

# Software is Dead

Death by a thousand  
cuts

May 2026

AUTHORS:



Jono Vickery

Vice President, Research



### About Activant

Activant is a research-led global investment firm that partners with high-growth companies. Since 2015, we have invested in category-defining businesses during their most critical phases of growth, partnering with founders who have won their initial battles and are ready for the next challenge.

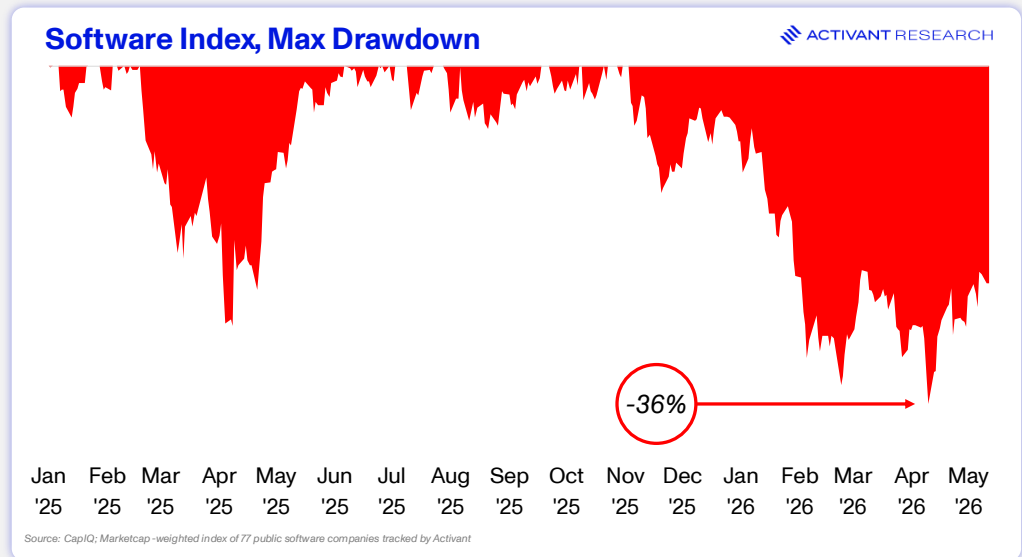
Our approach pairs deep, proprietary research with patient, flexible capital and hands-on operational partnership. We work alongside founders and leadership teams to refine strategy, strengthen operations, and accelerate sustainable growth.

Activant Research is dedicated to uncovering the most exciting emerging technologies, sectors, and companies we believe will shape the future. Our research-driven perspective informs everything we do, helping us invest at meaningful inflection points and support founders in building enduring, category-leading businesses.

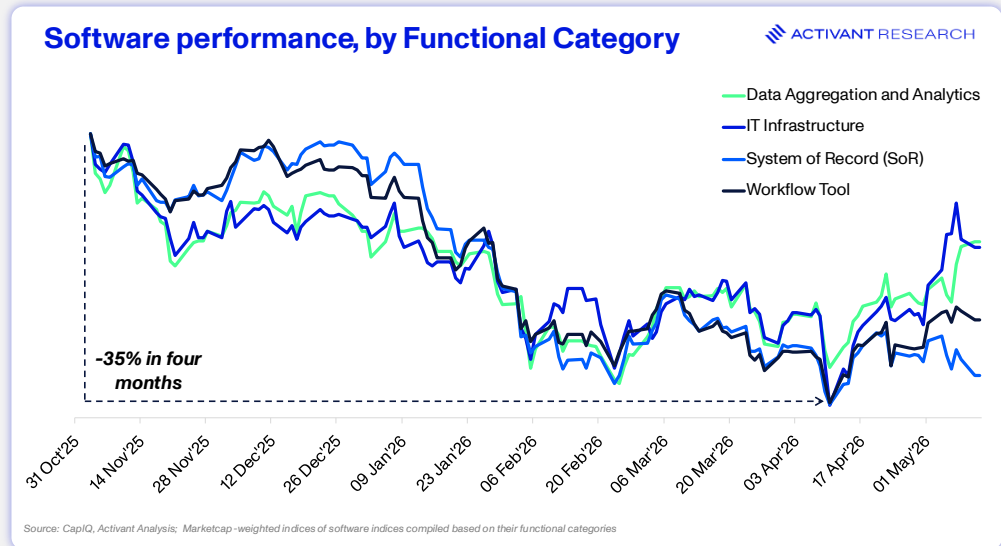
You can find out more about Activant and our research at <https://activantcapital.com/>.

# Software is Dead?!

An index of 77 public software companies tracked by Activant is down more than 20% since its last peak in November 2025. The median software company in the index now trades at 3.6x EV / NTM Revenues, 60% below the long term average.



In January 2025, we penned our [first public thoughts](#) on the shift from Systems of Record (SoR), to Systems of Intelligence (SoI) and how SoRs risked seeing the incremental growth of the industry captured by AI native firms playing the role of the SoI. **Now, the market is pricing every software category for significant AI disruption.** As we'll walk through in this article, the real risk for software is not any individual issue like vibe coding or seat contractions. Rather, it's death by a thousand cuts: software is facing four simultaneous category-defining disruptions as once, as a result of AI.



However, using stock charts to parse the news would suggest that **all** software will be uniformly disrupted by AI. That feels like “sell now and ask questions later” and not a well-thought-out response to the changes taking place. This article looks to cut through the noise and walk through **how** software really is changing.

