Aventisia

WHITE PAPER

PRICING & COMMERCIAL MODELS

The End of Per-License Pricing

Why enterprise software's oldest business model is broken — and what replaces it in the agentic automation era

Published by Aventisia | Makers of Avintra

July 2026

Executive Summary

For more than two decades, enterprise software has been sold the same way: a license, a seat, a robot - priced and paid for on a fixed schedule, irrespective of whether it is ever switched on. That model made sense when software was scarce and deployment was slow. It makes far less sense in an era of AI agents and automation, where the actual "work" being paid for happens in short, unpredictable bursts - a few hours a day, concentrated around business hours, pilots, and peak periods - while the bill keeps running nights, weekends, and idle stretches alike.

Independent research is remarkably consistent on the scale of the resulting waste: multiple 2025–2026 industry studies put unused or underutilized enterprise software licenses at roughly 43–55% of total spend, with average annual waste in the tens of millions of dollars for large enterprises. The structural reason is simple - the vendor is paid the moment the contract is signed, whether or not the software ever runs. That is not a rounding error. It is the business model working exactly as designed.

This paper examines why the per-license model is fundamentally misaligned between vendor and customer interests, surveys the pricing models now emerging to replace it - usage-based, outcome-based, hybrid, and credit-based - and makes the case that for AI agents and automation specifically, execution-hours pricing (pay only for the hours an agent is actually doing work) is the model best suited to how this technology actually behaves. We close with how Avintra, Aventisia's AI orchestration platform, has been built around this principle from the ground up.

1. The Silent Tax of Idle Capacity

Every enterprise software license carries an implicit assumption: that the thing being licensed is running, or at least available to run, all the time the meter is ticking. In practice, almost nothing in a business operates that way. Employees work roughly a third of the hours in a week. Departments run peak workloads around month-end, quarter-end, or seasonal spikes, and idle the rest of the time. Pilots and proofs-of-concept, by design, run at a fraction of production scale for months before (or instead of) going live.

AI agents and automation bots inherit this same rhythm, only more so - they are frequently built to handle a specific process (an invoice queue, a support escalation, a compliance check) that simply does not generate work 24 hours a day. Yet the traditional per-license or per-bot model charges as if it does. The result is a cost structure completely detached from actual utilization: an enterprise can be billed the same amount whether an agent executes for two hours a day or twenty.

This is not a hypothetical concern in robotic process automation - it is how the leading platforms are explicitly metered. UiPath's licensing, for example, measures Robot Units based on license allocation time, not execution time: a robot allocated for 24 hours consumes units for the full period regardless of how much of that time it actually spends running a process. The meter tracks whether the capacity was reserved, not whether it was used.

2. Just How Deep the Waste Goes

The scale of license waste is now well documented across independent research houses, procurement analytics firms, and SaaS management platforms. The figures vary by study and methodology, but they converge on the same order of magnitude: roughly half of enterprise software spend delivers little or no return.

Source	Metric	Figure
Zylo, 2026 SaaS Management Index	Enterprise software licenses that go unused	43%, ~$80.6M annual waste (large enterprises)
Vertice, 2025 SaaS Wastage Report	Applications that are complete shelfware / underutilized (combined)	21% shelfware + 45% underutilized = 66% of portfolio
Zylo, 2026 (separate benchmark)	SaaS licenses unused or underused; average annual waste	51–53%; ~$18–19.8M average enterprise waste/year
Ramp, 2026	Software licenses unused industry-wide; SaaS-specific waste	50% of all licenses unused; ~$45M/month wasted; 53% of SaaS underutilized
G2, License Management Statistics, 2025	Companies that deliberately over-license to avoid audit penalties	76%
Zylo / ASME, engineering software (CAD/CAE/EDA)	Licenses unused in engineering-heavy sectors	55%
Gartner, 2026 forecast	Global enterprise software spend, growth rate	$1.44 trillion, +15.1% YoY

