From SaaS to Agent-as-a-Service: The Economics of AI-Native Business Models

Joel Gonzalez Gomez
Independent working paper

September 2026

Abstract

This paper develops a microeconomic framework for understanding the transition from traditional Software-as-a-Service (SaaS) to Agent-as-a-Service (AaaS), a business model in which AI-native products sell completed work rather than mere access to software. Traditional SaaS is modeled as a labor-complementing technology: it raises the productivity of human workers but does not itself complete the task. Agentic AI is modeled as a task-executing input that can substitute for, complement, or reorganize human labor depending on its relative productivity, reliability, and cost. I first characterize the strategic distinction between seat-based SaaS pricing and task- or outcome-based agentic pricing. I then formalize the firm problem using a neoclassical production framework with labor, capital, and AI-agent services. In the baseline model, the firm minimizes the cost of producing effective task units using human labor and AI-agent services aggregated through a constant elasticity of substitution technology. The key comparative-static result is that the desired agent-to-labor ratio increases with the wage rate, the productivity of agents, and the elasticity of substitution, and decreases with the full cost of agentic execution. I then decompose the cost of agentic execution into inference cost, human-review intensity, integration cost, and expected error loss. A numerical simulation section illustrates the theory under transparent parameter assumptions, comparing a baseline workflow, a SaaS-enabled workflow, and an agentic workflow. The simulation shows how agent adoption rises as agentic costs fall and how gross margins can expand when human-review intensity declines. The paper concludes that the shift from SaaS to Agent-as-a-Service is not only a change in technology; it is a change in the unit of value creation, from access to execution.

Keywords: AI agents, SaaS, Agent-as-a-Service, software economics, artificial intelligence, automation, production theory, pricing models, gross margins.
JEL-style classification: D24, D22, L86, O33, M15.

Contents

1 Introduction

Software-as-a-Service changed the economics of software by turning packaged licenses into recurring subscriptions. Instead of buying a static product, customers paid for access to continuously improved software. The unit of monetization became the seat: a worker, user, or team member with access to a workflow tool. This model supported the modern SaaS playbook: high gross margins, low marginal distribution costs, recurring revenue, net revenue retention, and enterprise expansion through additional seats and modules.

AI-native products challenge this model. The emerging promise of agentic AI is not merely to provide better tools for workers, but to complete tasks that workers previously performed. An AI sales agent does not simply give a salesperson a dashboard; it may qualify leads, write outreach, update the CRM, and book meetings. An AI support agent does not simply provide a knowledge base; it may resolve tickets. An AI coding agent does not simply offer an IDE plugin; it may draft code, open pull requests, and debug failures. In this setting, the unit of value is no longer access to software. The unit of value is executed work.

This paper studies the economics of that shift. I use the term *Agent-as-a-Service* (AaaS) to describe AI-native business models in which customers pay for task execution, workflow completion, or measurable outcomes delivered by autonomous or semi-autonomous agents. This does not mean that all AI software will abandon seat-based pricing, nor that all agentic products will replace human labor. Rather, the argument is more precise: as AI systems move from assistive interfaces toward task-executing systems, the economics of software increasingly resemble the economics of production inputs. The firm must compare the cost and productivity of human labor, traditional software, and AI-agent services.

The paper makes three contributions. First, it distinguishes between three economic roles for software: software as access, software as augmentation, and software as execution. Traditional SaaS mostly monetizes access and augments workers. Agent-as-a-Service monetizes execution and can substitute for part of the task bundle. Second, it develops a neoclassical production model in which human labor and AI-agent services produce effective task units. The model yields a simple comparative-static result: the optimal agent-to-labor ratio rises when agents become cheaper, more productive, or more substitutable with labor. Third, it decomposes the cost of agentic execution into inference cost, human review, integration cost, and expected error losses. This decomposition connects microeconomic theory to the practical unit economics of AI-native companies.

The central claim is simple:

SaaS monetized access to software. Agent-as-a-Service monetizes completed work.

This distinction has implications for founders, managers, and investors. If an AI product is merely a better interface, seat-based pricing may remain appropriate. If an AI product completes measurable work, task-based or outcome-based pricing becomes economically natural. However, this opportunity comes with a cost-side challenge: unlike traditional software, AI-native products reintroduce meaningful variable costs through inference, orchestration, human review, and reliability management. As a result, some AI-native companies may initially exhibit lower gross margins than traditional SaaS firms even if their long-run software economics are attractive.

The paper is organized as follows. Section 2 reviews the shift from SaaS to AI-native software and situates the paper in the literature on productivity, automation, and AI adoption. Section 3 presents a business-model taxonomy. Section 4 develops a simple SaaS model in which software augments labor. Section 5 presents the core neoclassical model of Agent-as-a-Service. Section 6 adds human review, reliability, and expected error losses. Section 7 presents a numerical simulation that illustrates the model. Section 8 analyzes pricing and gross-margin dynamics. Section 9 discusses strategic implications for firms and investors. Section 10 concludes.

