
INTERNAL · PRICING & PACKAGING

Pricing Pre-Read

Agree the unit and the shape first, then the numbers.
Landing on 2 or 3 packages with a defensible minimum
spend at both SMB and Enterprise.

For the team, ahead of the pricing working session
Date 2026-08-15
Status Draft for discussion. No number in here is approved.

Internal · commercially confidential

Executive summary

The problem is not that we lack a price. It is that we have four, and they disagree.

1. We have discounted ourselves 73% in seven months on the same unit. Optibus signed at **\$11.67 per MAU per year**, Togal at **\$4.57**, Transparency Catalog at **\$3.20**. A 3.6x spread with no stated rule, so there is no floor a salesperson can defend.

2. We give away the economics of our own differentiator. In both signed contracts, **Task Agents and Voice run on the customer's model key**. That is the capability we call Execute and price at \$1.50 a run. Today it earns us nothing and we cannot optimise it. Cursor and GitHub do the exact opposite, keeping the high-frequency surface on their own models because that is where margin lives.

3. Our newest contract has the overage pointing the wrong way. Transparency Catalog charges **\$10.67 per 1M tokens committed and \$4.00 on overage**, so overaging is 2.7x cheaper than committing. Optibus already gets this right at **1.54x**. We do not need a new mechanic; we need to apply the one we already used.

4. MAU is the wrong meter and the market has already left it. It pays us per person who showed up while our cost is per interaction. Ably publishes the argument, Clerk retreated to a stricter metric, and Command AI, the closest predecessor to our category, was acquired and de-priced off MAU entirely. The published spread on "MAU" is **200x**, which is the tell that it was never a price. **Keep it as a contract fence, not the meter.**

5. Do not adopt per-resolution pricing. The category invented it and is walking away: **Ada abandoned it**, Decagon's customers mostly decline it, Sendbird argues it makes "success feel like a cost trigger." Only the deflection vendors still lead with it, and we are not one.

6. We are anchored against two markets that are a thousand times apart. As a product tool we get compared to Pendo and Chameleon at **\$279 to \$1,250 a month**. As a support agent we get compared to Sierra and Decagon at **\$150,000 to \$400,000 a year**. Same product. **Whoever defines the category defines the price**, and this deserves the first ten minutes of the session.

7. The comparable set that should actually set our price is three companies. Vendo, Thesys C1 and CopilotKit all sell an embedded agent to a SaaS company for its end users, and all three publish. Two land independently on a **\$49 / \$499** ladder.

What we recommend

A greater-of floor, where the floor buys the AI Gateway.

Your bill is whichever is larger: the plan price, or your usage at the published rates.

One sentence. The plan price **is** the minimum spend, there is no arbitrage because committed and overage rates are identical, and the buyer can compute their own bill. Vendo proves the mechanic works and also proves its failure mode: if the fee is defined as *usage*, BYOK collapses it to zero. **Ours is defined as the Gateway** — quotas, high availability, routing, failover — which Fin cannot sell and which holds its value when a customer brings their own key. Vercel has already proven you can charge for the control plane at 0% token markup.

Three packages: Resolve, Execute, Scale. Metered on what the agent does, fenced on MAU, governance sold as flat add-ons rather than meters.

The one number that blocks everything

What 1M weighted tokens actually costs us, blended across the current model mix. At a ~50% target AI gross margin, that number decides the floor, the overage rate, and which meters can carry a volume discount at all. Everything else in this document is structure, and structure is cheap to change. Get this number before the session if you can.

Part 1. How we have priced, so far

We have used three different commercial models in the last year. They are not variations on a theme; they meter completely different things.

1a. The live pricing page: feature tiers, no price

`Starter · Kinetic · Scale · Enterprise` on `foldspace.ai/pricing`. No published numbers, and the tiers are separated by **features** (Task Agents, voice, analytics, SSO). Every plan says "free trial available."

What it optimises for: sales-led conversations, no anchor given away. **What it costs us:** nobody can buy without talking to us, and feature-gating is the lower-converting design in every study we have.

1b. MAU with BYOK: the prior commercial model

Priced on **Monthly Active Users** of the customer's product who touch the agent, with the customer bringing their **own model provider key** so inference cost sits on their bill, not ours.

Unit		MAU
Who carries model cost	The customer, directly with OpenAI/Google	
Our gross margin	Very high, because we carry almost no COGS	
What the customer feels	Two bills, and full visibility into what the AI actually costs	

What the market actually does here (35 vendors checked, all vendor-published):

- **Flat fee under BYOK is the dominant practice: 19 of them.** Cursor, GitHub Copilot, Cody, Vapi, Retell, ElevenLabs, CopilotKit, Portkey, LangSmith, Botpress, Voiceflow, Zapier, n8n, Make, ServiceNow and others all hold the fee.
- **Only two discount, and both for a reason we do not share. Glean** discounts because BYOK there is bundled with *self-hosting*, so the customer absorbs compute and infrastructure too. **Salesforce** cut the multiplier 7 to 4 on its legacy token-proxy meter but **did not discount Flex Credits**: \$0.10 per action is unchanged.

- **Two vendors charge MORE.** Cursor applies its **\$0.25 per million token "Cursor Token Rate"** to BYOK traffic, on top of what you pay your provider — with first-party models exempt, which is the whole point. OpenRouter charges **5%** on BYOK.

The four ways a flat-fee vendor makes BYOK feel like a win without cutting price: 1. **Quota relief.** Copilot: BYOK does not count against premium-request quotas. 2. **Cap removal.** n8n: "you bring your own key, so no credit limits apply." 3. **Governance entitlement.** Zapier, ServiceNow, Cody, Voiceflow all gate BYOK to Enterprise and sell it as data control. 4. **Nothing at all, because the fee never contained token margin.** Vapi, Vercel, Botpress. **This is our answer.**

Two structural facts worth carrying into the room:

- **The high-frequency surface is never exposed to BYOK.** Cursor keeps Tab completion on its own models; Copilot keeps completions, embeddings and semantic search. Autocomplete is where volume and margin live, and nobody lets a customer key touch it. Our equivalent question: which surface do we never let BYOK reach?
- **Outcome pricing and BYOK are close to mutually exclusive.** Sierra, Decagon and Fin publish no BYOK at all, because the model is their COGS and they are underwriting the failure case. Salesforce proves the mechanism inside one company: it discounted the token-proxy meter and refused to discount the outcome meter.

The residual risk, and how the market handles it. Offering BYOK publicly concedes that the bundled price contained a markup. Cursor's answer is to make BYOK strictly worse on other axes (no zero-data-retention, no Tab, plus the token toll). Ours should be that the fee covers **routing, orchestration, quotas, failover, analytics and trust** rather than inference: **if the fee never contained token margin, BYOK removes a pass-through, not a margin, and there is nothing to give back.**

1c. The current order form: MAU cap + included tokens + overage

The Transparency Catalog order form, effective 2026-09-01, 12 month term. This is the most recent thinking and it is a genuine hybrid.