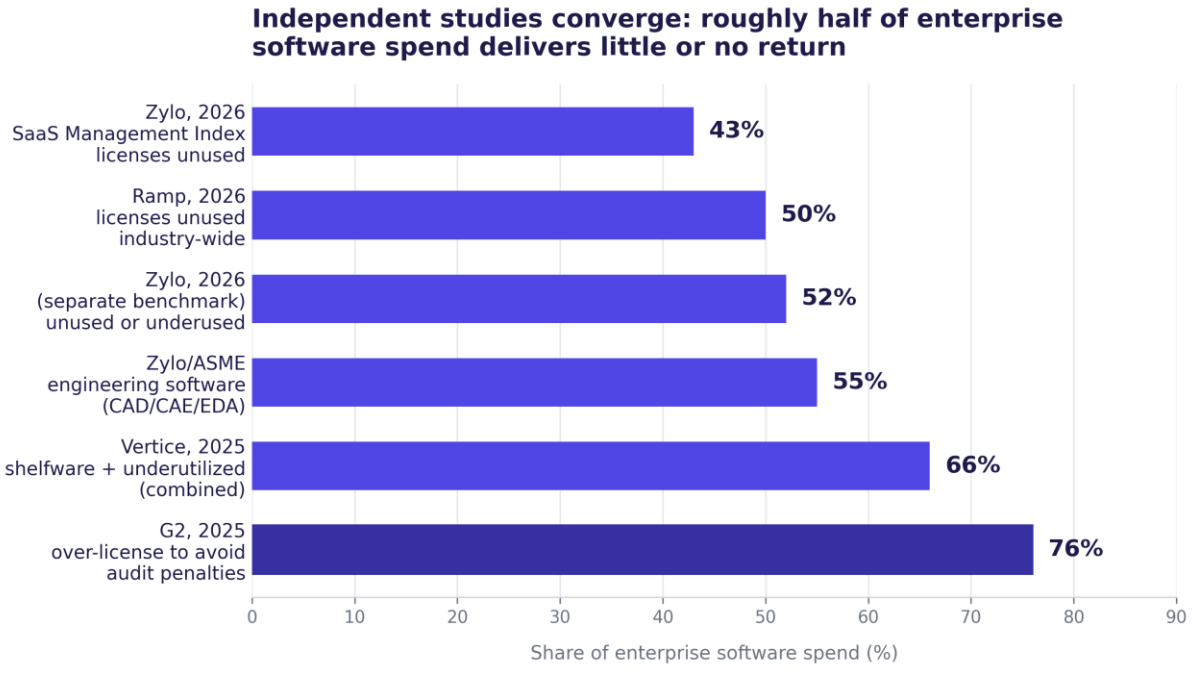

Figure 1. License-waste findings across six independent 2025–2026 industry studies (see Section 2 and Sources).

A few of these figures deserve particular attention. G2's finding that 76% of companies deliberately over-license to avoid audit penalties is perhaps the clearest illustration of how skewed the incentive structure has become: organizations are knowingly paying for capacity they do not need, not because it delivers value, but because the alternative — a compliance audit — is more expensive than the waste itself. That is a pricing model succeeding at extracting revenue while actively working against the customer's interest.

Feature bloat compounds the problem. Adoption research going back to Pendo's Feature Adoption studies, cited widely in more recent enterprise software cost analyses, has found that the large majority of features in a typical enterprise platform are never used by the customers paying for them — yet those unused capabilities are baked into the license price all the same.

3. Why the Per-License Model Is Structurally Skewed

The per-license model is not merely imperfect — it is designed in a way that systematically favors the seller over the buyer, for three reasons that compound each other:

- **Revenue is decoupled from value.** The vendor is paid in full the day the contract is signed. Whether the customer's teams adopt the tool fully, partially, or not at all has no bearing on what the vendor collects that year. There is little built-in commercial incentive for the vendor to help the customer use less of what they've bought, even when using less would be the more efficient outcome for the customer.
- **Allocation is billed as if it were consumption.** As the RPA example above shows, “licensing” in per-seat and per-bot models typically measures reserved capacity, not actual work performed. A bot sitting idle overnight, over a weekend, or during a slow quarter is billed identically to one running flat out.
- **Fear of penalties drives over-buying.** Audit and true-up clauses common in enterprise licensing agreements push customers toward provisioning well beyond their real usage - the G2 finding that three in four companies over-license specifically to avoid audit exposure is a direct consequence of this dynamic, and it is a cost that exists purely to manage vendor-side compliance risk, not to deliver business value.

Taken together, these dynamics explain why shelfware is not a temporary market inefficiency that will correct itself - it is the predictable output of a pricing model whose incentives are not aligned with the buyer's actual usage pattern.

