Sentra

Sentra technical report | November 2026

The Management Consultant That Never Leaves

Memory as Learning for a Whole Company

Ashwin Gopinath, Taran Singhania

Sentra | ashwin@sentra.app

Abstract

The essay “Services: The New Software” argues that AI that sells the work has a far larger market than AI that sells tools. We apply this services-as-software thesis to one role: the management and strategy consultant. Each consulting engagement starts with a diagnosis, which finds what is broken that nobody asked about. An AI that does this every day needs memory as learning, not only memory as search. A child who touches a hot stove once keeps the lesson, not the kitchen, and the lesson acts before anyone asks. Sentra keeps such lessons for a whole company, for its growth, efficiency and compliance, and in this study people wrote them.

Sentra applies its lessons as standing checks to every record. It raises what the checks find as situations, and it sends each situation about a customer to the people who work on that customer. We tested this diagnosis on two simulated companies with known answer keys. On the company where we wrote the checks, with its answer key available, Sentra raised 483 of 485 planted problems (99.6%, by two AI judges). On a second company, held out from development, the checks found 491 of 501 (98%) on their first run. An exact-match program scored this result, and we finished it with both companies’ results in view, so the test is not fully blind.

Claude Opus 5.5 read the same records but could not run code, and its briefs raised 8% of the planted problems (24.5% with a checklist). But on 25 judgments, which are conclusions about the planted scenarios, it came close: 68%, against 76% for Sentra. As memory for AI agents, Sentra helped them find more at about a third less agent cost. This figure does not include Sentra’s own cost. With Sentra, Opus did not answer direct questions better. The companies are simulated, and we did not test the rest of a consultant’s work: to learn lessons from experience, recommend changes and make them.

The authors work at Sentra. Eon generated the two companies, their records and the truth behind them with its Era platform. The answer keys, Sentra’s checks, the agent runs and the scoring are ours.

Executive summary

Sentra is paid for the work, not for a tool. This is the thesis of the essay “Services: The New Software” (1), applied to one role: the management and strategy consultant. A consultant’s first job is a diagnosis: how the company really runs, and what is broken that nobody asked about. This paper tests whether a memory that keeps lessons can make this diagnosis from every record, for the whole company, with nobody asking. In this study, people wrote the lessons.

From a one-day snapshot of all the records of Wisteria Group, a simulated payments company with 1,200 people, Sentra raised these problems with nobody asking. We wrote and refined its checks on Wisteria, with the answer key available.

Revenue cuts. Sentra found every cut in a customer’s annual recurring revenue (ARR) in the past year: 37 of 37, $2,044,169 to the dollar. For 31 of them, its keyword search of the deals, tickets and calls near the cut found no reason.

Promises that failed. In 117 cases, support told a customer that “the fix lands next week”, and customers reported the same failure more than a week later. In all, 218 customers reported these failures in 1,070 later tickets, and 247 of them are still open. All 117 cases were real, and they are 85% of the 137 broken promises in Eon’s truth.

Systems that disagree. Sentra found all 696 field conflicts between billing and Salesforce, and it flagged 110 customers ($4.7M of ARR) for a person to decide.

What people leave behind. Sentra found all 181 people who left, and the 79 open tickets and 29 help articles still in their names. Its checks listed 39 of the 40 app licences and admin roles that they still hold, and its reconciliation found all 40.

A model that stays current. On day one, the key fields of Sentra’s model of the company, which a language model built from the records, were 96% to 100% correct. After three months of change, it needed two fixes. Then its fields were 95% to 100% correct, in a check that was not blind.

Who hears. After one routing fix, 99.6% of the people that Eon’s truth and Sentra’s rules name received the customer situations that reached anyone. Of what people received, 58% came from other teams’ checks. Some situations about no customer, such as the 11 supplier renewals and 139 rows about what people who left still hold, reached no one.

Where AI agents stand. Claude Opus 5.5 wrote briefs from the same records and raised 8% of the 485 planted problems (24.5% with a checklist). But it came close to Sentra on 25 judgments about the planted scenarios (68% against 76%). When we asked it directly for the slipping deals, it found 183 of the 189. The gap is mostly in coverage, not in reasoning: an assistant answers the question that someone asks, but a consultant knows which questions to ask. Part of the gap is format: even with Sentra’s complete list, Opus’s briefs carried 22% of the planted problems.

Agents with Sentra. With Sentra as its memory, Opus raised nearly three times as many planted problems, at 35% lower agent cost. This figure does not include Sentra’s own cost, which we did not measure. The smaller Haiku with Sentra found about twice as many problems as Opus alone. Opus’s answers to direct questions did not improve.

A second company. On a second simulated company, held out from development, the checks found 98% of the planted problems on their first run. That company came from the same generator, so records in another format are untested. An exact-match program scored this result. We finished that program with both companies' results in view, so the test is not fully blind. We did not test a real company, agents that write their own code, lessons that the system learns from experience, or the rest of a consultant's work.

1 Introduction

The first large uses of generative AI help individuals. The standard test for coding agents is whether they fix real bugs in real software projects (2). In studies, AI assistants made support staff and developers more productive (3, 4). In millions of conversations with one language model, software development and writing were nearly half of all use (5).

Paid for the work. The essay "Services: The New Software" estimates that companies spend six dollars on services for every dollar on software (1). It argues that AI that sells the work has a far larger market than AI that sells tools: "A copilot sells the tool. An autopilot sells the work." We call this the services-as-software thesis. The essay separates intelligence, where "the rules are complex but they are rules", from judgement, which "requires experience and taste". It calls management consulting, a $300–400 billion market, "mostly judgement", and asks whether AI can do its intelligence layer while people keep the judgement. This paper places the diagnosis in that layer and tests it for the management and strategy consultant (from here on, "the consultant").

What nobody asks about. Every company cares about *growth* (sell more and keep it), *efficiency* (no money lost, no work done twice) and *compliance* (do things right). The failures that threaten these rarely sit inside one person's job. They sit between systems that disagree and teams that each hold part of a story. Support promises a customer that a fix lands next week, but engineering does not ship it. The account owner, whose customer renews in three weeks, never hears. Nobody asks about problems like this, because nobody knows to ask. The consultant's diagnosis must find them.

Memory as search. Most AI memory is storage and retrieval. Memory systems for language models return what one user or agent saw before (6, 7), and retrieval systems answer the question that they get (8, 9). This works when someone knows what to ask, but it fails when the answer must come from several systems. In Eon's tests, nine language models averaged 92.6% on simple questions about the records of one system (10, 11). They averaged 39.6% on questions that cross systems, and 3.7% on questions that take several steps. Memory like this works as an analyst does: it answers the question that someone asks.

Memory as learning. A child who touches a hot stove once keeps the lesson that hot surfaces burn, not the date or the kitchen. This lesson is small, applies to stoves that the child never saw, and acts before anyone asks. Psychology separates memory of events from general knowledge (12), and models of the brain describe how general structure comes slowly from many experiences (13). Organizations that work well put lessons into routines and rules (14, 15), and some AI agents now keep lessons, not only records (16, 17). Good consultants work the same way: lessons from many companies tell them where to look.

Memory for the organization. For a whole company, memory must keep three things: what is true now, which system to trust when systems disagree, and which patterns mean trouble. It must act on them for everyone, with nobody asking. A personal agent serves one person's tasks and sees what that person sees.

A company's memory reads every system and serves the whole company's growth, efficiency and compliance. A consultant also works for the company, not for one employee. This paper argues for this difference but does not test it.

Consultants with AI. In a field experiment at Boston Consulting Group, 758 consultants did tasks with or without GPT-4 (18). With GPT-4, they did more tasks, faster and better, if the task suited the model, but were less often correct if it did not. In our view, the team still samples records and interviews people for a few weeks, reports to the leaders and leaves. In a randomized trial, some Indian textile plants got four months of help to adopt better management practices. Against plants that got only a one-month diagnosis, their productivity rose by 17% in the first year (19). The authors concluded that missing information was the main reason why firms did not use these practices.