## Death by a Thousand Cuts

SaaS is being disrupted, that’s a fact. The CEO of the world’s largest software company is already on record stating that business applications will “collapse” in the AI era, surrendering logic execution to agents.<sup>1</sup>

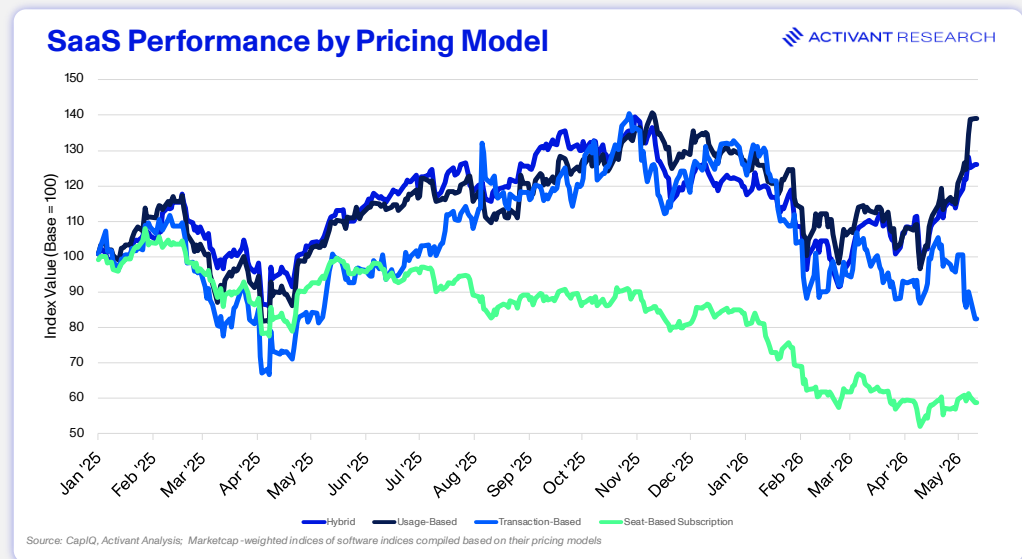
At this point, the question is where the risks are and which ones matter. The new disruption framework for software runs across misaligned pricing models, a potential shift from buy to build, intensified competition and disintermediation. While we see reason to question each pillar of the framework, the core issue for SaaS is not so much in the absolutes, but in the cacophony of risks emerging simultaneously. The real risk for SaaS that we see: **death by a thousand cuts**.

### 1. Seats Contract

For most of the past two decades, seat expansion (adding more users) provided a key part of the formula that powered software – land and expand. This was no problem for the enterprises paying more for each seat, because the value that they generated from the software was tied to the number of users clicking and typing in the platform.

The promise of having an agent perform work on your behalf completely breaks this model. When AI-driven software works, it increases value to the customer but compresses the time spent doing work. At an organization-wide level, fewer humans are needed to execute key functions like HR (Workday), ITSM (ServiceNow) or Sales (Salesforce). **Outcomes can improve while paid seats stay flat or even decline.**

This idea is not novel. We wrote about the required shift to [usage-based billing](#) in Q3 2024 and the market has been pricing this reality for over a year now.



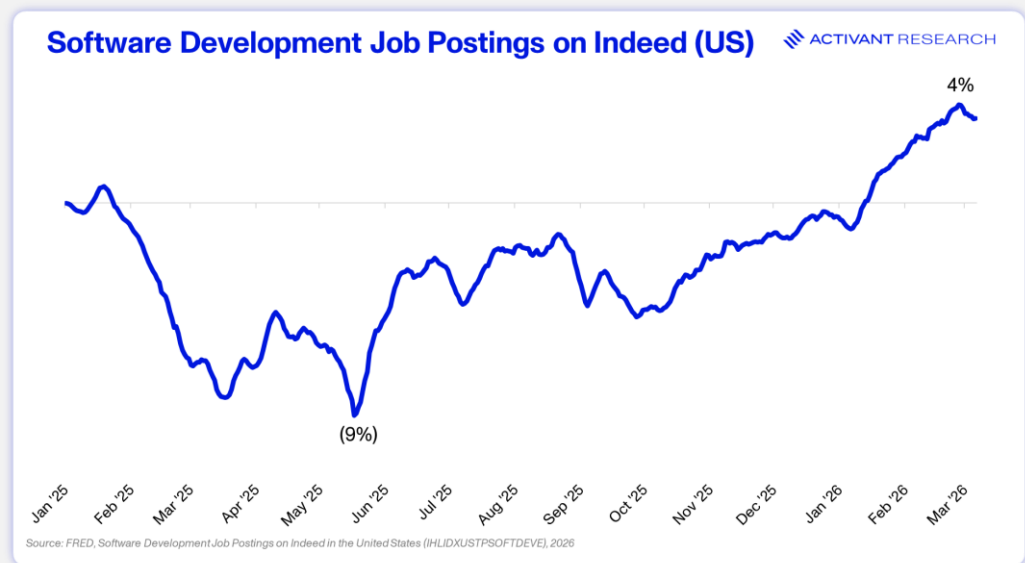
In our view, **even in the worst case, this will take much longer to play out than what the market is pricing today.** Consider that in a study in Denmark, a country with high AI adoption and low labor switching costs, researchers found a null impact on both hours worked and hiring trends two years after the release of ChatGPT.<sup>2</sup> Another study comparing jobs by their level of exposure to AI found no difference in employment outcomes for those most exposed to AI automation.<sup>3</sup>

This research suggests that we are still deeply in the phase of AI deployment where AI augments workers rather than replace them. This helps explain why some studies are finding hiring declines in early-career (age 22 – 25) cohorts despite no aggregate effects – employers want to put tools in the hands of the workers where acceleration provides the most benefit.<sup>4</sup>

Some researchers counter that we'll see a J-curve in hiring, where the initial roll-out of AI tools causes productivity disruptions, but as new processes emerge, productivity improves sharply, enabling companies to alter organizational structures.<sup>5</sup>

Yet trends in customer service and coding suggest that even as productivity improves, companies don't suddenly lay off their staff. When AI agents are deployed in customer service, firms see the service used more frequently (Jevon's paradox).<sup>6</sup> Customers have deeper conversations with humans, while agents handle mundane queries.<sup>7</sup> Software developers spend less time coding, but more time deciding on the right architecture and verifying or reviewing AI code, resulting in a zero change in total friction.<sup>8</sup>

Software development saw the strongest AI adoption of any function. 2025 was the year coding agents exploded in capability. SWE job postings still increased.