TERM	VALUE
Use limitation	Up to 5,000 MAU
MAU definition	A distinct end user of the customer's product who interacted with the agent at least once in a calendar month. Non-interacting users do not count.
Included AI usage	125,000,000 weighted tokens per month across all agents
Weighted tokens	Input + output. Compute consumed, not messages sent.
Model management	Foldspace-managed routing by default. Model costs included up to the allowance. Customer does not need a provider account.
Safety limit	Token quotas configured so consumption cannot exceed the allowance without notice. Automated notice at 80%.
At 100%	True-up at \$4.00 per 1,000,000 weighted tokens , or a mid-term upgrade. Service is not suspended for a customer in good standing.
BYOK	Optional, OpenAI or Gemini. Those costs bill direct to the customer and do not count toward the allowance. Electing BYOK does not change the fees.
Fee	\$16,000 for the initial term , paid annually in advance
Itemised	\$16,000/yr platform subscription including AI usage (recurring) · 90-day onboarding and agentic enablement valued at \$6,000 (non-recurring, included) · strategic support included with annual commitment
Other	Customer agrees to testimonials and logo usage

The implied ratios, which are the interesting part:

- $125\text{M tokens} \div 5,000 \text{ MAU} = \mathbf{25,000 \text{ weighted tokens per MAU per month}}$. The order form's own worked example confirms this: $2,000 \text{ extra users} \times 25,000 = 50\text{M tokens}$.
- $\$16,000 \div 5,000 \text{ MAU} = \mathbf{\$3.20 \text{ per MAU per year}}$, or about **\$0.27 per MAU per month**.
- $125\text{M/month} \times 12 = 1.5\text{B tokens/year}$. At the $\$4.00/\text{1M}$ overage rate that allowance is worth **\$6,000**, implying a **\$10,000/year platform fee** underneath.

Two things in 1c we should look at before reusing it

(i) **The overage rate is cheaper per token than the base fee.** \$16,000 buys 1.5B tokens, which is **\$10.67 per 1M tokens**. Overage is **\$4.00 per 1M**. Buying tokens through overage is roughly 2.7x cheaper per unit than buying them in the committed base.

That is backwards from every commit model in the market, where the committed rate is the discount and overage is the penalty. A procurement team will do exactly this arithmetic and conclude they should commit to the smallest possible base and overage into the rest. We should either reprice the base, raise overage above the effective committed rate, or state plainly that the base is a platform fee that happens to include tokens (which is what the itemisation already implies, but the worked example undercuts).

(ii) **25,000 tokens per user per month may be tight.** A single context-heavy agent turn can run 5k to 20k weighted tokens. At 25k/month a user gets roughly two to five real interactions before they are into the allowance. For a product where the agent is the interface, that is not much. If the agent succeeds, we hit the ceiling, and the customer experiences our success as a bill. Worth modelling against real conversation data before we set the ratio.

1d. What we have actually signed

Commercially confidential. This section stays internal and does not go into any customer-facing or marketing asset.

Three order forms, seven months apart. They are not three instances of one model; they are three different models.

	OPTIBUS (SIGNED 2026-02-17)	TOTAL AI (SIGNED 2026-04-15)	TRANSPARENCY CATALOG (EFF. 2026-09-01)
Term	12 months	12 months	12 months
MAU cap	3,000	3,500	5,000
MAU definition	Used the Optibus product at least once in a calendar month. Internal demo/test/QA users explicitly excluded	Used the product at least once in a calendar month	Interacted with the Foldspace agent at least once
Model management	Hybrid BYOK. Copilot on Foldspace's key; Task Agents and Real-Time Voice on Optibus's own key	Hybrid BYOK. Same split: Task Agents / Voice / Share state on customer key	Foldspace-managed routing. 125M weighted tokens/month included. BYOK optional, fees unchanged
Fee	\$35,000	\$16,000	\$16,000
Overage	\$1.50 per MAU per month above 3,000, billed quarterly. Access is never blocked	None specified	\$4.00 per 1M weighted tokens
Payment	Annual in advance	\$4,000 at kickoff, \$12,000 after 3 months. Year 2 annual in advance	Annual in advance
Exit	—	Termination option after first 3 months	—
Extras	Business continuity commitment; competitive exclusivity vs 7 named rivals; explicit end-user embedding right	Full onboarding, Slack support	90-day enablement valued \$6,000

(i) We have discounted ourselves 73% in seven months, on the same unit

DEAL	FEE ÷ MAU CAP	EFFECTIVE
Optibus, February	\$35,000 ÷ 3,000	\$11.67 per MAU per year
Togal, April	\$16,000 ÷ 3,500	\$4.57 per MAU per year
Transparency Catalog, September	\$16,000 ÷ 5,000	\$3.20 per MAU per year

A 3.6x spread on the same metric, trending down every deal. Some of that is real (different segments, different volumes) but none of it is *stated*, so there is no rule a salesperson can apply and no floor to defend. This is the single strongest argument for publishing a rate card, even internally.

(ii) We already have both overage structures live, pointing in opposite directions

- **Optibus: \$1.50/MAU/month overage = \$18 per MAU per year, against a committed rate of \$11.67.** Overage costs **1.54x** the commitment. **This is correct** and matches how every commit contract in the market works.
- **Transparency Catalog: \$10.67 per 1M tokens committed, \$4.00 per 1M on overage.** Overage costs **0.37x** the commitment. **This is inverted**, and it is the arbitrage flagged in 1c.

Optibus already contains the answer to the Transparency Catalog problem. We do not need to invent a mechanic, we need to apply the one we have used before.

(iii) The MAU definitions are not the same metric

Optibus and Togal bill on **anyone who used the customer's product**. Transparency Catalog bills only on **users who interacted with our agent**.

That is a fundamental difference, not wording: – Under the **Optibus/Togal** definition, our revenue is decoupled from whether the agent gets used at all. Safer, but it means adoption work earns us nothing. – Under the **Transparency Catalog** definition, revenue only appears when our agent is genuinely used. Aligned, but it means **low adoption reads as low revenue**, and every activation win is also a price increase the customer feels.

Only Optibus excludes internal demo, test and QA users. That exclusion should be standard in every contract.

(iv) The one that should worry us: we give away the economics of our differentiator

In **both signed deals**, the split is the same: **the Copilot runs on Foldspace's key, and Task Agents plus Voice run on the customer's key**.

Task Agents is the capability the PLG deck calls **Execute** and prices at **\$1.50 per execution**. In every signed contract, **it runs on the customer's key, generates no revenue for us, and we cannot optimise it because we do not control the routing**.

This is exactly backwards from the market pattern in 3c: Cursor keeps Tab completion on its own models, GitHub keeps completions and embeddings on its own, precisely because **the high-frequency surface is where volume and margin live and no vendor lets a customer key touch it**. We have done the reverse and pushed our most differentiated, highest-value surface onto customer keys.

This is the most consequential thing in the three contracts and it should be decision one.

(v) A contract hygiene problem worth fixing this week

The standard Terms grant use of the Platform *"solely for Customer's internal purposes."* Our entire business is the customer embedding the agent for **their own end users**, which is not an internal purpose.

Optibus caught this and added an explicit override: *"Notwithstanding anything to the contrary in the Terms (including Sections 2.1 and 4), Customer is explicitly authorized to make the Platform available to its end-users (customers) as an embedded feature within the Customer products."*

Togal and Transparency Catalog do not have that clause. In two of three contracts, the thing the customer is actually doing sits outside the licence grant. That carve-out should move into the base Terms rather than living in one negotiated order form.

(vi) Concessions we give away without pricing them

Across the three: **competitive exclusivity against seven named rivals** (Optibus), a **business continuity commitment** to keep the service running 12 months or hand over the cloud environment (Optibus), a **3-month termination right** and **staged payment** (Togal), and **90 days of enablement valued at \$6,000** (Transparency Catalog). Only the last one has a number attached anywhere.

Exclusivity in particular is a genuine constraint on our addressable market, granted for free.

Part 2. The menu: what you can meter, and what shape you can sell it in

Pricing decisions collapse into three independent choices. Most arguments happen because people are debating different ones at the same time.

