

MARKET RESEARCH BRIEFING · THE CONTEXT LAYER

Making AI Agents See Your Business

Why capable agents still get company work wrong, and when shared context pays off

An original cost model of shared business context for AI agents: where it pays back, what to build first, what to buy, and when not to do it

An evidence briefing for operations and technology leaders at enterprises whose core business is not software. Modeled figures are StratoFactory analysis; vendor-reported figures are labeled at the point of use.

1. Executive Summary

Why this matters

AI agents are software that carries out work inside company systems: checking an order, answering a supplier, reconciling an invoice. The models behind them have become capable. On a standard test of general computer tasks, the best agents now score 66.3%, close to the human baseline of 72.35% [E08]. What they usually lack is knowledge of the business they work in. They do not know that "ACME Ltd" in the ERP and "Acme Limited" in the CRM are the same customer, what a field in the finance system actually means, which documents a given employee may see, or which actions are allowed. We call this knowledge the agent's **context**. The problem is now widely recognized: in October 2026 a leading technology analyst firm listed "AI context platforms" among its ten strategic technology trends for 2027 [E47], and a global management consultancy reports that eight in ten companies cite data limitations as a roadblock to scaling agents [E44].

Without it, performance drops sharply. On a test built from large, messy company databases, the best AI system solved 21.3% of tasks; on an older test with small, tidy databases the same system solved 91.2% [E09]. When the same data was organized and explained before the AI saw it, accuracy rose by between 17 and 38 percentage points in three separate studies [E10–E12]. In most cases the AI is capable enough. **What it lacks is a view of the business.**



The idea, and why it rarely gets built

Most companies handle this one project at a time, with each agent team gathering and connecting the business knowledge its own agent needs. A **shared context layer** does that work once, to a common standard, so later agents can reuse it. The catch is that the first project would pay for something mainly later projects benefit from, and no single project budget can justify that. Few companies have built one: in a vendor survey, only 16% say they deliberately engineer it as a product [E02].

Where the value lies

We built a cost model (StratoFactory analysis, not measured data) to find out when the layer pays for itself. It uses US labor costs and mostly US evidence, so dollar figures will differ in other markets. Its value comes from three places:

- **Not paying for the same work again and again.** Built across the whole company up front, the layer costs USD 2.6 million over 3 years, about 6.1 times what one agent spends on its own context. It becomes cheaper than the project-by-project approach once the company has **11 agents**. By 30 agents it saves USD 4.6 million, or 39% of what the project-by-project route would cost.
- **Fewer wrong answers, so less human checking.** Agent output has to be checked by a person, and wrong output has to be redone. In an illustrative calculation, better context cuts that checking and rework time by 31%. If even a quarter of the benchmark improvement carried over to real work, the layer would pay back at agent 9 instead of 11. Nobody has measured this inside a company yet, so we treat it as plausible but unproven.
- **Control in one place.** Rules about who may see what are enforced once, in the layer, instead of separately inside every agent. In tests, AI agents showed almost no built-in sense of confidentiality [E13], so this matters for risk as much as for cost.

All three depend on scale. **With only a handful of agents, the layer costs more than it saves.**

What we recommend

- **Build it one business area at a time.** Start with one area, such as finance or customer service. The first area's share of the layer pays for itself once that area runs 6 agents; each later area needs only 3, because the company-wide parts are already paid for. Built this way, a company with several smaller areas breaks even at about agent 9 overall. It also avoids the familiar fate of company-wide data programs that take years and arrive out of date.
- **Build in a set order:** Observability → Permissions → Document extraction → Entity resolution → Tool surface → Golden questions → Metadata and lineage. Logging what agents do and controlling who may see what come first. Both are cheap, both reduce risk straight away, and the later steps depend on them.



- **Buy what is becoming standard, build what is specific to you.** Data cataloging (metadata and lineage) is cheaper to buy. Mapping your own customers, products and contracts is specific to your business, and the products that claim to do it are still immature, so build that part.
- **If you are not ready, keep two cheap habits.** Record access permissions on everything agents read, and log what agents do from day one. Both make a later layer far cheaper.

When it is the wrong choice

Do not build it if you cannot see at least 6 agents in a first business area, or about 9 across two areas, within three years. Do not build it if your planning horizon is shorter than about three years: a layer built up front needs at least 39 months to overtake the project-by-project route and delivers no agents in its first year, and even a layer built area by area needs 30 months. And do not run it as one company-wide program. If such a program's cost doubled, the break-even would move to agent 22. Separately, AI that reads very long documents directly could erode the value of some parts of the layer, which would push the company-wide break-even to agent 17.

How sure we are

Benchmark evidence that business context limits AI on company data is reasonably strong, and the economics of why no single project funds the layer are straightforward. Evidence on how many companies are held back by poor data all points the same way, but none of it is independent. The strongest, a Deloitte survey of 501 US leaders, found that 72% lack unified, accessible data [E45]; it states its sample but no margin of error. The most quoted figures (68% of technical leaders calling themselves AI-mature, while 93–96% say most of their data is not ready) are **vendor-reported** [E01]. We treat the direction as established and the size as uncertain. Our cost figures rest on stated assumptions, and readers can replace them with their own.

Headline predictions (2027–2029)

Section 9 sets out six dated predictions, each with a test that would prove it wrong. Among them: through 2029, major vendors will keep a managed knowledge layer in their agent designs rather than feeding agents whole document collections; by end-2028, at least two major platforms will sell a priced product for mapping business entities; and by end-2027, no independent survey will find most large companies running a shared context layer.

2. Where Enterprises Actually Are

The gap between how ready enterprises say they are and how ready their data is has been widely reported, but the evidence behind it is weaker than the coverage suggests. Almost every readiness figure comes from a vendor that sells the remedy or from an analyst firm that does not publish its method. What survives scrutiny is narrower: measured independently, AI use is still a minority activity, scaling agents even more so, and readiness is self-reported almost everywhere.

The perception gap, and who measured it

The most quoted recent figures come from Komprise, a vendor of unstructured-data management software. Its survey of 500 technical decision makers, fielded in September 2026, found that 68% describe their organization as maturing or mature in AI, while 93–96% report that most of their data is not ready for AI and lakehouses. Only 28% have quantified ROI from AI on average [E01]. These are **vendor-reported figures**. The release gives no country, fieldwork agency, margin of error, weighting method or exact question wording.

A second vendor source points the same way and adds context-layer questions. The Modern Data Company's third Modern Data Survey (more than 540 responses from data leaders and practitioners in 65+ countries, drawn from its own community) found that 57.3% are piloting or running AI agents but only 8.4% say the data feeding their AI is trustworthy enough for production. Sixty-one percent call a reliable context layer critical or very important, yet only 16% deliberately engineer one as a product [E02]. Organizations with agents in production were 3.6 times as likely to have engineered their context layer, though the company notes this does not establish causation. This is a **vendor-reported, self-selected sample**: consistent with the thesis, but unable to carry it.

The consultancy, analyst and official evidence

Gartner reported in February 2025 that "Sixty-three percent of organizations either do not have or are unsure if they have the right data management practices for AI", and added: "Gartner predicts that through 2026, organizations will abandon 60% of AI projects unsupported by AI-ready data." [E03] The same release gives two incompatible sample descriptions: 1,203 respondents in the body text and 248 in the page metadata. We carry the conflict rather than pick one. A separate Gartner survey of 1,303 organizations, fielded January to April 2026, found that "Only 22% of organizations have successfully scaled AI across multiple business units or adopted an AI-first approach" [E04].

A global consultancy's annual AI survey (1,719 participants, fielded May to June 2026) found about two in ten respondents reaching the scaling phase with AI agents. Among large organizations the share reporting scaled agents rose to 40%, from 27% a year earlier, while among smaller organizations it stayed at 22%. Only 37% attribute any EBIT impact to AI, against 39% a year earlier [E43] [E05]. Deloitte's August 2026 survey of 501 US leaders at organizations already piloting agents found that only 15% have scaled, orchestrated multi-agent adoption. In the same survey, 72% say they lack unified, accessible data and 67% say integration is too costly and complex [E45]. The same firm's April 2026 guidance on agent foundations reports that eight in ten companies cite data limitations as a roadblock, but it does not name the survey behind the figure [E44]. Gartner predicted in June 2025 that "Over 40% of agentic AI projects will be canceled by the end of 2027, due to escalating costs, unclear business value or inadequate risk controls" [E07].

The only probability-sample official statistic is the US Census Bureau's Business Trends and Outlook Survey. Between December 2025 and May 2026, AI use across US businesses hovered between 17% and 20%, and was 37% among firms with at least 250 employees [E06].

What this section can and cannot conclude

Finding: the direction of the context-readiness gap is established; its size is not. The Deloitte survey is the only one we found from a firm that does not sell data products and that states its sample and fieldwork period. It still gives no margin of error, and consultancies have their own interest in the answer. Two things can be said with some confidence. Scaling is a minority activity on every measure above, and the AI Index finds single-digit scaled agent use in nearly every business function [E53]. And every source that measures readiness, vendor or consultancy, points the same way. The rest of the report does not lean on these figures. The model's crossover depends on how much context work each agent needs, which is the same whether a reader's peers are well prepared or not.

3. What "Context" Decomposes Into

"Context" gets used for three different things, and most vendor material moves between them without saying so. **Context availability** means the data exists and an agent can reach it.

Context quality means the data is correct, current, permissioned and resolvable to the right entity. **Task success** means the agent does the right thing with it. A layer can deliver the first two and still fail on the third, in which case it has not paid for itself. The cost model in Sections 5 and 6 depends on keeping the three apart.

Availability is not enough

Spider 2.0 shows this most clearly. Its 632 enterprise text-to-SQL problems run against cloud databases "often containing over 1,000 columns". To solve them, an agent must search "database metadata, dialect documentation". The metadata is available to the agent. Even so, the best agent framework solved only 21.3% of tasks, against 91.2% on Spider 1.0 and 73.0% on BIRD [E09]. The agent framework and model were the same on all three tests; what changed was the size and disorder of the data.