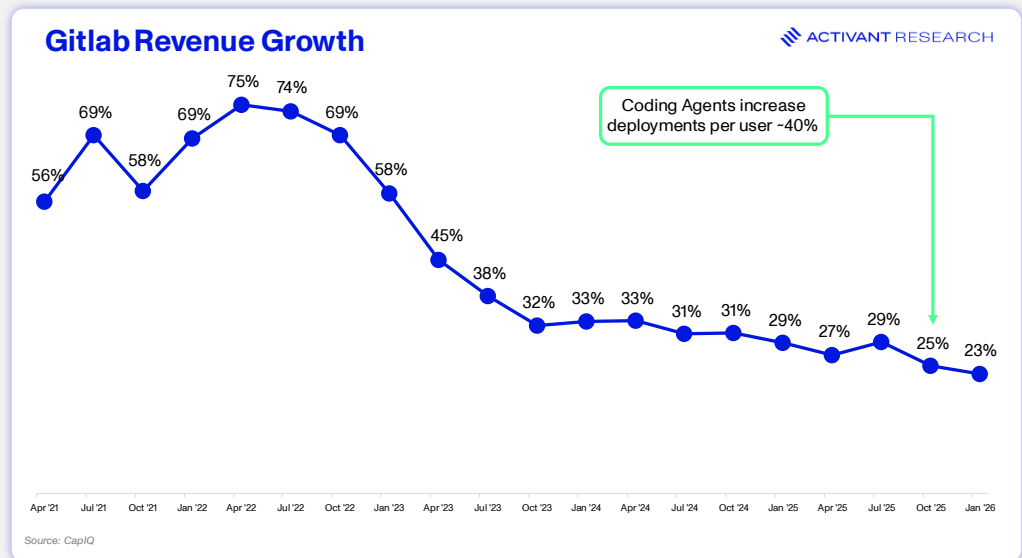


But that leaves the real question: if the number of workers holds, which companies are capturing the value of improved worker productivity?

### Seats Don't Equate to Value in the AI Era

Pricing-model disruption is moving from risk to reality. Consider [GitLab](#), whose developer tools power over 50 million registered users and more than 50% of the Fortune 500.<sup>9</sup> As coding agents drive developer productivity, the company disclosed that the number of deployments **per user** is up ~35% - 45%.<sup>10</sup>

Yet, GitLab isn't capturing the growing use of the platform. While a revolution in software development is underway, the company posted its slowest quarterly revenue growth as a public company and its weakest net revenue retention (118%).<sup>11</sup> Seats don't equate to value in the AI era.



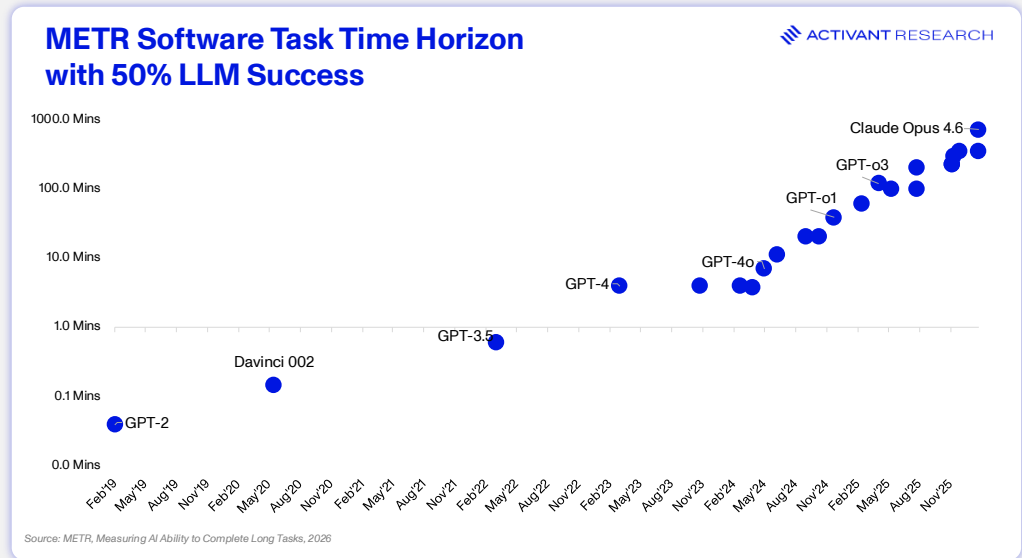
GitLab and many other software companies are still posting decent seat growth today. In fact Salesforce is still seeing seats “grow year on year” and ServiceNow posted 25% user growth in Q4 FY25.<sup>12,13</sup> While the risk of future seat contractions = still looms, **the more immediate issue we see is that the seat-based pricing model prevents traditional software companies from participating in the upside of AI adoption.**

Software companies must innovate their pricing models to stay relevant. The question for incumbents becomes: do you have a value proposition that truly suits a variable pricing model?

## 2. All Software Gets Vibe-Coded

[Andrej Karpathy](#) coined the term “vibe-coding” in early 2025. Describing the use of AI coding assistants, he characterized the practice as “[forgetting that the code even exists.](#)” Developers focus on architecture and end goals, describing specs or features and letting the LLM do the rest.

On 1 Jan 2025, the leading score on SWE-bench Verified was held by GPT-4o at 31%. By Dec 2025, that had improved to 77% (Opus 4.5).<sup>14</sup> That improvement provided the phase shift allowing Google to announce that over 25% of code across the entire company is written by AI, and led **some developers at Anthropic to declare that they no longer write code themselves.**<sup>15,16</sup> The most striking illustration of the improvement in AI coding capabilities is how long an AI system can successfully execute a task, as measured by METR. Opus 4.6 can code for ~12 hours, long enough to build a full-stack web application with authentication, database models, API routes, and a polished frontend.



The proposed problem for software companies is that if a trained developer is no longer the bottleneck to building software, then anyone can vibe-code their own version of popular software tools like [Monday.com](https://www.monday.com) (~\$3.5bn market cap<sup>17</sup>). Why pay \$50,000/yr when you could recreate something similar with a \$20/month Claude subscription and some persistence?<sup>18</sup>

This is, of course, too simplistic a view of what software actually is. Vibe coding can get you a quick prototype of the UI of a tool. But it doesn't get you security, compliance ongoing maintenance, or constant updates.