Choice A: the unit (what the meter counts)

UNIT	WHO USES IT	FIT FOR US	THE CATCH
Seat	Classic SaaS	Poor. Our users are the customer's <i>end users</i> , not their employees. Charging per seat means charging them per customer they have.	Only works for <i>builder</i> seats, which is a small number and not where the value is.
MAU	Auth0, Stream, Sendbird	Good commercially, bad structurally. Predictable and forecastable.	See 2a below: it is anti-correlated with value for an agentic layer. Also "active" is definable, therefore arguable.
Weighted token	Model providers, Cursor	Tracks COGS exactly.	The customer cannot forecast it, and it exposes our infrastructure inside their P&L. It is the unit buyers distrust most.
Outcome / resolution	Fin \$0.99, HubSpot \$0.50, Gorgias \$0.90	Comparable. Puts us on the axis buyers already shop on.	Definition disputes. Fin bills "resolution by silence" and it is their most criticised choice.
Action / execution	Nobody publishes this	Our differentiator. The agent doing work is what only we sell.	No market anchor, so no reference price. We would be setting it, not matching it.
Conversation / session	Salesforce \$2.00, Freshworks \$0.49	Simple to count.	Bills whether or not anything worked. Weakest promise in the category.

2a. Why MAU specifically breaks for us

The strongest critique is **Ably's own**, which is notable because they publish an argument against the metric their competitors use:

- To price an MAU you must **normalise a "typical" user**, allocating some assumed volume of activity to each one. **No real user fits that shape.** Most under-consume and pay full freight; a few blow through it.
- MAU **decouples the bill from the cost driver.** The things that actually cost us money (turns, tokens, context length) play no part in the invoice, so the customer has **zero incentive to optimise.**
- **Cost scales with headcount, value scales with usage.** For an interface layer those diverge violently.

Clerk conceded the point in public: it renamed the metric to **MRU** (monthly *retained* user) and made the first day free, so signup spikes and bots do not bill. That is a vendor admitting the definition was the problem.

And note who avoids MAU entirely: **Ably, Pusher and LiveKit all meter work, not headcount** — messages, connection-minutes, session-minutes. **LiveKit, the one actually selling agents, meters agent-session minutes.**

Two more nails, both from named sources:

- **Simon-Kucher: if the product works, the metric shrinks.** User-based metrics fail twice, because "customers may restrict the number of users to control costs" and because AI "may replace human labor, reducing the number of users." We would be betting revenue against our own success.

- **Bessemer names the structure:** "unlike traditional SaaS, where serving one more customer costs virtually nothing, every AI query incurs a non-trivial expense." MAU gives us **flat revenue per human against variable cost per interaction**. That is the whole indictment in one line.

And the tell: the published spread on the same unit is 200x. Effective cost at 100,000 MAU across vendors who publish it: WorkOS \$0, Supabase \$0.00025, Google Identity \$0.00275, Clerk ~\$0.0103, Agora Chat \$0.05. **MAU is not a price. It is a container.**

Where MAU sits on the value ladder: Simon-Kucher's rungs run **Resources (tokens, compute) → Activities (drafts, analyses) → Outputs (conversations, sessions) → Outcomes (cases resolved, revenue)**. MAU is not on this ladder at all; it is a *seat proxy*, which is exactly why it stopped tracking value. **The defensible rung for us today is Activities or Outputs:** actions taken, or completed sessions. Outcomes require attribution we would have to defend in a QBR.

The conclusion for us: MAU counts people who showed up. Our value and our cost both track what the agent did. MAU should be a **fence** (a ceiling in the contract, as the order form already uses it) rather than the meter.

Choice B: the shape (how the money arrives)

SHAPE	MECHANIC	WHERE IT FITS
Pure subscription	Flat fee, all-you-can-eat	Predictable for both sides, but we carry unbounded COGS
Pure usage	Meter only, no floor	No revenue floor, no forecast, terrible for us at SMB
Hybrid: platform + included allowance + overage	A fee that buys a bundle, then a rate beyond it	What our order form already does, and the 2026 default
Credits	Abstract currency, vendor sets exchange rate	Lets us reprice without changing the headline. Also the most criticised model of 2026
Commit + true-up	Annual committed volume, reconciled	How enterprise floors actually get built

Choice C: the gate (what you withhold to create the upgrade)

Ranked by how well it converts, from the evidence we have:

1. **Volume of the thing they already use** (best; usage gates)
2. **Depth of data** (analytics history, export, API access)
3. **Governance and assurance** (SSO, PII, residency, SLA)
4. **Service** (onboarding, named support, response times)
5. **Features** (worst; this is what our live pricing page does today)

Part 3. Competitive reference points

Who we are actually comparing against, and who we are only borrowing mechanics from

This distinction matters, because the two get confused. **Snowflake, Databricks, Datadog, Auth0, Clerk, Ably and LiveKit are not our competitors and are not price comparables.** They appear in this document only because they have solved a specific *mechanical* problem we also have. Reading them as comparables would produce a nonsense price.

TIER	WHO	WHY THEY ARE HERE
1. Direct structural comps	Vendo, Thesys C1, CopilotKit	Embedded agent, sold to a SaaS company, serving that company's end users . Same shape as us. All three publish. This is the set that should set our price.
2. Category-adjacent, sets the buyer's anchor	Pendo, Chameleon, Userpilot, Command AI (dead)	Not the same product, but the budget line a product buyer will compare us to
3. Unit-setters	Fin, HubSpot, Gorgias, Zendesk, Salesforce, Sierra, Decagon	They defined "per resolution" and a support buyer will quote them at us. We do not belong to this lineage (see 3b) but we have to answer for it
4. Mechanic donors only, NOT comparables	Cursor, Anthropic, OpenAI, GitHub Copilot (allowance and overage design) · Snowflake, Datadog (commit floors) · Auth0, Clerk, Stream (MAU definitions) · Vercel, Ably, LiveKit (fence design)	Borrow the <i>mechanism</i> , never the number

The honest summary of tier 1, which is the short list that matters:

	VENDO	THESYS C1	COPILOTKIT
Meters	Dollars of tokens, sandbox minutes, storage, automation runs	API calls	Threads + seats. No end-user metering at all
Entry paid	\$49/mo	\$49/mo	\$39/mo
Next rung	\$499/mo	\$499/mo	\$100/seat/mo, capped at 5 seats
Token cost	1.15x list, marked up	Pass-through at list, no markup	Not addressed on the pricing page
BYOK	\$0 on every meter	Yes, on the free tier	OSS runtime

Two of the three land independently on a **\$49 / \$499** ladder. That is the closest thing to a market price we have, and it is **two orders of magnitude below** the CX-agent contracts in tier 3. See 3g.

3a. The outcome-priced cohort (verified, Aug 2026)

These are the companies a buyer will put next to us if they think of us as a support agent. All figures below are **published by the vendor**.

VENDOR	UNIT	PRICE	WHAT COUNTS	PLATFORM FEE
Fin (ex-Intercom)	Resolution	\$0.99	Resolution, handoff or disqualification. 50/month minimum. A customer going quiet counts as an "assumed resolution."	None. "No setup, integration, or platform fees."
HubSpot Breeze	Resolved conversation	\$0.50	Not passed to a human within 72h. Halved the rate <i>and</i> moved off per-conversation in April 2026.	Service Hub seats, \$20 to \$150/seat/mo
Gorgias	Resolved interaction	\$0.90	Reverts if a human touches it within 72h	Priced on ticket volume, never per agent
Freshworks Freddy	Session	\$0.49	All interaction in 24h (chat) / 72h (email). Billed whether or not it resolves.	Freshdesk Omni seats, 500 free sessions on paid tiers
Salesforce Agentforce	Conversation, or Flex Credit	\$2.00 / conv. Credits \$500 per 100,000	Billed win or lose. Credits burn on every retrieval step.	Add-on from \$125/user/mo
Sierra · Decagon · Ada · Zendesk	Resolution	Not published	—	—

Three things that matter for us:

1. **The band is \$0.49 to \$2.00 and it is compressing.** HubSpot halved in April; Zendesk changed its definition in May so it bills *less*. Undercutting a floor that is already falling buys less share than it costs in margin.
2. **Fin has no platform fee.** If we lead with a platform subscription we are structurally more expensive at low volume than the category leader, regardless of unit price.
3. **The sales-led ones publish nothing.** Sierra, Decagon and Ada have no pricing page at all. Publishing a price is a declaration of motion before it is a pricing decision.

