BOOK ZERO · THE AI ECONOMY MONETIZATION SERIES

The Monetization Manifesto

for the AI Economy

Short. Sharp. Quotable.
The intellectual stake in the ground.

10 Laws · 10 Monetization Models · 5 Provocations

PREFACE

Why I Wrote This Book

Twenty-five years of watching the same mistakes. It is time to name them.

I have spent twenty-five years inside the engine room of software monetization. Not advising from a distance — inside. Running revenue operations for companies scaling from ten million to ten billion. Designing billing architectures that had to survive the scrutiny of Big Four auditors, the demands of enterprise procurement, and the unforgiving arithmetic of SaaS unit economics. Watching companies that got pricing right compound their advantage quarter after quarter, and watching companies that got it wrong spend those same quarters in crisis mode: billing disputes with major

customers, revenue recognition restatements, leakage that nobody could quantify, and forecasts that were wrong in ways nobody could explain.

I have watched three complete waves of software monetization. The first was the perpetual license era — enormous upfront payments, annual maintenance shakedowns, and a customer lock-in model so effective that it still supports the revenue bases of companies that haven't shipped a meaningful product update in a decade. The second was the SaaS revolution — the subscription model that democratized enterprise software, created the vocabulary of ARR and NRR and churn, and generated a generation of companies worth more than most countries. The third is the one we are living through now: generative AI as an economic force that does not merely change how software is delivered, but changes what software is, who can build it, what it costs to produce, and how its value can possibly be measured and captured.

Every pattern I have seen across twenty-five years is now playing out simultaneously, at ten times the speed, with ten times the stakes. And the monetization community — the people who price, bill, recognize revenue, forecast consumption, and govern AI spending — is not ready.

This book is my attempt to accelerate that readiness.

It is short by design. A manifesto is not a textbook. The other nine books in this series are textbooks — comprehensive, detailed, with implementation guides and worked examples and decision trees. This book is an argument. It makes claims. It names the problems with precision. It proposes frameworks and models. It ends with ten laws that I believe every person building, running, or financing an AI business needs to internalize before they make another pricing decision.

I wrote it because every conversation I have with a CPO, a CFO, or a CRO eventually arrives at the same place: they are managing a fundamentally new kind of product with the tools and mental models of the previous era. They are pricing AI like it is SaaS. They are forecasting AI consumption like it is predictable. They are invoicing AI services like the value is obvious and the customer agrees with the number on the page. None of those

assumptions are correct, and the cost of maintaining them compounds with every passing quarter.

The companies that will define the AI economy are not necessarily the ones with the best models or the most impressive demos. They are the ones that understand what their AI is worth — and have built the commercial architecture to capture it. That architecture is what this book is about.

PART ONE

The Problem: Why AI Monetization Is Broken

And who broke it.

PROVOCATION ONE

You Are Pricing Like It Is 2015

The seat-based model was brilliant for SaaS. It is actively harmful for AI. Here is the evidence — and the cost of doing nothing.

Let us be precise about what the seat-based pricing model was designed to do, because it was genuinely elegant and genuinely right for its time.

When Salesforce and its generation of SaaS pioneers needed to convince enterprise buyers to move off premise-installed software and onto cloud-delivered software, they faced a specific problem: the risk perception of the buyer. Asking a company to pay a large upfront license fee for software that would live on someone else's servers was a significant ask. What if the vendor went out of business? What if the software didn't work? What if you needed to get your data back? The subscription model solved this problem with elegant simplicity: pay monthly, cancel anytime, no upfront commitment. The seat became the unit because it mapped directly to something the buyer already

understood — headcount. You have five hundred salespeople, you buy five hundred seats. The price is predictable, the logic is transparent, and the CFO can approve it in a single meeting.

For roughly fifteen years, this model worked extraordinarily well. It created a generation of extraordinarily valuable software companies, a financial vocabulary that became the lingua franca of enterprise software — ARR, NRR, churn, expansion, logo count — and a set of investor mental models that produced some of the most efficient capital allocation in the history of technology. A company with 120% net revenue retention and low churn was mathematically a compounding machine. The ARR you sold this year was worth more than one dollar of ARR next year, because customers expanded. Investors paid thirty times revenue for these businesses. The math made sense, and everybody involved understood the math.

Here is what the seat model assumed, usually without stating it explicitly: that consumption was fixed. A seat was a seat. The person in that seat would use the software in broadly predictable ways. The variance between your heaviest and lightest users existed, but it did not affect your costs, because your underlying infrastructure was not consumption-based. Hosting five hundred users who each logged in twice a day cost roughly the same as hosting five hundred users who each logged in twenty times a day. The billing system was simple: count the seats, multiply by the price, generate the invoice. Finance could model the business with confidence.

Generative AI made consumption a variable. Not just variable in the way that Snowflake compute credits or Twilio API calls were variable — those were bounded by the use case and broadly predictable within a deployment. AI consumption is non-deterministic in a more fundamental sense: the cost of each interaction depends on the complexity of the input, the behavior of the model, and the nature of the task being performed, and all of these can change dramatically from one interaction to the next with no change in the customer's configuration.

Consider what happens when a company deploys an AI assistant to help its legal team. On Monday, the AI reviews a hundred standard non-disclosure agreements —

straightforward documents, predictable structure, moderate token consumption. On Tuesday, a partner asks the AI to review fifty complex merger and acquisition documents with extensive exhibits and jurisdictional issues — token consumption is four times higher per document. On Wednesday, the same partner asks the AI to research case law across eight jurisdictions and synthesize the findings into a briefing memo — token consumption is ten times the Monday baseline. Nothing in the customer's usage profile changed in a seat-based sense. The same user, the same seat, the same license. But the underlying compute cost to serve those interactions varied by an order of magnitude across three days.

This is what we mean by non-deterministic consumption. The cost varies with the work, not with the headcount. And when your revenue is decoupled from your costs in this way, margin arithmetic becomes treacherous.

What the Industry Did: Two Case Studies

The transition away from pure seat-based pricing is already happening across the industry, and the pattern is consistent regardless of company size or market position.

Salesforce introduced Einstein AI capabilities bundled into their existing cloud subscriptions starting in 2016, progressively expanding these features without meaningfully adjusting per-seat pricing to reflect AI consumption. By 2023, as generative AI features became central to the product, Salesforce faced a strategic repricing challenge: customers who were heavy AI users were generating dramatically more underlying compute cost than customers who were not, but both paid the same seat price. The company eventually introduced Einstein credits — a consumption-based overlay — precisely to address this problem. The seat model had taken them as far as it could.

Microsoft faced the same challenge with Copilot. Initially priced as a flat add-on to Microsoft 365 subscriptions, Microsoft subsequently moved to a more consumption-aware model for certain enterprise deployments as usage patterns clarified. The pattern

is consistent across the industry: seat-based pricing works well for initial adoption, then creates structural problems as AI usage deepens and consumption variance grows.

The Three Compounding Problems

The seat model applied to AI products does not produce a single error — it produces three compounding problems simultaneously, each of which makes the others worse.

The first problem is margin erosion. When a vendor bundles AI capability into a seat price, they are making an implicit bet about average consumption. Specifically, they are betting that the average token consumption per seat per month, multiplied by the per-token infrastructure cost, will remain below a threshold that keeps gross margins acceptable. If actual consumption comes in above that threshold — because customers are using the AI more effectively, or because the AI is getting better at doing more things, or because the heavy users are dominating the usage distribution — gross margins compress. The vendor's revenue is fixed by the seat price. Their costs are growing with consumption. The math deteriorates.