2 Background and Related Literature

2.1 Software economics and the SaaS model

The traditional software business is characterized by high fixed costs and low marginal distribution costs. Once built, software can be replicated at nearly zero cost. SaaS preserved this basic feature while changing the revenue model from perpetual licenses to recurring subscriptions. The customer paid for continued access, hosting, support, security, and ongoing improvement. In business terms, this created a focus on annual recurring revenue, churn, retention, expansion, and customer acquisition cost.

Economically, traditional SaaS can be interpreted as a productivity technology. A CRM, ERP, design tool, or analytics platform increases what workers can do, but the worker remains the actor responsible for task completion. The software is a complement to labor. It improves the marginal product of labor, reduces coordination costs, or increases the quality of decisions, but it does not generally replace the task bundle itself.

This is why seat-based pricing was natural. If each worker needs access to the tool, the seat is a convenient proxy for value. A customer with more employees or more users typically receives more value and pays more. The SaaS firm can charge per user while enjoying high gross margins because the incremental cost of serving one additional user is usually small relative to subscription revenue.

AI changes this logic because the cost of serving the next unit of work may not be near zero. Inference, context retrieval, tool calls, monitoring, and human review generate variable costs. More importantly, AI may change the unit of value from a person using software to a task being completed.

2.2 AI, productivity, and automation

A growing empirical literature studies generative AI at work. Brynjolfsson, Li, and Raymond (2023) find that access to a generative AI assistant increased productivity in customer support, with larger gains for less experienced workers. This evidence is important because it supports the idea that AI can raise worker productivity through knowledge transfer, assistance, and faster task completion. However, such systems are often assistive rather than fully autonomous. They improve labor productivity, but human workers remain central.

The broader economics literature distinguishes between technologies that complement labor and technologies that substitute for labor. Autor (2015) emphasizes that automation does not simply eliminate labor; it changes the task content of jobs and the allocation of labor across activities. Acemoglu and Restrepo (2018) discuss AI as a task-replacing technology whose distributional and productivity effects depend on whether it creates new tasks or merely substitutes for existing ones. In the context of AI agents, the relevant question is not whether AI is "good" or "bad" for labor in general, but whether a particular workflow is decomposable into tasks that agents can perform reliably and cheaply.

Recent market reports point to the same transition. The Stanford AI Index documents continued growth in generative AI investment and adoption, along with falling inference costs and rapidly changing AI capability frontiers (Stanford HAI, 2025). Venture and technology firms increasingly describe AI pricing as shifting from seats to usage, outcomes, or hybrid structures (Bessemer Venture Partners, 2025; a16z, 2024; Bessemer Venture Partners, 2026). Consulting reports similarly emphasize that enterprise value depends less on isolated model access and more on redesigning workflows around agents, data, governance, and operating models (McKinsey, 2026).

2.3 The missing theory of AI-native business models

Despite growing practitioner interest, the economics of AI-native business models remains undertheorized. Many discussions focus on valuation, market size, or product examples. Fewer formalize the production logic of an AI-native company. This paper fills that gap by modeling the agent as a production input.

The model is intentionally simple. It does not assume artificial general intelligence or fully autonomous firms. It asks a narrower question: when a firm can produce effective task units using either human labor or AI-agent services, what determines the optimal input mix? This framing allows us to connect business-model language - seats, tasks, outcomes, gross margin - to standard microeconomic concepts: production functions, input substitution, cost minimization, marginal products, and first-order conditions.

3 From SaaS to Agent-as-a-Service

3.1 Three economic roles of software

To clarify the shift, it is useful to distinguish between three roles of software.

(i) **Software as access.** The product provides access to a digital capability: a database, dashboard, design environment, collaboration tool, or workflow system. The user remains responsible for task execution.

(ii) **Software as augmentation.** The product increases the productivity of the user. It reduces search costs, improves communication, automates steps, or provides recommendations. The user still completes the task.

(iii) **Software as execution.** The product performs part of the task itself. It drafts, resolves, files, reconciles, schedules, codes, triages, or decides within a bounded workflow.

SaaS businesses historically operated mainly in the first two categories. AI-native agent businesses move toward the third. This matters because pricing, cost, and defensibility differ across categories. Access is naturally priced by seat. Augmentation can be priced by seat, usage, or module. Execution is naturally priced by task, workflow, or outcome.

3.2 Defining Agent-as-a-Service

Definition 1 (Agent-as-a-Service)**.** *Agent-as-a-Service is a business model in which a software provider sells AI-mediated task execution, workflow completion, or measurable outcomes through autonomous or semi-autonomous agents, rather than selling only access to a software interface.*

This definition has four components. First, the unit of delivery is a task or workflow. Second, the product performs work, not only recommendations. Third, the agent may require human oversight, especially in early deployments or high-stakes domains. Fourth, the pricing model can be usage-based, task-based, outcome-based, or hybrid.

Table 1 summarizes the distinction.