3b. The critical structural difference

Every vendor above bills work **a company already pays a human to do**. Their price replaces a salary line, so the ROI argument writes itself and the buyer is support.

We bill work **our customer's end users generate**. That makes our price their **COGS**, not their headcount saving. It is a harder sale and a better business, and it means the Fin comparison flatters us on price while misrepresenting what we do. Anchoring only against Fin caps us at the deflection budget.

3c. Token bands, overage and annual: how it actually works

This is the section you asked for. All figures verified against vendor-owned pages, Sept 2026.

The four reference structures

VENDOR	THE ALLOWANCE IS DENOMINATED IN	AT 100% OF ALLOWANCE	ANNUAL
GitHub Copilot	Dollars, pegged 1:1 to the subscription price. Pro \$10/mo = \$10 (1,000 credits) base + \$5 flex = \$15 included. 1 credit = \$0.01	Continue at \$0.01/credit. On by default for orgs. Budgets are notification-only unless the admin ticks "stop usage"	Legacy annual users stayed on the old premium-request model
Cursor	Dollars of API usage , amount not published per plan. Two pools: first-party (Grok/Composer, generous) and third-party (Anthropic/OpenAI/Google, at API rates)	Falls back to the first-party pool. Service never stops. On-demand overage on by default for Teams, opt-in for individuals	Toggle exists; rates did not render. ~20% is third-party reported, not confirmed
Anthropic	A multiple of a reference tier. Team Standard = 1.25x Pro, Premium = 6.25x Pro, Max = 5x/20x Pro. Deliberately abstract	Four exits offered in order: upgrade, enable usage credits, move to API credits, or wait for reset. Never auto-charges	Pro \$17/mo annual vs \$20 monthly (~15%). Team 20% off. Min 2 seats
OpenAI	Credits , ~\$0.04 each (derived from a promo, not published). Per-seat inclusion, pooled workspace overage wallet	Draws from the shared workspace pool, gated by both flexible pricing and spend controls	Business \$20/seat annual vs \$25 monthly (20%). Min 2 seats

The API-side ladders, for reference

OpenAI publishes a cumulative-spend ladder; Anthropic does not.

OPENAI TIER	UNLOCK	MONTHLY CEILING	ANTHROPIC TIER	MONTHLY CAP
Tier 1	\$5 paid	\$100	Evaluation	below standard
Tier 2	\$50	\$500	Start	\$500
Tier 3	\$100	\$1,000	Build	\$1,000
Tier 4	\$250	\$5,000	Scale	\$200,000
Tier 5	\$1,000	\$200,000	Custom	none

Note how cheap OpenAI's ladder is: **\$1,000 of cumulative spend unlocks a \$200,000/month ceiling.**

These ladders are credit-risk and anti-abuse devices, not monetisation devices. Anthropic advances you on "usage history and account standing," opaquely. If we want a self-serve-friendly ladder we copy OpenAI; if we want discretion we copy Anthropic.

The five mechanics worth stealing

1. GitHub's 1:1 peg. This is the direct fix for our order-form problem. Base credits are matched **1:1 with the subscription price** and never change: \$10/mo buys \$10 of credits. There is nothing for the buyer to translate, and **no arbitrage between committing and overaging** because the committed rate and the overage rate are the same. Our order form has a 2.67:1 ratio going the wrong way. Pegging removes the problem without lowering the price, as long as we separate the platform fee from the allowance explicitly.

2. GitHub's base/flex split. A margin release valve the buyer pre-agreed to. On top of the 1:1 base sits a **flex allotment** that GitHub states outright is variable, adjustable "as the economics of AI evolve, including model pricing, new models, and improvements in efficiency." Base is consumed first, then flex. It is a documented right to move the allowance when COGS moves, instead of a price rise that reads as betrayal. The live test: on **Sept 1, 2026** Business flex drops 3,000 to 1,900 credits (-37%) and Enterprise 7,000 to 3,900 (-44%) at unchanged prices.

3. Cursor's graceful fallback. The best mechanic in the whole survey. When the expensive third-party pool empties, users automatically fall back to the cheap first-party pool. **The product never stops working; it just runs on a cheaper substrate.** We have exactly this capability already: model routing in the AI Gateway. We could ship "at 100% you fall back to our efficient model tier" instead of "at 100% you get an invoice." Nobody in our competitive set can do that.

4. OpenAI's pooled overage on per-seat inclusion. Inclusion is per-seat, overage draws from a **shared workspace wallet**. One heavy user does not need their own upgrade; the account absorbs it. For us the analogue is per-agent inclusion with an account-level overage pool.

5. Free at the margin. Copilot's code completions **never consume credits** and stay unlimited on paid plans. Only the expensive agentic surface meters. Every vendor here keeps the *habitual* action out of the meter. A meter on the daily habit kills the daily habit.

The pattern across every 2025–26 repricing blowup

Cursor, Replit, GitHub Copilot and Salesforce Agentforce all repriced within twelve months, and all four moved toward dollar-denominated credits or token metering. The shared root cause: **an allowance denominated in a unit whose cost the vendor could not hold stable** (fast requests, flat-rate checkpoints, premium requests, per-conversation).

Three of the four paid customers back. Cursor refunded three weeks of charges. **Replit** had a billing bug on 2025-07-11 that overcharged ~6% of paying users, and **had shipped no spend cap at all** — it gave \$10 credits to every active account plus full reimbursement. GitHub added promotional credit boosts.

The design lesson is not "use credits." It is: denominate the allowance in something whose cost you control, and ship the spend cap in the same release as the meter. Vercel and GitHub both did that and took far less damage than Replit, which did not.

The Cursor lesson, which is the whole section in one line

Cursor's June 2025 blowup was **not** caused by charging for usage. It was caused by changing the unit of the allowance from something the buyer could count (requests) to something they could not see (dollars of tokens), with no in-product translation and no warning before the wall. Their own post-mortem says: "*We recognize that we didn't handle this pricing rollout well, and we're sorry.*" They refunded three weeks of charges.

Every vendor that has moved to allowance-plus-overage since has shipped **the dashboard, the alert thresholds and the plan ladder at the same time as the meter**. Our order form already has the 80% notice. That is the right instinct and it should be a product surface, not a contract clause.

Market state (Growth Unhinged, 2026 B2B SaaS & AI Monetization Report, May 2026)

- **Hybrid is now the most common model at 37%**, up from 25% a year prior.
- **Credits: 29% today, another 33% planning within 6 to 12 months.** About 50% of companies above \$50M ARR intend to launch credits this year.