This is not a hypothetical. Early-stage AI companies that priced their products on a per-seat basis before they had reliable consumption data regularly discovered, six to twelve months into deployment, that their heaviest customers were consuming five to ten times the tokens of their lightest customers while paying the same price. The per-seat price that was profitable for light users was loss-generating for heavy users. And in AI, the heavy users tend to be exactly the customers you most want to keep — they are getting the most value and are least likely to churn.

The second problem is value misalignment. A seat price tells the customer that what they are paying for is access to a tool. But what the customer is actually receiving — and what creates measurable economic value for them — is a set of outcomes: contracts reviewed at higher speed and lower cost, customer service tickets resolved without human intervention, code written faster and with fewer defects, financial analyses completed in hours rather than days. When the price is anchored to access rather than

to those outcomes, the vendor loses the ability to participate in the value they are creating. The customer captures 100% of the productivity gain while the vendor charges the same flat fee they charged before the AI made their product dramatically more valuable.

The third problem is expansion friction. The natural expansion motion for a seat-based product is to sell more seats. But AI productivity tools create a different dynamic: they make existing employees more productive, which reduces the need for headcount growth, which eliminates the natural trigger for seat expansion. A legal department that uses AI contract review to process twice as many contracts with the same team is not going to hire more lawyers to create an excuse to buy more seats. The seat-based vendor has designed a model that rewards their customers for demonstrating the value of the product by eliminating the mechanism by which the vendor could capture that value.

> ***"The seat model assumes consumption is fixed. Generative AI made consumption a variable. That single shift breaks everything downstream."***

The Predictable Three-Year Sequence

The consequences of maintaining seat-based pricing for AI products follow a predictable three-year sequence that is now visible across multiple cohorts of early AI adopters.

In year one, growth is excellent. Seat-based pricing is easy for buyers to understand, easy for procurement to approve, and easy for sales to close. Enterprise buying committees, trained on SaaS, process seat-based proposals quickly. Win rates are high, sales cycles are short, and the metrics look compelling. The internal narrative is that the product-market fit is proven.

In year two, the margin compression becomes visible. Engineering is asked to optimize inference costs. Product is asked to add consumption caps or usage limits to prevent the heaviest users from eroding margins further. Customer success is asked to manage usage expectations during renewal conversations. The financial model that looked so clean in year one is showing cracks.

In year three, the strategic crisis arrives. Competitors have entered the market with consumption-based or outcome-based pricing. These competitors can tell potential customers: pay us per contract reviewed, per ticket resolved, per deal closed. We share your risk. We grow when you grow. The vendor with pure seat-based pricing cannot have that conversation — their entire commercial architecture is built around access, not outcomes. And re-platforming a pricing model while preserving an existing customer base is a multi-year change management exercise that most organizations are not operationally prepared to execute.

The companies that will own the AI economy are the ones who design their monetization architecture correctly from the beginning — or who recognize the trap early enough to escape it before the installed base becomes an anchor. The first step is recognizing that the seat model is not a safe default. It is a choice with consequences, and those consequences compound.

The Seat Model vs AI Reality — Side-by-Side Comparison		
Assumption	Seat model assumes	AI reality reveals
Consumption unit	User × time period (predictable)	Token/task/outcome (highly variable)
Cost driver	Infrastructure per user (flat)	Compute per interaction (variable)
Expansion mechanism	More seats = more revenue	More value = same seats, same revenue
Invoice predictability	Perfectly predictable	Varies with usage intensity

Value captured	Access value (low ceiling)	Capability value (high ceiling — uncaptured)
Margin structure	Fixed revenue, fixed cost = stable	Fixed revenue, variable cost = compression risk

PROVOCATION TWO

Monetization Is a Data Problem in Disguise

Every billing error, every revenue recognition restatement, every leakage event traces to the same root cause. It is not a process problem. It is a data problem.

Walk me through the last time your company had a significant billing error. Not a rounding difference — a real error, the kind that required customer calls, credit memos, internal postmortems, and a conversation with the CFO about revenue that had been recognized incorrectly. Now trace it backward, past the billing system that generated the wrong invoice, past the process that failed to catch it, past the approval workflow that should have stopped it. What do you find at the root?

You find a data problem.

In twenty-five years, I have not found a single exception to this pattern. The billing error always traces to a data object that was wrong, missing, or inconsistent with another data object somewhere else in the commercial stack. A product record that did not capture the right pricing attributes for an AI consumption tier. An entitlement record that had not been updated when the contract was amended to include an additional AI workflow. A metering event that was logged against the wrong customer identifier after a customer reorganization. A token consumption record that was aggregated across the wrong time boundary before being passed to the billing engine. Something, somewhere in the chain of data objects that runs from the initial sale to the final invoice, was not right.

And because that data was wrong, everything downstream was also wrong. Not just the invoice — the revenue recognition entry, the deferred revenue calculation, the customer-level P&L, the consumption forecast, the renewal model. All of it derived from bad data. All of it wrong in ways that cascaded invisibly until the error surfaced — usually at the worst possible moment, with the worst possible audience.

This insight — that monetization failures are fundamentally data failures — is the most important and the least understood principle in commercial operations. It is the principle that this entire series is built on. And it is the principle that separates organizations that scale their monetization with confidence from those that survive quarter-close through heroic manual effort.

The reason this principle is so poorly understood is that it runs counter to the intuition of most executives and operators. When a billing error occurs, the natural instinct is to fix the process — add a review step, require an additional approval, build a reconciliation report. These interventions are not wrong. They catch some errors. But they do not address the root cause, which means the errors recur, in new forms, in new places, forever.

You cannot fix a data problem with a better process. You can route around it temporarily, but the data problem will find a new way to surface. The only durable fix is to correct the underlying data model.

> ***"You cannot fix a data problem with a better process. You can only route around it temporarily. The data problem will find a new way to surface."***

The Correct Order of Operations

The correct order of operations for building a monetization capability that can survive the complexity of AI products is: data first, process second, integration third.

Define your monetization objects — the canonical data structures that represent products, prices, contracts, entitlements, metering events, invoices, and the relationships between them — before you design any process. Build your processes around those objects, so that every step in every workflow is operating on precisely defined data. Then, and only then, evaluate which software systems and integrations you need to support those processes.

Most organizations do this in exactly the reverse order. They select a billing platform. They implement it. They discover, three months into implementation, that the billing platform's data model does not match the actual complexity of their AI product and pricing. They build workarounds. The workarounds create new data inconsistencies. Those inconsistencies cause billing errors. They hire more operations staff to catch the errors manually. Some errors are caught and some are not. Leakage accumulates invisibly. Quarter-close becomes a recurring crisis. And the root cause — the fundamental mismatch between the data model the business actually needs and the data model the systems enforce — is never resolved.

This pattern is extraordinarily common and extraordinarily expensive. The typical cost of a major billing platform implementation that fails due to data model mismatch — including the platform licensing, the implementation services, the internal engineering time, the rework, and the revenue that leaked during the period of dysfunction — runs into eight figures for mid-market companies and nine figures for large enterprises. The correct architecture, designed data-first, would have cost a fraction of that and produced a system that could actually support the business.

The Three-Layer Monetization Architecture — Why Order Matters		
Layer 1: DATA	Define monetization objects with precise schemas before anything else	Without this: every process operates on ambiguous inputs; every system

		encodes the wrong model; errors compound invisibly
Layer 2: PROCESS	Design billing, recognition, and governance workflows around the defined objects	Without this: data is correct but workflow exceptions create manual workarounds that re-introduce errors
Layer 3: INTEGRATION	Select and integrate systems that can support the defined processes	Without this: duplication and translation between systems introduce new data inconsistencies

The Thirteen Monetization Objects

A complete monetization data model for the AI economy requires thirteen canonical objects. These are not software concepts — they are business concepts that must be precisely defined before any software is selected or any process is designed. Every company that has successfully scaled its AI monetization has done so because it defined these objects with unusual precision.