Quality moves accuracy on the same data

Three benchmarks hold the data constant and change only how it is represented. GPT-4 answering enterprise insurance questions directly over SQL scored 16%. Over a knowledge-graph representation of the same database it scored 54% [E10]. A 2026 benchmark by authors at Cube, a semantic-layer vendor, found that adding a semantic layer lifted three frontier models from 45.5–50.5% to 67.7–68.7% on 99 retail analytics questions [E11]. This is **vendor-authored**: the semantic document was written by an analyst who had seen the dataset, and the judge model shares a family with two of the models tested. A third 2026 preprint, on industrial maintenance and with no vendor affiliation stated, found that a typed knowledge graph lifted the same GPT-4 agents from 65% to 82–83% on 139 scenarios. Its authors conclude that "the data layer -- not the LLM orchestration -- is the primary lever" [E12].

These are benchmark accuracies, not production acceptance rates, and each study changes the data differently. Three teams in three domains agree on the direction; they disagree on the size, and we have not averaged them.

Task success needs more than context

Even with the data reachable through tools, agents fail at business processes. On CRMArena-Pro, leading agents reach about 58% success in single-turn settings and drop to 35% in multi-turn settings. They also show "near-zero inherent confidentiality awareness" [E13]. On τ -bench, the best agent in 2024 succeeded on fewer than 50% of tasks, and only about 25% of the time across eight repeated trials in retail [E14]. These failures come from following rules, gathering missing information and staying consistent. They are task-success failures that a context layer only partly addresses.

The confidentiality result matters most for design, and controls are rare: a 2026 survey of more than 1,300 executives by a global consultancy found that only 5% of companies have the full set of critical controls in place for agents [E46]. Researchers at Microsoft argue that only "deterministic and rigorous enforcement of fine-grained access control" prevents leakage in retrieval systems, and that any content used must be authorized for "all users involved in the interaction" [E16]. Permission has to be enforced by the layer; the model cannot be relied on to respect it.

Is the model still the constraint?

On generic tasks, much less than it used to be. OSWorld computer-use accuracy reached 66.3% in the 2026 AI Index data, against a 72.35% human baseline [E08]. Set beside Spider 2.0, this is consistent with the thesis: capability is close to human on generic interfaces and far from it on enterprise data. Two benchmarks published just before our cut-off sharpen the point. On Argo-Bench, built on an ERP-style warehouse of 235 tables and 7.5 billion rows, the strongest of 14 models fully solved only 34.8% of 210 analytics tasks [E49]. On Era by Eon, when the answer depended on a fact that no record states directly, the best agent got 18 of 24 attempts right, and the hardest questions were answered correctly in only 1 of 84 attempts [E50].

Knowledge that a business holds only implicitly is exactly what a context layer has to make explicit. A Gartner analyst made the same point in September 2026: "The success of agentic AI depends less on the model and more on data context." [E23]

The seven components

We decompose the layer into seven components and say which of the three properties each mainly serves. The list is close to the foundations a global consultancy described in April 2026, which include shared definitions, built-in access controls, stable interfaces for agents and observability of how agents use data [E44].

- **Entity resolution and master-data mapping.** One customer, one part, one contract across systems. Serves quality.
- **Access control and permission propagation.** Source permissions carried to every chunk, row and tool call. Serves quality, as a hard precondition.
- **Document extraction and chunking.** Turning files into retrievable units. Serves availability.
- **Metadata, classification and lineage.** What a field means, how current it is, where it came from. Serves quality.
- **Tool and action surface.** The governed operations agents can call. Serves task success.
- **Retrieval evaluation set (golden questions).** The test that tells you whether the layer works. Measures all three.
- **Trace capture and observability.** The record of what the agent saw and did, and the raw material for measuring task success in production.

The decomposition lets the model separate two kinds of value. **Build-reuse value** is the saving from not rebuilding each component for every agent. **Quality value** is the saving from agents getting more tasks right first time. Section 5 models both, separately.

4. Why No Single Project Can Fund It

StratoFactory's September report found that internal building pays on the portfolio, not the application. Cumulative cost crossed over "at 1.1 applications, or 1.2 if only half the assumed reuse is realized". The reason was that "The second application on an internal platform inherits identity, the data model, the integration fabric, CI/CD, observability and increasingly the agent orchestration layer." [E35] When the applications are agents, the context layer is the thing that gets inherited. We do not re-derive that crossover here. The question for this section is why, if the logic is the same, so few context layers exist.

The answer is the size of the first bill

The September platform was cheap relative to the application that first paid for it. The context layer is not. Built to enterprise standard across all seven components, it costs USD 2.6 million over a 3-year lifecycle. That is about 6.1 times the USD 432k that one agent spends on its own context slice. No project business case can carry six times its own context cost. In the model, the enterprise layer does not pay back until agent 11. At five agents, the layer path has cost USD 3.4 million against USD 2.0 million without it.

That is why the September report's crossover arrives at the second application and this one does not. It is also why the finding is fragile in a way September's was not. Halving the reuse shares moves September's crossover from 1.1 to 1.2 applications [E35]. Halving them here moves the context-layer crossover from agent 11 to agent 27. The September report warned that its crossover "is a property of a platform team, a component registry and enforced reuse standards". For the context layer, that condition binds much harder.

Why projects do not coordinate their way to it

Each agent team sees its own slice. Building to a shared standard costs that team more than building for itself, and most of the saving goes to the teams that come after it. Project-based funding, where each agent must justify itself within its own budget, will therefore under-invest in the layer even when the portfolio would gain. The vendor survey figure that only 16% of organizations deliberately engineer a context layer [E02] fits this mechanism, though it does not prove it.

The fix is mostly organizational. Someone has to own the layer as a product, with a budget that does not come out of any one agent's business case, and with the authority to stop teams building duplicate context work. In practice that usually means a centrally funded platform team for the first domain, with later agent teams paying only for their own residual work. The September report reached a similar conclusion about application platforms. The difference is that the context layer's first bill is larger, so the central funding has to be committed before the business case is visible.

Analysts now frame the layer as infrastructure rather than as a project. Gartner's list of strategic technology trends for 2027, published on 20 October 2026, includes "AI context platforms" [E47]. One of the firm's analysts warned: "Without a dedicated context layer, organizations face severe operational risks, including overprivileged agents," "compliance failures, low trust and excessive token consumption" [E47b]. All of this is advice and forecasting rather than evidence that the layer pays. Our model gives the conditions under which the advice pays off: an expected estate of at least 11 agents within the lifecycle for an enterprise-wide build, or at least 6 agents in the first business domain for a seeded one.

A note on the house view

Most of StratoFactory's earlier reports found the optimistic case overstated; this one comes out mildly positive. We built the model before writing the conclusion and kept its answer: the case is real, but only for an estate of at least 6 agents in a first domain and a horizon of roughly three years.

5. Original Analysis: The Marginal Cost of the Nth Agent

Every agent needs some business context: which records refer to the same customer, who may see what, how to read the company's documents, which actions it may take. Each team can build that context for its own agent, or the company can build it once, in a shared layer that every later agent reuses. The shared layer costs more up front and less per agent. This section asks how many agents it takes before the shared route is cheaper, and which inputs decide the answer.

Every modeled number comes from the model script (key assumptions in Appendix A.2); the few external figures are cited. Costs are our judgment wherever no benchmark exists, which is almost everywhere. The model shows **which way** the answer goes and **what decides it**, not the costs of any particular enterprise.

How the model works

The model takes the seven components of Section 3 and asks three questions of each. What does it cost to build once, for the whole company (B)? What does one agent team spend building its own version, its **slice** (s)? A slice is the work a team does for its own agent alone: mapping the entities it touches, wiring the permissions of its sources, parsing its documents, writing its tools. And what share of that slice does a later agent avoid when the shared layer exists (the **reuse share**, r)?

- **Without a layer**, the first agent pays its full slice, and each later agent pays it less 10% incidental reuse from copying earlier teams' work.
- **With a layer**, the enterprise pays B once. Every agent, including the first, then pays only the thin residual $(1 - r)$ of each slice.



All costs are lifecycle costs: the build plus 3 years of maintenance, at 20% a year for slices and 25% for the shared layer, which drifts faster because it is curated. Over the lifecycle, a person-month of build work therefore costs 1.6 person-months in a slice and 1.75 in the layer.

Engineering effort is converted at USD 15,864 per person-month: the Bureau of Labor Statistics median software-developer wage of USD 135,980 (May 2025) [E33] with a 1.4 loading factor, not adjusted for inflation. **The cost base is American:** labor costs, and most surveys and benchmarks behind the inputs, come from the US market. Where engineering is cheaper, dollar figures shrink roughly proportionally, but the break-even agent counts and build order do not, because they compare effort with effort. The build-or-buy verdicts can shift, since vendor subscriptions are priced in dollars.

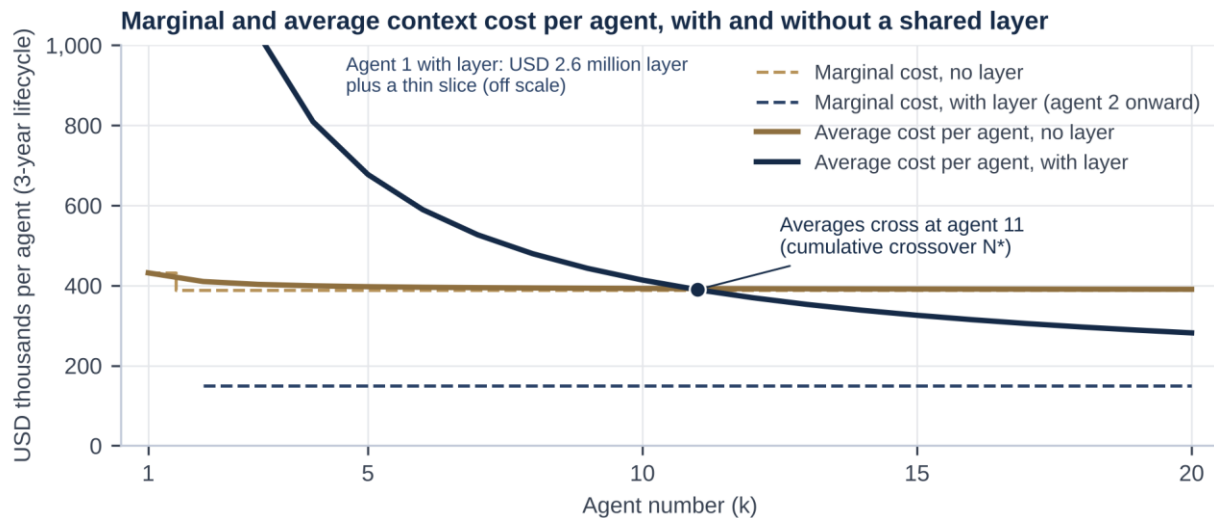
For one component, the layer path is cheaper once the number of agents N exceeds $(B/s - i) \div (r - i)$, with B and s as lifecycle costs and i as incidental reuse. Observability is a worked example. Its shared version costs 5.5 times one agent's slice, and later agents avoid 85% of the slice, so it pays back after $(5.5 - 0.10) \div (0.85 - 0.10) \approx 7.2$ agents, that is, at agent 8.