- **Median target AI gross margin is ~50%**, against 70 to 80%+ for classic SaaS. Only 12% target 80%+.
- **Correction worth knowing:** the famous "AI gross margins are 50 to 60%" line is a **2020 a16z estimate drawn from founder interviews**, not audited data. Bessemer restated it in Feb 2026, which gives it currency but not independence. The sample-backed numbers are **ICONIQ (n=269): AI product gross margin 41% in 2024, 45% in 2025, 52% projected 2026, 59% projected 2027**, with model inference running **20 to 23% of AI product cost** (roughly 11% of revenue at a 52% margin). a16z has since partly inverted its own thesis: *"it's an orange flag if your gross margins are 85, 90, sky-high... there's probably not much AI usage in your product."*
- **54% named "internal costs and margins" the single most important factor** when pricing AI. This is a COGS problem before it is a value problem, which is why question 1 in Part 7 blocks everything.
- **70% say AI spend comes out of the existing technology budget.** We are displacing, not expanding.
- **29% now offer buyers a choice of pricing model**, up from 21%. Salesforce runs four in parallel.
- Poyar's tactical notes: seats plus usage-based overage outgrow both pure models; **rename "overage" to "flex" or "on-demand"**; give a grace period before charging.

Bessemer's published hybrid formula, which is a direct anchor for our order form: a platform fee set at roughly **2x calculated delivery cost, including an allotment, then outcome credits on top.** Their worked example is **\$12,000/year including 100 resolutions, then \$5,000 per additional 100.** Our order form is \$16,000/year including 1.5B tokens. Same shape, and close enough in magnitude that the comparison is worth running properly once we have the COGS number.

Bessemer's anti-complexity test, which is the best single filter for the session: **"identify one model that works at both 10 and 1,000 customers."** Anything that needs a different explanation for SMB and Enterprise has already failed.

On credits specifically (Metronome): credits exist because *costs are knowable and value is not*, so vendors mark up cost and ship. They break on customer confusion, opacity, and because they **mask the value signal so the vendor never learns what customers actually pay for.** Their verdict: **"credits are useful, but not loved."** The observed maturity path is abstract credits to **tangible units** (Synthesia sells video minutes, Fireflies sells transcription minutes). Credits are the fast way to ship, not the end state. We should not start there.

3d. vendo.run

Aisle Technologies, Inc. YC S26, two employees, San Francisco. Founders Yousef Helal (CTO) and Nour Zahzah (CEO). Open source, Apache-2.0. GitHub `runvendo/vendo` : 598 stars, created 2026-06-30. **Nine weeks old.**

What it does, and why it matters to us. A SaaS company installs it and its *end users* describe what they want in plain English; Vendo builds working micro-apps (real UI, data, actions) inside the host product. The agent executes as the signed-in user through the host's existing APIs. This is adjacent to our territory, and their positioning lines are aimed squarely at it:

"A chatbot talks about your product; an embedded agent operates it." "A copilot helps your customer finish the tasks you already shipped; an embedded agent lets them build the task you never shipped."

Their pricing is fully published, and it is the most transparent in the category.

PLAN	PRICE	INCLUDES	DEFAULT SPEND CAP
Free	\$0	\$5 usage/mo (~100 agent turns)	\$0. Hard stop: "a clear error, never a bill"
Pro	\$49/mo (\$490/yr)	\$49 usage/mo	\$100/mo
Teams	\$499/mo (\$4,990/yr)	\$499 usage/mo	\$1,000/mo
Enterprise	Custom	Committed usage	None

The core mechanic, from their Terms §4: "A plan's price is the amount of usage it includes, and your bill for a period is whichever is larger: the plan price, or your usage at the published rates."

A greater-of floor. There is no separable platform fee anywhere in the structure. This is the single most reusable idea in the research: **the plan price is the minimum spend**, stated in one sentence, with no arbitrage possible because committed and overage rates are identical.

Published rate card, identical on every plan including Free: AI at \$1.15/\$5.75, \$2.30/\$11.50 and \$5.75/\$28.75 per M in/out across three tiers; sandbox \$0.01/min; storage \$0.25/GB-month; automations \$3/1,000 runs. **BYOK on model, sandbox or infra reads \$0.**

Two derived observations, both ours rather than theirs:

- 1. Those AI rates are exactly 1.15x Anthropic's published Haiku 4.5, Sonnet 5 and Opus 5 rates.** A flat 15% markup. Vendo does not publish the multiplier or name the models: their gateway maps names to models server-side so they can "retune without a client release." The abstraction is the strategy, not the 15%.
- 2. Their BYOK path makes the fee look like rent.** Because the plan price is *only* a usage prepayment, a BYOK customer on Pro pays \$49 for \$49 of usage they will never consume. What they are actually buying is the feature tier. Vendo has pushed all its defensibility into multi-tenant governance, and made itself indifferent to token revenue.

This is the sharpest available argument about our own BYOK posture. We must decide deliberately whether our fee is framed as a **platform fee** (which stays defensible when a customer brings a key) or a **usage allowance** (which does not). Our order form currently says BYOK "does not change the Fees" while itemising the fee as "platform subscription, including AI usage." That is both framings at once.

One more thing worth reading before the session. On Launch HN, with the clearest rate card in the category, Vendo still got this:

cube00: "Consider tightening up the pricing page. Paying \$49/month for '\$49 of usage a month' doesn't tell me what I'm getting."

Transparency is not the same as legibility.

3e. The rest of the embedded-agent cohort: three findings that change our thinking

(i) Per-resolution pricing is in retreat, and the retreat is vendor-led. **Ada** — the loudest outcome-pricing evangelist in the category — has moved from per resolution to **per conversation**. **Decagon** publishes both and its own glossary notes most customers pick per-conversation. **Sendbird** (now delight.ai)

argues outcome pricing makes "*success start to feel like a cost trigger*." Only Intercom and Zendesk still lead with it, and **both sell deflection, which we do not**.

This contradicts the current PLG deck, which proposes \$0.49 per resolution. See "what this changes" below.

(ii) **The real per-interaction anchors are 20x lower than Fin.** The only two published per-interaction rates in the entire cohort are **\$0.001 to \$0.002** (Thesys, per generated-UI call) and **\$0.05 to \$0.10** (Crisp, per fully-handled conversation). Vendo's stated typical agent turn is **\$0.05**. Fin's \$0.99 is per *resolution*, meaning a whole conversation closed out, not per interaction. **We have been anchoring against the wrong granularity.**

(iii) **Vercel AI Gateway validates the module lever directly.** Vercel charges **0% markup on tokens, including BYOK**, and monetises the control plane instead: \$0.075 per 1K tag writes, \$5 per 1K reporting queries, \$0.10 per 1K requests for team-wide zero-data-retention. They give away the commodity meter entirely and charge for governance. **That is our AI Gateway lever, already proven by someone else.**

Thesys C1 is the other close comp, and it publishes. Vendo itself names it "the closest like-for-like product." Free (3,000 API calls/mo, **BYO key**), Build **\$49** (25,000 calls, \$0.002 overage), Grow **\$499** (500,000 calls, \$0.001 overage). **It meters API calls, not users, and passes LLM cost through at list with explicitly no markup.** Note how closely the \$49/\$499 ladder matches Vendo's, arrived at independently.

And the cleanest dual-track structure published anywhere: Vercel Agent charges \$0.25 per million tokens *plus* pass-through token cost. A small, named platform margin sitting on top of transparent COGS. That is the shape to copy if we want a visible markup at all.

Also worth knowing: CopilotKit, our closest positional comparable, prices on **seats plus stored threads and does not meter end-user volume at all** (\$39/mo Pro, \$100/seat Team capped at 5 seats). And across the whole DAP cohort (Pendo, Chameleon, Userpilot, Amplitude) **AI is either bundled free or metered on an unpublished unit. There is no published per-conversation price anywhere in that category.**

3f. Three numbers that bound our options

(i) **The disclosed token markup band is 0% to 5%, not 2x.** Vercel AI Gateway **0%**, including on BYOK. OpenRouter **5%**. Cursor a flat **\$0.25 per million tokens**. **No survey or benchmark anywhere supports a 2x or 5x application-layer token markup** — any such claim is unsourced. **Vendo's 15% sits above the disclosed band**, which is survivable only because their SKU names hide the underlying model. If we mark up visibly, we will be benchmarked against 0%.