4. The Pricing Evolution Already Underway

The market has not been blind to this. Pricing in enterprise software has moved through several distinct eras, and the transition is accelerating as AI agents make usage easier to meter and outcomes easier to define. Usage-based pricing rose from a minority practice to the majority approach among SaaS companies through the early 2020s. Gartner has projected that outcome-based components would be present in roughly a third of enterprise SaaS deals by 2025 (up from about 15% in 2022), and separately forecasts that at least 40% of enterprise SaaS spend will shift to usage-, agent-, or outcome-based pricing by 2030. More recent industry

surveys report seat-based pricing's share among SaaS companies falling from around a fifth to roughly 15% within a single year, with hybrid models rising sharply over the same period.

4.1 What the market leaders are actually doing

- **Intercom's Fin AI Agent** charges $0.99 only for a customer support conversation the AI fully resolves — a clear, attributable outcome, with no charge for a failed attempt.
- **Zendesk** has gone further still, billing customers only when a ticket is completely resolved by its AI, transferring essentially all of the performance risk onto the vendor.
- **Salesforce's Agentforce** charges roughly $2 per AI conversation, explicitly positioned against the $30–$50 typical cost of a human-handled interaction, and has introduced “Agentic Work Units” as a consumption credit to track AI-driven actions.
- **SAP's leadership** has been unusually direct about why the old model no longer fits: CEO Christian Klein has publicly argued that continuing to charge on a per-seat subscription basis makes little sense once AI is powerful enough to automate away the very seats being charged for. SAP's Joule agents are increasingly billed on consumption and outcome units - for example, the number of autonomous reconciliations performed, rather than the number of accountants logged into the system.
- **ServiceNow** reports that the software license itself is now often only around a quarter of total cost of ownership, with pricing increasingly tied to the volume of workflows actually automated.
- **Automation Anywhere** has shown more willingness than its RPA peers to strike consumption-based deals - pricing tied to bots deployed, processes automated, and labor hours replaced - in contrast to committed bot-count enterprise agreements.

Table 2 summarizes the landscape as it stands in 2026, including where each model's incentives can still break down.

Model	How it works	Representative examples	Where it breaks down
Per-seat / per-license	Fixed fee per user or per bot, regardless of actual usage or output	Legacy RPA licensing; traditional enterprise SaaS seats	Zero correlation to value delivered; penalizes idle time; encourages over-provisioning to avoid audit penalties
Pure usage / consumption	Pay per unit of work — API call, token, compute-second	LLM APIs; Automation Anywhere's consumption-leaning deals	Hard to forecast; 78% of IT leaders report unexpected charges and 90% of CIOs cite cost forecasting as their top AI deployment challenge
Outcome-based	Pay only when a defined result is achieved; zero cost for failure	Intercom Fin ($0.99/resolved conversation); Zendesk (pay only on full resolution); Salesforce Agentforce ($2/conversation)	Attribution and “what counts as resolved” disputes; and structurally, efficiency gains accrue to the vendor rather than the customer once an outcome price is fixed

Hybrid	Base platform/subscription fee plus a usage or outcome layer on top	Adobe, ServiceNow, and most AI-native platforms in 2026 (43% of SaaS companies today, projected 61% by year end)	More moving parts in contracting and billing infrastructure
Credit-based	Prepaid consumption units, redeemable across actions/agents/departments	Clay and other AI-native tools	Credit-to-value conversion can obscure true unit economics if not transparent

4.2 The case against pure outcome-based pricing

Outcome-based pricing is often presented as the purest form of "pay for value," and for well-bounded, easily attributable tasks it can work well. But it carries a structural weakness worth naming plainly: once an outcome price is fixed, any efficiency gain the vendor achieves in delivering that outcome — a faster resolution, a cheaper model, a shorter conversation — accrues entirely to the vendor's margin, not the customer's bill. The customer pays the same $0.99 or $2 whether the underlying work took three exchanges or thirty. This is a textbook principal-agent misalignment: the party doing the work benefits from its own optimization in a way the party paying for it does not automatically share in.

> *"If the AI agent is supposed to make your business more efficient, you should keep the gains; pricing should scale with transformation, not work against it." — a critique increasingly voiced by practitioners building consumption-first alternatives to outcome pricing*

Outcome pricing also introduces its own friction: what exactly counts as "resolved" has to be negotiated, documented, and often re-litigated, and disputes over attribution are common wherever a result depends on multiple systems working together (the CRM, the routing logic, the underlying data quality) rather than the AI acting in isolation.