Table 1: Economic taxonomy: SaaS versus Agent-as-a-Service

Dimension	Traditional SaaS	Agent-as-a-Service
Unit of value	Access to software	Completed task or workflow
Typical pricing	Per seat, per month	Per task, per usage unit, per outcome, hybrid
Economic role	Labor complement	Labor substitute and complement
Cost structure	High fixed cost, low marginal serving cost	High fixed cost plus variable inference, review, and orchestration costs
Margin risk	Customer acquisition and support	Compute intensity, human review, reliability, error costs
Defensibility	Workflow lock-in, integrations, data, brand	Workflow ownership, proprietary data, reliability, orchestration, trust

4 A Simple Model of SaaS as Labor Augmentation

Before modeling agents, consider a traditional SaaS tool. A firm produces output Y using capital K, labor L, and SaaS intensity S. Let SaaS augment labor productivity through a function $\theta(S)$:

$$Y_S = K^{\alpha} \left[\theta(S)L\right]^{1-\alpha}, \qquad 0 < \alpha < 1, \quad \theta'(S) > 0, \quad \theta''(S) < 0. \tag{1}$$

The firm maximizes profits:

$$\max_{K,L,S} \Pi_S = pK^{\alpha} \left[\theta(S)L\right]^{1-\alpha} - rK - wL - P_S S, \tag{2}$$

where p is output price, r is the rental cost of capital, w is the wage rate, and P_S is the price of SaaS intensity. The first-order condition for S is:

$$p(1-\alpha)K^{\alpha}L^{1-\alpha}\theta(S)^{-\alpha}\theta'(S) = P_S. \tag{3}$$

Equation (3) states that the firm buys SaaS intensity until the marginal revenue product of software-enabled labor productivity equals the price of software. The worker remains the task executor. SaaS raises the worker's productivity, but Y_S still depends multiplicatively on L. If $L = 0$, output is zero.

This framework explains why per-seat pricing fits SaaS. If software productivity is embodied through workers, and if seats are proportional to workers, then charging per user is a practical way to capture value. The SaaS vendor monetizes the number of people equipped with the tool.

5 A Neoclassical Model of Agentic Production

5.1 Effective task units

Agent-as-a-Service differs because agents can produce effective task units directly. Let a firm produce final output using capital K and an effective task aggregate E:

$$Y_A = K^{\alpha}E^{1-\alpha}, \qquad 0 < \alpha < 1. \tag{4}$$

The effective task aggregate is produced using human labor L and AI-agent services A. A flexible specification is the constant elasticity of substitution (CES) aggregator:

$$E(L, A) = \left[L^{\rho} + (\phi A)^{\rho}\right]^{1/\rho}, \qquad \rho = \frac{\sigma - 1}{\sigma}, \quad \sigma > 0. \tag{5}$$

Here $\phi > 0$ is the productivity-adjusted effectiveness of agentic AI relative to labor, and σ

is the elasticity of substitution between labor and agents. If σ is high, labor and agents are close substitutes. If σ is low, they are complements. This matters because not all workflows are equally automatable. A customer-support ticket, invoice reconciliation, or sales follow-up may be more substitutable than a complex negotiation or high-stakes legal decision.

5.2 The firm's problem

The firm chooses K, L, and A to maximize:

$$\max_{K,L,A} \Pi_A = pK^{\alpha}E(L,A)^{1-\alpha} - rK - wL - c_A A, \tag{6}$$

where c_A is the full per-unit cost of agentic execution. This cost is not merely the price of model inference. It includes compute, orchestration, retrieval, monitoring, human review, integration, and expected error losses.

A useful way to solve the input-choice problem is in two stages. First, for a given effective task requirement $\bar{E}$, minimize the cost of producing $\bar{E}$:

$$\min_{L,A} wL + c_A A \quad \text{s.t.} \quad [L^{\rho} + (\phi A)^{\rho}]^{1/\rho} \geq \bar{E}. \tag{7}$$

The Lagrangian is:

$$\mathcal{L} = wL + c_A A + \lambda \left\{ \bar{E} - [L^{\rho} + (\phi A)^{\rho}]^{1/\rho} \right\}. \tag{8}$$

The first-order conditions are:

$$w = \lambda E^{1-\rho} L^{\rho-1}, \tag{9}$$

$$c_A = \lambda E^{1-\rho} \phi^{\rho} A^{\rho-1}. \tag{10}$$

Dividing equation (10) by equation (9) gives the optimal agent-labor mix:

$$\frac{A}{L} = \phi^{\sigma-1} \left(\frac{w}{c_A} \right)^{\sigma}. \tag{11}$$

Equation (11) is the central result of the paper. It says that the optimal agent-to-labor ratio increases with the wage rate w, decreases with the full agentic cost c_A, increases with relative agent productivity ϕ, and is amplified by the elasticity of substitution σ.

Proposition 1 (Agent adoption intensity)**.** *In the CES task-production model, the cost-minimizing agent-to-labor ratio is increasing in labor cost w, increasing in agent productivity ϕ, decreasing in agent cost c_A, and more sensitive to relative costs when the elasticity of substitution σ is high.*

Proof. The result follows directly from equation (11). Taking logs:

$$\log\left(\frac{A}{L}\right) = (\sigma - 1)\log\phi + \sigma\log w - \sigma\log c_A. \tag{12}$$

Thus $\partial\log(A/L)/\partial\log w = \sigma > 0$, $\partial\log(A/L)/\partial\log c_A = -\sigma < 0$, and $\partial\log(A/L)/\partial\log\phi = \sigma - 1$, which is positive when agents and labor are substitutes with $\sigma > 1$. The response to relative cost changes is scaled by σ. □

5.3 The perfect-substitutes limit

In the limiting case where labor and agents are perfect substitutes, the effective task aggregate becomes:

$$E = L + \phi A. \tag{13}$$

The firm compares the cost of one unit of effective labor from humans to the cost of one unit of effective task output from agents. One unit of labor costs w and contributes one effective unit. One unit of agent service costs c_A and contributes ϕ effective units. Agents dominate labor when:

$$\frac{c_A}{\phi} < w, \tag{14}$$

or equivalently:

$$c_A < \phi w. \tag{15}$$

Equation (15) is the intuitive adoption condition: AI agents are economically attractive when the cost of agentic execution is below the productivity-adjusted labor cost of performing the same task.