Object	Definition and role in AI monetization
Product	Defines what is being sold — attributes, pricing logic, entitlement rules, and relationship to the underlying AI capability. In AI, a product may be a token bundle, an agent workflow, an outcome-based service, or a composite. The product object must accommodate this complexity without forcing simplification that loses information.
Price	Defines how the product is charged — per unit, per tier, per outcome, as a subscription, as a reservation, or as a combination. AI billing often requires multiple simultaneous price objects for a single product.
Entitlement	Defines what the customer has the right to consume. For AI products this includes token budgets, agent call limits, model access rights, and outcome delivery commitments — each tracked separately and enforced in real time.
Meter	Defines how consumption is measured. Must capture token counts, task completions, API call durations, model inference times, and outcome delivery events — attributed to the correct customer, product, and billing period with zero ambiguity.

Event	The atomic record of a single unit of consumption. Every token processed, every agent task completed, every outcome delivered generates an event. These are the raw material from which all billing is constructed. If event data is wrong, everything downstream is wrong.
Invoice	The formal request for payment, derived from event data aggregated according to billing rules. Must be traceable back to every event that contributed to every line item.
Contract	The commercial agreement governing what is sold, at what price, under what conditions, for what duration. The source of truth for entitlement and billing rules.
Credit	A reduction in amount owed, applied at the invoice or line level with documented authorization. Every credit must trace to an approved adjustment request.
Allocation	The assignment of revenue to accounting periods, business units, and cost centers. Critical for multi-party AI stacks where revenue must be split between model providers, platform operators, and application developers.
Token Budget	The governance constraint on AI consumption — a defined allowance that can be monitored in real time and enforced through throttling. The CFO's primary financial control for AI spending.
Agent Task	A discrete unit of agentic work — a single completion of a defined workflow by an AI agent. The billing unit for agent-layer products.
Outcome	A verified business result — a resolved ticket, an approved contract, a closed deal — that triggers a billing event in outcome-based pricing models.
Asset	The per-customer record of every AI product, model, agent, and entitlement that a customer has purchased and is currently consuming. The register from which renewals, expansions, and accurate billing derive.

Together, these thirteen objects form the complete vocabulary needed to describe any AI monetization scenario. A company that can model its commercial operations in terms of these objects — with precise definitions for each, clear relationships between them, and defined lifecycle states for each object — has the data foundation required to scale its monetization with confidence. A company that cannot will find that every addition to its product portfolio, every new pricing tier, and every new customer segment creates new opportunities for billing errors, recognition failures, and revenue leakage.

PROVOCATION THREE

Five Layers. Five Pricing Logics. One Company Using One.

The AI economy is not one market. It is five overlapping markets, each with its own cost structure, value proposition, and pricing logic. Using one pricing model across all five is five mistakes compounded.

The single most common strategic error in AI monetization is treating the AI economy as a single market with a single pricing logic. It is not. The AI economy is a five-layer stack, and each layer has fundamentally different economics, fundamentally different cost structures, and fundamentally different relationships between the vendor and the customer. A pricing model designed for one layer will create commercial distortions, margin problems, or customer trust issues at any other layer. Understanding this is not complexity for its own sake. It is the foundation of every good pricing decision in the AI economy.

The image that has emerged from early AI monetization strategy work is a stack diagram that has become a reference point for practitioners in this field: at the bottom, the Compute Economy; above it, the Model Economy; then the Token Economy; then the Agent Economy; at the top, the Outcome Economy. Each layer is a distinct market with distinct participants, distinct unit economics, and distinct pricing logic.

What makes this map genuinely useful is not the labels — it is the recognition that companies operating at different layers are not competing with each other, but they are building on each other. An outcome-based AI healthcare service company is a customer of the agent economy, which is a customer of the token economy, which is a customer of the model economy, which is a customer of the compute economy. Value flows up the stack. Cost flows down. Margin is created at the layer where value can be captured more efficiently than cost accumulates.

And here is the critical observation: most companies operating at higher layers of the stack are pricing as if they are operating at a lower layer. An outcome-based legal AI that charges per token is a company that has built a higher-layer capability but is capturing value as if it were a lower-layer infrastructure provider. The premium they have earned by taking on the complexity of outcome delivery, SLA commitment, and quality verification is being left entirely on the table.

The Five-Layer AI Economy — Layer Characteristics

Layer	What it is	Natural pricing unit	Example participants	Key economic dynamic
Compute Economy	GPUs · AI factories · raw inference infrastructure	Time-based: per GPU per minute, per instance per hour, with reservation discounts	CoreWeave, Lambda Labs, Crusoe, AWS, Azure, GCP at the infrastructure layer	Extreme capital intensity. Scale economics decisive. Margins thin. Utilization is everything.
Model Economy	Foundation models · LLMs · image generators · code models	Consumption-based: per million input tokens, per million output tokens, with quality tiers	OpenAI, Anthropic, Google DeepMind, Mistral, Meta AI	Quality and capability differentiation. Context window length as premium feature. Fine-tuning as value-add.
Token Economy	Enterprise AI consumption governance · token flows · budget management	Usage-based with governance controls: token budgets, hierarchical spend limits, chargeback	Every enterprise AI deployment. The layer most CFOs currently lack visibility into.	The CFO's primary financial exposure. Token Factory concept: organizational capability to govern token flows.
Agent Economy	Autonomous AI workflows · copilots · multi-step task execution	Per-task or per-workflow: price per completed	Coding agents (Devin, Copilot), legal agents (Harvey),	Task definition quality is the key commercial risk. Unclear task

		unit of agentic work	finance agents, CS automation	boundaries = billing disputes.
Outcome Economy	Verified business results · SLA-committed AI service delivery	Value-based: price per outcome achieved, with SLA terms and variable consideration	Emerging category. Healthcare AI, legal AI, sales AI moving toward outcome models.	Maximum alignment. Maximum complexity. Requires measurement infrastructure and attribution methodology.

The five-layer diagram above is not a hierarchy in the traditional sense — it is not the case that companies at higher layers are inherently more valuable or more sophisticated than companies at lower layers. CoreWeave and Lambda Labs at the compute layer are building some of the most defensible businesses in the AI economy, precisely because compute infrastructure has massive barriers to entry and strong utilization-based economics. The layers are distinct not in terms of prestige but in terms of economics: each layer has its own cost structure, its own value proposition, and its own natural pricing unit, and confusing these creates commercial problems that compound over time.

> ***"Every layer of the AI economy needs its own pricing logic. A company that applies one model across all five has not simplified its pricing — it has guaranteed it is wrong in at least four places."***

The practical implication is not that companies need to implement all five pricing models simultaneously. Most companies operate primarily at one or two layers. The implication is that companies need to be honest about which layer they primarily inhabit, design their pricing model for that layer's specific economics, and resist the

temptation to borrow pricing logic from adjacent layers where the fundamental economics do not hold.

An outcome-based healthcare AI that charges per token is a company that has built a higher-layer capability but is capturing value as if it were a lower-layer infrastructure provider. The premium it has earned by taking on the complexity of outcome delivery, SLA commitment, and clinical quality verification is being left entirely on the table — potentially hundreds of dollars of captured value per interaction being billed at cents.

PART TWO

The Models: Ten Monetization Models for the AI Economy

Each model: definition · economic logic · when it works · when it destroys value · right-conditions checklist.

Ten models follow. They are not mutually exclusive. Most successful AI businesses use two or three in combination. The art is knowing which models to combine, at which layer, for which customer segment — and when to migrate from one combination to another as your product and market mature. What makes a model 'right' is not its elegance or its familiarity — it is the degree to which it aligns vendor economics with customer value creation at your specific layer of the stack.