5. The Case for Execution-Hours Pricing

For AI agents and automation specifically - as distinct from consumer-facing or single-task AI products - execution-hours pricing (billing for the hours an agent is actually running, rather than the number of agents deployed, the seats provisioned, or a negotiated definition of "success") resolves the weaknesses of every model surveyed above, without introducing new ones of its own:

- **It eliminates the idle tax outright.** Nights, weekends, low-activity periods, and pilot phases cost nothing, because nothing is executing. This directly answers the core complaint against per-license pricing: cost tracks usage, not allocation.
- **It uses a unit finance teams already understand.** "Execution hours" maps cleanly onto how enterprises already think about cloud compute and RPA bot-hours - it is far easier to forecast and audit than counting tokens, API calls, or the shifting definition of an "AI conversation."
- **It keeps efficiency gains with the customer, not the vendor.** If a well-tuned agent completes the same work in less time, the customer's bill goes down - fewer hours, lower cost - rather than the vendor

quietly keeping the margin on a fixed per-outcome fee. This is the direct fix for the principal-agent problem inherent in pure outcome-based pricing.

- **It scales with elastic AI workloads without penalizing growth.** Because the meter is hours of execution rather than number of agents or number of users, an organization can build and deploy as many agents as a process requires, and onboard as many users as needed, without every additional agent or every additional employee adding a new license line item.

No pricing model is entirely free of trade-offs, and execution-hours pricing is no exception - it does require enterprises to get comfortable with variable, usage-linked billing rather than a single flat number, and it depends on transparent, auditable metering so that "how many hours were actually run" is never in dispute. Structured well, however - with a modest fixed platform fee for governance and support, and a low, published per-hour rate for the metered portion - this trade-off is manageable in a way that pure outcome-based or pure token-based pricing often is not.

6. Avintra in Practice

Avintra, Aventisia's AI and automation orchestration platform, is built around execution-hours pricing as its central commercial principle, precisely to close the gap this paper has described between what enterprises are billed for and what they actually use.

The mechanism is straightforward: a fixed platform fee covers hosting, governance, and support — the cost of having the capability available — while the dominant cost driver is the metered hours the agents genuinely spend working. There is no per-bot license, no per-seat charge, and no negotiated definition of an "outcome" standing between the customer and the bill they actually owe. This section explains the concept behind how that metered rate is structured and why it produces a benefit that holds across every market Avintra is deployed in — without reference to any specific published price point, which varies by tier, region, and commercial agreement.

6.1 Locking the Per-Hour Rate to a Volume Tier

Execution-hours pricing is not a single flat per-hour number that applies uniformly regardless of scale. It is structured as a series of volume tiers, each with its own per-hour rate that is locked for the entire band of usage within that tier. As an organization's cumulative execution-hour volume grows and it moves into a higher tier, the rate for every hour inside that new tier steps down — it is not renegotiated hour by hour, and it is not applied retroactively in a way that penalizes early or exploratory usage.