6 Human Review, Reliability, and the Full Cost of Agentic Execution

6.1 Decomposing agentic cost

The variable c_A is the key bridge between theory and business reality. For traditional SaaS, the marginal cost of one additional user is often small relative to revenue. For AI agents, the marginal cost of one additional task can be material. Define:

$$c_A = c_I + c_O + hc_H + \pi(h)D + c_R. \tag{16}$$

where:

- c_I is inference and model-serving cost;
- c_O is orchestration cost, including retrieval, tool calls, memory, and system monitoring;
- $h \in [0, 1]$ is human-review intensity;
- c_H is the cost of human review per fully reviewed task;
- $\pi(h)$ is the probability of an economically relevant error after review;
- D is the expected damage or penalty from an error;
- c_R is residual integration and compliance cost per task.

Human review reduces errors, so assume:

$$\pi'(h) < 0, \qquad \pi''(h) > 0. \tag{17}$$

The firm chooses review intensity h to minimize the review-error component of cost:

$$\min_{h\in[0,1]} \; hc_H + \pi(h)D. \tag{18}$$

For an interior optimum, the first-order condition is:

$$c_H = -\pi'(h^*)D. \tag{19}$$

Equation (19) says that the optimal review intensity equates the marginal cost of human review with the marginal expected error reduction. If errors are costly, review should be high. If models become more reliable, $\pi(h)$ shifts downward and the optimal h^* declines. This decline is a major path from service-like margins to software-like margins.

Proposition 2 (Reliability and margin expansion)**.** *If model reliability improves such that the error probability function $\pi(h)$ shifts downward for all h, then the full cost of agentic execution c_A weakly decreases. If the improvement also lowers the marginal value of review, optimal review intensity h^* decreases.*

6.2 Scale and learning effects

Agentic cost may decline with cumulative volume. Let cumulative task volume be Q and define inference-orchestration cost as:

$$c_I(Q) + c_O(Q) = \bar{c}Q^{-\mu}, \qquad \mu \geq 0. \tag{20}$$

The parameter μ captures learning, caching, routing, model optimization, and infrastructure efficiency. If $\mu = 0$, there is no scale effect. If $\mu > 0$, cost per task falls with volume. Substituting into the full cost gives:

$$c_A(Q, h) = \bar{c}Q^{-\mu} + hc_H + \pi(h)D + c_R. \tag{21}$$

As Q rises and h falls, the adoption condition in equation (15) becomes easier to satisfy. This generates a dynamic path for AI-native companies: early deployments can be expensive and human-intensive, while later deployments may become scalable if reliability and cost improve.

7 Numerical Simulation: From SaaS to Agentic Workflows

The previous sections derive the theoretical conditions under which a firm shifts from labor-only production to SaaS-enabled labor augmentation or agentic task execution. This section adds a simple numerical simulation. The objective is not empirical estimation. Instead, the simulation is a transparent illustration of the model under plausible parameter values. Its purpose is to make the comparative statics visually interpretable.

I compare three workflow regimes. First, a baseline firm uses labor and capital but no SaaS and no agents:

$$Y_0 = K^\alpha L^{1-\alpha}. \tag{22}$$

Second, a SaaS-enabled firm pays a software cost and uses the remaining budget on labor, but SaaS raises labor productivity:

$$Y_S = K^\alpha (\theta_S L_S)^{1-\alpha}, \qquad \theta_S > 1. \tag{23}$$

Third, an agentic workflow uses labor and AI-agent services as effective task inputs:

$$Y_A = K^\alpha (L_A + \phi A)^{1-\alpha}. \tag{24}$$

The benchmark parameters are normalized for interpretation: total workflow budget is set to $B = 100$, the wage is $w = 1$, capital is fixed at $K = 100$, and $\alpha = 0.35$. In the SaaS case, the firm pays a software cost of 10 budget units and receives a labor-productivity multiplier of $\theta_S = 1.25$. In the agentic case, the firm allocates part of its workflow budget to agentic execution at cost $c_A = 0.75$ and relative productivity $\phi = 1.25$. These values are not calibrated to a specific company. They are chosen to show the direction and magnitude of the model's mechanisms under a simple normalization.