Direction: the crossover rises as the layer gets more expensive or as incidental reuse rises, and falls as reuse shares rise or as slices get more expensive. The script checks all four directions on every run, and confirms that a component never pays back if its reuse share is no higher than incidental reuse.

The base case puts the whole layer at 95 person-months (USD 1.5 million to build) and one agent's slice at 17 person-months (USD 270k). Reuse shares range from 0.30 for golden questions to 0.85 for permissions and observability. Where a component resembles one in the September report, we anchored its reuse share there; tool-surface reuse, for example, matches that report's 45% integration reuse [E35].



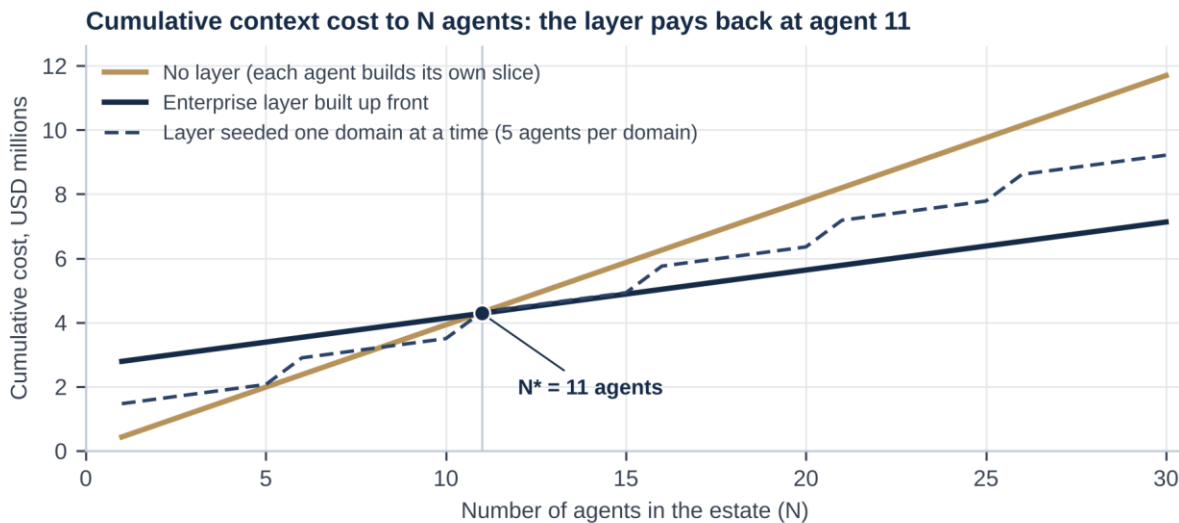
Results



Source: StratoFactory analysis (model v1.0), October 2026. Lifecycle cost in USD 2026 at US labor costs. Modeled, not measured.

Figure 1. Marginal and average context cost per agent, with and without a shared layer, agents 1 to 20. The first agent on the layer path carries the full layer build (USD 2.6 million lifecycle cost, off scale). Every later agent costs USD 150k, against USD 388k without a layer. Average cost per agent crosses at agent 11. StratoFactory analysis: modeled, not measured; 3-year lifecycle cost in USD 2026.

The first agent with a layer carries the full build cost. Every later agent costs USD 150k over its lifecycle, against USD 388k without a layer (Figure 1). Average cost per agent on the layer path falls as the build is spread over more agents and meets the no-layer average at agent 11.

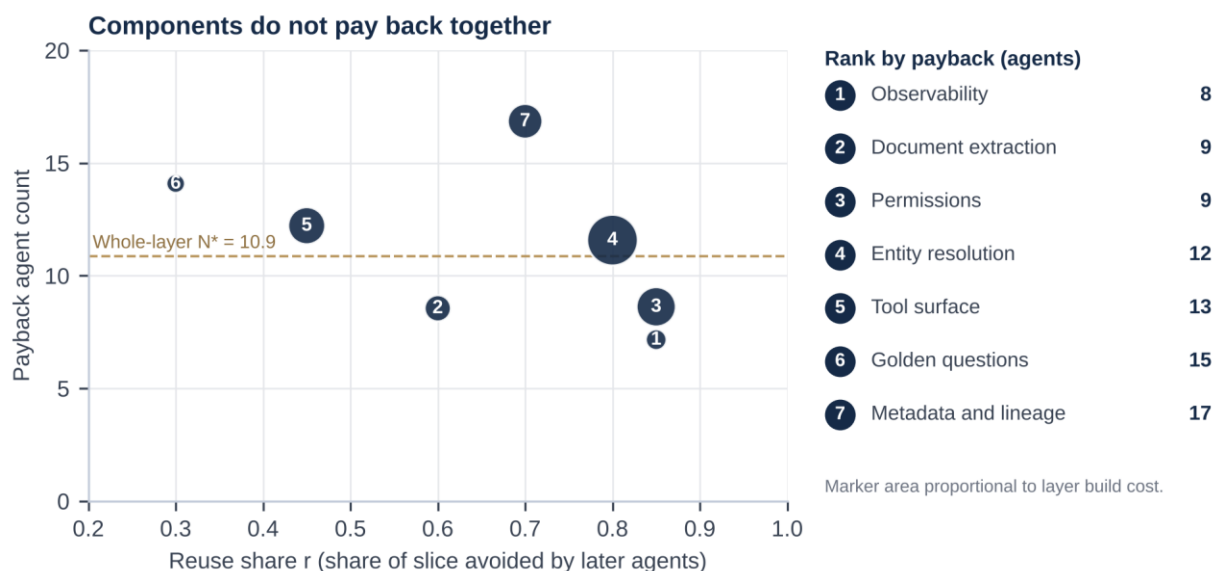


Source: StratoFactory analysis (model v1.0), October 2026. 3-year lifecycle cost, USD 2026, US labor costs. Modeled, not measured.

Figure 2. Cumulative context cost to N agents on three paths: no layer, an enterprise layer built up front, and a layer seeded one domain at a time (5 agents per domain). The up-front layer crosses at $N^* = 11$. The seeded path first breaks even at agent 9 but costs more at scale, because domain-scoped components are rebuilt for each domain. StratoFactory analysis: modeled, not measured.

Cumulatively (Figure 2), the two paths cost about the same at agent 11, roughly USD 4.3 million each. By 30 agents the layer path is USD 4.6 million cheaper.

The dashed line shows the variant the model favors: a layer **seeded** one business domain at a time, with roughly 5 agents per domain. Permissions and observability are built once for the whole enterprise; the other five components are built for each domain as it arrives. The first domain pays back once it holds 6 agents, because it also carries the enterprise-wide components. Each later domain pays back at 3. With domains of about 5 agents, the first domain does not quite get there on its own; the estate as a whole first breaks even at agent 9, late in the second domain, once permissions and observability start being reused. Opening a third domain at agent 11 briefly puts it USD 11k above the no-layer path; from agent 12 onward it stays below. The price of seeding is a higher bill at scale, because domain-scoped work is repeated for each domain.

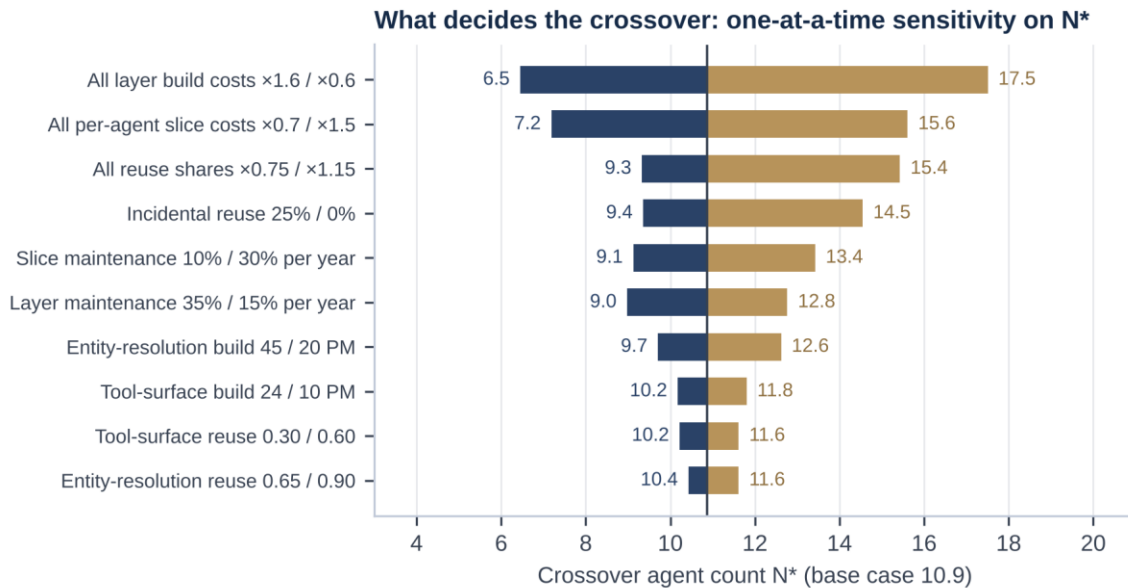


Source: StratoFactory analysis (model v1.0), October 2026. Payback = smallest N at which the component's layer path is cheaper. Modeled, not measured.