MODEL 1 · PER-TOKEN PRICING	
Definition	You charge for every token processed — input tokens entering the model and output tokens returned. The most granular unit of AI consumption becomes the unit of billing.
Economic logic	Aligns cost and revenue at the infrastructure layer. Every unit of compute cost maps to a unit of billable activity. No consumption means no charge and no cost.

When it works	At the model and token layers for API-first businesses. When customers have predictable, bounded use cases and technical sophistication to manage consumption.
When it fails	When customers cannot predict or control consumption — especially with agentic workflows. When billing complexity erodes trust. When your AI capability has grown far beyond what per-token pricing can capture.
Right conditions	API-first product · technical customer base · bounded use cases · robust metering · token budget governance · customer ability to set hard limits

Per-Token Pricing: The Deeper Argument

The per-token model is the entry point of AI monetization — the model that most API-first AI businesses start with, and the one that most companies stay on longer than they should. Understanding precisely when it works and when it fails is foundational knowledge for anyone pricing an AI product.

The economics of per-token pricing are straightforward. The vendor's underlying compute cost is roughly proportional to token consumption. Charging per token creates a direct link between cost and revenue, which protects margins in a way that flat-rate pricing cannot. For an early-stage API business with minimal sales infrastructure and a technically sophisticated customer base, per-token pricing offers several advantages: zero friction on adoption (customers pay only when they use), automatic scaling of revenue with usage, and natural usage data for building consumption forecasting models.

The OpenAI API pricing model is the canonical example of per-token pricing executed well. Distinct price points for input versus output tokens (output tokens typically cost two to four times more than input tokens because generation is computationally more expensive than context processing). Premium pricing for higher-capability models. Volume discounts for committed usage. Cached input tokens at a discount (because cached context costs less to process). This pricing architecture rewards efficient prompting, scales naturally with customer growth, and creates a clear commercial rationale that technical buyers understand and can explain to their finance teams.

Where per-token pricing creates problems is at the agent and application layers. When an AI agent is executing a multi-step autonomous workflow — researching a topic, synthesizing findings, drafting a document, reviewing it, revising it, and outputting a final result — the token consumption is substantial and highly variable. The same task might require 50,000 tokens one day and 200,000 tokens another day depending on the complexity of the source material and the number of revision cycles the agent requires. For a business customer who bought the agent to perform a defined function, this variance is deeply uncomfortable. They cannot budget for it. They cannot predict it. And when a month-end invoice arrives that is four times last month's invoice with no corresponding four-times increase in business value, the billing conversation becomes a relationship problem.

The right conditions for per-token pricing: your product is an API accessed by technical developers who can manage consumption; the use cases are bounded and the consumption variance is limited; you have robust token budget governance features that allow customers to set hard limits; and your pricing is calibrated to remain below the point where customers feel compelled to optimize aggressively against your meter.

MODEL 2 · PER-TASK / PER-AGENT PRICING	
Definition	You charge per unit of autonomous work completed — per task, per workflow, per agent execution. The agent does a job; the customer pays for the job.
Economic logic	Aligns value delivered at the agent layer with value captured. Customers pay for what was accomplished, not for underlying resource consumption. Agent efficiency gains benefit the vendor, not the customer.
When it works	When tasks are clearly definable and completable. When completion can be verified programmatically. When there is natural agreement on what 'a task' means.
When it fails	When task definition is ambiguous or contested. When tasks vary dramatically in complexity. When completion verification is expensive or disputed.
Right conditions	Well-defined task boundaries · programmatic completion verification · relatively homogeneous task complexity · cost attribution per task · billing cadence matching customer cash flow

MODEL 3 · OUTCOME / VALUE-BASED PRICING	
Definition	You charge for a defined business outcome achieved — a resolved support ticket, an approved contract, a closed sales opportunity — not for the process or resources involved.
Economic logic	Maximum alignment between vendor revenue and customer value. The vendor captures a share of value created, scaling with quality and quantity of outcomes. The customer pays only for confirmed results.
When it works	When outcomes are measurable and attributable. When there is shared agreement on what constitutes success. When the vendor has confidence in AI performance to accept outcome risk.
When it fails	When outcomes are poorly defined. When attribution is contested. When variable consideration creates revenue recognition complexity. When AI reliability is insufficient to accept performance risk.
Right conditions	Precisely defined outcome · agreed measurement methodology · reliable attribution model · outcome verification infrastructure · ASC 606 variable consideration framework

Outcome Pricing: The Deeper Argument

Outcome-based pricing is where AI monetization ultimately wants to go — and where most of the industry is not yet operationally ready to arrive.

The intellectual case is simple. If an AI product resolves a customer support ticket, the value created is roughly equal to the fully-loaded cost of a human agent resolving that ticket plus the customer satisfaction improvement from faster resolution plus any reduction in escalation costs. If that value is twenty-five dollars per ticket, and the AI costs the vendor three dollars per ticket to operate (compute, model, overhead), and the vendor charges eight dollars per ticket resolved, the customer captures seventeen dollars of value, the vendor earns a five dollar margin, and everyone's incentives are aligned. Compare this to a per-token model where the vendor charges for the tokens consumed by the resolution regardless of whether the ticket was actually resolved — which aligns nobody's incentives, rewards inefficient agents, and creates billing disputes the moment quality degrades.

The practical challenge is the prerequisite stack. Outcome-based pricing requires that you can answer five questions with high confidence and with evidence that a skeptical customer will accept. First: what is the outcome? The definition must be precise enough that both parties agree unambiguously on whether it occurred. 'Customer satisfaction' is not an outcome — it is a sentiment. 'Ticket closed without escalation and customer CSAT score above 4.2 within 24 hours' is an outcome. Second: how is the outcome measured? Who measures it, with what data, at what frequency, and who adjudicates disputes about the measurement? Third: was the AI causal? The counterfactual question — would this outcome have occurred without the AI — is the hardest to answer and the most frequently contested. Fourth: who verifies? Ideally, verification is programmatic and based on data both parties can access rather than on the vendor's assertion. Fifth: what happens when the outcome is not achieved? The SLA breach protocol, credit mechanism, and remediation process need to be as carefully designed as the outcome pricing itself.

Cognition's Devin, the autonomous software engineering agent, has navigated these questions in the enterprise market with a subscription-plus-outcome hybrid that commits to a defined scope of engineering tasks (the subscription floor provides budget predictability) while adding outcome-based charges for delivered features (aligned with business value). This structure gave enterprise customers the budget predictability they needed for procurement approval while demonstrating the vendor's confidence in their AI's performance.

MODEL 4 · SUBSCRIPTION FLOOR + CONSUMPTION OVERAGE	
Definition	A committed monthly or annual fee giving the customer a defined allocation, plus usage-based charges for consumption above that allocation.
Economic logic	Gives the vendor revenue predictability (the floor) while preserving upside from heavy users (the overage). Gives the customer budget certainty for core usage while preserving scaling flexibility.
When it works	For enterprise customers who need to budget predictably but whose AI usage is growing. As a transitional model migrating from pure subscription to consumption-based billing.

