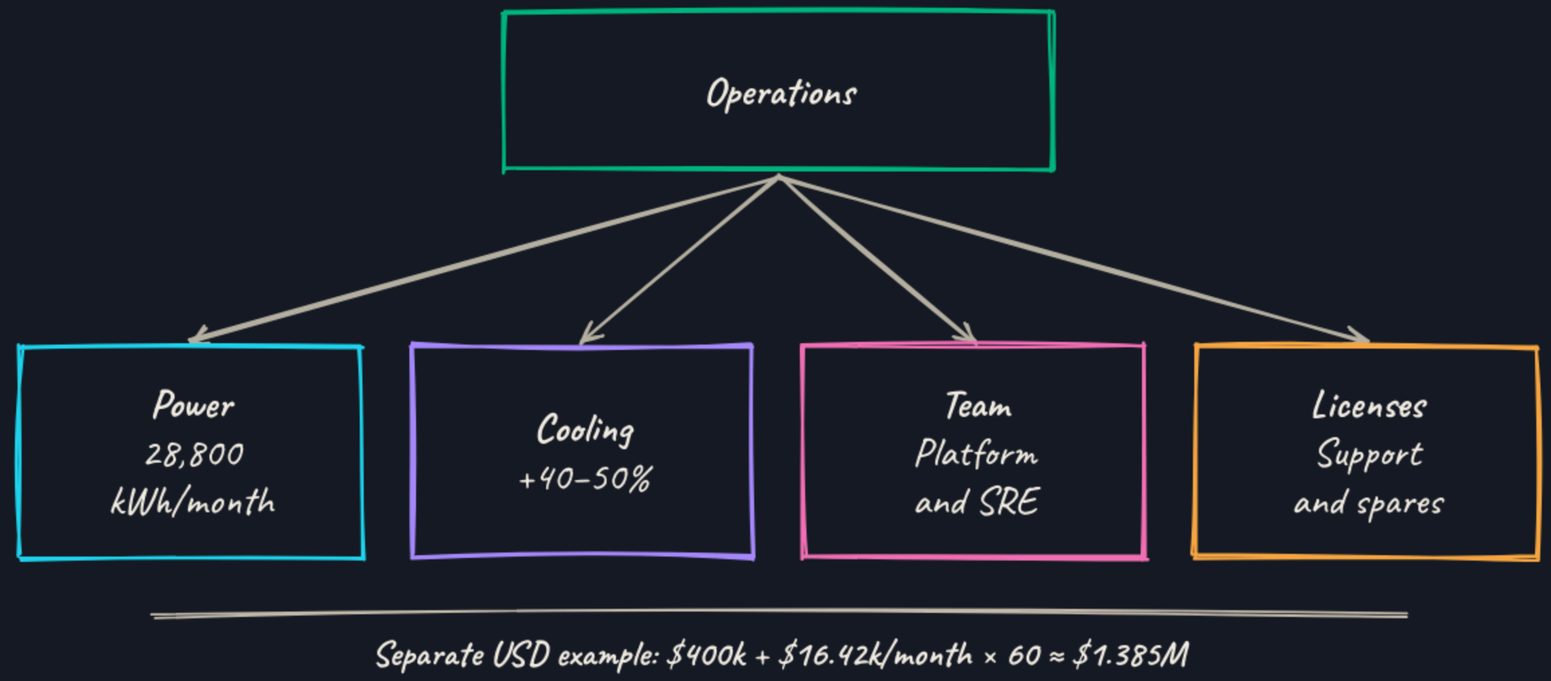


≈202M RUB per configuration; ×3 ≈606M RUB

After CAPEX, operations begin

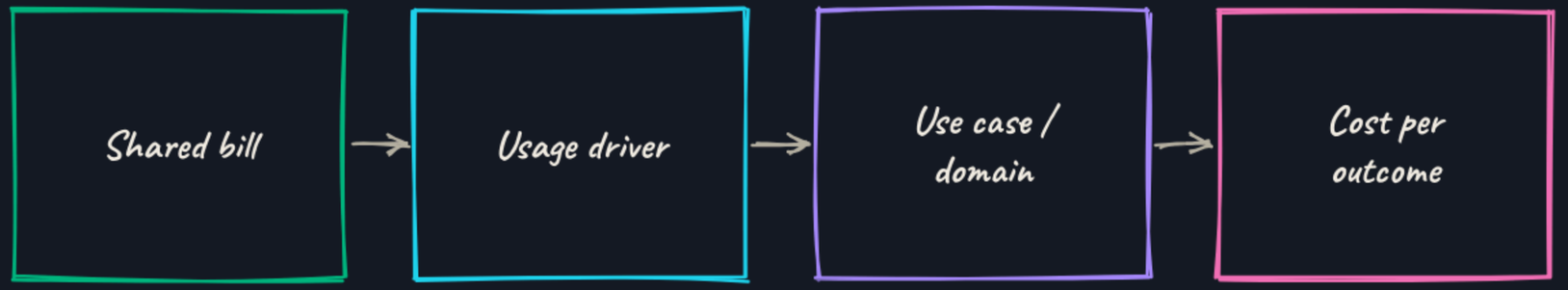
FinOps · OPEX

After purchase



Shared platforms need clear allocation drivers

FinOps · Chargeback Who pays



What managers see

The dashboard connects quality, risk and money

Quality and risk

Use cases
Fallback and policy
Incidents

Economics

Cost per outcome
Cache and retries
Route mix

120 days: scenario to scale

Final · Roadmap 120 days



Final Core question

No measurement, no management

- Without evals, quality looks higher.
- Without fallback, risk grows.
- Without unit economics, economics break.
- Without ownership, disputes begin.

- Managed AI starts with manager decisions

References

AIOps, risk and economics

- OpenAI pricing and docs.
- OWASP, NIST, Langfuse.
- Habr TCO, FinOps Foundation.
- ↗cloud.ru price list, Qwen2.5 benchmark.

HSE University

Day two feedback

Scan the QR code to open the feedback form.