Figure 3. Component payback: reuse share against payback agent count, ranked. Paybacks range from agent 8 (observability) to agent 17 (metadata and lineage). Marker area is proportional to layer build cost. StratoFactory analysis; costs and coefficients are analyst judgment (S03 to S23).

Components do not pay back together (Figure 3). Observability pays back at agent 8, document extraction and permissions at agent 9, entity resolution at agent 12, the tool surface at 13, golden questions at 15, and metadata and lineage at 17. The spread has two sources: how expensive each component's shared version is compared with one agent's slice, and how much of the slice later agents actually avoid. Metadata reuses a respectable 70% of the slice but pays back last, because its shared version costs 10.2 times one agent's slice; observability's costs 5.5 times. Golden questions are the reverse case: their shared version costs only 2.9 times a slice, but later agents avoid just 30% of the work, so they pay back late. Section 6 turns this spread into a build order.

What decides the answer



Source: StratoFactory analysis (model v1.0), October 2026. One input varied at a time. Blue: favorable setting; gold: unfavorable. Modeled, not measured.

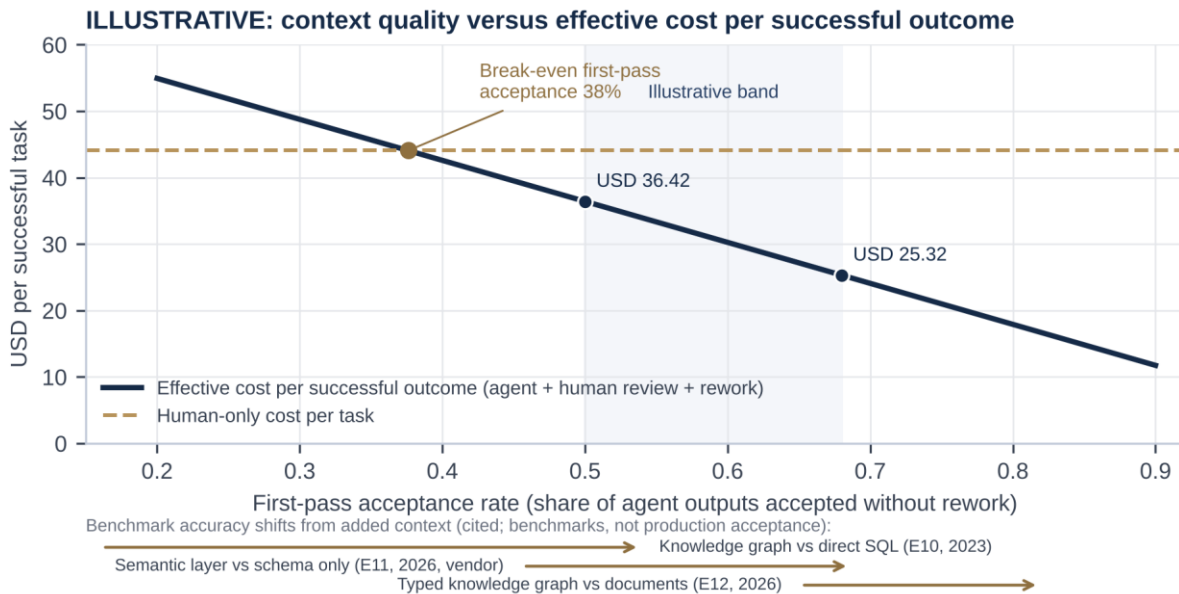
Figure 4. One-at-a-time deterministic sensitivity (tornado) on the crossover agent count N, base case 10.9. Bar ends are continuous values; the text rounds them up to whole agents. The size of the layer relative to an agent's slice decides the answer. Individual components' reuse coefficients matter least. StratoFactory analysis: modeled, not measured.*

The tornado chart (Figure 4) varies one input at a time. **What decides the crossover is the size of the layer relative to an agent's slice.** The reuse shares matter less. Moving all layer build costs between 0.6 and 1.6 times base moves the crossover from agent 7 to agent 18. Per-agent slice costs come second (8 to 16), and the reuse shares third (10 to 16). Any single component's reuse share moves the answer by less than two agents. The practical lesson is to estimate, before anything else, what the layer would cost in units of one agent's context work. If that ratio is well below 6.1, the crossover comes much sooner.

Quality coupling: the link to the verification tax (illustrative)

Our June report defined the verification tax as "the additional human effort required to read, test, debug, and gain confidence in AI-produced changes". It showed that review queues diverge as utilization approaches one [E36]. Neither that report nor any production source we could find gives a first-pass acceptance rate, so the extension below is **illustrative**. Its inputs are S26 to S33 in Appendix A.2, and none of them is field-calibrated.

Take a task that costs USD 44.07 when a person does it (25 minutes at USD 106 an hour). An agent attempt costs USD 0.29 in tokens (at the May report's USD 4.75 per million tokens [E39]) plus 3 minutes of review, which is always paid. A failed output adds 10 minutes of diagnosis plus the manual task. Effective cost per successful outcome then depends only on first-pass acceptance.



Source: StratoFactory analysis (illustrative; parameters S26-S33 are not field-calibrated); benchmark arrows from E10-E12. October 2026.

Figure 5. ILLUSTRATIVE. First-pass acceptance against effective cost per successful outcome (agent attempt, plus mandatory human review, plus diagnosis and manual rework on failure), compared with a human-only baseline of USD 44.07 per task. The shaded band spans 50% to 68% acceptance. Arrows show benchmark accuracy shifts from added context reported in E10 to E12. They are benchmark accuracies, not production acceptance rates. StratoFactory analysis; parameters S26 to S33 are not field-calibrated.

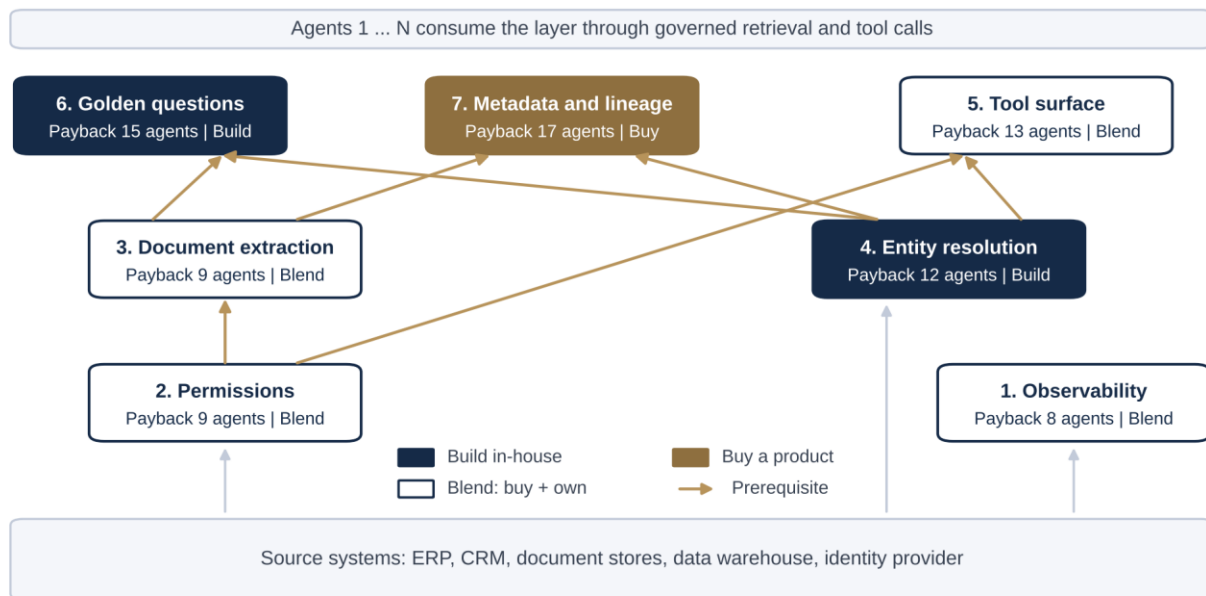
The agent breaks even with a person at 38% acceptance (Figure 5). At 50% acceptance, near the schema-only end of the semantic-layer benchmark, a successful outcome costs USD 36.42, a 17% saving. At 68%, near that benchmark's with-layer end, it costs USD 25.32, a 43% saving. **The link to the June report is in the human minutes.** Per 100 tasks, review plus rework falls from 2,050 minutes to 1,420. That is 31% less load on the kind of review queue the June report showed saturating.

Folding this into the crossover requires knowing how much of a benchmark uplift reaches production. If a quarter of it does, the layer's quality benefit is worth USD 50k per agent over its lifecycle, and the crossover falls from agent 11 to agent 9. If all of it does, the benefit is USD 200k and the crossover falls to agent 6. Gartner is far more optimistic. It predicts that by 2027 organizations that prioritize semantics will "increase their agentic AI accuracy by up to 80%" [E48]. That is a ceiling in an analyst prediction, well above anything the benchmarks show, and we have not used it. **The parameter that needs measuring is production first-pass acceptance with and without the layer, on the same task.** Until someone measures it, the quality case remains a hypothesis and the case we can defend rests on reuse alone.

6. Original Analysis: Build Order and Prerequisites

Section 5 showed that components pay back at different points, which suggests building the fastest payers first. But some components only work on top of others, so ranking by payback alone would give the wrong order. This section starts from the payback ranking and applies explicit prerequisites. The prerequisites are our engineering judgment, written down so they can be challenged.

Reference architecture of a shared context layer, with prerequisite edges and build order



Source: StratoFactory analysis, October 2026. Numbers in boxes give build order (payback order subject to prerequisites). Verdicts at a 15-agent estate.

Figure 6. Reference architecture of a shared context layer, with prerequisite edges. Numbers give the build order: payback order subject to prerequisites. Fill shows the sourcing verdict at a 15-agent estate: Build (develop in-house), Buy (license a product) or Blend (costs within 10%; buy the standard parts and own what is specific). StratoFactory analysis; the prerequisite graph is engineering judgment.