Memory that works as a consultant. A company memory that works as a consultant changes the form of the work, not only its speed:

- It reads every record, not a sample, so its lists can be complete for the kinds of problem that it has lessons for.
- It looks every day, not once, so it can find problems early and keep its findings current.
- It tells each person whom a problem affects, in any team, so the people who can act hear.
- It keeps its lessons, so they do not leave with people, and the next company can use them.
- People and AI agents can ask it at any time.

Box 1 shows which of these gains this study measured.

What we test. Sentra, the system that we build, keeps lessons as standing checks, and in this study people wrote the checks. Its aim is to analyse the company all the time, find what must change and, in time, help to make the change. This study tests only the diagnosis, on two simulated companies that Eon, an independent company, generated with the truth behind every planted problem. We developed Sentra's checks on the first company and compared Sentra there with AI agents (language models that do a task on their own). Then we ran the checks on the second company. We also measured who heard about each finding, whether Sentra helps other agents as their memory, and whether its model stays current.

2 How Sentra works

Sentra works in four stages. The first two learn how the company runs, the third finds what is broken, and the fourth tells the people who can act.

1. **Memory.** Sentra's product reads every record, keeps a copy and extracts dated facts from it. Each fact names who and what the record mentions and how they connect, and it cites its source. Google's Gemini 3.5 Flash extracts the facts, and Gemini 3.5 Flash-Lite scores how much each fact matters. OpenAI's text-embedding-3-large indexes the facts for search.
2. **Model.** A language model, Gemini 3.5 Flash, combines the facts about each customer, deal and person into their current state, one entity at a time. It also reads up to four of the entity's own records from its systems of record, such as its Salesforce account and billing records. Every field cites the

facts that it came from. The language model's instructions name the company and say which system to trust for each field. Agents query Sentra's model when Sentra is their memory, and section 7 compares it with Eon's truth.

3. **Reconcile and check.** For each customer, Sentra records what each system says, where they disagree, and which system to trust for each field, by a policy fixed in advance. Standing checks then run over three things: the structured fields of the stored records, the reconciliation, and the model's record of who left. The fields include a deal's close-date history, a ticket's priority and the customers that an incident affected. Each check is a short program that encodes one lesson, such as "a deal whose close date keeps moving is in trouble". On day one the checks raised 349 situations of 23 kinds, each with its evidence (S12 lists every check and its rule). A situation can hold one problem, such as a slipping deal, or many, such as every licence that people who left still hold.

Box 1. A consulting team and a memory that works as a consultant. Each row is one part of a consultant's work. The middle columns describe how each side does the work, and only the last column reports measurements.

The work	A consulting team, with or without AI	A memory that works as a consultant	What this study measured
Learn how the company runs	Interviews and samples, over weeks	Every record, updated as records arrive	Model 96% to 100% correct on day one, and 95% to 100% at quarter end after two fixes (sections 2 and 7)
Find what is broken	What the team has time to examine	Every record, every day, with nobody asking	483 of 485 planted problems at the development company; Opus alone 8% (section 4)
Judge what a finding means	The team's experience	People, or language models that read its situations	25 judgments: Sentra 76%; Opus 68% alone, 80% with Sentra, a gain that can be chance (section 6)
Tell the people who can act	The leaders who hired the team	Each person whom the problem affects, in any team	After a routing fix, situations about customers reached 99.6% of the people that Eon's truth and Sentra's rules name; 58% of feed items came from other teams' checks (section 5)
Bring lessons from other clients	The consultants' experience	Lessons kept in the system and run at the next company	491 of 501 planted problems at the held-out company, from the same generator, on the first run, by an exact-match program finished with both companies' results in view (section 4)
Help people and agents who ask	Meetings and reports	Any person or AI agent, at any time	As Opus's memory: nearly three times as many problems in its briefs, at 35% lower agent cost; its direct answers no better. Sentra's own question-answering agent: 34% (section 6)
Rank findings, recommend and make changes	The team ranks and recommends; the company acts	A later stage	Not tested (sections 5 and 9)

4. **Route.** Sentra sends each situation about a customer to the people who work on that customer, in any team, within the company's permissions.

The model updates continuously: Sentra adds new records as they arrive and recompiles only what changed. In this study, stages 2 to 4 ran as research code over Sentra's memory, and the product does not have them yet.

Who wrote the lessons. People did. We wrote the checks while we studied the first company, with its answer key available, and refined them where they missed (S4). Most checks read fields and phrases in Era's record formats, such as a ticket subject that starts with "Cancel", or "the fix lands next week". The broken-promise check takes its words from the records: the product areas and failures that recur in ticket subjects. A lesson, unlike a fact in memory, must work on a company that it was not written on, and the second company tests this. For records in another format, or a real company's language, someone must adapt the checks, or the system must learn them.

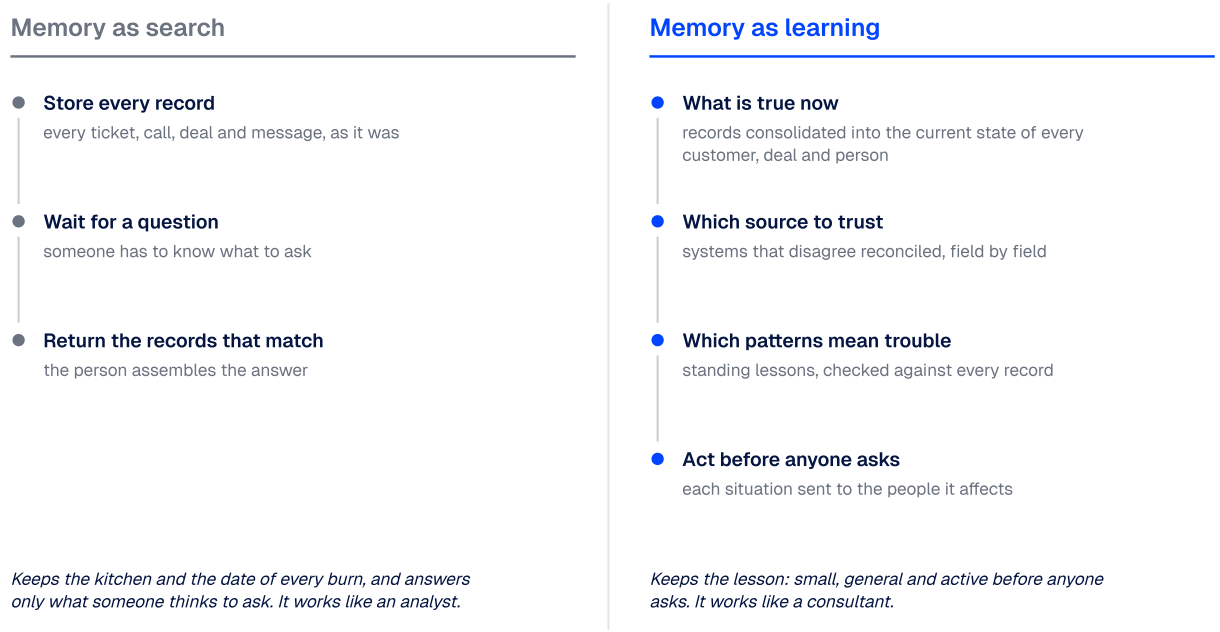


Figure 1. Two views of memory. Memory as search keeps every record and waits for a question, as an analyst does. Memory as learning keeps what is true now, which source to trust and which patterns mean trouble. It acts on them with nobody asking, as an experienced consultant does. In this study, people wrote Sentra's lessons.

Memory in 102 minutes. Sentra's product built its memory of Wisteria from the raw records (21,815 files, 63.7 MB of text) in 102 minutes. The memory holds 6,694 entities joined by 76,306 links, each with its evidence. The model took another 37 minutes to compile. The same person or customer often has different names in different systems. Sentra resolved 9,719 name variants to 2,817 identities, and recovered 1,114 of 1,199 reporting lines (93%) from the records alone. Against Eon's truth, the model's key fields were 96% to 100% correct (section 7).