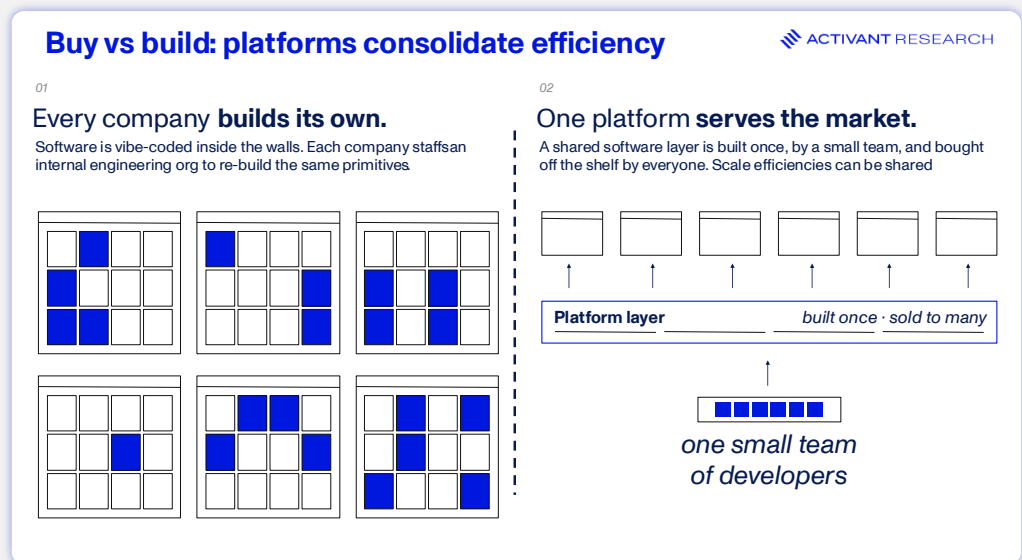
Large, listed SaaS companies manage customer data, security, and compliance frameworks like SOC2 across jurisdictions worldwide. We've recently learned that Claude Mythos exploits 72% of Firefox 147 vulnerabilities (compared to Opus 4.5 at <1%).<sup>19</sup> The future of security is extremely high stakes, and enterprises will not tolerate customer data being hosted on apps thrown together as a weekend project.

Beyond security, software platforms need deep integrations across numerous systems with constant maintenance for any changes, bugs, or upgrades. Over time, maintenance becomes 60%+ of total software lifecycle cost.<sup>20</sup> For companies looking to save costs by coding internal apps, this is their biggest hidden risk – you scope out the cost of “build” versus the SaaS cost, but don't consider the business impact when your staff are distracted fixing bugs in your custom CRM.

In other words, a sales team with Claude Code can replicate a CRM, but can they ensure that customer data is secure and compliant, and can they maintain their sales quotas when the system needs maintenance or upgrades?

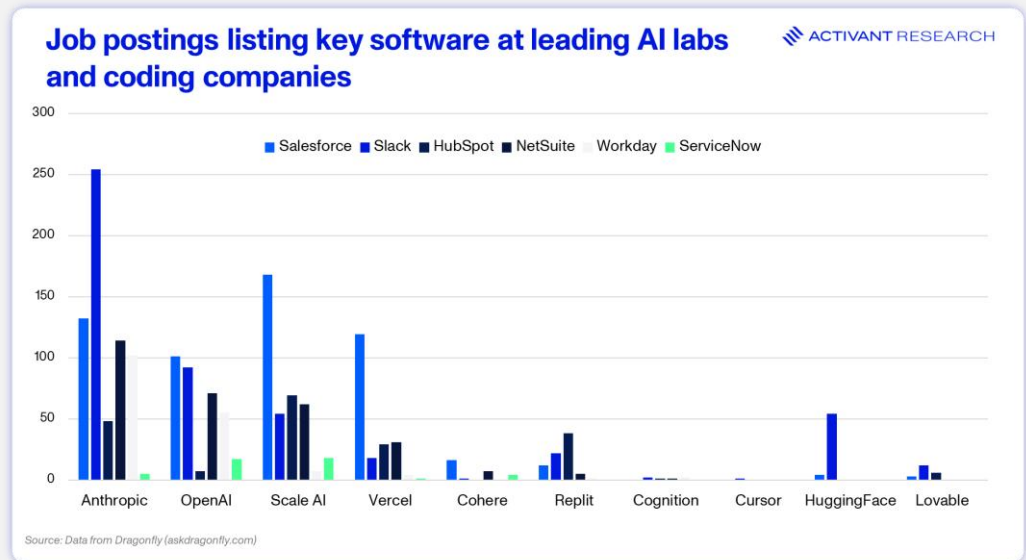
Therein lies the fallacy of vibe-coding as a substitute: software is already cheap. Yes, large enterprises can see their Salesforce bills reach \$10 million or more, but the cost per seat is \$350/month.<sup>21</sup> When comparing that to the salary of a sales rep or the cost of a lost deal, the idea of having valuable salespeople hindered by poorly performing vibe-coded software feels misguided. [Focus on what makes the beer taste better.](#)

Finally, if coding makes software development 10x cheaper, a shared platform can pass those savings onto its entire customer base. Hundreds of companies building internally can't see that benefit.



At least at enterprise level, we see the risk of companies vibe-coding their own software as unlikely. There are businesses out there that amount to little more than some CRUD operations with a nice UI, and those will undoubtedly struggle as AI coding tools improve. But most public and large late-stage software companies are far more than that.

If you need a stronger datapoint, consider that even the most sophisticated AI Labs and AI Coding tools are actively hiring users of standard software tools like Salesforce, Slack and HubSpot.



The above data was provided by [Dragonfly](#), an AI-powered discovery and decision intelligence platform that draws on a database of over 250,000 tools to help businesses identify and implement tailored, unbiased software recommendations instantly.