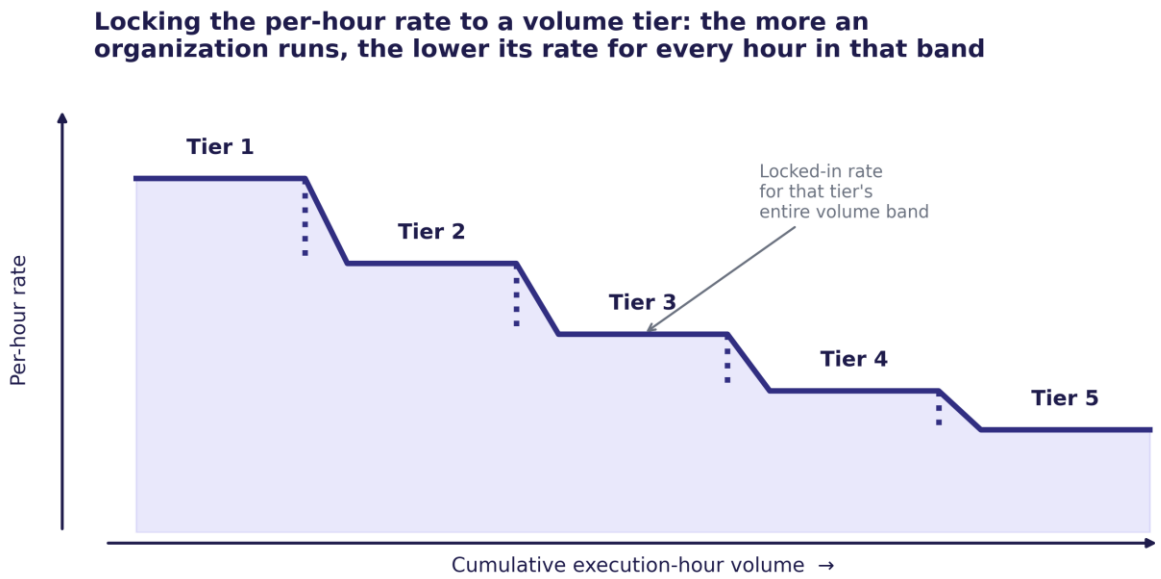

Figure 2. Conceptual illustration of tiered, locked execution-hour pricing — rates shown are illustrative, not published figures.

The effect of this structure is to align the pricing curve with how automation actually scales inside a business. A team piloting a handful of agents pays the entry-tier rate on a modest volume of hours. As adoption spreads — more processes, more departments, more agents running more of the working day — the organization moves through the tiers, and its effective blended rate falls. Growth in usage is rewarded with a lower marginal cost, rather than simply generating a larger bill at an unchanged rate, which is the opposite of how allocation-based per-license pricing behaves.

6.2 A Rate That Travels: Comparing Agent-Hours to Human-Hours, Anywhere

Because the per-hour rate at each tier is locked and published rather than derived from local wage benchmarks, it becomes a fixed, known reference point — one that can be compared directly against the fully-loaded hourly cost of a human employee performing equivalent work. Finance and operations teams already make this comparison routinely for outsourcing, contracting, and offshoring decisions; execution-hours pricing simply gives them the same comparison to make for AI agents.

That comparison holds in every country Avintra is deployed in, because the agent rate does not change with geography while the human labor cost it is being measured against does — often substantially. In a market where fully-loaded labor cost is relatively low, the gap between the locked agent rate and the human rate is real, but comparatively modest. In a market where fully-loaded labor cost is high, the same locked agent rate is being compared against a much larger number, and the resulting saving is correspondingly larger.

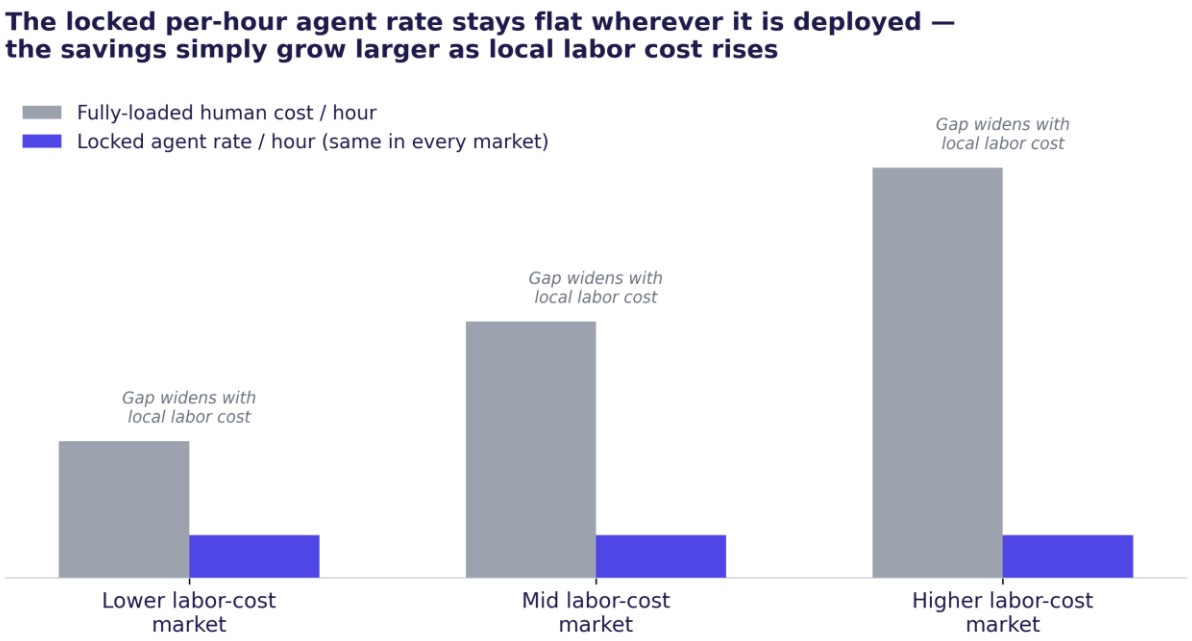

Figure 3. Conceptual illustration of how a locked, geography-independent agent rate compares against fully-loaded human labor cost across markets — relative scale shown for illustration only.