(ii) **Deflation is real but it does not reach the models we need.** a16z's "LLMflation": for an LLM of equivalent performance, **cost falls roughly 10x per year**. But flagship *launch* prices have stayed roughly flat, and the line that matters is **"the frontier never gets cheaper."** Deflation applies to trailing capability. An agent product that needs frontier reasoning does not get the discount. Meanwhile agentic workloads are pushing token volumes up: ICONIQ shows inference *rising* as a share of cost even as unit prices fall.

(iii) **Usage cost is brutally concentrated.** Poyar: **the top 5% of users drove about 75% of usage costs** at a company on flat-fee pricing. This is the single strongest argument for offering BYOK, and for a spend cap: the tail that destroys the margin is the same tail that already has committed model spend and wants its own key.

3g. The anchor problem, which is worth more than any single number

We can be measured against two markets that are three orders of magnitude apart.

IF THE BUYER IS...	THE COMPARISON SET	THE ANCHOR
Product / growth (a DAP replacement)	Chameleon \$279 to \$1,250/mo, Userpilot \$299 to \$849/mo, Pendo free to 500 MAU then quote-only	Hundreds of dollars a month
Support / CX (a deflection agent)	Sierra, Decagon, Glean, Kore.ai	\$150,000 to \$400,000 a year

Same product, same effort, **a thousandfold difference in what the room thinks is a reasonable number.** Whoever we let define the category defines the price. This is the strongest form of question 5, and it deserves the first ten minutes of the session.

One more datapoint that supports the MAU argument: **Command AI / CommandBar**, the nearest predecessor to our category, was acquired by Amplitude and de-priced off MAU entirely — commandbar.com now redirects to Amplitude's guides product, which meters **events with unlimited seats**. The end-user copilot was dropped. The closest thing to us that reached an exit did so by leaving MAU behind.

3h. What this changes

PREVIOUSLY PROPOSED	WHAT THE RESEARCH SAYS	RECOMMENDATION
\$0.49 per resolution	The category is walking away from per-resolution, and we are not a deflection product	Do not lead with per-resolution. It borrows a unit from a lineage that is abandoning it
Fin's \$0.99 as our anchor	Wrong granularity. Per- <i>interaction</i> anchors are \$0.001 to \$0.10	Anchor per-interaction against Vendo/Crisp, and reserve the Fin comparison for the deflection conversation only
MAU as the commercial unit	Actively anti-correlated for an agentic layer (see below)	Keep MAU as a <i>fence</i> , not the meter
Platform fee is hard to justify (Fin has none)	Vercel proves you can monetise the control plane at 0% token markup	Charge for the AI Gateway. It is the defensible floor

Part 4. The levers, as modules

The instruction was to find levers at the **module** level, not the feature level. Going through the product as it actually ships, Foldspace is **three apps plus a gateway plus the distribution layer**:

MODULE	WHAT IT CONTAINS	WHAT IT IS FOR
Agent Studio	Actions, Task Agents, Knowledge, Navigation, Chatterblocks, Style & Voice, shared state, share screen	Build the agent
Analytics	Conversational Analytics (GA), product analytics, Event Explorer, Cost & tokens, session recording, audience explorer, subscription explorer, Mixpanel/Amplitude export	Measure and mine it
Trust Lab	Test automation, evals, regression catching (<i>early access</i>)	Trust it before users see it
AI Gateway	Token Quotas, High Availability, AI Provider Keys, model routing, failover	Control what it spends and keep it up
Distribution	SDK embed, MCP servers, A2A, integrations	Put it in the product and connect it

The five module-level levers

L1. Reach. How much of their user base the agent serves. This is the MAU dimension and it is the one that grows on its own as the customer succeeds. Best expansion lever we have because it requires no new sale.

L2. Depth of work. What the agent is permitted to do: answer, then navigate, then act on the backend, then run multi-step tasks. This is the Four Levels ladder and it is the clearest value ladder in the product.

L3. Intelligence. Analytics and intent signals. This compounds, cannot be rebuilt by a competitor from scratch, and is worth more the longer they stay. The retention moat and the natural top-tier anchor.

L4. Control. The AI Gateway plus governance. **This is the underrated one.** Token Quotas is not a safety feature, it is *pricing infrastructure we hand our customer so they can tier their own product*. The docs say it outright: "If you plan to monetize on AI, quotas are where a free tier ends and a paid one begins." We are selling the customer the thing they need to build their own AI business model. Nobody else in our set sells that.

L5. Service. Onboarding and agentic enablement (already valued at \$6,000 on the order form), strategic support, SLA, response times. Currently given away with an annual commitment. It is a real cost and a real lever.

The shape of the answer: L1 and L2 should be *metered* because they scale with the customer's success. L3 and L4 should be *subscribed* because they compound and are not consumption. L5 should be *attached to commitment level*, which is what we already do.

Part 5. Minimum spend, at both ends

The constraint we cannot design around

Fin has no platform fee. Their published line is "no setup, integration, or platform fees," with a 50-outcome monthly minimum (about \$50). If we lead with a platform subscription we are structurally more expensive than the category leader at low volume, whatever our unit price is.

So our floor cannot be justified as "access." It has to be justified as **something Fin does not give you**. We have three candidates, and they are all modules: the AI Gateway (they do not have one), Analytics and intent data (they do not sell it), and enablement (they do not do it).

How floors are actually built, from the research

MECHANISM	HOW IT WORKS	WHERE IT FITS
Seat minimum	2 seats at Claude Team and ChatGPT Business; both cut to 2 in 2026	Not available to us. Our users are the customer's end users.
Monthly minimum	Fin's 50 outcomes/month	Cleanest SMB floor. Low friction, no negotiation.
Platform fee	Decagon's reported ~\$50k/yr; Aibly \$29 to \$399/mo on top of usage	Works if the fee buys a named module, not "access"
Annual prepaid commit	Snowflake capacity commitments, drawn down against usage, discounted vs on-demand	The enterprise floor. Behaves like subscription revenue in the P&L while burndown behaves like usage revenue, which is why finance likes it
Cumulative-spend ladder	OpenAI Tier 1-5	Anti-abuse, not a floor. Do not confuse the two

The cleanest floor mechanic found: "greater of"

Vendo's Terms §4, in one sentence: *"A plan's price is the amount of usage it includes, and your bill for a period is whichever is larger: the plan price, or your usage at the published rates."*

The plan price *is* the minimum spend. No separate platform fee to justify, no arbitrage between committing and overaging (the rates are identical), and the buyer can compute their own bill. It is GitHub's 1:1 peg and a revenue floor in the same mechanic.

The catch, and it is real: on Launch HN, Vendo was told *"paying \$49/month for '\$49 of usage a month' doesn't tell me what I'm getting."* **A greater-of floor needs the fee to buy something nameable, or it reads as rent** — which is exactly what happens to Vendo under BYOK.

Our version fixes their problem: greater-of floor, where the floor also buys **the AI Gateway**. Vercel has already proven you can charge for the control plane at 0% token markup. That gives us a floor that survives a customer bringing their own key, which Vendo's does not.

Recommended floor design

SMB floor: greater of the plan price or metered usage, where the plan price buys the AI Gateway. Not "access to Foldspace." The fee buys the **AI Gateway**: quotas, high availability, model routing, failover. A real thing Fin cannot sell, worth money to anyone embedding an agent in production, and it holds its value under BYOK. Usage bills at the same published rate whether it is inside the floor or above it, so there is no arbitrage.