When it fails	When the floor is set so high that light users feel they are paying for unused capacity. When overage pricing suppresses usage. When billing complexity erodes the relationship benefit of the subscription base.
Right conditions	Clear allocation definition · bimodal usage patterns (base plus peaks) · billing infrastructure tracking floor vs overage separately · overage pricing perceived as fair

MODEL 5 · TOKEN BUDGET / PREPAID CREDITS	
Definition	The customer purchases a defined token budget or credit pool in advance. Consumption draws down the budget. When exhausted, customers top up or service is throttled.
Economic logic	Converts open-ended consumption risk into a finite, pre-approved expenditure. The CFO can approve a token budget through existing procurement — a bounded commitment with a defined ceiling.
When it works	When enterprise customers need to govern AI spend through existing budget approval processes. When the customer is early-stage and not yet confident in consumption forecasts.
When it fails	When budgets are set so conservatively they throttle productive AI usage. When the top-up process creates workflow friction. When customers feel trapped by breakage (unused tokens they cannot recover).
Right conditions	Clear consumption reporting · easy self-serve top-up · fair roll-over policy · governance dashboard for finance teams · alerts before budget exhaustion

MODEL 6 · RESERVED COMPUTE / GPU-AS-A-SERVICE	
Definition	The customer commits to defined GPU capacity for a defined period — a number of GPUs, a duration, and a performance specification — at a discount versus spot pricing.
Economic logic	Converts variable infrastructure cost to fixed cost for the vendor, improving margin predictability. Creates utilization pressure for the customer — committed capacity they are incentivized to use.
When it works	At the compute layer for customers with predictable, high-volume AI workloads. For AI factories running continuous inference at scale. For production deployments with known baseline capacity.
When it fails	When customers over-commit to capacity they cannot utilize. When model efficiency improvements make reserved capacity over-specified before the contract allows re-sizing.

Right conditions	Proven production workload · reservation discount justifying commitment risk · flex provisions for capacity adjustment · customer FinOps capability to manage utilization

MODEL 7 · MARKETPLACE / REVENUE SHARE	
Definition	A platform operator charges a take rate on transactions between model providers, application developers, and end customers transacting through the platform.
Economic logic	The platform creates value through aggregation, reduced transaction costs, and trust infrastructure. The take rate is the price of access to that aggregation. Value compounds through network effects.
When it works	When the platform has genuine multi-sided network effects. When the platform provides real integration, trust, or discovery value that participants cannot easily replicate outside it.
When it fails	When the take rate incentivizes bypass. When the platform lacks genuine network effects and is simply a toll booth. When royalty accounting creates complexity that erodes participant trust.
Right conditions	Genuine multi-sided network effects · transparent revenue share calculation · take rate below the value of aggregation · strong governance and dispute resolution · audit-quality royalty accounting

MODEL 8 · GAIN-SHARE / RISK SHARING	
Definition	Vendor and customer share the financial upside of AI-delivered results. The vendor charges a base fee plus a percentage of measured value created above a defined baseline.
Economic logic	Perfect alignment between vendor revenue and customer value creation. The vendor has skin in the game. Trust is embedded in the commercial structure — the vendor is accountable for results.
When it works	In high-value, measurable use cases where AI contribution to business outcomes can be reliably quantified. When the vendor has confidence in AI performance sufficient to accept downside risk.
When it fails	When outcome measurement methodology is disputed. When the base fee does not cover costs if the AI underperforms. When administrative complexity exceeds the value of alignment created.

Right conditions	Agreed, auditable outcome measurement methodology · accepted baseline performance · AI performance history to model expected gain range · legal framework for gain calculation disputes

MODEL 9 · FREEMIUM TO ENTERPRISE PLG	
Definition	The product is offered free at limited usage levels, with premium tiers unlocked by consumption growth or enterprise requirements. Usage itself is the primary expansion trigger.
Economic logic	Acquisition cost is minimal because customers self-acquire. Expansion is natural — customers who get value use more, which creates a natural upgrade conversation. The free tier generates metering data that identifies expansion candidates.
When it works	For AI products where individual practitioners can demonstrate value independently before organizational adoption. When the product has a natural usage-driven conversion point.
When it fails	When the free tier is so generous that users never need to upgrade. When the product requires enterprise integration that makes self-serve conversion impossible. When enterprise CAC economics don't support free-tier acquisition.
Right conditions	Clear free-to-paid conversion trigger · usage analytics identifying expansion candidates · self-serve upgrade path · enterprise tier with genuine differentiation · freemium COGS that do not erode margins

MODEL 10 · AGENT-TO-AGENT COMMERCE	
Definition	AI agents acting as autonomous economic agents — purchasing services from other AI agents, negotiating SLAs, processing micropayments, and settling accounts — without human involvement in individual transactions.
Economic logic	As AI agents become more capable, the volume of machine-to-machine transactions will dwarf human-initiated ones. Human procurement economics cannot scale to the transaction volumes agent commerce will generate.
When it works	For well-defined, high-volume, low-complexity service exchanges between agents. When transaction value is small enough that human oversight of individual transactions adds no value.
When it fails	When services being exchanged are complex enough that automated SLA negotiation cannot capture the relevant terms. When regulatory environment requires human oversight of spending decisions.

Right conditions	Machine-readable pricing APIs · AI agent identity and authentication infrastructure · programmatic credit assessment · automated dispute resolution · settlement rails at micropayment scale

PART THREE

The Strategy: Choosing, Combining, and Evolving Your Model

The right model is not the one your competitors use. It is the one your economics require.

PROVOCATION FOUR

The Model Selection Matrix

Match pricing to your position in the stack. Not to what your competitors charge. Not to what your customers expect. To what your economics actually require.

The model selection matrix is a decision tool, not a decision. It narrows the field of viable pricing models based on two variables — where you sit in the five-layer stack and what kind of company you are — and eliminates the category errors before they are made. From the constrained set it produces, you make a choice based on your specific product, customers, and growth stage. But you make that choice with the confidence that you are at least choosing among viable options rather than between viable and catastrophic ones.

The matrix has two dimensions. The horizontal dimension is your primary layer in the AI economy stack. The vertical dimension is your company archetype. The intersection defines which pricing models are viable for you, which are available but suboptimal, and which represent category errors that will create commercial problems regardless of how well they are executed.

The Five Company Archetypes:

AI Infrastructure Providers sit at the compute layer. They sell GPU time, AI factory capacity, and the raw substrate of inference. Their customers are model companies, enterprise AI teams, and AI application developers. The commercial relationship is purely about capacity — the infrastructure provider has no visibility into what their compute is producing and bears no responsibility for its quality. The right pricing models are time-based: reserved compute, spot compute, and hybrid reservation-plus-spot. Per-token pricing is a category error here because the infrastructure provider cannot measure tokens. Outcome pricing is a category error because the infrastructure provider cannot define, measure, or guarantee outcomes. The companies that have tried to introduce outcome pricing at the compute layer have universally discovered that they cannot operationalize it.

AI Model Companies sit at the model and token layers. They sell access to foundation models through APIs. Per-token is the natural primary model, with subscription floor plus overage for enterprise customers who need budget predictability. Model companies that attempt per-task pricing face a fundamental measurement problem: tasks are defined by the application layer, not the model layer, and the model has no visibility into whether a task was completed. Outcome pricing is theoretically interesting at the model layer but practically unworkable for the same reason.

AI Application Companies operate across the widest range of layers — token, agent, and outcome — and have access to the widest range of viable pricing models. This is where the model selection decision has the largest strategic consequence, because application companies are the ones who interact directly with enterprise buyers and have the opportunity to capture value at the highest levels of the stack. Early stage application companies typically start with subscription or freemium — necessary for acquisition. Growth stage companies should be migrating toward per-task or outcome models that capture expansion value. Mature application companies with proven AI performance and strong measurement infrastructure should be exploring gain-share.

Enterprise AI Buyers are the mirror image of vendors in the pricing conversation. Their monetization challenge is internal: how do they govern AI spending, allocate costs to business units, measure ROI, and ensure that AI investments generate the financial returns that justified the budget approval. The FinOps OS for AI — token budgets, chargeback models, agent cost attribution, utilization dashboards — is the enterprise buyer's answer to the vendor's pricing strategy.