When systems disagree. Every company runs systems that disagree, and an AI that only searches records can repeat a disagreement as a confident answer. Sentra treats a disagreement as a finding. By Eon's truth, Salesforce and billing disagree about 200 of Wisteria's 407 accounts. Sentra's reconciliation compared 400 customers and found all 696 planted conflicts with no false alarms. These include the 97 about whether a customer is live. The reconciliation trusts each system for what that system owns: Salesforce for a customer's profile and owner, billing for money and renewals.

The reconciliation flagged 110 customers, with $4.7M of annual recurring revenue (ARR), for a person to decide (some for more than one reason):

- 64 whose status changed more recently in the other system than in the system of record

- 32 that asked to cancel while billing is still live
- 27 whose owner in billing left
- one whose owner in Salesforce left.

It also found all 181 people who left, and all 40 app licences and admin roles that they still hold.

3 Two companies with known answer keys

Eon's Era platform builds simulated companies for tests of AI. Each company has connected records across the systems that a real company runs, with exact answers to questions about them (10, 11). We used two of these companies.

Wisteria Group is the *development company*: we wrote and refined Sentra's checks on it. It is a payments company that sells to other businesses. Its HR system lists 1,200 people, and 181 of them left. It has about 400 customer and prospect accounts (193 of them paying), 4,675 recorded sales calls and 16,676 support tickets. These are in twelve systems, from Salesforce (sales) and Stripe (billing) to Zendesk (support tickets), PagerDuty (incidents), Okta (app access) and HiBob (HR) (S1).

Eon planted six business scenarios in Wisteria and gave us files that record the truth of each (Table 1). We have two snapshots of the company, on day one (16 October 2026) and at quarter end (15 January 2027), and the records between them.

Norell is the *held-out company*: a second Era company with different people, customers, competitor and scenarios. Sentra's checks first ran on it on 18 October 2026, after we developed them on Wisteria. (One of Wisteria's customers has the name Norell too. In this paper, the name always means the second company.) All people and companies in the Wisteria and Norell records are part of the simulation. Figure 2 shows how the study worked.

The answer key. From Eon's truth files, we listed the problems that someone at Wisteria must act on that morning: 485 in all, of nine kinds (section 4). Each problem became a line that a particular morning brief must contain. The seven briefs are for the CEO, sales, customer success, support, engineering, IT and security, and finance. We also wrote 25 judgments about the planted scenarios: conclusions that someone who knows the company will reach. And we wrote 109 graded points for the answers to 14 direct questions (S2). The key uses the same definitions as the checks, so in both a slipping deal is one whose close date moved three or more times.

During the study, we found that our key counted "the next 90 days" from the end of the quarter, not from day one. So it covered about six months. We corrected the key and scored Sentra and every agent setup again. Earlier, we widened Sentra's own renewal check to match the faulty key, and we returned it to 90 days (S4). We built Norell's answer key, 501 problems, with the same code, two hours before Sentra first read Norell's records.

Who looked, and how we scored them. Sentra and every agent setup saw the same thing: the company's day-one records, one text file per record, grouped by system. We blocked the agents from the answer key, the scoring criteria and the company's later records. Each agent setup ran the same 37 tasks (S5).

- **Sentra** read these records through its product and then ran the research code that section 2 describes. We wrote its checks with Wisteria's answer key available. We first wrote and tuned its record layer (S3) and reconciliation on Wisteria's end-of-quarter records, before we loaded the day-one records. We wrote its situation checks on the day-one records (S4).
- **AI agents.** Two Claude models ran as agents in Claude Code, Anthropic's tool that runs Claude on a task. They were Opus 5.5, a leading frontier model, and the smaller Haiku 4.5. They could read and search the files, but they could not run code, use the web or write files. Each task had

Table 1. What Eon planted in Wisteria Group. From Eon's truth files. ARR is annual recurring revenue.

Scenario	Day one (16 Oct 2026)	End of quarter (15 Jan 2027)
Enterprise deal	Diabase Bank, $1,875,000, in negotiation and blocked by two open security findings; the CFO signs after security approves; the close date moved four times already	Won on 3 November
Customers leaving	68 accounts gave notice, 32 of them paying ($2.56M ARR); their cancellation tickets give the reasons, most often consolidation	9 churned, $588K ARR lost
After an outage	A severe data-pipeline outage left 90 urgent tickets open across 90 accounts ($3.1M ARR); 5 renew within 90 days ($102K)	6 of the 90 churned
A competitor	Vertex is in 87 open deals ($12.6M); 88 sales calls say it wins on price while Wisteria's security review keeps Wisteria in; 1 deal already lost	41 lost ($8.2M), 26 won
Conflicting records	Salesforce and billing disagree about 200 of 407 accounts ($6.9M ARR): 696 field conflicts, 97 of them on whether the customer is live at all	
Messy identities	12 duplicate accounts and 142 duplicate contacts	
Also in the records: people who left	181 people left; they still hold 40 app licences or roles	70 more left, 42 promoted
Also in the records: unwatched spending	11 supplier contracts renew within 90 days ($6.2M a year) with no owner recorded	
The quarter as a whole		Bookings $10.1M against a plan of $11.3M; 121 deals won, 189 lost; 34 hires, 70 people left

a fixed budget ($6 for Opus, $3 for Haiku), and no run used all of it. For each team, we asked an agent for that morning's brief, for example: "Nobody has asked you anything. Write the brief the VP of Sales should read this morning: every problem in their area that needs action, who holds each one, and what it is worth in dollars where the records show it."

- **Agents with a checklist** got the same tasks and a list of every kind of problem in the key.
- **Agents with Sentra** got the same tasks and Sentra's tools. These gave access to its memory, its model of the company, its situations and each person's feed.

Two AI judges, OpenAI's GPT-5.5 and Anthropic's Claude Sonnet 5.5, checked each brief against the lines of the key that belong in it (20). They used the criterion prompt of Harvey's open-source Legal Agent Benchmark (21). A line counted only if both judges passed it, and a problem counted as raised if any

brief where it belongs raised it. On the agent runs, the judges agreed on 96% of 4,008 decisions (Cohen's kappa 0.91, where 1 means perfect agreement) (22). We also checked every claim in a brief against the company's records, first those that the brief cited (S6).

The two sides did different jobs (Table 2).

So the comparison measures how completely a system built for this task covers known kinds of problem, with lessons written on this company. The agents, by contrast, must read files to find and report problems. The comparison does not test agents that write and run their own programs. That is the most important comparison that we still must make (section 9).

Table 2. What the comparison tests, and what it does not. Output lengths are in S13.

	Sentra	AI agents
How it works	**Standing checks, run as code over every record**	Reads and searches files; cannot run code
What it was told	**Nothing in the task; we wrote its checks and its model's instructions on Wisteria**	A brief to write; with the checklist, every kind of problem in the key
Length of output	**Not limited: a complete list (its CEO brief joined four teams' lists, about 22,000 words)**	A brief (the CEO's limited to one page; about 900 words for Opus)
Where it ran	**Wisteria (development) and Norell (held out)**	Wisteria only