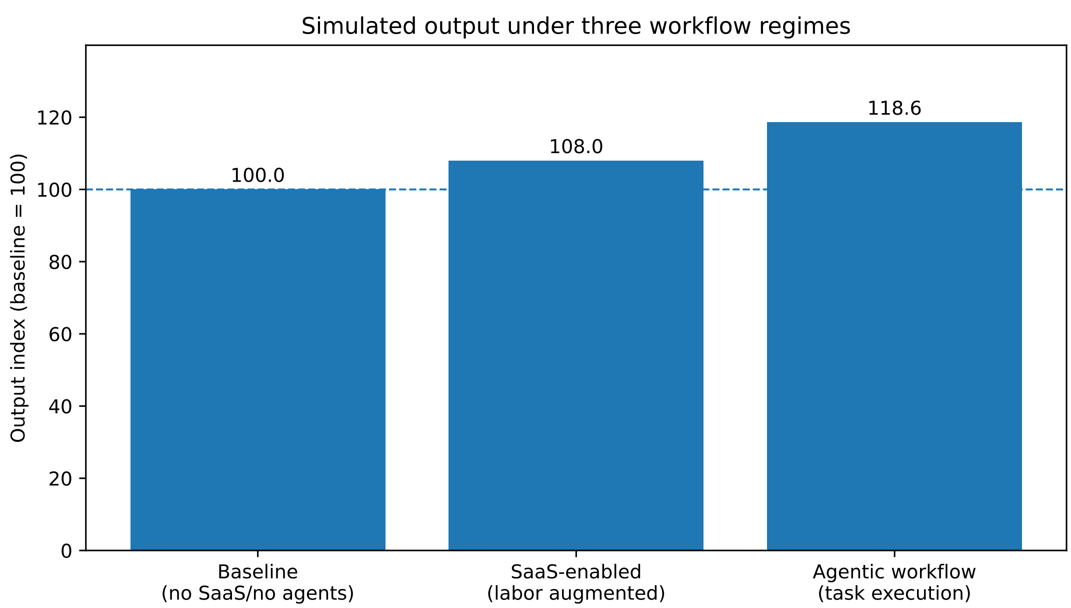

Figure 1: Simulated output under baseline, SaaS-enabled, and agentic workflow regimes.

Figure 1 shows the basic economic distinction. SaaS improves output by making labor more productive, but the human worker remains the execution unit. Agentic workflows expand effective task capacity by adding a task-executing input. Under the illustrative parameters, the SaaS-enabled workflow raises output to about 108 on a baseline index of 100, while the agentic workflow raises output to about 119. The point is not that agentic workflows always dominate SaaS. Rather, the simulation shows the condition under which agentic execution becomes economically meaningful: agents must be productive enough relative to their full cost.

The second simulation focuses on the central adoption condition. From equation (11), the desired agent-to-labor ratio is:

$$\frac{A}{L} = \phi^{\sigma-1}\left(\frac{w}{c_A}\right)^{\sigma}. \tag{25}$$

Holding $w = 1$ and $\phi = 1.25$, Figure 2 varies agentic cost c_A and the elasticity of substitution σ. When agents and labor are closer substitutes, the agent-labor ratio responds more strongly to changes in relative cost. This is why agentic adoption is likely to be highly uneven across workflows. In automatable, repeatable, and measurable workflows, a fall in agentic costs can generate a rapid shift toward AI-agent execution. In judgment-heavy or high-liability workflows, substitution is slower.

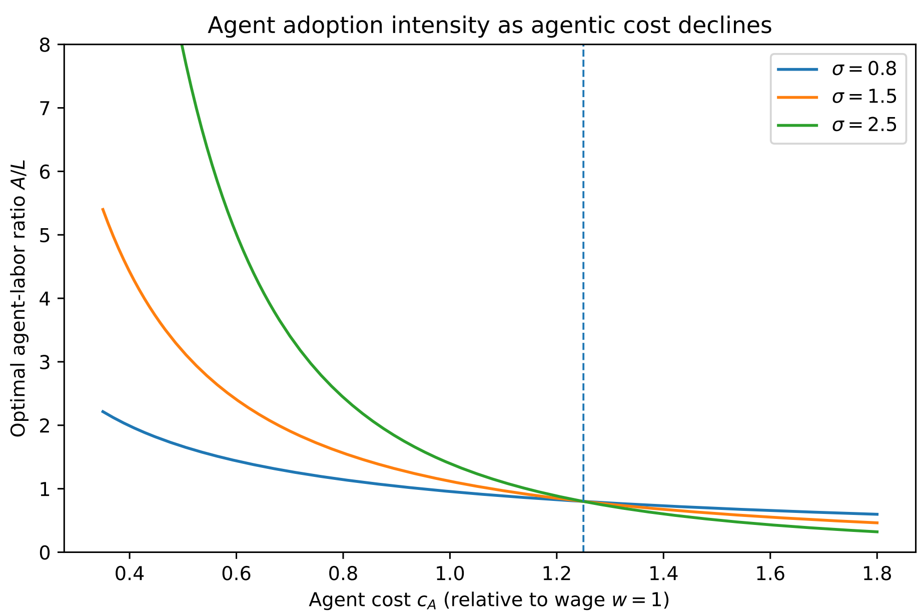

Figure 2: Simulated agent-labor ratio as the full cost of agentic execution changes.