However, one of the more cogent concerns from AI-driven software development is that as the cost of producing software approaches zero, competitive intensity in every software category will rise, eroding margins in the process.

### 3. Barriers to Competition Fall Away

Cursor made waves when they hit \$100 million in ARR with just 20 employees.<sup>22</sup> We met a six-person team last quarter that is replacing billion-dollar incumbents in the enterprise.

Vibe coding may not be a viable alternative to buying out-of-the-box software, but it certainly is an accelerant to the production of said software. When a previously scarce element becomes widely available, it tends to trigger a wave of business formation as start-ups look to capitalize – from cloud start-ups building on AWS to mobile apps launching on Apple’s App Store.

The question is whether this is automatically bad for existing businesses. As we’ve seen repeatedly, the economy is not necessarily zero-sum. Consider two key precedents: Oracle and Salesforce.

Oracle did just under \$70 billion dollars in annualized revenue last quarter, but the foundational component of their business, the relational database, has been free for ~25 years.<sup>23</sup>

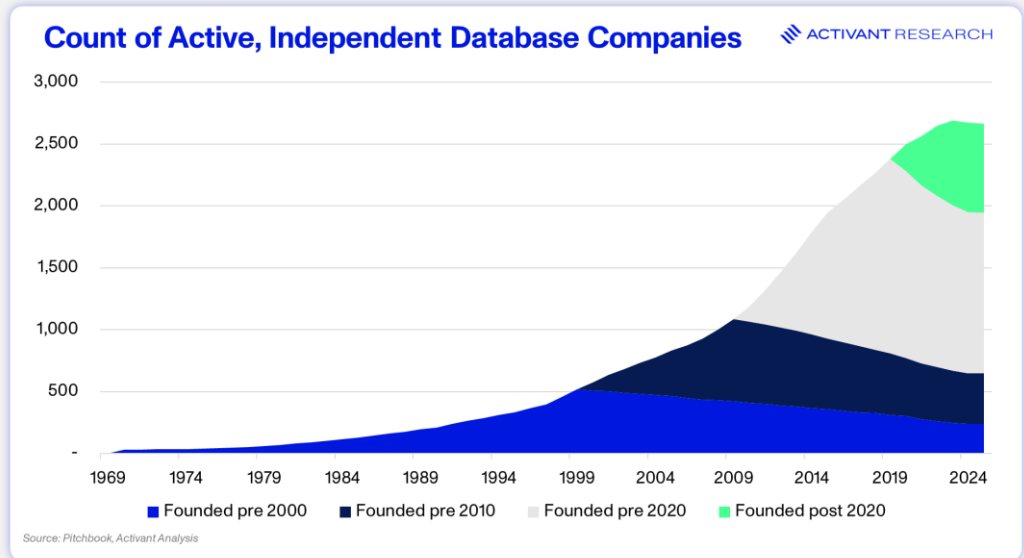
MySQL was released as a free, open-source relational database in 1995, quickly followed by PostgreSQL in 1996.<sup>24</sup> Neither was initially capable of threatening Oracle in serious enterprise deployments. But that missed the point. What they represented was a structural disruption to the cost of entering the database market. The marginal cost of software production fell not incrementally, but by orders of magnitude.

A startup could now fork open-source code, contribute enhancements, and build a credible commercial product at a fraction of the original investment. The technical complexity that had protected Oracle for fifteen years disappeared.

What followed should be instructive for today's enterprise software companies. Oracle was largely able to protect its enterprise business through **high switching costs, enterprise trust, and aggressively bundling** the database with upstream applications like Peoplesoft and NetSuite. Start-ups like Heroku and, more recently, Supabase built on top of Postgres but primarily sold to smaller customers or used the now commoditized database to serve different use cases, rather than trying to compete head-on with Oracle.

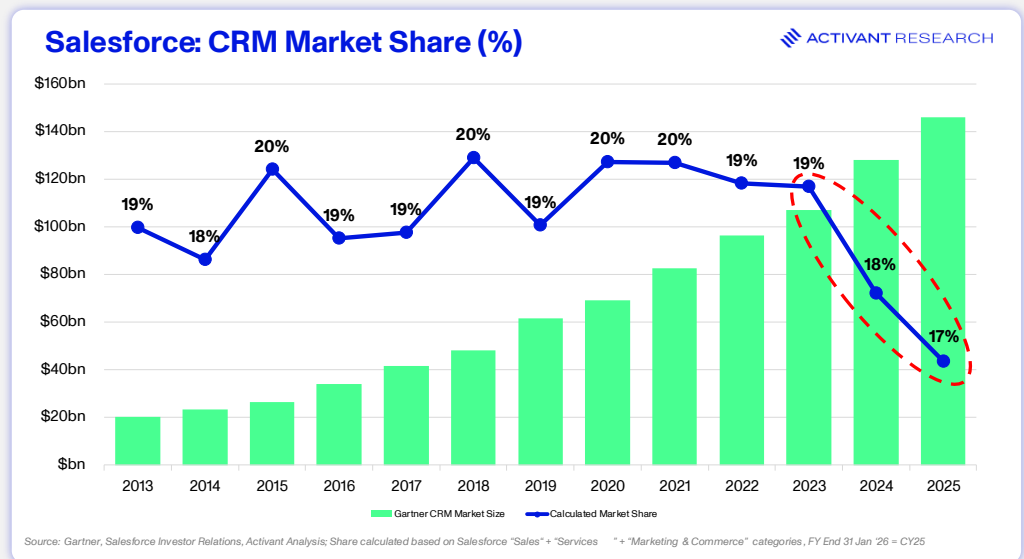
Similarly, when NoSQL databases like MongoDB rose in popularity, they rode the growth of the internet and social media, rather than ripping and replacing Oracle. In fact, [polyglot persistence](#) meant that customers could run dynamic product catalogues on MongoDB while routing transactions back to Oracle as the core.