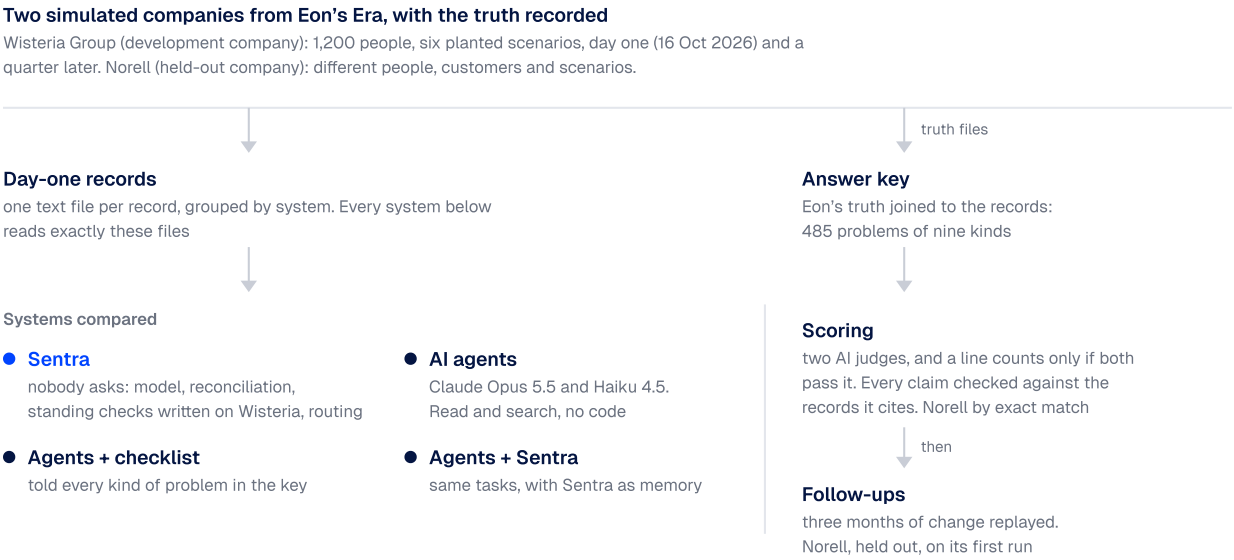


Figure 2. How the study worked. Eon generated the companies and the truth behind them. Sentra and every agent setup read the same day-one records, and we scored every brief against the same answer key. The company's next three months and a second, held-out company came after.

4 Problems that nobody asked about

On the development company, Sentra raised 483 of 485, and the agents far fewer. With nobody asking, Sentra raised 483 of the 485 planted problems at Wisteria (99.6%, 95% interval 98.5% to 99.9%) (23). The two AI judges scored this result. The two problems that Sentra missed were lines about licences

that people who left still hold. In one, ingestion took no facts from the person's HR record, so the check that reads Sentra's model did not know that he left. The reconciliation, which reads HR records directly, found the licence (S9).

When we grouped Sentra's situations into a brief for each team, the briefs made 358 claims. Of these, 343 (96%) were valid: the cited records supported them, and they needed action.

The AI agents found far fewer (Figure 3). Opus 5.5 raised 39 of the 485 (8%). With the checklist, it raised 119 (24.5%). Haiku 4.5 raised 11 (2%).

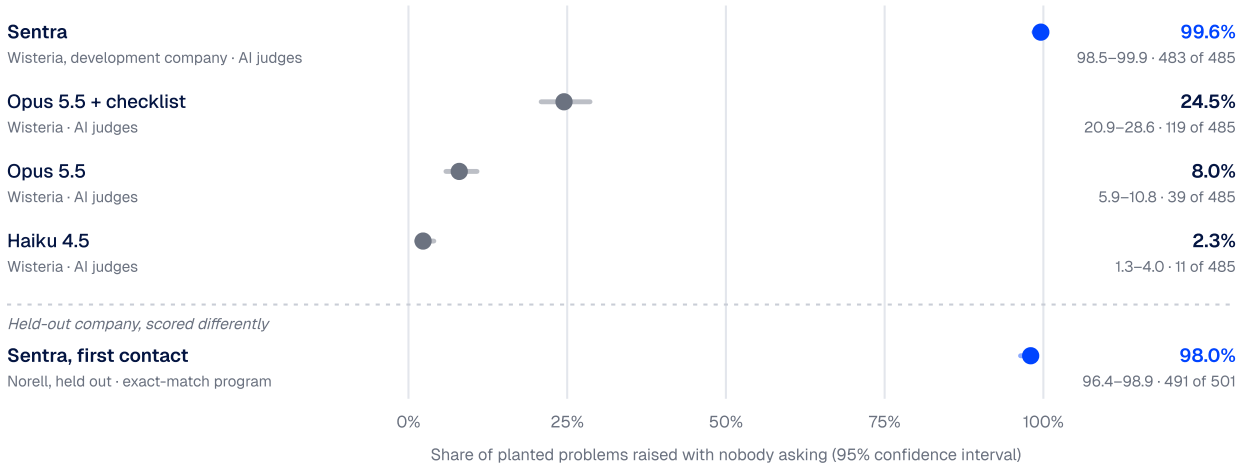


Figure 3. Planted problems raised with nobody asking. Each dot is the share of a company's planted problems that a setup raised in its morning briefs. The bar around it is a 95% confidence interval: the range of underlying rates that agree with the count. At Wisteria, the two AI judges scored every setup, and even the best agent's upper bound (28.6%) is far below Sentra's lower bound (98.5%). An exact-match program scored Norell, below the line. We finished that program with both companies' results in view, and it is slightly more lenient than the judges. No agent ran on Norell.

The gap holds under every way to count that we tried. The largest kind of problem in the key is the slipping deal (189 of the 485). If each of the nine kinds counts equally, Sentra scored 98.9%, Opus 32.8%, Opus with the checklist 57.6% and Haiku 11.1%. Without the slipping deals, the figures are 99.3%, 12.8%, 37.2% and 3.7%. The problems are also not independent: a brief that skips a kind skips all of it. If each brief, or each kind, counts as one unit, the intervals become much wider. But even then, the best agent's upper bound (53% or 63%) stays far below Sentra's lower bound (98%) (S13).

Where the agents failed. The agents did well on short lists. Opus found all 11 supplier contracts that renew within 90 days and 19 of the 21 licence problems. With the checklist, it went from none to all 19 renewals at risk, and from none to 25 of the 29 help articles whose owners left. The agents failed in two places.

The first is long lists, where the agent must count across the whole company. The 189 slipping deals are in one system, Salesforce, where any one deal is easy to see. Yet Opus's sales brief named 20 deals in all, and passed 1 of the 189 slipping-deal lines. With the checklist, it found 9 of the 189. A brief that an agent writes from files picks examples, but Sentra's checks list every case.

The second is joins across systems. The agents' engineering briefs discussed the incidents. But no agent setup, with or without Sentra, passed any of the 36 lines that list each incident with its customers and revenue at stake. To pass, a brief must match every affected customer to its billing record. Haiku added a third failure, about money. It told customer success that "$3.3B+ in ARR" was at stake, in a company that booked $10.1M of deals over the next three months. It seems that it counted its customers' own revenue as Wisteria's (S10).

Table 3. Planted problems raised with nobody asking, by kind, at Wisteria (the development company). A problem counts when both judges pass it. *Systems* is the number of systems that a reader must combine to see the problem. Eight of the 36 incidents hit no customer, and a brief had to say so. Agents with Sentra are in section 6 and S9.

Kind of problem	Planted	Systems	Sentra	Opus 5.5	Opus + checklist	Haiku 4.5
Deals whose close date keeps moving	189	1	**189**	1	9	0
Customers whose status in billing disagrees with Salesforce	97	2	**97**	2	15	0
Open tickets assigned to people who left	79	2	**79**	2	17	0
Incidents, with the customers and ARR they hit	36	2	**36**	0	0	0
Help articles owned by people who left	29	2	**29**	0	25	0
Licences and admin roles held by people who left	21	2	**19**	19	19	0
Renewals within 90 days at customers with urgent tickets open	19	2	**19**	0	19	0
Supplier contracts renewing within 90 days	11	1	**11**	11	11	11
Production services owned by people who left	4	2	**4**	4	4	0
All problems	**485**		**483 (99.6%)**	**39 (8.0%)**	**119 (24.5%)**	**11 (2.3%)**
Average over the nine kinds			**98.9%**	**32.8%**	**57.6%**	**11.1%**

Reasoning was not the main gap. When we asked Opus direct questions about the same company, it earned 86% of the graded points. When we asked directly for the slipping deals, it found 183 of the 189, against 20 deals in its unasked sales brief, which passed 1 of the 189 lines. On the 25 judgments about the planted scenarios, it met 76% when we asked about the scenario and 68% when we did not. Sentra's situations, which nobody asked for, met 76% (section 6). When Opus looked, it reasoned well, but by reading files it could not cover the whole company.