Enterprise floor: an annual committed volume at a discounted unit rate. Snowflake's architecture. The commitment buys a better rate; the rate is the reward for the floor. Add pooled allowance across agents (Cursor charges for pooling explicitly, so it is worth real money) and the governance modules.

The finding that should shape our whole rate card

Where the commit discount sits tells you where the vendor's margin is, and every mature vendor withholds the discount exactly where COGS is real.

- **Datadog** charges a **1.0x** multiple on log *ingest* (the commodity), but **1.5x** on indexed logs and **1.3x** on AI credits.
- **Snowflake's** storage commit discount reaches **40% at \$40M ACV**, but **AI Credits compress only ~6%, and are explicitly carved out of the Platform Credit Discount.**
- **Sentry's** newest, COGS-heaviest SKUs (logs, profiling, application metrics) are **pay-as-you-go only, with no reserved rate offered at all.**

The rule: be generous with the discount curve where COGS is near zero, and absent where COGS is real. For us that means the AI Gateway, analytics and governance can carry real volume discounts; the interaction meter should carry almost none. That is the opposite of how our order form currently behaves.

Two prices for the same unit, published side by side

Only **Datadog** and **Sentry** publish committed and on-demand rates next to each other. **Sentry's is a flat 1.25x across every single category**, which converts "I don't want to forecast" into a quantified 25% premium a buyer can accept or decline. Datadog's varies 1.16x to 1.5x by SKU. Everyone else (Snowflake, Databricks, MongoDB, Twilio, Vercel) keeps the curve off the page entirely, **which is exactly what makes procurement adversarial**: the option to stay uncommitted is priced, but invisible.

Recommendation: publish both rates. A flat, single multiple is easier to defend than a per-SKU table.

Metronome's fix for our inverted overage, which is better than raising the rate

"Overage was often the stick, because it was priced punitively and was intended to push customers into committing to larger upfront purchases."

Their prescription: **kill the overage rate entirely and extend the commit rate past the commit**, "rewarding them for using it more than they had initially planned." **MongoDB already does this** — run out of credits and you transition to list billing with **no premium at all**. Combined with Monthly Commitment, which *banks the shortfall within the term* so unused credit can offset a later month's overage.

That is a cleaner answer to our \$10.67-vs-\$4.00 problem than raising the overage rate: **make the two rates the same and let the commit buy the discount.**

The public floors, for calibration

Nobody publishes a minimum commit. **Not Snowflake, Databricks, Datadog, MongoDB, Twilio, or Vercel.** It is universally sales-gated. The only floors that are public are platform fees:

VENDOR	PUBLIC FLOOR
Cloudflare Workers	\$5/month minimum charge per account , spanning five products
Vercel Pro	\$20/month per team (not per seat)
Supabase Pro	\$25/month per org
Sentry Team	\$26/month billed annually
Fin	50 outcomes/month , about \$49.50
Twilio	\$8,000/month is where card payment stops and invoicing begins

Metronome's segment guidance: monthly minimums for SMB and mid-market; prepaid credits with an access schedule for enterprise; **commit discounts land in a 10 to 50% band.**

Three sub-decisions, with a recommendation on each

(a) What happens at 100%? The split in the market is **not by segment, it is by what the failure costs the customer.** Where losing the service is worse than a surprise bill (Snowflake, Datadog, MongoDB, Cloudflare, Vercel) overage is uncapped and billed. Where a surprise bill is worse than losing the service (**Sentry, Supabase, Lovable**) the hard cap is the *default*.

For an agent that is our customer's live interface, losing the service is clearly worse. So: **ship fallback**, which we are one of very few vendors able to do because model routing already exists in the AI Gateway. "At 100% your agent keeps working on our efficient model tier" beats any invoice. Pair it with opt-in true-up.

Two things to be honest about in the room: – **Every published hard cap is lagged or partial.** Vercel checks spend "every few minutes" and tells customers to set the cap below their true maximum; Cloudflare's budget alerts process usage **once daily, for the prior day, and do not pause anything**; Supabase's cap exempts about eleven line items. Sentry's is the only true real-time cap, and it achieves that by **dropping data rather than settling money.** – **Nobody auto-upgrades to the next tier at 100%.** It could not be documented at a single named vendor. If we want it, we are inventing it.

Supabase's design rule is the one to copy: cap what scales with unpredictable traffic; do not cap what the customer deliberately provisioned. Compute, IPv4 and replicas keep billing even with their spend cap on, and they say why.

(b) Rollover? Open at **monthly reset, no rollover** for the included allowance; it is the universal self-serve default and m3ter's reason is honest: rollover disincentivises usage inside the period and creates revenue-recognition problems. But adopt Anthropic's split explicitly: **prepaid balances do not expire, included allowances do.** Concede capped, renewal-conditional rollover at enterprise only, as a closing lever, which is exactly how Snowflake uses it.

(c) Buffer the allowance. m3ter's guidance, stated precisely: for **stable** usage, structure it so customers "are likely to buy 30 to 50% more than they're likely to use in a given period." For **volatile or growing** usage, sell less buffer and upsell more often. Deliberately give more headroom than you expect to be consumed: breakage protects the margin and the generosity protects the relationship. Directly relevant to the 25,000 tokens-per-MAU ratio, which currently looks buffered the wrong way.

What Fin's floor actually is: a 50-resolution monthly minimum at \$0.99, so roughly **\$49.50/month**, with no seats and no platform fee when used over the API. That is the number our SMB floor gets compared against. We are not going to beat it on price, so the floor has to buy something Fin does not have.

One number to keep in view: Zylo's SaaS Management Index found **66.5% of IT leaders reported unexpected charges due to consumption-based or AI pricing models**. Bill shock is now the default expectation our buyer walks in with. Whoever removes it wins the deal.

The trust mechanic worth copying outright

Intercom credits the resolution back when a customer reopens the conversation, even across billing periods. It converts "you charged me for a bot that failed" into a non-event. Our equivalent: do not bill an interaction the agent did not complete, and say so on the pricing page.

And the cautionary one: Figma's extra AI credits worked out roughly **5.6x more expensive than adding a seat**, produced a public forum thread and a EUR 24.40 to EUR 627.28 invoice complaint, and Figma retreated on **2026-08-25** by granting 2x credits at the same price. **If your meter is more expensive than your seat, customers will find out and say so publicly.**

Part 6. Package options

Three structures, with the trade-off stated. **All numbers illustrative.** The point of the session is to pick a structure; the numbers move once we have the COGS answer.

Structure A: Floor + published meter + module gates

An evolution of the order form we already have, with the arbitrage fixed.

	LAUNCH	SCALE	ENTERPRISE
Floor	\$500/mo, card, self-serve	\$2,500/mo	Annual committed volume
Modules	Agent Studio, embed, AI Gateway (quotas, HA, routing)	+ full Analytics, intent signals, Conversational Analytics	+ Trust Lab, SSO, PII, residency, pooled allowance across agents
Included	Allowance pegged 1:1 to the fee	Larger allowance + flex allotment	Committed volume at a discounted rate
At 100%	Falls back to efficient model tier, or opt-in true-up	Same	Same, plus burndown against commit
Service	Docs and community	Onboarding	90-day enablement, named support, SLA

Why this one. It keeps what the order form already does right, fixes the peg, gives a floor at both ends, and differentiates on **modules** rather than features. The AI Gateway at the entry tier is the answer to "Fin has no platform fee": we are not charging for access, we are charging for the thing that keeps their agent up and their spend bounded.

The risk. Three tiers plus a meter plus an overage rate is four things to explain. It works only if the pricing page translates the allowance into something countable, which is the Cursor lesson.

Structure B: Thin platform + pure pass-through

Anthropic Enterprise's shape: a small fee that buys governance, with 100% of usage metered at published rates.