The final simulation illustrates the gross-margin logic. Using the cost decomposition in equation (16), suppose the agent charges $P_A = 1$ per task and faces inference cost, orchestration cost, residual integration cost, fixed cost per task, and human-review cost. For simplicity, hold inference, orchestration, residual error loss, and fixed cost per task constant, and vary review intensity h:

$$GM_A(h) = 1 - \frac{c_I + c_O + hc_H + c_{error} + c_R}{P_A} - \frac{F_A}{QP_A}. \tag{26}$$

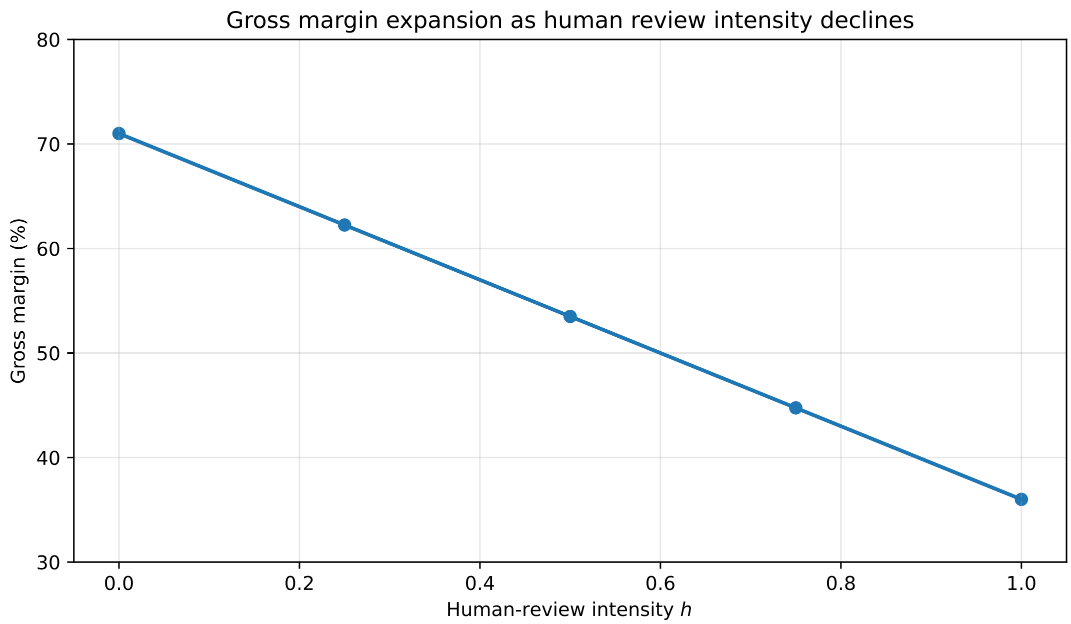

Figure 3: Simulated gross-margin expansion as human-review intensity declines.

Figure 3 captures a key investor and founder implication. Agentic companies may begin with service-like margins because they require human review, monitoring, and exception handling. As reliability improves, the review share can decline. When h falls, the full cost of agentic execution declines, making equation (15) easier to satisfy and raising gross margins through equation (31). This creates a possible transition path: early AI-native firms may look operationally intensive, but the best ones should become more software-like as reliability, routing, and workflow integration improve.

The simulations reinforce three theoretical lessons. First, SaaS and agents raise output through different mechanisms: augmentation versus execution. Second, the adoption of agents depends on quality-adjusted relative input prices, not on model capability alone. Third, gross-margin expansion depends less on the existence of AI and more on whether the firm can reduce inference cost, human review, and error losses per completed task.

8 Pricing and Gross-Margin Dynamics

8.1 SaaS revenue

Traditional SaaS revenue is usually modeled as:

$$R_S = NP_S, \tag{27}$$

where N is the number of seats and P_S is the subscription price per seat. Let marginal serving cost per seat be m_S and fixed cost be F_S. Gross margin is:

$$GM_S = \frac{NP_S - Nm_S - F_S}{NP_S} = 1 - \frac{m_S}{P_S} - \frac{F_S}{NP_S}. \tag{28}$$

When m_S is small and N is large, gross margins can be high.

8.2 Task-based agent pricing

For Agent-as-a-Service, revenue can be modeled as:

$$R_A = QP_A, \tag{29}$$

where Q is the number of completed tasks and P_A is the price per task. Gross margin is:

$$GM_A = \frac{QP_A - Qc_A(Q,h) - F_A}{QP_A} = 1 - \frac{c_A(Q,h)}{P_A} - \frac{F_A}{QP_A}. \tag{30}$$

Substituting equation (21) into equation (30):

$$GM_A = 1 - \frac{\bar{c}Q^{-\mu} + hc_H + \pi(h)D + c_R}{P_A} - \frac{F_A}{QP_A}. \tag{31}$$

Equation (31) explains why AI-native margins can initially disappoint. Even if the product is software, the cost of task execution may include compute, review, and error risk. The path to margin expansion requires one or more of the following:

(i) lower inference and orchestration cost $\bar{c}Q^{-\mu}$;

(ii) lower human-review intensity h;

(iii) lower error probability $\pi(h)$;

(iv) higher task price P_A;

(v) higher task volume Q over which fixed costs are spread.