On the held-out company, the checks found 98% on their first run. Norell tests whether the checks depend on Wisteria's particular records. Eon generated Norell on 17 October 2026, and we loaded its end-of-quarter snapshot. We copied the checks unchanged from Wisteria's day-one code, and they first ran on Norell the next day. The code that builds the model differed in two lines. One line gives the company's name in the instructions, and the other gives the path segment from which the code reads each record's source system.

To remove AI judges from the result, an exact-match program scored Norell. It counts a planted problem when Sentra's situations name the record, account, person or incident that makes the problem unique. It checks the names, not every count or amount, so it is slightly more lenient than the judges. On Wisteria,

it gave credit for all 485 problems, including the two licence lines that the judges failed. We finished this program with both companies' results in view. Its fixes raised Wisteria's count from 471 to 485 (the judges passed 483) and Norell's from 434 to 491.

On its first run on Norell, Sentra found 491 of the 501 planted problems (98.0%, 95% interval 96.4% to 98.9%). Norell had three kinds of problem that Wisteria's answer key did not contain, but we wrote checks for all three while we developed them on Wisteria. Sentra found two of these kinds on the first run. They were people who left but are still on on-call schedules (32 cases), and a customer whose owner left. The third kind accounts for 9 of the 10 misses: 9 open deals ($980,650) at customers that gave notice. The check counted a request to cancel as notice only while its ticket was open, and support closed every such ticket before Norell's quarter end.

We changed this rule for every company, so that notice holds until the customer leaves. Sentra then found 500 of 501 (99.8%), and Wisteria's output did not change. The one problem that it still missed is a service whose owner left.

Norell shows that the checks do not depend on Wisteria's particular customers, people or numbers. It cannot show that they work on records in another format. Both companies come from the same generator and its record templates, and we built both answer keys with the same definitions. No AI agent ran on Norell.

Beyond the key. We checked two findings outside the answer key directly against Eon's truth, on both companies (Table 4).

- **Promises that failed.** On 192 tickets, support told customers that "the fix lands next week". Sentra matched each promise to later tickets about the same failure, from any customer. For 117 promises, customers reported the same failure again more than a week later, in 1,070 later tickets from 218 customers. Of those tickets, 247 are still open. All 117 were real broken promises, and all 1,269 links from a promise to a later ticket joined reports of the same failure. Eon's truth holds 137 broken promises, so Sentra found 85% of them, with no false ones.
- **Revenue cuts.** Billing shows when revenue fell, but the reason, if there is one, is in conversations. Sentra found every cut in a customer's ARR in the past twelve months, at renewal or mid-contract: 37 of 37, $2,044,169 to the dollar. It found a reason on record for 6 of these cuts, such as a competitor with a lower price, and none for the other 31. The check searches the deals for words about price, budget or consolidation. It also searches the tickets and calls from 30 days before to 120 days after the cut. So it misses a reason given later or in different words.

Two revenue cuts show the second finding. Doric Worldwide's ARR fell from $1,250,000 to $480,000 at its November 2025 renewal, and by another $159,000 in April 2026. An expansion in September then took it to $775,000. Nexora Payments' ARR fell from $976,900 to $644,800, and Nexora shows what the search misses. In October 2026 it gave notice and said that its parent company will consolidate onto another vendor. This came outside the window that the check searched.

5 Who hears about each situation

To find a problem is half the job. The other half is to make sure that the right people hear about it. For each customer, Sentra's routing reaches the account owner, the open-deal owners, the open-ticket assignees and the incident responders.

When we checked this paper, we found that the research code read each account's owner from the wrong field. So on the first run, account owners heard about their accounts only through another link, such as their own deals and tickets. We fixed the field and ran the routing again from the same inputs. Routing is deterministic and uses no language model, so the same inputs give the same output. The figures below come from the corrected run, and S7 compares them with the first run.

Table 4. The same checks on the development company and on the held-out company. The judges and the exact-match program scored the problems found. We checked the other rows against Eon's truth. A link joins a broken promise to one later ticket. Norell's problems found come from the matcher's saved results, and its other figures come from the team's internal report.

	Wisteria (development)	Norell (held out)
Planted problems found	483 of 485 (judges); 485 of 485 (exact match)	491 of 501 on the first run; 500 of 501 after one general change (exact match)
Revenue cuts, found against Eon's truth	37 of 37, $2,044,169 exact	29 of 29, $1,751,970 exact
Broken promises flagged that were real	117 of 117	293 of 293
Links that joined reports of the same failure	1,269 of 1,269	2,461 of 2,461

One customer, four systems. Doric Worldwide renews on 9 November, and Laura Blackmoor in customer success owns its renewal. Facts that she needs to know are in systems that other teams mostly use (Figure 4). Billing shows that Doric cut $770,000 at its last renewal and $159,000 more in April, before it expanded in September. Sentra's search found no reason for either cut. In February 2025, support told Doric that the fix for a payout-schedule failure "lands next week". Since then, 28 other customers reported the same failure in 29 tickets, and 7 of these are still open.

In all, seven of support's broken promises involve Doric: two made to Doric, and five made to other customers whose failures Doric later reported too. Engineering's incident records put Doric among the customers that three incidents hit, including both of September's most severe. In Salesforce, three people own open deals at Doric, worth $7,151,700: Laura, a colleague in customer success and a sales rep. The colleague's $1,771,700 expansion is open while two urgent or high-priority tickets stay unresolved, and the close date of the renewal itself moved 19 times. No single system shows all of this. On day one, Sentra put all of it in her feed: 23 items about Doric, 20 of them from checks that belong to other teams.

Across the company. On day one, 281 people received a feed. Of the items in those feeds, 58% came from checks that belong to a team other than the recipient's (S12).

Did the right people hear? To check the routing, we used Eon's truth about these four roles for each customer. Sentra's own rules say which kinds of situation each of these roles needs. For each situation and customer that reached anyone, 99.6% of the people that the truth and those rules name heard about it.

And 99.1% of deliveries went to someone whom the truth ties to that customer. In a comparison where each person gets only their own team's situations, 40.6% of what they must hear reaches them (S7).

What the fix changed, and what still reaches no one. On the first run, the situations of 52 of the 97 customers whose status in billing disagrees with Salesforce reached no one. Neither did 17 of the 37 revenue cuts. After the fix, all of them reached the people who work on those customers. The owners of the five customers that renew soon after the outage got none of those customers' situations on the first run. Now they get between 16 and 30 items about them. What still reaches no one is mostly restricted, or it concerns no customer:



Figure 4. One customer, four systems. On day one, four systems held parts of the story about Doric Worldwide, and none of them held all of it. Sentra sent every part to the person who owns Doric's renewal.

- the 11 supplier renewals, which name no owner and no customer
- 139 rows about what people who left still hold, which concern no customer and are only for People, IT and security, and the CEO
- one row about a declined discount.

In People, IT and security, and the CEO's office, only one person, in People, has a feed. The score covers only situations that reached someone, so it does not count these. Routing that follows customers needs a second path for situations that concern the whole company.

How much each person received. A feed holds about 97 items on average, but items are rows inside situations. The median person received 29 situations (90th percentile, 52), or 21 if each incident's engineering and support copies count once. Support staff received the most: 13,771 items across 88 people. About half of these were broken promises at customers whose tickets they hold. We did not test how many items people will read, or how best to rank them (S11).

Permissions. Sentra applies permissions when it builds the feeds. Situations about what people who left still hold contain HR detail. So outside People, IT and security, and the CEO, they go only to people who work on the customers involved. The policy keeps the full list of revenue cuts for Finance and the CEO,

but no one in those functions had a feed. Instead, 43 people received the rows for their own customers, and these rows covered all 37 cuts (S7).