The critical lesson is that Oracle had to react. It recognized the commodification of the core database and moved aggressively into apps, benefitting from their distribution advantage. But as the database market grew, it continued to support an increasing number of independent database companies without past generations needing to die. **Software is not always zero-sum.**



The Salesforce story reinforces this. It watched the rise of numerous new participants to the CRM market in the 2010s: vertical CRM tools like Veeva, vertical operating systems with bundled CRM like [Procore](#), [Service Titan](#) and [Toast](#), and niche sales tools like [Gong](#), [Outreach](#), [Clari](#) and [ZoomInfo](#) serve use cases like sales pipeline forecasting and customer profile enrichment. Rather than eat Salesforce's lunch, the entire market expanded more than fivefold in the period and, until AI, Salesforce held its share of the market.

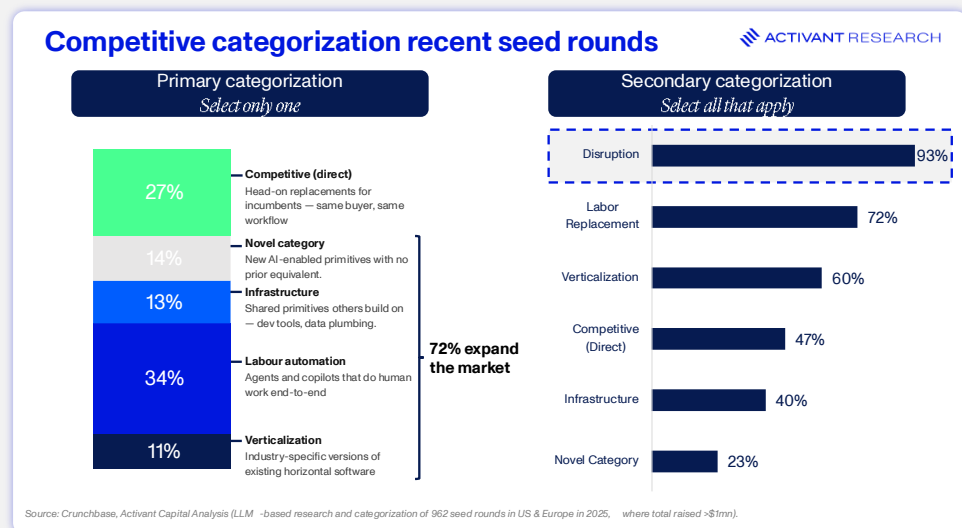
The company responded by launching competing industry clouds for financial services, healthcare, and manufacturing, and acquired IQVIA's pharma CRM to compete directly with Veeva. The lesson that the market is not zero-sum holds once again, but there is a concerning read-through: who captures the growth?



## It's the *Indirect* Competition

For Salesforce, the story turned in 2024. While the CRM market continues to grow at an attractive ~19% YoY, Salesforce has grown at ~9% for the past two years.<sup>25</sup> Its execution on the agent product has been commendable, closing 29,000 cumulative deals and reaching \$800 million in ARR. But that's just 2% of the business as they've been too [expensive and complex](#). AI Natives like Sierra are not trying to build a new CRM, but their customer support agents are capturing share that Salesforce needed to command a growth multiple.

As we show in our analysis below, this trend is accelerating significantly. New companies are (*largely*) not trying to build a new CRM, HRIS, or ERP but they are disrupting them. We categorized every US or European seed round that took place in 2025 over \$1 million, and only 27% are trying to directly compete with an incumbent software company.<sup>26</sup> **72% are expanding the market**, with the biggest primary category being labor automation.



Among ServiceNow competitors from the dataset, [Echelon](#) is automating the implementation consultants who deploy the platform itself. [Kovr.ai](#) is replacing the 1,000+ hours of manual FedRAMP documentation that compliance teams currently track through ServiceNow's GRC module. [Collate](#) is building purpose-built quality management for life sciences, pulling a vertical that ServiceNow serves generically into a domain-specific tool with regulatory-native workflows.

None of these companies are trying to build a better ITSM platform. None compete with ServiceNow for the same budget today and that is the critical competitive point. The cost of building software is falling towards zero, but that doesn't mean founders want to build the same software we built twenty years ago.

Removing the friction of creating software, and increasing its capabilities in parallel, massively expands the number of companies built and use cases served. The market expands, and software eats labor. ServiceNow doesn't die.

But here's the disruption threat. If [Console](#) succeeds in resolving 50–70% of IT support volume before a ticket is ever created, the platform that manages tickets becomes less valuable. If [Ravenna's](#) AI agents handle access provisioning, device diagnostics, and onboarding autonomously, fewer humans open ServiceNow every morning. If [Trustible](#) captures the emerging AI governance budget as a standalone category, that's a growth vector ServiceNow never gets to monetize.

Individually, none of these companies will kill ServiceNow. But 93% of the seed-stage companies in our dataset pose some form of competitive threat to incumbent software and the most likely outcome is that it becomes impossible to defend against every best-of-breed vendor in every niche that plugs into or picks away at your footprint. Death by a thousand cuts doesn't require any single cut to be fatal. For most incumbent software companies, the implication is that their market share charts start to look like Salesforce's over the next few years: **the absolute business survives, but the share of incremental value captured shrinks as the ecosystem around them explodes.**

This is precisely the future we mapped out for platforms like ServiceNow when we first penned our [systems of intelligence](#) thesis, and it's a natural segue to the final AI risk: the platform transition.

## 4. Platform Transitions

In the shift that we're seeing, users don't log into Salesforce to find prospects to reach out to, they tell their AI SDR to do it. As the system of record increasingly offboards work to the system of intelligence, the SoR's UI becomes invisible. The risk that follows is that after becoming invisible, they slowly disappear.

If the valuable piece of your software stack is the one doing the work, does it really matter which one holds the data? Perhaps organizations would be willing to switch to the cheaper alternative. Why pay \$10 million for a database when you could just dump all the data in a cheap S3 bucket and let the Sol figure out the rest.