AI Marketplace Operators create the infrastructure connecting model providers, developers, and enterprise buyers. Their model is the platform take rate — calibrated carefully enough to attract participation without incentivizing bypass. The strategic error for marketplace operators is setting take rates based on what the market will bear rather than on the value the platform creates. Platforms whose take rates exceed the value of their aggregation will be bypassed. Platforms whose take rates are below the value of their aggregation are leaving strategic opportunity unrealized.

The Model Selection Matrix — Archetype × Layer × Viable Models

Company archetype	Primary layer	Viable primary models	Category errors to avoid
Infrastructure Provider	Compute	Reserved compute · Spot compute · Hybrid reservation	Per-token is a category error — no token visibility. Outcome is a category error — no outcome visibility.
AI Model Company	Model / Token	Per-token with quality tiers · Subscription floor + overage · Marketplace participation	Per-task is problematic — task boundaries defined above model layer. Outcome requires attribution not available at this layer.
AI Application Company	Token / Agent / Outcome	Per-task · Outcome / value-based · Subscription + overage · Gain-share · Freemium → enterprise	All models viable depending on stage. Seat-only pricing is the category error — misses expansion value entirely.

Enterprise AI Buyer	Internal governance	Token budgets · Chargeback by team · Agent cost attribution · Utilization dashboards	This archetype does not sell — it governs. The right 'model' is FinOps OS, not a pricing model.
Marketplace Operator	Platform	Take rate / revenue share · Royalty splits · Transaction fees	Take rate above platform value triggers bypass. Take rate below platform value is foregone strategic revenue.

The matrix is a constraint, not a prescription. Within the viable models for your archetype and layer, the right choice depends on variables the matrix cannot determine: your AI's reliability, your measurement infrastructure, your customers' procurement processes, and your own risk tolerance. But the matrix eliminates the category errors — the choices that will create commercial distortions regardless of how well they are executed.

> ***"The worst pricing decision a company can make is to copy a competitor's model without understanding whether that model fits their own layer and archetype."***

The reason this error is so common is that pricing decisions are made under competitive pressure. When a significant competitor announces a pricing change, the natural instinct is to respond. But a competitor's pricing model was designed for their specific position in the stack, their specific cost structure, and their specific customer relationships. Importing it wholesale, without checking whether it fits your own economics, is how companies end up with pricing models that look right in the pitch deck and fail in the P&L.

PROVOCATION FIVE

The Monetization Maturity Ladder

Seat to usage to outcome. The three-rung ladder that every AI company must climb — and the reason most companies get stuck on the first rung longer than they should.

The monetization maturity ladder describes the evolutionary path that most AI companies follow as their products, customer relationships, and operational capabilities develop. It has three rungs. Almost every company starts on the first. Most get stuck there longer than they should. The ones that climb deliberately, with the right prerequisites at each step, compound their advantage in ways that become nearly impossible for competitors to replicate.

Understanding where you are on the ladder is important. Understanding why you are there — whether it is a deliberate strategic choice or a failure to recognize that the prerequisites for moving up are in place — is more important. And understanding what the climb costs, what it requires, and what it compounds is the most important understanding of all.

The first rung is access-based pricing. Seats. Subscriptions. Platform fees. This is the appropriate starting point for most AI products, for three good reasons. First, it is what buyers know how to purchase and what procurement committees know how to approve. Second, it requires minimal billing and measurement infrastructure. Third, it reflects genuine uncertainty about consumption patterns and outcome delivery in an early-stage AI product — you do not yet know enough to price more precisely, and pretending you do creates promises you cannot keep.

The problem is not starting here. The problem is treating it as a permanent home rather than a launchpad. The seat model captures access value. AI creates outcome value. The gap between those two numbers — the open-claw effect — is the strategic opportunity that every AI company on the first rung is currently failing to monetize.

The second rung is consumption-based pricing. Per-token, per-task, per-API call, per-workflow execution. Moving to this rung requires three things: metering infrastructure

that captures consumption accurately at the granularity needed for billing, consumption data sufficient to price reasonably (ideally twelve months of deployment history), and the willingness to have a different kind of customer conversation where the invoice varies with usage.

The triggers for making the move from rung one to rung two are specific and recognizable. You know it is time when your heaviest users are generating five to ten times the token consumption of your lightest users but paying the same price. When customers are asking for usage-based options because they want to expand flexibly without renegotiating contracts. When your gross margins are compressing because AI infrastructure costs are consumption-based but your revenue is not. When a competitor has introduced consumption-based pricing and is winning the value conversation with enterprise buyers who prefer to pay for what they use.

The migration from rung one to rung two is primarily a change management challenge, not a technical one. The technical infrastructure for consumption billing is available. The challenge is the installed base: customers who bought predictable seat pricing and are now being told their invoice will vary. The best migrations use a hybrid approach — preserve the subscription floor as a committed minimum (which customers can budget as before), introduce consumption overage pricing for usage above the floor (which gives the vendor upside from heavy users), and invest heavily in billing transparency (which makes customers comfortable with variable invoices because they can see exactly what they are being charged for and why).

The third rung is outcome-based pricing. This is the frontier — the rung where the most value is available and where most companies are not yet ready to operate. Moving to outcome-based pricing requires five things that must all be in place simultaneously: a precisely defined outcome that both vendor and customer agree describes success, a measurement methodology that both parties accept as accurate, attribution logic that determines whether the AI caused the outcome or whether it would have happened anyway, verification infrastructure that confirms outcomes programmatically rather

than through human assertion, and a contractual framework sophisticated enough to handle variable consideration under ASC 606 or IFRS 15.

Most AI companies are not yet ready to move to this rung because they have not built the measurement infrastructure. The temptation is to design outcome-based pricing first and build the measurement capability afterward. This is exactly backwards. The companies that successfully reach rung three are the ones that built their measurement infrastructure while still on rung two — they tracked outcomes without charging for them, building credibility and trust with customers, and demonstrating the link between AI performance and business results before proposing to capture a share of that value commercially.

The Ladder Visualized

Level	Unit	Strategic value	Prerequisites
Rung 3 Outcome-Based	Price per verified business result	Resolves open-claw effect. Maximum alignment. Highest revenue ceiling.	Outcome definition · verification infra · attribution methodology · ASC 606 variable consideration framework
Rung 2 Consumption-Based	Price per unit of AI consumption — token, task, workflow	Captures expansion value automatically. Aligns cost and revenue. Eliminates seat expansion friction.	Real-time metering · 12+ months consumption data · billing transparency · subscription floor hybrid
Rung 1 Access-Based	Price per seat, user, or platform subscription	Easy to sell. Familiar to buyers. Appropriate for early-stage uncertainty.	Contract management · user provisioning · basic entitlement tracking

> ***"Most companies overestimate how quickly they can reach outcome-based pricing and underestimate how***

> ***much the climb compounds their competitive advantage once they get there."***

The timing question — when to move from one rung to the next — depends on three things being true simultaneously. First, the operational prerequisites are in place: the metering infrastructure, the consumption data, the measurement capability required for the higher rung. Second, there are visible signals in the business that the current rung is suboptimal: margin compression, expansion friction, competitive pressure from higher-rung competitors. Third, the customer relationship has matured enough to support the commercial conversation that the higher rung requires.

The most common mistake is attempting the migration before the operational prerequisites are ready. A company that announces outcome-based pricing before it can reliably measure and verify the outcomes it is charging for will face billing disputes, customer trust damage, and revenue recognition issues that set the entire monetization programme back by years. The ladder must be climbed in order. There are no shortcuts.