6 Sentra as memory for other AI agents

With Sentra as memory, the same AI agents found more of what nobody asked about, at lower cost. On direct questions, only the smaller agent improved. We gave the agents Sentra as extra tools, with the same budgets, and ran all 37 tasks again (Figure 5, S9).

- **Cheaper and faster.** All 37 tasks cost Opus $34.81 with Sentra, against $53.53 without it (35% less), and took 75 minutes of agent time instead of 113. Haiku's cost fell from $10.09 to $6.36 (37% less). The agents processed 42% to 48% less text, probably because they could start from Sentra's model, not from the raw records.
- **Costs not counted.** These figures count only the agents' own costs. We did not meter the dollar cost to build Sentra's memory (102 minutes) and model (37 minutes). Nor did we meter Sentra's question-answering agent, which uses a language model and which the agents could call. Search uses embeddings, and the other tools use no language model.
- **More found, with nobody asking.** With Sentra, Opus raised 22% of the 485 planted problems, against 8% without it, and Haiku went from 2% to 16%. Neither gain is likely to be chance (S6). With Sentra, Haiku found about twice as many problems as Opus alone (76 against 39), for about an eighth of the cost. With Sentra's memory and model but not its situations or feeds, Opus found about as many (22%), and its cost fell by 14%.
- **Claims.** With Sentra, the records supported Haiku's claims more often (56% against 41%). Opus's claims had about the same support either way (89% with Sentra, 92% without).
- **Direct questions: no gain for Opus.** On the 109 graded points for direct questions, Opus scored 86% alone and 84% with Sentra. Haiku rose from 39% to 61%. Sentra's own shipped question-answering agent, which answers from a sample of its memory, scored 34%.
- **Judgments about the planted scenarios.** With nobody asking, Sentra's own situations met 76% of the 25 judgments. Opus met 68% alone and 80% with Sentra, and 76% and 88% when we asked about the scenario directly. With 25 judgments, a gain of three can easily be chance ($p = 0.25$). Haiku's score with nobody asking rose from 12% to 52%, and this gain is unlikely to be chance ($p = 0.002$).

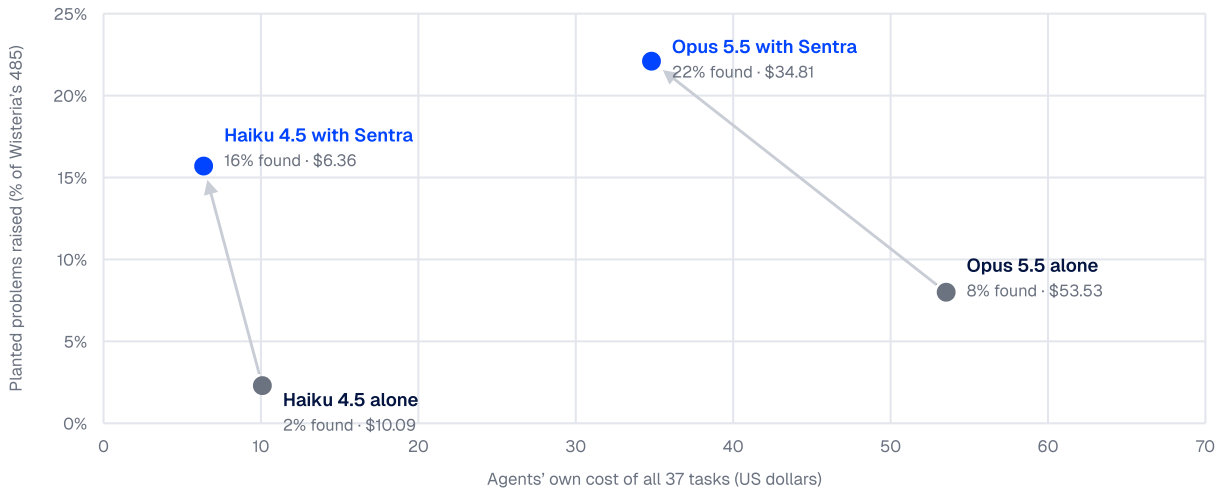


Figure 5. AI agents with and without Sentra as memory. Each dot is one agent setup. Each arrow goes from an agent that works alone to the same agent with Sentra. Higher means more problems found, and further left means cheaper. Costs are the agents' own and do not include Sentra's. Sentra's own checks, with no agent, raised 99.6%.

An agent with Sentra still writes a brief, and a brief omits things. With every situation in front of it, Opus's briefs carried 22% of the planted problems, and the complete list is Sentra's. This suggests a division of work: Sentra tracks the whole company, and agents, including those that people already use, read from it.

Sentra's checks are also complete only for what we wrote them to see. On day one, the check for the outage's aftermath counted 74 urgent tickets still open. It missed 16 whose subject lines had none of its search words. Opus with Sentra counted the outage's tickets in eight answers, and in six it checked the ticket files and found all 90. Only two of those six also corrected the revenue at stake, and the remaining two of the eight answers repeated Sentra's 74 (S10). New wording, or a new kind of problem, needs a new lesson, or an agent that also reads the records.

7 A model that stays current

The model stayed current through three months of change, after two fixes. We replayed Wisteria's next quarter into Sentra in monthly batches. Each month brought between 2,646 and 3,209 new or changed records. Sentra took 6.6 to 30.5 minutes to add them, and it recompiled the affected parts of the model in about 20 minutes (S8).

The first comparison with the end-of-quarter truth showed two faults (Table 5). First, when Sentra read an updated deal record again, the new facts attached to the customer, not to the deal. This left 70 of 835 deals frozen in an old stage. Second, Sentra marked 48 accounts as customers because they had a won deal, although Salesforce still listed them as prospects.

We fixed both faults. Now Sentra compiles a deal from every fact that cites it, and a customer's status is what its system of record says. After the fixes, the model was 95% to 100% correct. We found these faults by comparison with the end-of-quarter truth, so the fixed figures show what the corrected system does. They are not a blind test.

8 Related work

Memory for AI. Section 1 describes memory and retrieval systems for language models (6–9). Closer to our view, generative agents reflect on what they observe and store higher-level conclusions (16). Reflexion agents keep lessons from feedback on earlier attempts and do not change their weights (17). Both write their own lessons, for one agent. Sentra keeps lessons for an organization and acts on them for everyone that they affect, but in this study people wrote its lessons.

Agents at work. Recent benchmarks test agents on work tasks in simulated workplaces. These include a small software company (24), customer-relationship management in Salesforce (25), common knowledge work in enterprise software (26) and customer service under business rules (27). These benchmarks ask whether an agent can complete a task that it gets. We ask a different question: does a system find the problems that nobody asked about, and tell the people that each one affects?

Rules over company records. Companies already run rules over their records: business-rule engines, alerts in business intelligence (BI) tools, and process mining, which uses event logs to show how processes run (28). Expert systems encoded the rules of human experts, and the central problem was to get these rules from the experts (29). Sentra's lessons have the same problem as long as people write them. What Sentra adds is the combination of three things: a model of the company that a language model builds from many systems, reconciliation across them, and routing across teams.

Table 5. Accuracy of Sentra's model against Eon's truth. Day one, and the end of the quarter before and after the two fixes. ARR and deal amounts count as correct within 5%, and the date a person left within 7 days. On day one, we measured deal stage, amount and close date together, and a dash marks a figure that we did not report. On day one, we scored the fields for 399 customers (190 live ones for ARR and renewal date), 1,196 people and 549 deals. At quarter end, we scored them for 405 customers (189), 1,226 people (249 for the date they left) and 583 deals.