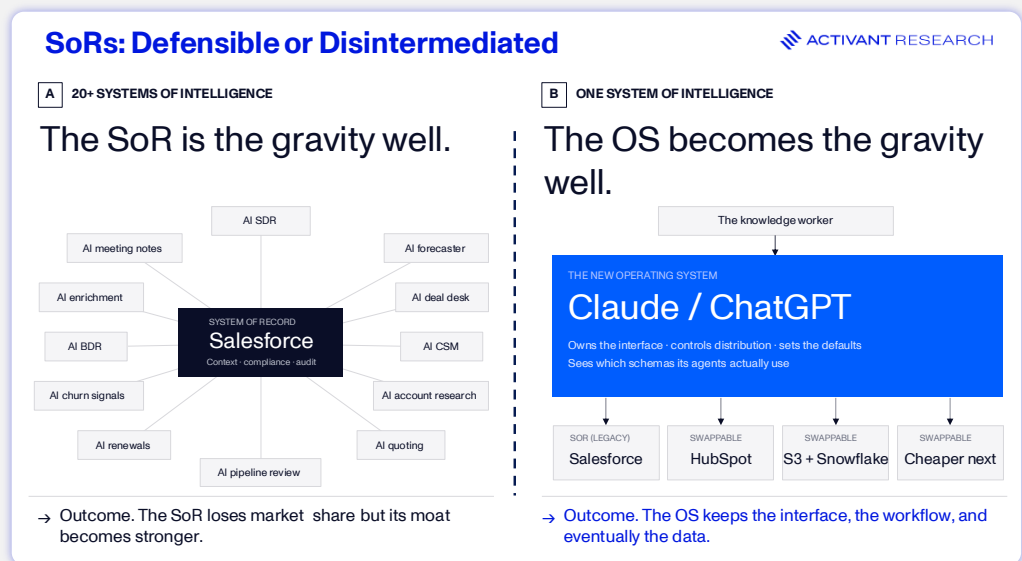
This is exactly what we're seeing in security: [Splunk](#) is being hollowed out by more intelligent architecture: instead of dumping every byte of raw telemetry, teams are using [Cribl](#) or [Tenzir](#) for intelligent filtering, offloading bulk storage to S3, or "security data lakes" powered by Snowflake or Databricks. Smart query engines and automated detection layers from companies like [Panther Labs](#), [Hunters](#), and [Anvillogic](#) then find threats without ever needing anyone to log into a legacy SIEM.

To be fair, stateless Splunk logs are the easiest to move, whereas expecting a different system to perform all the data control work that Salesforce does on a stateful opportunity is much harder.<sup>27</sup> That doesn't mean it won't happen, it just means that the disintermediation risk is proportional to the value that your system brings to the data.

Both Marc Benioff and Satya Nadella are confident enough in their platforms' value to declare that Salesforce and Microsoft can succeed as headless platforms. For Marc, it's the depth of customer and workflow data together with all the compliance infrastructure, security, and audit that can't be replicated by another system.<sup>28</sup> For Satya, it's the enterprise context housed in Microsoft's various data sources, which creates an opportunity for "the next Office" to be even bigger than the prior one, even if its headless.<sup>29</sup>

We think they could be right. In a world where you plug 20+ intelligence systems into the SoR for various use cases, it's the SoR's positioning that becomes **stronger** – because no individual system can perform the comprehensive context capturing while maintaining enterprise trust. That gives the SoR the right to layer agents on top of their platform and go after point solution Sols.

**In that world, the platform transition favors systems of record.** But what if there aren't 20 Sols? What if there's just one?



## Disintermediation is the Platform Transition Risk

Anthropic and OpenAI are both releasing connectors to every critical software system on Earth. Their goal is clearly to become the default interface for knowledge work, **the new operating system**. Both now connect to Microsoft 365

and apps including Figma, Notion, Jira, Monday.com, and more. They're creating a potential future where work on these platforms is orchestrated from Claude/ChatGPT, not from inside the systems themselves.

The risk for the connecting systems is that the operating system eventually becomes a competitor.

In the PC era, Microsoft did not dominate enterprise software by writing the best applications. It dominated by controlling the layer every application had to reach through. That position translated into three specific levers.

- **Distribution:** pre-installation and default app status for Windows-native tools.
- **Defaults:** Internet Explorer shipped with Windows and broke Netscape in roughly eighteen months, not because it was a better browser but because it was the one users didn't have to think about.
- **Developer gravity:** applications were built to Windows APIs, meaning Microsoft shaped which capabilities were first-class and which had to work around platform constraints.

In the same way, if Anthropic or OpenAI become the horizontal operating system where work takes place, the economics of the SoR invert. Salesforce stops being the mega-source-of-truth that 20 intelligence layers orbit around and becomes one database among several that the operating system chooses to reach into. The OS gets to decide which CRM is first-class. It sees exactly which data structures and features its agents actually need. And in time, it has everything required to build its own AI-native alternative.

The terminal case is where this gets existential. AI-native architectures do not want relational CRM schemas, they want unstructured document stores, embeddings, and memory layers that match how agents reason over data. Once the OS has enough leverage and enough traffic flowing through it, the rational move is to migrate the underlying customer data into an architecture built for the way that data is now used. At that point, the SoR does not just become invisible, it becomes redundant. Anthropic keeps the interface, the workflow, and eventually the data. Salesforce goes away entirely.

The platform transition risk is clear: for SoRs that don't build a deep ecosystem around themselves or perform sophisticated, proprietary logic on customer data, the long-term path is disintermediation.

# Charting the Path Forward

Individually, none of the risks we have laid out are legitimate enough, or progressed enough to warrant the SaaS apocalypse the market is pricing today. **Seat contraction** is on a much longer timeline than current multiples imply. **Vibe-coded substitution** runs into the realities of maintenance, security, and compliance. **Pricing model disruption** is the more pressing version of the seats problem, but it is a required business model shift, not the end of already successful businesses. **Platform transitions** cut both ways: SoRs that become the connective tissue across many intelligence systems get stronger, while those with a single dominant Sol on top get hollowed out.

**Indirect competition** is the most pervasive risk in our framework: 93% of seed-stage companies in our dataset pose some competitive threat to incumbent software and AI is going to significantly accelerate the pace of new entrant creation.

The problem for today's software companies is that all of these risks are appearing simultaneously.