8.3 Outcome-based pricing

Many agentic products may price not per attempted task, but per successful outcome. Let μ_A be the probability that the agent produces a chargeable outcome. Expected revenue is:

$$\mathbb{E}[R_O] = Q\mu_A P_O, \tag{32}$$

where P_O is the outcome price. The break-even condition is:

$$Q\mu_A P_O \geq Qc_A + F_A. \tag{33}$$

Solving for P_O:

$$P_O \geq \frac{c_A}{\mu_A} + \frac{F_A}{Q\mu_A}. \tag{34}$$

Outcome pricing transfers performance risk from the customer to the vendor. It is attractive when outcomes are measurable and the vendor has confidence in agent reliability. It is dangerous when success is hard to observe, heavily dependent on customer inputs, or exposed to error costs outside the vendor's control.

9 Strategic Implications

9.1 The new unit of monetization

The most important strategic implication is that AI-native companies must choose the unit of monetization carefully. A seat is appropriate when value scales with the number of users. A task is appropriate when value scales with work completed. An outcome is appropriate when value is measurable and attributable.

This creates a hierarchy:

$$\text{Access} \to \text{Usage} \to \text{Task} \to \text{Outcome}. \tag{35}$$

The movement from left to right increases alignment with customer value, but also increases operational and performance risk. A vendor that sells seats does not guarantee work. A vendor that sells outcomes must manage execution risk.

9.2 When does Agent-as-a-Service beat SaaS?

Agent-as-a-Service becomes strategically superior when three conditions hold:

$$c_A < \phi w, \tag{36}$$

$$\sigma \text{ is sufficiently high,} \tag{37}$$

$$\mu_A P_O \text{ is observable and contractible.} \tag{38}$$

Condition (36) is the economic adoption condition. Condition (37) states that the task must be substitutable: the agent must be able to perform the work without excessive human complementarity. Condition (38) is the monetization condition: the firm must be able to charge for the work in a way that customers accept and that covers cost.

9.3 Defensibility in AI-native business models

The model also clarifies why AI-native defensibility is unlikely to come from model access alone. If model capabilities become widely available and inference costs fall, then c_I becomes less differentiated. Competitive advantage shifts to other components of c_A and ϕ:

- proprietary workflow data that increases ϕ;
- integrations that reduce c_O and c_R;
- trust and governance that reduce expected error losses $\pi(h)D$;
- product design that lowers human-review intensity h;
- domain specialization that raises success probability μ_A.

This view is consistent with the idea that the durable AI company may not be the one with the most impressive demo, but the one with the best workflow ownership and cost control.

9.4 Implications for emerging markets

The framework has special relevance for emerging markets. In countries where professional services are expensive relative to small-business purchasing power, agents may expand access to capabilities that were previously unaffordable. However, the adoption condition depends on local wages. If wages are low, the threshold $c_A < \phi w$ is harder to satisfy. This implies that emerging-market AI adoption will be strongest where agents do something that local labor cannot easily do, where speed matters, where quality is scarce, or where coordination costs are high.

This point is important for Latin America. Agentic AI may not simply replace low-cost labor. Its larger opportunity may be to expand the frontier of affordable services: legal intake, accounting support, customer service, logistics coordination, event planning, sales operations, loan processing, and SME administration. In those settings, the agent is not merely a substitute for existing labor. It can create new feasible services by lowering fixed costs and reducing organizational complexity.

10 Discussion: Three Regimes of AI-Native Software

The framework suggests three regimes.

10.1 Regime 1: Copilot SaaS

In the first regime, AI is embedded into SaaS as a productivity feature. The model remains close to equation (1). The AI feature raises $\theta(S)$ but does not produce E independently. Pricing may remain seat-based, perhaps with usage caps or premium tiers. This regime is natural for writing assistants, analytics copilots, and productivity features embedded in existing workflows.

10.2 Regime 2: Hybrid agentic workflow

In the second regime, agents perform part of the workflow but require human review. The cost function in equation (16) is central. Gross margins depend on the review intensity h and expected error loss $\pi(h)D$. Many AI-native startups are likely to live in this regime during early commercialization. They sell automation, but operationally they manage a hybrid production process.

10.3 Regime 3: Outcome-executing agent

In the third regime, agents complete measurable outcomes with low review intensity and reliable performance. Pricing shifts toward tasks or outcomes. The business begins to resemble software again in margin structure, but with a different revenue base. It is no longer monetizing seats; it is monetizing completed work.

11 Limitations

This paper is theoretical and intentionally stylized. It does not estimate a structural model using firm-level data, and it does not claim that all AI-native companies will converge to the same business model. The CES framework abstracts from many real-world complications: multi-agent orchestration, customer-side adoption frictions, legal liability, data privacy, security constraints, model degradation, and strategic competition.

The model also treats agentic productivity ϕ as a parameter, when in practice it is endogenous to product design, domain data, model choice, prompt architecture, workflow integration, and human oversight. Similarly, the cost of human review may depend on labor markets, compliance requirements, customer expectations, and failure tolerance.

Finally, the paper does not assume that AI agents will universally replace labor. In many settings, the elasticity of substitution σ may be low. In those workflows, agents remain complements to workers rather than substitutes. The model's purpose is not to predict full automation, but to show how the economics of AI-native firms change as software moves from access to execution.