Field	Day one	Quarter end, first run	Quarter end, after fixes
Customer status	100%	88.1%	99.8%
Customer owner	99.5%	99.3%	99.5%
Owner has left	100%	100%	100%
Customer ARR	98.4%	97.9%	99.5%
Renewal date	100%	99.5%	99.5%
Asked to cancel	96.0%	95.3%	95.1%
Person still employed	100%	99.8%	99.8%
Department	99.3%	99.1%	99.1%
Date a person left	—	95.6%	95.6%
Deal stage	98.9%	88.3%	99.0%
Deal amount	98.9%	98.5%	99.3%
Deal close date	98.9%	85.6%	99.3%

Organizational memory and learning. Organizations store lessons in rules, routines, structures and people (14, 15). A classic concern is that they lose these lessons when people leave, or that the lessons never reach the people who need them. Section 5 measures a version of that second failure.

Simulation with known truth. Simulations of people with AI agents (16, 30) showed how much researchers can study with a simulated population. Era does the same for companies, so we can check every answer here against the truth.

9 Discussion and limitations

The services-as-software thesis expects AI to do work that companies now buy as services. On simulated companies, these results suggest that a memory that applies lessons can do the first step of one such job, the consultant's diagnosis: to find what is broken. Here, people wrote the lessons.

What the agents lacked. When we asked Opus about a scenario, it met as many judgments as Sentra did (76%). So, as far as these tests show, Opus lacked what the essay calls intelligence more than judgement: to apply known lessons to every record. Part of the gap is also format, because a brief picks examples (section 6). The essay expects that "today's judgement will become tomorrow's intelligence" (1). A lesson is one way for this to happen: a person uses judgement once and writes it as a lesson, and the system then applies it everywhere.

What comes from a memory that works as a consultant. The gains in Box 1, with their limits, are gains in the diagnosis. For a company, the gain that matters is in its results: revenue kept, money saved and risk removed. In the India trial, the 17% gain came from help to adopt better practices, beyond a diagnosis that the control plants also got (19). So the value of a diagnosis depends on what the company does next. Our hypothesis: a memory that finds such problems daily, tells those who can act and keeps its lessons gives larger and longer-lasting gains than a one-off diagnosis. This study does not test this hypothesis, and live deployments must.

What is and is not shown. This paper makes three claims:

- Memory must keep lessons, not only records.
- A company's memory must work for the whole organization.
- Such a memory can do a consultant's diagnosis, and keep doing it.

On simulated companies, the study supports part of the first claim and part of the third. Standing lessons, applied to a reconciled view of a company, found far more of what nobody asks about than agents that searched the same records. After one routing fix, situations about customers reached 99.6% of the people that Eon's truth and Sentra's rules name. Some situations about no customer, such as the 11 supplier renewals, reached no one.

We scored one day's diagnosis, and the quarter replay tested only the model. The study does not test the rest. People wrote the lessons, so nothing here shows a system that learns them. Every agent could read every record, so nothing here compares a company's memory with a personal agent that sees only one person's work.

Limitations.

- **Simulated companies.** Era's companies are realistic, but a real company is messier. Its people write in many styles, its systems hold more mistakes, and its problems come with no answer key. Both companies here come from one generator and share its record templates, and most of Sentra's checks read their fields and phrases (S12). For a real company, we must adapt the checks, or the system must learn them, and our next results will come from live deployments.
- **Tuned to the key.** We refined Sentra's checks on Wisteria after we saw what they missed against its key, and S4 lists every such change. For example, we changed the incident check until all 28 incidents with customers matched the key exactly. We also first wrote the record and reconciliation code on Wisteria's end-of-quarter records. So the Wisteria results describe a system tuned to that key.
- **The held-out test.** Norell is the cleaner test. We finished its checks before they first ran on it, apart from one rule that we changed afterwards. But we finished its scoring program with its results in view, and Norell came from the same generator.
- **What the lessons cover.** The checks find only the kinds of problem that we wrote them to find (section 6).
- **The comparison.** The agents could not run code, their output was a brief (the CEO's limited to one page), and each task ran once. Sentra ran code over every record with no length limit. It is possible that an agent that writes programs over the records, with more time, lists far more. Differences of a few points between agents are within chance. The differences that we report for Sentra's coverage are tens of points under every way to count that we tried.
- **The scoring.** Two AI judges, not people, scored the briefs, with a prompt written for legal work. They agreed closely, but a human audit is still necessary. The exact-match program that scored Norell checks names, not counts, and is slightly more lenient than the judges.
- **Research code, not yet product.** The model, checks and routing tested here ran as research code over Sentra's memory, and we do not release their code (see below). Sentra's shipped question-answering agent scored 34% on the direct questions. In the clearest test, its permissions held: when we asked about supplier renewals, an administrator received all 11 and a sales representative none. But it told both that nobody left after 1 September, and said that it could not reach Salesforce (S7).
- **Routing.** Routing follows customers and Sentra's own rules for who needs what. So its score measures whether Sentra found the right people, not whether the rules are right. Situations that concern no customer, such as supplier renewals, can reach no one. The figures come from a second run, after we fixed a fault that skipped account owners (section 5).
- **Costs.** The cost comparison counts only the agents' own costs. We did not meter the dollar cost to build Sentra's memory and model, or the cost of the agents' calls to Sentra's question-answering agent. We also did not compare costs with a consulting engagement.
- **Not prediction.** Sentra raises situations and routes them. It does not predict which customers will leave or which deals will close, and we make no such claim. On day one, Sentra flagged all 9 customers that churned during the quarter, but it also flagged 192 of the 193 paying customers.
- **Diagnosis, not advice.** We call Sentra a consultant because of the work that we build it to do, and this study tests only the diagnosis. Consultants also frame the question that leaders want answered, rank findings by value, write a recommendation, get leaders to agree and assess the management team. Sentra's lists are not ranked, and Sentra did none of the other things here. The planted prob-

lems are also operational, closer to an operations audit than to a strategy question. We did not measure whether anyone acted on a finding or whether a company improved, and we did not compare Sentra with a consulting team.

What would make the evidence stronger.

- Above all, lessons that the system learns from experience, not lessons that people write
- Agents that can write and run code, and three to five runs of every setup
- Sentra limited to one person's data, to test a company's memory against a personal agent
- Sentra without reconciliation, or with one system removed at a time
- A human audit of the judges
- A blind test on a third company, in another format, scored by Eon
- For each finding, whether its owner acts, how fast, and what the action is worth
- Recommendations from Sentra, judged by experienced operators
- The same company diagnosed by a human team with the same records and time.

Items two to six need only new runs (S11). The first item needs new software, and the last three need new kinds of measurement.

10 Conclusion

A good consultant learns how a company runs, brings lessons from other companies, finds what nobody inside saw and tells the people who can act. This study measured coverage, one part of that work. On two simulated companies, lessons that people wrote found nearly every planted problem when Sentra ran them on every record. This held on the company where we wrote them and on a held-out company from the same generator. On the first company, AI agents that read the same files found far fewer. Sentra's model of the company stayed current through a quarter, after two fixes. When we asked Opus about a scenario, it met as many judgments as Sentra did, and both agents found more problems with Sentra as their memory.

Three tests will decide whether this memory becomes the consultant that never leaves. The tests are a comparison with agents that write their own code, a real company's records, and lessons that the system learns from experience.

Data, code and access

The companion repository is private, and researchers can ask for access at ashwin@sentra.app. It holds:

- the 53 task definitions with their judging criteria (37 that every agent setup ran, plus 16 check-ins)
- every agent run's answer, cost, time and token count, with the judges' verdicts and the claim checks
- the judges' verdicts and the claim checks on Sentra's briefs
- the answer keys and the code that built them
- Sentra's day-one outputs: situations, compiled model, per-person feeds from both routing runs, reconciliation, and the shipped agent's answers in the permissions test

- the model's accuracy, the quarter replay, and the checks of promises and revenue against Eon's truth
- Norell's matcher and results
- a plain description of every check, of the reconciliation policy and of the routing rules
- a script that recomputes the paper's numbers that those files support.