Worse, the defenses against each risk amplify exposure to the others. Innovating pricing models compresses near-term revenue at exactly the moment the market is already repricing the multiple. Buying or building agentic capability into adjacent workflows requires investment at exactly the moment when the equity currency for those acquisitions has been marked down. Opening the platform to retain ecosystem gravity accelerates the disintermediation it is meant to prevent; closing it accelerates the competitive substitution it is meant to defend against. And every dollar of management attention spent on one front is a dollar not spent on the others, against a wave of AI-native entrants whose entire surface area is the new paradigm, and who carry none of the legacy tradeoffs.

It's clear that only the strongest of incumbent software companies will come through this era as successful as they went in. That's what we explore in our next article.

- 
- <sup>1</sup> [BG2 Podcast. Satya Nadella | BG2 w/ Bill Gurley & Brad Gerstner. 2025](#)
  - <sup>2</sup> [Humlum, Vestergaard. Large Language Models. Small Labor Market Effects. 2025](#)
  - <sup>3</sup> [Chen, et al. The \(Short-Term\) Effects of Large Language Models on Unemployment and Earnings. 2025](#)
  - <sup>4</sup> [Brynjolfsson, et al. Canaries in the Coal Mine? Six Facts about the Recent Employment Effects of Artificial Intelligence. 2025](#)
  - <sup>5</sup> [Chen, et al. The \(Short-Term\) Effects of Large Language Models on Unemployment and Earnings. 2025](#)
  - <sup>6</sup> [Stripe, Bret Taylor of Sierra on AI agents, outcome-based pricing, and the OpenAI board. 2026](#)
  - <sup>7</sup> [Ibid](#)
  - <sup>8</sup> [Google Cloud. 2025 State of AI-assisted Software Development. 2025](#)
  - <sup>9</sup> [GitLab, Investor Relations 2026](#)
  - <sup>10</sup> GitLab, Q4 FY 26 Earnings Call, 2025
  - <sup>11</sup> [Ibid](#)
  - <sup>12</sup> [Salesforce. Q4 FY26 Earnings Conference Call. 2026](#)
  - <sup>13</sup> ServiceNow, Q4 FY25 Earnings Conference Call, 2026 (via CapIQ)
  - <sup>14</sup> [Epoch AI. Benchmarks: SWE-bench verified. 2026.](#) *SWE-bench measures the ability of AI models and agents to automatically resolve real-world software engineering issues by navigating, modifying, and fixing bugs in actual, open-source GitHub repositories*
  - <sup>15</sup> Alphabet, Q3 2025 earnings call, 2025
  - <sup>16</sup> [Entrepreneur. Anthropic CEO Says AI Could Replace Software Engineers in 6 to 12 Months. 2026](#)
  - <sup>17</sup> CapIQ, 10 March 2026
  - <sup>18</sup> Monday.com has 4,281 customers in excess of \$50,000 ARR; [Q4 FY25 Earnings Release](#)
  - <sup>19</sup> [Anthropic. Claude Mythos Model Card. 2026](#)
  - <sup>20</sup> [Vention. Software maintenance costs: Tips for keeping costs low and quality high. 2026](#)
  - <sup>21</sup> Salesforce, Investor Relations, 2026
  - <sup>22</sup> [Indiehackers.com. AI startups are speedrunning to \\$100M ARR with barely any people on board. 2025](#)
  - <sup>23</sup> [Oracle. Q3 FY26 Earnings press release. 2026](#)

<sup>24</sup> MySQL, Wikipedia, 2026

<sup>25</sup> Salesforce, company reports, 2025 - 2026

<sup>26</sup> Based on Activant categorization, denotes primary categorization vector

<sup>27</sup> Stateless vs. Stateful Data: Splunk logs are "stateless," meaning they are independent snapshots of events that are easy to move because they don't rely on previous context. Conversely, Salesforce opportunities are "stateful," involving complex, interconnected data (like history, permissions, and stage transitions) that must be maintained across systems to remain functional.

<sup>28</sup> [VentureBeat, Morgan Stanley Technology, Media & Telecom Conference, 2026](#)

<sup>29</sup> Microsoft, Morgan Stanley Technology, Media & Telecom Conference, 2026

**Disclaimer:** The information contained herein is provided for informational purposes only and should not be construed as investment advice. The opinions, views, forecasts, performance, estimates, etc. expressed herein are subject to change without notice. Certain statements contained herein reflect the subjective views and opinions of Activant. Past performance is not indicative of future results. No representation is made that any investment will or is likely to achieve its objectives. All investments involve risk and may result in loss. This newsletter does not constitute an offer to sell or a solicitation of an offer to buy any security. Activant does not provide tax or legal advice and you are encouraged to seek the advice of a tax or legal professional regarding your individual circumstances.

This content may not under any circumstances be relied upon when making a decision to invest in any fund or investment, including those managed by Activant. Certain information contained in here has been obtained from third-party sources, including from portfolio companies of funds managed by Activant. While taken from sources believed to be reliable, Activant has not independently verified such information and makes no representations about the current or enduring accuracy of the information or its appropriateness for a given situation.

Activant does not solicit or make its services available to the public. The content provided herein may include information regarding past and/or present portfolio companies or investments managed by Activant, its affiliates and/or personnel. References to specific companies are for illustrative purposes only and do not necessarily reflect Activant investments. It should not be assumed that investments made in the future will have similar characteristics. Please see "full list of investments" at [activantcapital.com/companies/](http://activantcapital.com/companies/) for a full list of investments. Any portfolio companies discussed herein should not be assumed to have been profitable. Certain information herein constitutes "forward-looking statements." All forward-looking statements represent only the intent and belief of Activant as of the date such statements were made. None of Activant or any of its affiliates (i) assumes any responsibility for the accuracy and completeness of any forward-looking statements or (ii) undertakes any obligation to disseminate any updates or revisions to any forward-looking statement contained herein to reflect any change in their expectation with regard thereto or any change in events, conditions or circumstances on which any such statement is based. Due to various risks and uncertainties, actual events or results may differ materially from those reflected or contemplated in such forward-looking statements.