The practical implication is that the case for execution-hours pricing is not contingent on where a customer happens to operate — the model produces a genuine benefit everywhere — but the size of that benefit is not uniform either. It scales upward with the local cost of labor, which means the advantage is largest in precisely the economies where the cost of human labor, and therefore the incentive to automate, is already greatest.

6.3 Why the Benefit Compounds: An Hour of Agent Work Is Not an Hour of Human Work

The comparison above, cost per hour against cost per hour, actually understates the real benefit, because it implicitly assumes an hour of agent work and an hour of human work produce a comparable amount of output. They do not. A human hour is bounded by a single person's attention and working speed, by the need for breaks and context-switching, and by largely sequential handling of one task at a time. An agent hour is not bounded in the same way: within that hour, an agent can work through a queue of transactions concurrently, sustain the same pace across the full sixty minutes without fatigue, and typically complete a volume of comparable work that would take a human meaningfully longer than an hour to match.

This means the more relevant unit of comparison is not simply cost per hour, but cost per unit of completed work — and on that basis, the effective advantage of a locked, tiered execution-hour rate is larger still than the headline hourly comparison alone suggests. As with the geographic effect above, this gap widens further, not narrows, in higher labor-cost markets, since both the rate differential and the throughput differential are working in the customer's favor at the same time.

Taken together, these three properties — a rate that is locked and gets cheaper with scale, a rate that can be benchmarked directly against local labor cost in any market, and an hour of work that reliably outproduces its human equivalent — are what make execution-hours pricing more than a billing mechanic for Avintra. It is the

commercial expression of the principle this paper has argued for throughout: price should track real work, not reserved capacity, and the customer, not the vendor, should keep the benefit of every efficiency gain.

7. Addressing the Fair Objections

A fair reading of this paper should acknowledge where execution-hours pricing asks something of the buyer, too. Enterprises accustomed to a single flat annual number will need to build forecasting discipline around expected execution volumes, much as they already do for cloud infrastructure spend. Minimum-hour commitments — as in Avintra's own tiers — exist for good reason: they give the vendor a predictable revenue floor and the customer a straightforward way to budget, while still leaving the variable portion of the bill tied to real usage above that floor rather than to seats, bots, or negotiated outcomes.

It is also fair to note that no metering-based model removes the need for good FinOps practice. Just as usage-based AI pricing elsewhere has drawn criticism for occasional bill surprises, an execution-hours model still requires clear dashboards, usage alerts, and transparent rate cards — without which any consumption-based approach can erode the trust it depends on.

8. Conclusion

The per-license model persisted for as long as it did because software distribution was slow, usage was hard to meter, and “one flat number” was operationally simpler for both sides of the table. None of those conditions hold in the same way for AI agents and automation. Usage is trivially metered. Outcomes are inconsistently defined. And the actual pattern of use — concentrated bursts of work, not continuous operation — makes allocation-based billing an increasingly hard model to defend, as the shelfware statistics in this paper make clear.

Execution-hours pricing is not a marketing repackaging of usage-based billing — it is a deliberate choice of unit, one that avoids the idle-time waste of per-license models and the misaligned incentives of pure outcome-based pricing, while remaining simple enough for finance and procurement teams to forecast and trust. That is the principle Avintra has been built around, and it is, in our view, the model the next decade of enterprise automation will be priced on.

Sources

This paper draws on publicly available industry research current as of mid-2026, including: Zylo's 2026 SaaS Management Index; Vertice's 2025 SaaS Wastage Report; Ramp's analysis of unused software subscriptions; G2's license management statistics; Gartner's enterprise software spending forecasts; AIMultiple's RPA pricing benchmarks; CheckThat.ai and Vendor Benchmark's UiPath and Automation Anywhere pricing analyses; Flexprice, Pickaxe, Bessemer Venture Partners, Deloitte, and Futurum Research on AI and agentic pricing models; and public statements from SAP, Salesforce, ServiceNow, Intercom, and Zendesk regarding their AI pricing structures. Figures are presented as reported by these sources at the time of publication and are subject to change as the market continues to evolve.