- **Floor:** the platform fee alone, which can be genuinely small.
- **Pro:** the cleanest and most honest structure in the survey. Total separation of license from consumption. Very easy to explain, and the buyer can forecast their own usage.
- **Con:** almost no revenue floor from usage, and it exposes our COGS directly, which recreates the BYOK problem. Also the hardest to hold margin on if consumption is lumpy.

Structure C: Outcome-priced with a monthly minimum (*weakened by the research*)

Fin-comparable: per resolution, with a minimum monthly commitment.

- **Floor:** the monthly minimum, exactly as Fin does with 50 outcomes.
- **Pro:** directly comparable, and the buyer already understands the unit. Still the right answer if the buyer turns out to be **support**.
- **Con, and it got worse:** the band is \$0.49 to \$2.00 and **compressing**, and **the category is walking away from the unit**. Ada moved off per-resolution to per-conversation. Decagon's own glossary says most customers pick per-conversation. Sendbird argues outcome pricing makes "success start to feel like a cost trigger." We would be adopting a unit its inventors are abandoning, in a lineage (deflection) we do not belong to.

The recommendation

Structure A, with Vendo's greater-of mechanic as the floor and the AI Gateway as what the floor buys.

	LAUNCH	SCALE	ENTERPRISE
Floor	Greater of \$500/mo or usage	Greater of \$2,500/mo or usage	Annual committed volume, discounted rate
What the floor buys	AI Gateway: quotas, HA, model routing, failover	+ full Analytics and intent signals	+ Trust Lab, SSO, PII, residency, pooled allowance
Meter	Published per-interaction rate, same inside and above the floor	Same	Same, discounted, drawn down against commit
Fences (not metered)	MAU ceiling, history retention, 1 agent	Higher ceilings, more agents	Unlimited
At 100%	Falls back to the efficient model tier	Same, or opt-in true-up	Burndown against commit
Service	Docs, community	Onboarding	90-day enablement, named support, SLA

Why this survives contact with the market: – The floor is a **minimum spend at both ends** without inventing a platform fee we cannot defend. – It **holds up under BYOK**, because the floor buys the Gateway, not tokens. This is the specific failure in Vendo's model. – It **avoids the retreating unit**. We meter interactions, not resolutions. – MAU survives as a **fence** in the contract, which is what the order form already does, rather than as the meter. – **Fences are free margin**. Retention windows, concurrency ceilings, agent counts: they cost us almost nothing, they are binary so they never produce a surprise invoice, and Ably, LiveKit and Vendo all use them this way.

Five design rules the research is unanimous on

- 1. Three visible axes is the ceiling.** Intercom, Vercel, Clerk, Supabase and Sentry all sit at exactly three. HubSpot sits at four (seats + contact tier + credits + edition) and is the least readable page in the set.
- 2. Package on needs, meter on volume** (Monetizely). Do not let the pricing metric be the thing that differentiates packages, or you signal that quantity is the only difference between them.
- 3. Free the seats that do not carry value.** Sentry, Vercel, Retool and Intercom all give unlimited or free read-only seats deliberately. If usage is the value metric, charging for access double-taxes.
- 4. Use the tier as a multiplier, not a fourth axis** (Snowflake: the edition *is* the credit multiplier). The most underused simplification in the whole survey.
- 5. Publish one worked example.** Every legible pricing page in the set has one. Every vendor without one shows up in a forum thread about their invoice.

The filter to apply to whatever we choose (Bessemer): *"If the math doesn't work at 10 customers, it won't at 1,000."* And Tunguz's line for the room: **"Cost-plus is a payment processor with a dashboard. Value-based is software."**

The one number that decides the floor levels: our blended COGS per interaction. Vendo's typical agent turn is \$0.05 and Crisp's fully-handled conversation is \$0.05 to \$0.10. If our cost per turn is meaningfully above that, the floors go up or the model changes.

Which to choose still depends on one answer

If the buyer is **support**, Structure C, because comparability wins and the deflection budget already exists. If the buyer is **product or engineering**, Structure A, because the modules are the value and there is no incumbent line item to be compared against.

That is question 5 in Part 7, and it decides more than it looks like it does.

Part 7. What I need from you

These are the questions where I cannot pick a defensible answer without your input, ordered by how much they change the model.

The blocking ones

1. **What does 1M weighted tokens actually cost us, blended across the current model mix?** Everything hangs on this. The \$4.00/1M overage is either a healthy margin or below cost depending on whether we are mostly on Haiku/Flash-class models or mostly on Sonnet/GPT-class ones. I cannot design a floor without it.
2. **What is the real token consumption per active user per month?** We assumed 25,000 on the order form. What does the Cost & tokens report say for Optibus, Mixmax, RoofSnap and Transparency Catalog? If the real number is 60k, the order form is underpriced by more than 2x.
3. ~~What are the three existing customers actually paying?~~ **Answered by the order forms, see 1d.** The follow-on question is sharper: **which of the three structures is the one we standardise on?** Optibus has the correct overage direction, Togal has the softest commitment, Transparency Catalog has the only real token model. They cannot all be right.

One thing to decide before we touch the decks

The current PLG deck proposes **Resolve at \$0.49 per resolution**. The research says the category is abandoning that unit and that our per-interaction anchor should be roughly **\$0.05, not \$0.49**, because Fin's \$0.99 is per resolved *conversation*, not per interaction. **The deck's pricing slides should not go further until we settle the unit.** The naming (Resolve, Execute, Scale) and the module structure survive; the unit and the numbers do not.

The strategic ones

4. **Do we want BYOK to survive?** The research answers most of this. **Holding the fee flat is the dominant published practice (19 of 35 vendors), and two vendors charge *more* for BYOK.** Cost is concentrated enough to make it worth offering: the **top 5% of users drive ~75% of usage cost**, and that tail is exactly the cohort with committed model spend that wants its own key. **So: keep BYOK, keep the fee flat, and pair it with a non-price concession** (quota relief, cap removal, or an Enterprise governance entitlement) as every flat-fee vendor does. The one thing we must fix: **the fee has to be visibly denominated in capability, not inference.** Our order form currently says both at once, and that is the Vendo failure mode.

5. **Is the buyer the product team or the support team?** This decides whether we price against Fin (support deflection) or against Pendo/Command AI (product adoption). We cannot anchor against both.
6. **What is the smallest deal we are willing to service?** The \$16,000 order form includes 90 days of hands-on enablement. That is not a self-serve motion. If SMB is real, something has to be genuinely self-serve.
7. **Do we want a published price at all, or stay sales-led?** Every sales-led agent company (Sierra, Decagon, Ada) publishes nothing. Every developer-adjacent one publishes a ladder. This is a motion decision before it is a pricing decision.

How to prompt me on this

You asked. Most useful, in rough order:

- **Give me the constraint, not the answer.** "Minimum \$12k ACV at enterprise, must be self-serve under \$500/mo" is far more useful than "make three tiers." I can generate structures; I cannot invent your revenue targets.
 - **Tell me what you have already rejected and why.** The two most useful things you said this week were "I can't do free" and "Learn is scale." Both killed a branch instantly and saved a full rebuild.
 - **Point me at the artefact.** The order form was worth more than an hour of description. Same for the Four Levels doc. If a document exists, attach it.
 - **Separate the unit question from the number question.** Ask me to fix the structure first. Numbers move; structure is what we have to live with.
 - **Say who the audience is.** A board pre-read, a sales enablement doc and a customer-facing pricing page are three different documents. This one is written as an internal working doc.
 - **When you reject something, one clause of why is enough.** "Names are weak, doesn't feel researched" was enough to rebuild the whole naming system correctly.
-