How to read Figure 6

Each box is one component. The number before its name is its place in the recommended build order; the line below gives its payback agent count and its sourcing verdict. Gold arrows run from a component to the components that depend on it, and gray arrows show which components draw directly on source systems. The fill color shows how to source each component, using the cost comparison in Section 8 at a planning estate of 15 agents:

- **Build** (dark fill): develop the component in-house. Building is more than 10% cheaper than buying over the lifecycle, and the component encodes knowledge specific to the business. Here: entity resolution and golden questions.
- **Buy** (gold fill): license a vendor product and integrate it. Buying is more than 10% cheaper than building. Here: metadata and lineage.



- **Blend** (white fill): build and buy cost within 10% of each other, so price does not decide. The usual answer is to buy the standard plumbing and own the parts specific to you, for example buying document parsing while keeping control of chunking rules and permission tags. Here: observability, permissions, document extraction and the tool surface.

A verdict says how to source a component, not whether to have one. Section 8's fourth verdict, Defer, applies to the layer as a whole when the estate is too small.

The prerequisites

- **Permissions before document extraction.** Chunks must carry their source permissions at ingestion. Retrofitting permissions onto an existing index means rebuilding it. Agents show "near-zero inherent confidentiality awareness" [E13], so the layer cannot delegate this to the model.
- **Document extraction and entity resolution before metadata and lineage.** Classification and lineage attach to extracted documents and resolved entities.
- **Entity resolution and permissions before the tool surface.** A cross-domain action needs to know which customer or asset it acts on, and whether the caller may act on it.
- **Document extraction and entity resolution before golden questions.** A golden question refers to real entities and needs a corpus to retrieve from.
- Observability, permissions and entity resolution have no prerequisites inside the layer.

The resulting sequence

At each step the model picks, from the components whose prerequisites are complete, the one that pays back soonest. Pure payback order would be Observability → Document extraction → Permissions → Entity resolution → Tool surface → Golden questions → Metadata and lineage. With prerequisites applied, the build order is **Observability** → **Permissions** → **Document extraction** → **Entity resolution** → **Tool surface** → **Golden questions** → **Metadata and lineage** (Figure 6). The two differ in one place: document extraction pays back marginally sooner than permissions (8.55 against 8.62 agents on the continuous measure), but building it first would mean rebuilding it later, so permissions move ahead.

The order holds up under sensitivity. Across the 20 one-at-a-time variants in Figure 4, observability and document extraction rank among the first three by payback in all 20, and permissions in 17. Metadata and lineage rank last in 16. The middle of the order (entity resolution, tool surface and golden questions) is less stable, and the model should not be taken to settle it.

Where we override the model

Golden questions pay back late on cost, at agent 15, because they are mostly task-specific and reuse little (30%). But they are also the only instrument that can measure whether the rest of the layer works, which is the quality parameter Section 5 could not calibrate. **We recommend building a minimal golden-question set alongside the first agent, ahead of its payback position.** Here our judgment overrides the cost model, which values only build reuse; the main value of golden questions is measurement.

Putting the order into practice

In practice the sequence falls into four phases:

- **With the first agent:** observability, permissions and a minimal golden-question set. They are cheap, reduce risk at once, and provide the measurements everything later depends on.
- **As the first domain fills up:** document extraction and entity resolution for that domain. These make cross-system questions answerable.
- **Once agents start acting, not only answering:** the tool surface, exposing governed actions through a common standard such as MCP [E29].
- **As the estate approaches the metadata payback, or earlier if bought:** metadata and lineage, preferably from a product that exports definitions in an open format [E25].

In a seeded layer the same phases repeat for each new domain, except that permissions and observability are built only once. The order also limits the damage if the program stops early. A layer abandoned after the first phase still leaves every agent with traces and enforced permissions, and the components most exposed to the long-context objection (Section 7) are the ones built last.

7. The Case Against

There are four serious objections. We state each as strongly as we can and test it in the model where possible. None defeats the case outright, but each wins somewhere: against the big-bang build, on curated retrieval, on short horizons, and on metadata.

1. "This is master data management again"

The objection. Enterprise-wide curated-data programs have a long and poor record. They run for years, consume budgets sized for transformation, and arrive describing a business that has since changed. A context layer is a master-data program with a new label and a more fashionable consumer. A peer-reviewed study of a master-data program found that data ownership shaped how the program developed and how roles evolved [E40]. Agents do nothing to settle who owns a customer record. (The widely repeated failure-rate figure for such programs could not be traced to a dated primary source, so we have not used it.)

The test. With no reliable overrun statistic, we used a stress test rather than a forecast: double the layer's build cost. The enterprise crossover moves from agent 11 to agent 22. A big-bang layer that overruns this badly needs twice the estate to justify itself.

What is different, and what we concede. Two things are different. A context layer has users from day one, so its quality can be tested with golden questions while it is being built. And it does not have to cover the whole enterprise at once. Seeded by domain, the same overrun moves the first domain's payback to agent 11 and later domains' to agent 6. The damage shows up within one domain rather than after years of enterprise-wide spending, but it is real: with domains of 5 agents, a seeded layer that overran this badly would not pay back at any estate size. **We concede the objection against the big-bang layer, and partly against the seeded one.** Anyone proposing an enterprise-wide context program should be held to the master-data record. A seeded layer survives it only if cost is tracked domain by domain and domains are large enough, at least 6 agents each, to absorb an overrun.

2. "Long context and agentic search will make curation obsolete"

The objection. If the model can read everything and search for itself, curated retrieval depreciates fast. The evidence is real. A Google DeepMind study found that "when resourced sufficiently, LC consistently outperforms RAG in terms of average performance", although "RAG's significantly lower cost remains a distinct advantage" [E19]. Anthropic describes agents that skip pre-processing in favor of "just in time" retrieval through tools [E20]. A 2026 PwC study found that grep "generally yields higher accuracy than vector retrieval" in its comparisons, and that overall scores "still depend strongly on which harness and tool-calling style is used" [E21]. On this evidence, whatever retrieval method a company builds today may not last.

The evidence on the other side. Accuracy degrades when relevant information sits in the middle of long inputs, "even for explicitly long-context models" [E17]. Across 18 models, "model performance varies significantly as input length changes, even on simple tasks" (Chroma, a vector-database vendor) [E18]. Spider 2.0's metadata was available, and agents still failed [E09]. Anthropic itself suggests that "the most effective agents might employ a hybrid strategy" [E20].

The test. Suppose agentic search halves the reuse value of document extraction, metadata and golden questions, and drives layer maintenance up to 40% a year as curated artifacts go stale faster. The crossover then rises to agent 17. Document extraction's payback moves to agent 27, metadata's to 51 and golden questions' to 72. **On curated retrieval, the counter-case wins.** It does not win on permissions or entity resolution. An agent that searches for itself still has to know what the user may see and which "Acme" a record refers to. If anything, agentic search makes those two components more valuable.

3. "Discount rates favor point solutions"

The objection. The agent ships this quarter; the layer ships next year. Value delivered sooner is worth more. A 2024 analyst survey of 644 organizations, as reported in the trade press, put the prototype-to-production cycle at around eight months on average, with less than half of AI projects reaching production [E32]. Adding a year of layer build on top is a hard sell.

The test. We simulated a team of fixed capacity (8 person-months a month) building agents over 5 years, with maintenance consuming capacity as the estate grows. Because both paths spend the same, the comparison reduces to discounted agent-months delivered, and the value per agent drops out. Without a layer, the first agent ships in month 3. With an up-front layer it ships in month 13, and at month 12 the layer path has 0 agents against 5. The layer path catches up in month 24 and ends the horizon with 34 agents against 20. Over 5 years the layer path wins at any annual discount rate below 85%, and the seeded path below 120%.

What we concede. The discount rate barely matters; the planning horizon does. At a 10% discount rate, the up-front layer delivers more discounted agent-months only if the horizon is at least 39 months. The seeded layer needs 30. At 25% the thresholds are 41 and 31 months. **For an organization planning on a horizon shorter than about three years, the counter-case wins against a layer built up front, and below two and a half years it wins against a seeded one too. A layer built up front also loses the first year whatever the horizon.** A seeded layer keeps pace with the no-layer path, with 5 agents at month 12, which is one reason we recommend it.

4. "Someone will sell you the layer"

The objection. Platform vendors are assembling exactly these components. AWS announced AWS Context, which maps relationships across existing data into a knowledge graph and has each call "inherit the calling user's IAM and Lake Formation permissions". It was listed as "coming soon" in June 2026 [E26]. Microsoft announced Fabric IQ as an intelligence layer for agents in November 2025 [E27]. Salesforce's Agent Context Engine is slated for general availability in December 2026 [E28]. The Open Semantic Interchange specification is available under Apache 2 [E25]. In October 2026 Gartner named "AI context platforms" as a category [E47], and Neo4j put a "Context Engine" into private preview [E51]. If the layer can be bought, the case for building it weakens.

The test. We gave each component a buy option with partial coverage, an integration cost and a subscription (Section 8). Taking the cheaper of build and buy for each component leaves the crossover at agent 11. Buying everything moves it to agent 12. **Buying changes who builds each component but not the threshold**, because the threshold comes from fixed costs and per-agent residual work, and both exist on the buy path too.

What we concede. Buying wins on metadata and lineage. Gartner analysts predicted in September 2026 that "Through 2028, heads of AI, data science and data management will attempt to build their own unstructured metadata solutions," "incurring costs more than 300% higher than they would if they used existing document and records solutions," "skills and practices." [E23] Entity-level products are another matter. As of October 2026, Fabric IQ's ontology component "remains in preview" [E27], AWS Context had not been released, and the newest entrants were in private preview or still announced. Today you can buy the plumbing, but not yet the meaning.

8. Build It, Buy It, or Defer It

The September report divided the choice into build, buy, blend and orchestrate; here, orchestrating agents across the existing estate without changing it is simply the no-layer path. On the data layer its rule was absolute: "At the data layer, own — always, and including the right to extract." [E35] This section applies that frame to each of the seven components. We compare three lifecycle costs at a planning estate of 15 agents: no layer, build the component, and buy it. Each includes the fixed cost (layer build or subscription and integration) plus every agent's residual work, so these totals are larger than the build-only figures in Table 1. A component is marked Buy or Build when one option is more than 10% cheaper, Blend when they are within 10%, and Defer when neither beats doing without. Coverage, integration and subscription inputs are analyst judgment (S37 to S40). We could not verify vendor pricing, which is not public.