Migration Triggers — When to Move to the Next Rung		
Signal	Move from Rung 1 to 2	Move from Rung 2 to 3
Financial	Gross margin compression · heavy users subsidized by light users	Consumption pricing ceiling visible · value created dramatically exceeds price
Operational	Metering infrastructure live · 12 months consumption data available	Outcomes measurable · attribution methodology validated · verification infra operational
Commercial	Competitors offering usage-based options · customers requesting flex pricing	Competitors offering outcome pricing · customers asking for accountability-based contracts
Strategic	AI usage patterns stable and predictable	AI reliability sufficient to accept outcome performance risk

THE MONETIZATION MANIFESTO

Ten Laws of AI Economy Monetization

One page each. The quotable core. Designed to travel.

These are not guidelines. They are laws in the sense that matters most: violating them has consequences that compound over time, regardless of whether the violation was intentional. Every company in the AI economy will either internalize these laws or learn them the hard way. The goal of this manifesto is to spare you the tuition.

LAW 1 · PRICE THE OUTCOME, NOT THE ACCESS

Value is created at the outcome layer. Capture it there.

Access is the floor, not the ceiling. When you charge for a seat, you are charging for the right to use a tool. When you charge for an outcome, you are charging for the value that tool creates. These are different numbers — not marginally different, but categorically different. The first is a cost that the customer tolerates. The second is a share of a benefit that the customer welcomes, because both parties understand the logic. Every AI product that delivers measurable business results has the potential to charge for those results. The obstacles to doing so are real — outcome definition, measurement infrastructure, attribution methodology, variable consideration accounting — but they are surmountable obstacles, not fundamental barriers. The companies that are investing now in the infrastructure required for outcome pricing are buying a commercial option that will be extraordinarily valuable when they exercise it. The companies that are not investing in that infrastructure are allowing their open-claw gap to widen with every improvement in their AI's capability. Outcome pricing is not just a better commercial model. It is the commercial model that tells the truth about what AI is worth. Every other model is an approximation,

a proxy, a placeholder. The seat approximates access. The token approximates consumption. The outcome measures value. Only the last one is the thing itself.

LAW 2 · MONETIZATION IS DATA, NOT PROCESS

Fix your objects before you fix your workflows.

The organizations that scale AI monetization without crisis are the ones that defined their data model before their billing process. They know exactly what a product object is, what an entitlement record contains, what events their metering system captures, and how those events flow through to an invoice. Every billing error, every revenue recognition restatement, every leakage event can be traced to a data model that was not precise enough. Process automation on top of imprecise data does not fix the problem — it automates the error at scale, making it harder to detect and more expensive to unwind. Fix the data first. Define the thirteen objects with precision. The processes will follow naturally, and the systems will have something coherent to implement.

LAW 3 · THE TOKEN IS THE NEW SEAT

The atom has changed. Your metrics must change with it.

For a generation of SaaS businesses, the seat was the atom — the irreducible unit of commercial measurement that billing, forecasting, and investor reporting were built around. In the AI economy, the token is that unit. Not because per-token pricing is the right model for every situation — it is not — but because token consumption is the foundational measurement from which all other AI billing derives. You cannot price outcomes without understanding the token cost of achieving them. You cannot build token budget governance without understanding token flows. You cannot detect revenue leakage without metering tokens accurately. You cannot forecast AI spending without modeling token consumption curves. The token is not just a pricing unit. It is the data standard of the AI economy, the way that the seat was the organizational unit of the SaaS economy.

LAW 4 · EVERY LAYER NEEDS ITS OWN ECONOMICS

One pricing model spanning five layers is five mistakes compounded.

The compute economy, the model economy, the token economy, the agent economy, and the outcome economy each have their own cost structures, their own value propositions, and their own natural pricing units. A pricing model that fits one layer perfectly creates commercial distortions at any other layer. Infrastructure providers who try to charge per outcome are making a category error — they have no visibility into outcomes and no accountability for them. Outcome-based service companies that charge per token are leaving their most powerful commercial lever completely unused. Know your layer. Design your pricing for that layer's specific economics. Resist the competitive pressure to borrow pricing logic from adjacent layers where the fundamental dynamics do not hold.

LAW 5 · THE GOLDEN THREAD MUST NEVER BREAK

Trace concept to cash without a gap. Every gap is silent revenue loss.

The golden thread is the unbroken data lineage from the first moment an AI product is conceived to the last dollar of revenue recognized in the general ledger. Every link in that chain — product definition, offer design, contract, entitlement, provisioning, metering, invoice, payment, recognition — must be traceable. When the thread breaks — when an invoice line cannot be traced back to the metering events that generated it, or when a recognized revenue number cannot be tied to a specific contract and performance obligation — you have a leakage event in progress and an audit risk accumulating simultaneously. Traceability is not a compliance requirement that you build to satisfy an auditor. It is a commercial operating standard that you build because the alternative is flying blind with real money.

LAW 6 · YOUR PRICING MODEL IS YOUR GO-TO-MARKET

How you charge determines who buys, how they expand, and why they leave.

The pricing model is not a downstream commercial decision made after the product is built. It is a strategic signal that shapes the entire customer journey. A per-token model attracts developers and technical buyers who can manage consumption and who value the flexibility of paying only for what they use. A per-outcome model attracts business leaders who want accountability and are willing to pay a premium for a vendor who shares their risk. A subscription model attracts buyers who need budget predictability above all else. Each pricing model self-selects a customer profile, creates a specific expansion dynamic, and sends a signal about what kind of partner you intend to be. Choose it with that awareness. And recognize that changing it — once a customer base has been acquired under one model — is among the most difficult commercial transformations a company can undertake.

LAW 7 · AGENT CONSUMPTION IS YOUR BIGGEST UNMANAGED RISK

Autonomous agents can exhaust a quarterly budget in hours. Govern them.

A human user who consumes too many tokens does so at human speed — one interaction at a time, over hours or days. An agent that consumes too many tokens does so at machine speed — thousands of interactions per second, over minutes. The difference between a correctly configured agent workflow and a misconfigured one can be the difference between a normal billing period and a monthly invoice that is one hundred times larger than expected. Every organization deploying AI agents without token budgets, without hierarchical spend controls, without real-time monitoring dashboards, and without automatic throttling mechanisms is carrying a financial risk that has not yet manifested — but will. The first time it does, the customer relationship will not survive it. Govern your agents. Build the financial controls before you experience the failure they are designed to prevent.

LAW 8 · THE CFO IS YOUR BEST AI ALLY — SPEAK THEIR LANGUAGE

Finance unlocks AI deployment. Give them the tools to say yes.

Finance leaders are not obstacles to AI deployment — they are its most important enablers. The CFO who can see AI spending clearly, attribute it accurately to business units, forecast it reliably, and demonstrate its ROI to the board is the CFO who approves expanding AI budgets quarter after quarter. The CFO who cannot see it clearly is the one who caps spending, demands proof of value before releasing budget, and creates the organizational friction that slows AI adoption to a crawl. Speaking the CFO's language means providing them with the Billing Health Index they can report to the board, the token budget governance framework they need to approve spending, the Usage Adoption Score they can use as a leading indicator of retention, and the customer-level P&L they need to evaluate where AI is actually generating return. These are not operational metrics. They are CFO metrics. Build them deliberately.

LAW 9 · LEAKAGE IS SILENT — AND IT COMPOUNDS

You will never receive a bill for revenue you forgot to capture.

Revenue leakage does not announce itself. It accumulates in the gaps between what was sold and what was billed, between what was billed and what was collected, between what was collected and what was recognized. Every gap in the golden thread is a potential leakage point — and in AI billing, the golden thread has more links and more potential gap points than in any previous generation of software monetization. The typical sources of leakage in AI billing are specific and recurring. Metering events that were not logged due to a system error or API timeout — consumption happened, cost was incurred, revenue was never captured. Entitlements that were not enforced — a customer exceeded their contractual token budget because the governance mechanism was not configured correctly, but the overage was never billed. Token consumption attributed to a test environment rather than a production environment due to a misconfigured customer identifier.