12 Conclusion

The shift from SaaS to Agent-as-a-Service is a shift in the unit of economic value. SaaS sells access to software that helps people work. Agent-as-a-Service sells work performed by software. This change is large enough to require a different economic framework.

The paper models traditional SaaS as a labor-augmenting technology and Agent-as-a-Service as a task-executing input. In the core CES production model, the optimal agent-to-labor ratio rises with the wage rate, the productivity of agents, and the elasticity of substitution, and falls with the full cost of agentic execution. In the perfect-substitutes case, the adoption condition is especially simple: agents are adopted when $c_A < \phi w$.

The most important practical contribution is the decomposition of c_A. The cost of an

AI agent is not only model inference. It includes orchestration, human review, integration, and expected error losses. This explains why AI-native companies may initially look less like traditional SaaS businesses and more like technology-enabled services businesses. Their margins improve only if task volume scales, inference costs fall, review intensity declines, and reliability improves.

The strategic implication is that AI-native founders and investors should not ask only whether an AI product is useful. They should ask: What is the unit of value? What is the full cost of execution? How much human review is required? Can the outcome be measured? Does the agent substitute for labor or merely augment it? Is the product priced for access, usage, tasks, or outcomes?

SaaS monetized the seat. Agent-as-a-Service monetizes the task. The companies that understand this distinction may be best positioned to build the next generation of AI-native business models.

A Mathematical Appendix

A.1 Derivation of the CES agent-labor ratio

Let:

$$E(L, A) = [L^\rho + (\phi A)^\rho]^{1/\rho}. \tag{39}$$

The partial derivatives are:

$$\frac{\partial E}{\partial L} = E^{1-\rho} L^{\rho-1}, \tag{40}$$
$$\frac{\partial E}{\partial A} = E^{1-\rho} \phi^\rho A^{\rho-1}. \tag{41}$$

Cost minimization implies:

$$\frac{w}{c_A} = \frac{\partial E/\partial L}{\partial E/\partial A} = \frac{L^{\rho-1}}{\phi^\rho A^{\rho-1}}. \tag{42}$$

Since $\rho = (\sigma - 1)/\sigma$, we have $1 - \rho = 1/\sigma$. Therefore:

$$\frac{A}{L} = \phi^{\sigma-1} \left(\frac{w}{c_A}\right)^\sigma. \tag{43}$$

A.2 Cost function for effective task units

The unit cost of producing one effective task unit under the CES aggregator can be written as:

$$C_E(w, c_A, \phi) = \left[w^{1-\sigma} + \left(\frac{c_A}{\phi}\right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}}. \tag{44}$$

This is the dual cost function associated with the effective task aggregator. It implies that the firm chooses between labor and agents based on quality-adjusted input prices. The relevant price of agentic effective labor is c_A/ϕ.

A.3 Comparative statics

Taking logs of equation (11) gives:

$$\log A - \log L = (\sigma - 1)\log\phi + \sigma \log w - \sigma \log c_A. \tag{45}$$

Thus:

$$\frac{\partial \log(A/L)}{\partial \log w} = \sigma, \tag{46}$$

$$\frac{\partial \log(A/L)}{\partial \log c_A} = -\sigma, \tag{47}$$

$$\frac{\partial \log(A/L)}{\partial \log \phi} = \sigma - 1. \tag{48}$$

B Business Model Summary Table

Table 2: Summary of SaaS and Agent-as-a-Service economics

Model	Unit of value	Main cost risk	Most natural pricing
Traditional SaaS	User access and productivity enablement	Customer acquisition, support, infrastructure	Seat subscription
AI-enhanced SaaS	User productivity plus AI feature usage	Inference cost and usage overages	Seat plus usage caps
Hybrid agentic workflow	Partial task completion with human review	Review intensity, orchestration, error costs	Usage or task-based
Outcome agent	Measurable completed outcome	Outcome attribution, liability, failure risk	Pay-per-outcome

References

Acemoglu, D., and P. Restrepo. 2018. Artificial Intelligence, Automation, and Work. *NBER Working Paper* No. 24196.

a16z. 2024. AI Is Driving a Shift Towards Outcome-Based Pricing. Andreessen Horowitz Enterprise Newsletter.

Autor, D. H. 2015. Why Are There Still So Many Jobs? The History and Future of Workplace Automation. *Journal of Economic Perspectives* 29(3): 3–30.

Bessemer Venture Partners. 2025. The State of AI 2025.

Bessemer Venture Partners. 2026. The AI Pricing and Monetization Playbook.

Brynjolfsson, E., D. Li, and L. R. Raymond. 2023. Generative AI at Work. *NBER Working Paper* No. 31161.

Eloundou, T., S. Manning, P. Mishkin, and D. Rock. 2024. GPTs are GPTs: Labor Market Impact Potential of LLMs. *Science* 384(6702): 1306–1308.

McKinsey & Company. 2025. What is an AI Agent? McKinsey Explainers.

McKinsey & Company. 2026. Building the Foundations for Agentic AI at Scale.

Stanford Institute for Human-Centered Artificial Intelligence. 2025. *Artificial Intelligence Index Report 2025*. Stanford University.

Varian, H. R. 2010. Computer Mediated Transactions. *American Economic Review* 100(2): 1–10.