Build: entity resolution and golden questions

Entity resolution costs USD 1,137k to build at 15 agents, against USD 1,419k to buy and USD 1,381k without a layer. We assume a bought platform would cover only about 30% of it. Entity semantics are where your business differs from every other, and the products are immature. It is also where the September report's rule applies most directly: the entity model is your data model. **Golden questions** are task-specific, and no platform writes them for you. On cost alone they only just beat doing without at 15 agents (USD 511k against USD 518k); we recommend them mainly because they are how you measure whether the layer works (Section 6).

Buy: metadata and lineage

Metadata and lineage cost USD 493k to buy against USD 560k to build. **Bought, they pay back at agent 14; built, at agent 17.** Catalog and classification products are mature. The Gartner prediction quoted in Section 7 points the same way [E23]. In fairness, our buy-side inputs for this component were informed by that prediction, so the verdict partly restates it. Metadata is the one component where the "someone will sell it to you" objection clearly wins.

Blend: observability, permissions, document extraction and the tool surface

For these four, build and buy are within 10% of each other at 15 agents, so the decision should rest on things the model does not price:

- **Observability** (USD 196k built, USD 195k bought). OpenTelemetry conventions are emerging [E30]. Use a standard so traces stay portable. Whether you host the store yourself matters less.
- **Permissions** (USD 671k built, USD 615k bought). Buy where your content sits mostly in one suite whose permission model the platform already understands. Build a mapping service where permissions span many systems. The July report's sovereignty layers apply here: where permissions and residency are a regulatory question, owning the mapping is part of owning the answer [E37].
- **Document extraction** (USD 527k built, USD 493k bought). Extraction services are commoditizing. Buy the parsing, own the chunking policy and its permission tags.
- **Tool surface** (USD 1,282k built, USD 1,196k bought). Vendor MCP servers cover standard actions on standard software [E29]. Actions specific to your processes still have to be built.

Defer: when the estate is small

If you cannot realistically expect 6 agents in a first domain, or about 9 across two, within the lifecycle, **defer the layer**. Build each agent's context the cheap way, but keep two disciplines that make a later layer possible: record source permissions on everything you index, and capture traces from day one. Both are cheap, and they are the first two steps in Figure 6, so a later layer starts with them in place.

Table 1 brings the component inventory, paybacks, prerequisites and verdicts together.

Component	Layer build (lifecycle)	Per agent: with / without layer	Reuse share (r)	Payback (agents)	Prerequisites	Verdict
1. Observability	USD 139k	USD 4k / USD 23k	0.85	8	None	Blend
2. Permissions	USD 500k	USD 11k / USD 69k	0.85	9	None	Blend
3. Document extraction	USD 222k	USD 20k / USD 46k	0.60	9	Permissions	Blend
4. Entity resolution	USD 833k	USD 20k / USD 91k	0.80	12	None	Build
5. Tool surface	USD 444k	USD 56k / USD 91k	0.45	13	Entity resolution, Permissions	Blend
6. Golden questions	USD 111k	USD 27k / USD 34k	0.30	15	Document extraction, Entity resolution	Build
7. Metadata and lineage	USD 389k	USD 11k / USD 34k	0.70	17	Document extraction, Entity resolution	Buy

Table 1. Component inventory in build order: lifecycle layer build cost, per-agent lifecycle increment with and without a layer, reuse share r (the share of one agent's own context work that later agents avoid because the layer exists), payback agent count, prerequisites, and verdict at a 15-agent estate. StratoFactory analysis. All costs and coefficients are analyst judgment (S03 to S23, S37 to S41), in USD 2026 over a 3-year lifecycle, at US labor costs.

Lock-in is the price of buying semantics

Buying the plumbing carries little risk, because standards make it portable. Buying the semantics is a different matter. A vendor's ontology or context engine becomes the place where your business's meaning is defined, and exit costs grow with every agent that depends on it. Vendors understand this well. Presenting Salesforce's Data 360 at Dreamforce '26, an executive called it "the investment that you can make today that accrues forward as you build out your entire agentic enterprise" [E28]. What accrues to the buyer as reuse accrues to the vendor as lock-in. The September report's rule, own the data layer "including the right to extract", is the right condition to put on any bought semantic component. In practice that means insisting on export in an open format such as the OSI specification [E25] before signing.

9. Predictions 2027–2029

These are StratoFactory's judgments, not sourced facts. Each has a date and a test that would show it wrong (Table 2). They follow from the analysis: the plumbing will be bought, the semantics will stay hard, and nobody will be able to say whether the layer improves task success until acceptance rates are measured.

Each follows from the analysis: P1 from the long-context evidence (Section 7); P2 from product status, with AWS Context, Fabric IQ's ontology and Salesforce's Agent Context Engine announced or in preview at our cut-off; P3 from Sections 2 and 5; P4 from the gap Section 5 could not fill; P5 from the OSI specification's stated plan for foundation governance; and P6 from the confidentiality results in Section 3 and a 2024 analyst survey, reported in the trade press, in which oversharing concerns led 40% of 132 IT leaders to delay AI assistant rollouts by three months or more [E15].

#	Prediction	What would prove it wrong
P1	Through 2029, published reference architectures for enterprise agents from Microsoft, AWS, Google and Salesforce keep a governed retrieval or semantic layer for enterprise-scale corpora.	Any of the four publishes, before 31 Dec 2029, a reference architecture recommending no retrieval index or semantic layer for corpora above one million documents.
P2	By 31 Dec 2028, at least two of AWS Context, the Microsoft Fabric IQ ontology and Salesforce's Agent Context Engine (or their successors) are generally available with published pricing.	Fewer than two are generally available with public list pricing on 31 Dec 2028.
P3	By 31 Dec 2027, no independent survey with a published methodology finds that a majority of large enterprises run a shared context layer used by five or more agents.	A non-vendor survey with published sampling and margin of error reports 50% or more, by 31 Dec 2027.
P4	By 31 Dec 2027, at least one major agent platform reports first-pass acceptance (or an equivalent accepted-without-rework rate) as a standard production metric.	No such metric in the generally available product documentation of Microsoft, Salesforce, ServiceNow, Google or AWS by 31 Dec 2027.
P5	By 31 Dec 2028, the Open Semantic Interchange specification is governed by a neutral foundation.	The specification is still hosted and governed by its vendor sponsors on 31 Dec 2028.
P6	By 31 Dec 2028, at least one Fortune 500 or equivalent enterprise publicly attributes a data exposure to an AI assistant or agent retrieving content beyond the requesting user's permissions.	No such attribution in a filing, regulator notice or company statement by 31 Dec 2028.

Table 2. StratoFactory predictions for 2027 to 2029, with falsifiers. These are judgments, not sourced facts.

10. Limitations and What Would Change Our View

The readiness evidence is not independent (Section 2). The benchmark evidence is better but has its own limits. Spider 2.0 is peer-reviewed (ICLR 2025), but of the three studies that measure the gain from curated context, one is vendor-authored, one tests a single schema with a 2023 model, and the third is a preprint with no stated affiliation. None measures acceptance in production.

Component costs are analyst judgment. No published benchmark gives the engineering effort for entity resolution, permission propagation or the other components. The crossover is a direct function of the ratio of layer build cost to per-agent slice cost. The tornado shows the crossover moving from agent 7 to agent 18 across a plausible range of that ratio. A reader with their own estimates should re-run the script rather than adopt our number. The quality coupling is illustrative throughout.

We could not count agents per enterprise. No independent source reports how many production agents a typical large enterprise runs. The crossover therefore has to be compared with the reader's own pipeline, not with a market average.

Retrieval constraints. Two sources we wanted, the MIT NANDA "GenAI Divide" report and the TheAgentCompany benchmark, could not be opened, so claims based on them were cut.

Market. The model is built on US data and for the US market: the labor rate is a US median wage, and most surveys cited are US-led. Where labor costs, vendor prices or regulation differ, the dollar figures and some build-or-buy verdicts will differ; the break-even agent counts will not.

Scope. The model does not price security incidents avoided, regulatory exposure, or the option value of a layer that makes new agents faster to build. Each would lower the crossover. It also does not price organizational failure, which would raise it.

What would change our view. If a major vendor's reference architecture drops governed retrieval for enterprise-scale corpora (P1 fails), the layer narrows to permissions and entity resolution. Production evidence that first-pass acceptance does not improve with curated context would reduce the case to reuse alone. Entity-level products at general availability with transparent pricing would move entity resolution from build to blend. If the layer cost three agents' worth of context work instead of 6.1, the enterprise crossover would fall to agent 6; it would take a ratio of about 1.2 to reach the September report's second-application result. A horizon shorter than two and a half years makes the layer the wrong choice whatever else holds, and one shorter than about three years rules out building it up front.

A. Appendix: Methodology, Assumptions, Glossary and Sources

A.1 Methodology

Corpus. We read the StratoFactory research index and the May, June, July and September 2026 reports and the W40/2026 weekly report. This report cites the June report (verification tax, review queues), the July report (sovereignty layers), the September report (portfolio crossover; build, buy, blend, orchestrate) and the May report (token price) [E35–E39].

Evidence rule and verification. A figure, forecast or quotation was admitted only after its source document had been opened and the passage read, never from a search snippet. The fetch tool quotes at most 125 characters at a time, so long quotations appear in verbatim parts. In total 60 searches were run and 70 pages opened, producing 72 evidence rows. Before writing, every cited source was re-opened and each figure, attribution and quotation re-checked; the pass produced 6 corrections. Sources that could not be opened were cut (evidence table, §3).

Freshness. A sweep covered the 90 days to the first cut-off (20 July to 18 October 2026). A second sweep on 22 October 2026, aimed at consultancies, analyst firms and academic sources, found five relevant items the first had missed and several published in the final days. These are recorded in the evidence table (§9). They changed the evidence in Sections 1 to 4, 7, 9 and 10 but no modeled number.

Labeling and use of sources. The body distinguishes measured, modeled, vendor-reported, analyst-predicted and illustrative figures at every point of use. Third-party figures are limited to a few per source and are not reproduced as tables.