Outcome delivery that was verified internally but never triggered a billing event due to a broken integration between the verification system and the billing system. Agent task completions that were billed at a base rate rather than the contracted premium rate because a pricing rule was not updated when the contract was amended. None of these leakage events is dramatic. Each one is small. But they compound. The typical AI company that conducts a rigorous revenue assurance audit for the first time discovers leakage in the range of three to eight percent of revenue. At ten million dollars of ARR, that is three hundred thousand to eight hundred thousand dollars per year leaving through gaps that nobody was watching. At one hundred million dollars of ARR, it is three to eight million dollars. At a billion dollars, it is thirty to eighty million. Leakage compounds with revenue. And it compounds in silence, because it never appears as a line item on any report. Build the traceability infrastructure. Quantify the leakage. Then fix it — not by patching individual gaps as they are discovered, but by fixing the data model that allows gaps to exist.

LAW 10 · THE MACHINE WILL EVENTUALLY PRICE ITSELF

Agent-to-agent commerce is not science fiction. Prepare now.

Agent-to-agent commerce is the logical endpoint of a trajectory that is already visible. AI agents that browse the web, call APIs, process data, and execute tasks are already transacting with external services on behalf of their operators. As agents become more capable, as the volume of machine-to-machine interactions grows, and as the economic value flowing through agent interactions compounds, the need for a formal monetization protocol for agent commerce becomes urgent. The agents will negotiate SLAs, assess creditworthiness, and settle accounts — but only if the infrastructure exists to support them. That infrastructure — machine-readable pricing APIs, AI agent identity standards, programmatic credit assessment, automated settlement rails — is being built now. The companies that contribute to defining it will shape the rules of the AI economy for a generation. Do not wait for the standard to be handed to you. Participate in creating it.

CODA

Close the Claw

What twenty-five years teaches you about the companies that win.

Twenty-five years of watching this industry teaches you that the companies that endure are not always the ones with the most sophisticated technology or the most aggressive sales machines. They are the ones that understand, with unusual clarity, what they are selling and what it is worth to the people buying it — and that have built commercial architectures capable of capturing that value sustainably, at scale, across customer relationships that last not quarters but years.

The AI economy is creating an opportunity for a generation of companies to build exactly those architectures, on a foundation of data precision and commercial intelligence that was not available to previous generations of software companies. The tools exist. The frameworks are being developed, this series among them. The capability to price AI correctly, to bill it accurately, to recognize its revenue in compliance with accounting standards, to forecast its consumption with analytical rigor, and to govern its spending with financial discipline — all of this is achievable. None of it is easy.

What stands between where most AI companies are today and where they need to be is not primarily a technology gap. It is a clarity gap. A gap in the precision with which executives think about what they are selling and what it is worth. A gap in the seriousness with which commercial architecture is treated as a strategic discipline rather than an operational afterthought. A gap in the investment in measurement infrastructure that would make outcome-based pricing not just theoretically attractive but practically achievable.

This manifesto cannot close those gaps alone. That is the work of the series. But it can name them. It can make them visible. And it can ensure that anyone who reads it goes back to their business with a sharper question: not 'how do we grow?' but 'how do we price what we have actually built?'

The open claw opens silently. Every quarter you wait, the gap between what your AI can do and what you are charging for it grows wider. The question is not whether you have an open-claw problem. You do. The question is how big yours already is — and what you are going to do about it.

Close the claw.

THE OPEN-CLAW EFFECT — The widening capability-capture gap			
Quarter 1	AI capability: strong baseline	Revenue capture: 100% of capability priced	**Gap: minimal — pricing reflects current capability**
Quarter 2	AI capability: 40% improvement	Revenue capture: 5% increase (contract renewal)	**Gap: opening — customers receive 40% more value, pay 5% more**
Quarter 4	AI capability: 120% of launch	Revenue capture: 10-15% above launch price	**Gap: wide — customers receive 2x value, pay 10-15% more**
Year 2	AI capability: 300%+ of launch	Revenue capture: flat or modestly above launch	**Gap: claw fully open — capability is vastly underpriced**

The table above shows how the claw opens across a typical two-year deployment cycle. The numbers are illustrative — the exact trajectory differs by company, product, and market. The pattern does not. AI capability grows faster than revenue capture in every company that has not deliberately built the commercial architecture to close the gap. And the gap, once open, does not close by itself.

The Four Things That Close the Claw

- Elastic contracts: consumption ratchets, expansion clauses, and usage-based upsells that allow revenue to grow automatically as AI delivers more value — without requiring a renegotiation.

- Outcome anchoring: pricing structures that tie revenue to measurable business results, so that as AI capability improves and outcome delivery rates increase, revenue captures a share of the increasing value.
- Measurement infrastructure: the systems, data, and processes required to measure AI value before you can price it. You cannot charge for value you cannot quantify. Build the instruments first.
- Moat-building: data advantages, workflow integration, and trust infrastructure that create switching costs, defend pricing power, and prevent the gap from reopening under competitive pressure.

> ***"The open claw opens silently. Every quarter you wait, the gap between what your AI can do and what you are charging for it grows wider. Close the claw."***

The AI Economy Monetization Series continues in Book One:

The AI Revenue Imperative

APPENDIX

The Five Core Frameworks

Quick reference for practitioners and leaders.

Framework F1 — The Golden Thread

The unbroken data lineage from product concept to recognized revenue. Traced across: Idea → Product object → Price object → Offer → Quote → Contract → Entitlement → Provisioning → Meter → Event → Invoice → Payment → Revenue recognition → Allocation. Every link must be traceable. Breaks represent leakage and audit risk.

Framework F2 — The Monetization Objects Model

Thirteen canonical data objects that must be precisely defined before any billing process is designed or any billing software is selected: Product, Price, Entitlement, Meter, Event, Invoice, Contract, Credit, Allocation, Token Budget, Agent Task, Outcome, Asset. These are business concepts, not software concepts.

Framework F3 — The Five-Layer AI Economy

Compute → Model → Token → Agent → Outcome. Five distinct economic layers, each with its own cost structure, value proposition, customer set, and natural pricing unit. Strategic positioning means choosing your primary layer and designing your commercial architecture specifically for that layer's economics. Category-error pricing — applying the wrong layer's logic — creates margin, trust, and expansion problems that compound.

Framework F7 — The Monetization Maturity Ladder

Three rungs: access-based pricing (seats/subscriptions), consumption-based pricing (per-token/per-task), outcome-based pricing (per verified result). Each rung requires specific operational prerequisites before a company can sustainably operate there. The correct migration sequence: build measurement infrastructure while on rung two, prove outcomes before proposing to charge for them, introduce outcome pricing only when the verification infrastructure is operationally ready.

Framework F8 — The Model Selection Matrix

Maps company archetype (infrastructure provider, model company, application company, enterprise buyer, marketplace operator) against primary stack layer to

identify viable pricing models and eliminate category errors. The worst pricing decision is copying a competitor's model without verifying that their archetype and layer match yours.

Framework F22 — The Open-Claw Effect

The continuously widening gap between what AI can do and what companies are able to charge for it. Opened by four forces: pricing lag (prices set at launch do not update with capability), measurement gap (value cannot be charged for if it cannot be measured), contract lock (fixed contracts prevent expansion capture), and race to zero (competitive pressure drives price below value). Closed by four strategies: elastic contracts, outcome anchoring, measurement infrastructure, and moat-building.

Book Zero · The Monetization Manifesto for the AI Economy
Series Plan Version 4.0 · 10-Book Edition