Model. A deterministic Python script (`context_layer_model.py`) writes every modeled number to a canonical JSON file, and the prose takes its numbers from that file. A mechanical sweep compared every number in the rendered PDF with the model output and the evidence table. The script checks on every run that each threshold moves in the stated direction. The capacity simulation uses monthly steps, unlimited agent demand, and maintenance that consumes capacity as the estate grows.

Source conflicts. Where sources disagree, or one source contradicts itself, both values are reported and neither is averaged or silently chosen. The conflicts are logged with their handling in the evidence table (§2). The ones that bear on the argument are discussed where they arise: the two sample sizes in one Gartner release (Section 2), the spread of context-uplift results (Sections 3 and 5), the analyst accuracy prediction set against benchmark gains (Section 5) and long context against retrieval (Section 7).

A.2 Key assumptions

All assumptions are StratoFactory analyst judgment unless a basis is given. Money is in USD 2026. **Market basis:** the inputs are set for the US market, with labor costed at the US median software-developer wage. Readers in other markets should substitute local labor costs and vendor prices; the break-even agent counts are unaffected by the labor rate, but dollar figures and build-or-buy verdicts are not. Table 3 groups them; the full registry, with every parameter, unit and source note under its S-label, is written by `context_layer_model.py` to `model_outputs.json`, which accompanies this report.

Group	S-labels	Base values	Basis
Cost of effort	S01, S02	USD 15,864 per person-month (median wage × 1.4 / 12)	Derived from BLS wage data [E33]
Layer build cost	S03 to S09	4 to 30 person-months per component; 95 in total	Analyst judgment
Per-agent slice cost	S10 to S16	1 to 4 person-months per component; 17 in total	Analyst judgment
Reuse	S17 to S24	Coefficient r 0.30 to 0.85 by component; incidental reuse 0.10	Judgment; three coefficients anchored to the September report [E35]
Lifecycle and maintenance	S25a to S25c	3 years; maintenance 20% a year (slices), 25% (layer)	Analyst judgment
Seeded layer	S46, S47	Domain-scoped components need 0.30 to 0.40 of the enterprise build per domain; permissions and observability 1.00; 5 agents per domain	Analyst judgment
Long-context scenario	S48	Reuse of document extraction, metadata and golden questions halved; layer maintenance 40% a year	Stress test
Delivery simulation	S34 to S36	Capacity 8 person-months a month; horizon 60 months; discount rates 10% and 25% for horizon tests	Analyst judgment
Quality coupling	S26 to S33	Manual task 25 min; review 3 min; diagnosis 10 min; 60,000 tokens per task; acceptance 0.50 without and 0.68 with the layer; 25% of uplift realized	Illustrative; not field-calibrated [E11] [E36] [E39]
Buy option	S37 to S41	Coverage 0.10 to 0.85; integration 1 to 6 person-months; subscription USD 20,000 to 120,000 a year; verdicts at 15 agents	Judgment; no public pricing verified

Table 3. Key model assumptions, grouped by S-label range. StratoFactory analyst judgment except where a basis is cited; US market basis. The full registry is written by `context_layer_model.py` to `model_outputs.json`.

A.3 Glossary

- **Context layer.** Shared, governed infrastructure that gives agents access to an organization's entities, documents, permissions, definitions and actions.
- **Slice.** The context work one agent team does on one component for its own agent alone.
- **Reuse share (r).** The share of a slice that later agents avoid because the layer exists (also called the reuse coefficient).
- **Incidental reuse (i).** The share of a slice avoided without a layer, by copying earlier teams' work.
- **Crossover (N^*).** The smallest number of agents at which cumulative cost with the layer is at or below cumulative cost without it. Computed for one component on its own, it is that component's payback agent count.
- **Seeded layer.** A layer built one business domain at a time, with permissions and observability built once for the enterprise.
- **First-pass acceptance.** The share of agent outputs accepted without rework.
- **Verification tax.** In the June 2026 report's definition, the additional human effort needed to read, test, debug and gain confidence in AI-produced work.
- **Golden questions.** A curated evaluation set of questions with known correct answers, used to test retrieval and agent quality.

- **MCP.** Model Context Protocol, an open protocol for connecting models to tools and data, now hosted by the Linux Foundation's Agentic AI Foundation.
- **OSI.** Open Semantic Interchange, a vendor-neutral semantic model specification published under Apache 2.

A.4 Sources

Every row below was opened and read in this session. Rows from the first research pass were retrieved and re-verified on 18 October 2026; rows E43 to E53 on 22 October 2026. Where the text describes a source generically (for example, "a global consultancy" or "an analyst survey reported in the trade press"), the full citation appears here under its evidence ID. Full extraction notes, confidence ratings and contradicting evidence are in the accompanying evidence-table.md.

ID	Source	Type	URL (link)
E01	Komprise, survey of 500 technical decision makers (press release), 15 Oct 2026	Vendor survey	globenewswire.com/news-release/2026/10/15/3381565/0/en/komprise-survey-of-unstructured-data-and-ai-reveals-wide-gap-between-perceived-ai-maturity-and-ai-results.html
E02	The Modern Data Company, Modern Data Report (third Modern Data Survey), 5 Oct 2026; interim release 19 Aug 2026	Vendor survey	cioandleader.com/57-of-organisations-are-piloting-or-running-ai-agents-but-only-8-say-their-data-is-production-ready-the-modern-data-company-report/
E03	Gartner Press Release, "Lack of AI-Ready Data Puts AI Projects at Risk", 26 Feb 2025	Analyst survey	gartner.com/en/newsroom/press-releases/2025-02-26-lack-of-ai-ready-data-puts-ai-projects-at-risk
E04	Gartner Press Release, "Gartner Survey Finds Only 22% of Organizations Have Successfully Scaled AI Across Multiple Business Units", 1 Sep 2026	Analyst survey	gartner.com/en/newsroom/press-releases/gartner-survey-finds-only-22-percent-of-organizations-have-successfully-scaled-ai-across-multiple-business-units
E05	McKinsey, "The state of AI in 2025: Agents, innovation, and transformation", 5 Nov 2025	Consultancy research	mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-2025
E06	US Census Bureau (Grundy, Breaux, Khatiwoda), "Large Firms With at Least 20 Employees Biggest AI Users", 26 May 2026	Primary research	census.gov/library/stories/2026/05/ai-use-businesses.html
E07	Gartner Press Release, "Gartner Predicts Over 40% of Agentic AI Projects Will Be Canceled by End of 2027", 25 Jun 2025	Analyst prediction	gartner.com/en/newsroom/press-releases/2025-06-25-gartner-predicts-over-40-percent-of-agentic-ai-projects-will-be-canceled-by-end-of-2027
E08	Stanford HAI, The 2026 AI Index Report (web page, OSWorld chart)	Academic	hai.stanford.edu/ai-index/2026-ai-index-report
E09	Lei et al., "Spider 2.0: Evaluating Language Models on Real-World Enterprise Text-to-SQL Workflows", ICLR 2025 (arXiv 2411.07763v2, 17 Mar 2025)	Academic	iclr.cc/virtual/2025/oral/31826
E10	Sequeda, Allemang, Jacob, "A Benchmark to Understand the Role of Knowledge Graphs on Large Language Model's Accuracy for Question Answering on Enterprise SQL Databases", arXiv 2311.07509, 13 Nov 2023	Academic preprint	arxiv.org/abs/2311.07509v1
E11	Rumiantsau and Fokeev (Cube), "Semantic Layers for Reliable LLM-Powered Data Analytics", arXiv 2604.25149, 2026	Vendor technical document	arxiv.org/pdf/2604.25149
E12	Mandarapu and Kunkunuru, "Knowledge Graphs as the Missing Data Layer for LLM-Based Industrial Asset Operations", arXiv 2605.26874 v3, 29 Jun 2026	Academic preprint	arxiv.org/abs/2605.26874
E13	Huang et al. (Salesforce AI Research), "CRMarena-Pro", arXiv 2505.18878, 24 May 2025	Vendor research lab, academic preprint	arxiv.org/html/2505.18878v1



ID	Source	Type	URL (link)
E14	Sierra, "T-Bench: Benchmarking AI agents for the real-world", 20 Jun 2024	Vendor technical document	sierra.ai/blog/benchmarking-ai-agents
E15	Computerworld (M. Finnegan), Gartner survey of Microsoft 365 Copilot rollouts, 27 Sep 2024	Analyst survey via journalism	computerworld.com/article/3542000/microsoft-365-copilot-rollouts-slowed-by-data-security-roi-concerns.html
E16	Bhatt et al. (Microsoft Research), "Enterprise AI Must Enforce Participant-Aware Access Control", arXiv 2509.14608	Vendor research lab, academic preprint	microsoft.com/en-us/research/?p=1150688
E17	Liu et al., "Lost in the Middle: How Language Models Use Long Contexts", TACL 12:157–173, 2024	Academic	aclanthology.org/2024.tacl-1.9/
E18	Hong, Troynikov, Huber (Chroma), "Context Rot: How Increasing Input Tokens Impacts LLM Performance", 14 Jul 2025	Vendor technical document	trychroma.com/research/context-rot
E19	Li, Li, Zhang, Mei, Bendersky, "Retrieval Augmented Generation or Long-Context LLMs? A Comprehensive Study and Hybrid Approach", arXiv 2407.16833, 23 Jul 2024	Academic / vendor research lab	arxiv.org/html/2407.16833v1
E20	Anthropic Engineering, "Effective context engineering for AI agents", 29 Sep 2025	Vendor technical document	anthropic.com/engineering/effective-context-engineering-for-ai-agents
E21	Sen et al. (PwC), "Is Grep All You Need? How Agent Harnesses Reshape Agentic Search", arXiv 2605.15184, 14 May 2026	Consultancy research	arxiv.org/abs/2605.15184
E23	Gartner Press Release, "Gartner Data & Analytics Summit 2026 Mumbai: Day 2 Highlights", 22 Sep 2026	Analyst statement	gartner.com/en/newsroom/press-releases/2026-09-22-gartner-data-and-analytics-summit-2026-india-day-2-highlights
E25	Snowflake, Open Semantic Interchange press release, 23 Sep 2025; "Open Semantic Interchange specs finalized" blog, 27 Jan 2026	Vendor technical document	snowflake.com/en/news/press-releases/snowflake-salesforce-dbt-labs-and-more-revolutionize-data-readiness-for-ai-with-open-semantic-interchange-initiative snowflake.com/en/blog/open-semantic-interchanges-specs-finalized/
E26	AWS (M.-L. Tomsen Bukovec), "Context intelligence for your data and AI agents at scale", 17 Jun 2026	Vendor technical document	aws.amazon.com/blogs/machine-learning/context-intelligence-for-your-data-and-ai-agents-at-scale/
E27	Cloud Wars, Microsoft Work IQ, Fabric IQ and Foundry IQ, 19 Nov 2025; Directions on Microsoft, Fabric IQ roadmap, 1 Oct 2026	Journalism	cloudwars.com/ai/microsoft-debuts-work-iq-fabric-iq-and-foundry-iq-a-unified-intelligence-layer-for-the-ai-powered-enterprise/ directionsonmicrosoft.com/roadmaps/ref/fabric-iq-roadmap/
E28	Salesforce Ben (T. Combridge), "Top Salesforce Data Cloud updates from Dreamforce '26", 17 Sep 2026	Journalism	salesforceben.com/top-salesforce-data-cloud-updates-from-dreamforce-26/
E29	Linux Foundation, formation of the Agentic AI Foundation, 9 Dec 2025	Primary	linuxfoundation.org/press/linux-foundation-announces-the-formation-of-the-agentic-ai-foundation
E30	OpenTelemetry, "OpenTelemetry for Generative AI" (blog)	Standards body	opentelemetry.io/blog/2024/otel-generative-ai/
E32	CIO Dive, Gartner survey of 644 organizations, 17 May 2024	Analyst survey via journalism	ciodive.com/news/generative-ai-adoption-barrier-project-value/716504/
E33	US Bureau of Labor Statistics, Occupational Outlook Handbook: Software Developers (May 2025 wage data)	Primary	bls.gov/ooh/computer-and-information-technology/software-developers.htm
E34	BVResources, "Damodaran posts his first data update for 2026", 21 Jan 2026	Journalism reporting academic data	bvresources.com/articles/bvwire/damodaran-posts-his-first-data-update-for-2026
E35	StratoFactory, "Enterprise Software After AI: Build, Buy, or Let AI Build It?", Sep 2026	StratoFactory	stratofactory.com/wp-content/uploads/2026/09/Enterprise-Software-After-AI-StratoFactory-2026.pdf
E36	StratoFactory, "AI in the SDLC: The Productivity Paradox", Jun 2026	StratoFactory	stratofactory.com/wp-content/uploads/2026/06/AI_in_the_SDLC_Research_by_Stratofactory.pdf
E37	StratoFactory, "AI Sovereignty for the European Enterprise", Jul 2026	StratoFactory	stratofactory.com/wp-content/uploads/2026/07/AI_Sovereignty_for_the_European_Enterprise.pdf



ID	Source	Type	URL (link)
E39	StratoFactory, "AI Token Cost Forecast 2026–2030", May 2026	StratoFactory	stratofactory.com/wp-content/uploads/2026/05/AI_Token_Cost_Forecast_2026-2030_Market_Research_by_Stratofactory.pdf
E40	Vilminko-Heikkinen and Pekkola, "Changes in roles, responsibilities and ownership in organizing master data management", Int. J. Information Management 47:76–87, 2019	Academic	ideas.repec.org/a/eee/ininma/v47y2019icp76-87.html
E43	McKinsey, "The State of AI: Global Survey 2026", 25 Aug 2026	Consultancy research	mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai
E44	McKinsey Technology, "Building the foundations for agentic AI at scale", Apr 2026	Consultancy research	mckinsey.com/~media/mckinsey/business%20functions/technology/our%20insights/building%20the%20foundations%20for%20agentic%20ai%20at%20scale/building-the-foundations-for-agentic-ai-at-scale_vf.pdf
E45	Deloitte, agentic AI readiness survey (press release), 12 Aug 2026	Consultancy research	deloitte.com/us/en/about/press-room/deloitte-survey-examines-ai-readiness-agentic-ai-success.html
E46	BCG, "The Formula for Winning Enterprise AI Adoption in 2026", 30 Sep 2026	Consultancy research	bcg.com/publications/2026/the-formula-for-agentic-ai-value
E47	Gartner, Inc., "Top Strategic Technology Trends for 2027" (article), 20 Oct 2026; Network World (A. Bednarz), 20 Oct 2026 (Alvarez quotation, E47b)	Analyst prediction	gartner.com/en/articles/top-technology-trends-2027 networkworld.com/article/4238030/gartners-top-tech-trends-for-2027-expand-and-constrain-ai-capabilities.html
E48	Gartner Press Release, "Gartner Says Lack of Semantics Causes Inaccurate AI Agents and Wasted Spending", 11 May 2026	Analyst prediction	gartner.com/en/newsroom/press-releases/2026-05-11-gartner-says-lack-of-semantics-causes-inaccurate-artificial-intelligence-agents-and-wasted-spending
E49	Tomitsuka et al., "Argo-Bench: Evaluating Data Agents on Enterprise-Scale Workflows", arXiv 2610.02122, 1 Oct 2026	Academic preprint	arxiv.org/abs/2610.02122
E50	Gruenbaum et al., "Era by Eon: Benchmarking Enterprise Agents on Hidden Knowledge", arXiv 2609.30055, 24 Sep 2026	Academic preprint	arxiv.org/abs/2609.30055
E51	Neo4j, Context Engine launch (press release), 20 Oct 2026	Vendor technical document	financialcontent.com/article/bizwire-2026-10-20-neo4j-context-engine-launches-to-create-ai-knowledge-layer-and-enhance-agent-performance
E53	Stanford HAI, AI Index Report 2026, Chapter 4: Economy	Academic	hai.stanford.edu/assets/files/ai_index_report_2026_chapter_4_economy.pdf

Table 4. Sources cited, keyed to evidence-table IDs. Every source was opened and read, and re-verified on 18 October 2026 (rows E43 to E53 on 22 October 2026).

B. Disclaimer and Notices

B.1 Author, editorial responsibility and how this report was produced

This briefing is independent market research published by StratoFactory.com. Piotr Jurowiec conceived the report, defined its thesis, research questions, structure and evidence standards, directed the analysis, made the editorial judgments, and reviewed and approved the final text. He is its author and takes editorial responsibility for its content. AI tools supported the work under his direction, helping with source retrieval, modeling code, drafting and consistency checks. Every cited figure and quotation was verified against its source (A.1). The briefing is not the product of any vendor, consultancy or advisory engagement.

B.2 Modeled figures are StratoFactory analysis

The cost model and everything derived from it (crossover counts, paybacks, build order, delivery simulation, buy comparison and quality coupling) are StratoFactory analysis, not measured market data. Their inputs are analyst judgment, stated in Appendix A.2 and the full model registry, and the quality coupling is illustrative. They are built on US data and for the US market, and may differ materially elsewhere. They show direction and sensitivity and must not be quoted as measured costs or outcomes for any organization.

B.3 Forward-looking statements and data currency

Predictions, scenarios and statements about future products, prices or adoption are forward-looking and rest on assumptions that may prove wrong. Announced or preview products may change, be delayed or be withdrawn. Actual outcomes may differ materially. The evidence cut-off is 22 October 2026. This document is not updated automatically; treat it as historical context after six to nine months and check current sources before relying on any figure.

B.4 Reliance on third-party, vendor-reported and analyst content

This briefing relies on third-party sources, including vendor surveys and consultancy research by firms that sell related products or services. Vendor-reported figures are labeled where used, and missing methodology is noted. We have not audited any third-party data. Analyst predictions and statements, including Gartner's, are the opinions of the firms and individuals concerned, not statements of fact, and may change without notice. Citing them implies no endorsement of this briefing, StratoFactory.com or any product. Readers who rely on a figure should consult the original source.

B.5 Quotations, data and graphics

Quotations from third-party sources are short, verbatim (where split, each part is verbatim) and attributed, and are used for commentary, criticism and review. Third-party figures are cited as facts with their source, a few per source, and are not reproduced as tables or in the original layout. All figures, tables and diagrams are original StratoFactory work. Where they plot third-party data points (Figure 5), the caption gives the source. No third-party chart, graphic, logo or proprietary methodology is reproduced. Sources are listed with title, publisher, date and URL in the source list (Table 4), and third-party material remains the property of its owners.

B.6 Third-party names and trademarks

Company, product and service names in this briefing (among them Amazon Web Services, Anthropic, BCG, Chroma, Cube, Deloitte, Gartner, Google, Komprise, McKinsey, Microsoft, Neo4j, PwC, Salesforce, Snowflake and The Modern Data Company, and products such as AWS Context, Microsoft 365 Copilot, Fabric IQ, Data 360 and Agent Context Engine) are trademarks or registered trademarks of their owners. GARTNER is a registered trademark of Gartner, Inc. Names are used nominatively for identification and commentary only, and no association, endorsement or sponsorship is implied.

B.7 Not investment, procurement or legal advice

Nothing here is investment, procurement or legal advice, or a recommendation to buy, sell or contract for any product, service or security. Build, buy and defer verdicts are illustrative model outputs, not recommendations for a particular organization.

B.8 No warranty; limitation of liability

This document is provided "as is", without warranty of any kind, express or implied, including as to accuracy, completeness or fitness for purpose. To the maximum extent permitted by law, the author and StratoFactory.com disclaim all liability for any loss arising from its use.

B.9 Independence and conflicts of interest

The author and StratoFactory.com received no payment, sponsorship or other consideration from any organization named in this briefing, and none of them commissioned, reviewed or approved it.

B.10 Sharing, attribution and corrections

Readers may share this briefing and quote brief portions with attribution to "Making AI Agents See Your Business (The Context Layer) — Market Research Briefing, StratoFactory.com, October 2026", without altering figures or their labels (modeled, vendor-reported, illustrative). Third-party material should be cited to its original source. Corrections are welcome through the publisher's website.

© 2026 StratoFactory.com / Piotr Jurowiec. All rights reserved, except third-party material quoted or cited, which remains the property of its owners. Modeled figures are StratoFactory analysis; third-party figures are sourced as listed in A.4 and in evidence-table.md. Comments and corrections are welcome through the publisher's website.