We do not release the code of Sentra's model, checks, reconciliation and routing. Researchers who want to reproduce or extend the Sentra results can contact Sentra at the same address. Eon agreed that we can share its records, truth files and task files with researchers. A few results come only from the team's internal report. These are Norell's checks of revenue and promises, the memory's size, the identity and reporting-line figures, and the count of paying customers with a situation. Table 1's figures come from Eon's truth files as that report gives them, with Diabase Bank's close-date count corrected.

Materials and methods in brief

Environment. Wisteria Group is Era company seed 1352647494, and Norell is seed 2718281828. Era describes Wisteria as 1,234 people, but its HR system holds 1,200 person records, and we use this figure. We exported each record as a text file and grouped the files in 11 folders by system: CRM, billing, support, calls, Slack, HR, engineering, IT, knowledge base, Drive and finance. Each file starts with a header that names its system, type, identifier and dates, and Sentra and every agent setup saw exactly these files. The runs took place from 16 to 19 October 2026 (S1).

Answer key. We joined Eon's truth files to the records. The day-one key holds 485 problems, written as 497 lines across the seven morning briefs, because the CEO's brief repeats 12 of them. We scored each of the 25 scenario judgments twice: once when we asked about its scenario and once when we did not (S2).

Agents. Each agent run was one headless call to Claude Code 2.1.301 per task. The agent had only read and search tools: no code execution, web access, file writing or sub-agents. The models were claude-opus-5-5 (budget $6 per task) and claude-haiku-4-5-20251001 ($3 per task). A deny list blocked the answer key, the tasks, the results and every other snapshot of the company. We later rebuilt the agents' settings file from the commands that wrote it (S5).

Sentra. Sentra's product, as of 16 October 2026, ingested the records as uploaded files. Fact extraction used gemini-3.5-flash, importance scoring used gemini-3.5-flash-lite, and embeddings came from OpenAI's text-embedding-3-large. The compiled model, reconciliation, checks and routing ran as research code over Sentra's memory.

Judging. GPT-5.5 used OpenAI chat completions, with JSON output and default temperature. Claude Sonnet 5.5 used Anthropic messages, with up to 1,024 output tokens and default temperature. Both judges used the criterion prompt from Harvey's open-source Legal Agent Benchmark (21), unchanged, except that the request to Claude ended with "Respond with JSON only." We could not parse six of Claude Sonnet's verdicts, and they count as fails. We saved only the parsing error, so we cannot read these verdicts again (S6).

Claim checks. GPT-5.5 listed every problem that a brief claimed, with the records that it cited. A Claude Sonnet 5.5 agent then checked each claim against the records, first those cited (S6). An exact-match program scored Norell (section 4).

Statistics. Shares in Figure 3 and for Norell have 95% Wilson score intervals (23). We compare two set-ups scored on the same criteria with an exact McNemar test (31). Problems cluster within briefs and kinds. So we also report averages over kinds, and intervals from a bootstrap that resamples whole briefs or whole kinds (20,000 resamples, S13). Agreement between judges is Cohen's kappa (22). Each agent ran once per task.

References

1. Services: The New Software. Essay (5 March 2026); https://sequoiacap.com/article/services-the-new-software.
2. C. E. Jimenez, J. Yang, A. Wettig, S. Yao, K. Pei, O. Press, K. Narasimhan, SWE-bench: Can language models resolve real-world GitHub issues? *International Conference on Learning Representations (ICLR)* (2024); arXiv:2310.06770.
3. E. Brynjolfsson, D. Li, L. Raymond, Generative AI at work. *Q. J. Econ.* **140**, 889–942 (2025); doi:10.1093/qje/qjae044.
4. S. Peng, E. Kalliamvakou, P. Cihon, M. Demirer, The impact of AI on developer productivity: Evidence from GitHub Copilot. arXiv:2302.06590 (2023).
5. K. Handa et al., Which economic tasks are performed with AI? Evidence from millions of Claude conversations. arXiv:2503.04761 (2025).
6. C. Packer, S. Wooders, K. Lin, V. Fang, S. G. Patil, I. Stoica, J. E. Gonzalez, MemGPT: Towards LLMs as operating systems. arXiv:2310.08560 (2023).
7. P. Chhikara, D. Khant, S. Aryan, T. Singh, D. Yadav, Mem0: Building production-ready AI agents with scalable long-term memory. arXiv:2504.19413 (2025).
8. P. Lewis et al., Retrieval-augmented generation for knowledge-intensive NLP tasks. *Adv. Neural Inf. Process. Syst.* **33**, 9459–9474 (2020).
9. D. Edge et al., From local to global: A graph RAG approach to query-focused summarization. arXiv:2404.16130 (2024).
10. D. Porat, Meet Era: Simulated companies to benchmark models and eval agents. Eon (updated 26 October 2026); https://www.eon.io/blog/era-build-and-test-ai-agents.
11. B. Gruenbaum, D. Porat, A. Natanzon, R. Zavida, C. Dinachi, O. Itzahary, The Era by Eon benchmark: A generated enterprise estate with exact ground truth for benchmarking LLM agents. arXiv:2609.09853 (2026).
12. E. Tulving, Episodic and semantic memory. In *Organization of Memory*, E. Tulving, W. Donaldson, Eds. (Academic Press, New York, 1972).
13. J. L. McClelland, B. L. McNaughton, R. C. O'Reilly, Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychol. Rev.* **102**, 419–457 (1995).
14. J. P. Walsh, G. R. Ungson, Organizational memory. *Acad. Manage. Rev.* **16**, 57–91 (1991).
15. B. Levitt, J. G. March, Organizational learning. *Annu. Rev. Sociol.* **14**, 319–338 (1988); doi:10.1146/annurev.so.14.080188.001535.
16. J. S. Park, J. C. O'Brien, C. J. Cai, M. R. Morris, P. Liang, M. S. Bernstein, Generative agents: Interactive simulacra of human behavior. *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*, 1–22 (2023); doi:10.1145/3586183.3606763.
17. N. Shinn, F. Cassano, A. Gopinath, K. Narasimhan, S. Yao, Reflexion: Language agents with verbal reinforcement learning. *Adv. Neural Inf. Process. Syst.* **36**, 8634–8652 (2023).
18. F. Dell'Acqua, E. McFowland III, E. R. Mollick, H. Lifshitz-Assaf, K. Kellogg, S. Rajendran, L. Krayer, F. Candelon, K. R. Lakhani, Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality. Harvard Business School Working Paper 24-013 (2023); doi:10.2139/ssrn.4573321.
19. N. Bloom, B. Eifert, A. Mahajan, D. McKenzie, J. Roberts, Does management matter? Evidence from India. *Q. J. Econ.* **128**, 1–51 (2013); doi:10.1093/qje/qjs044.
20. L. Zheng et al., Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. *Adv. Neural Inf. Process. Syst.* **36** (Datasets and Benchmarks Track), 46595–46623 (2023).

21. N. Grupen, G. Pereyra, J. Pereyra, Open-sourcing Harvey's long horizon Legal Agent Benchmark. Harvey (6 May 2026); https://harvey.ai/blog/introducing-harveys-legal-agent-benchmark; code and criterion prompt (lab_core/evaluation/prompts/rubric_criterion.txt) at https://github.com/harveyai/harvey-labs.
22. J. Cohen, A coefficient of agreement for nominal scales. *Educ. Psychol. Meas.* **20**, 37–46 (1960); doi:10.1177/001316446002000104.
23. E. B. Wilson, Probable inference, the law of succession, and statistical inference. *J. Am. Stat. Assoc.* **22**, 209–212 (1927); doi:10.1080/01621459.1927.10502953.
24. F. F. Xu et al., TheAgentCompany: Benchmarking LLM agents on consequential real world tasks. arXiv:2412.14161 (2024).
25. K.-H. Huang, A. Prabhakar, S. Dhawan, Y. Mao, H. Wang, S. Savarese, C. Xiong, P. Laban, C.-S. Wu, CRMArena: Understanding the capacity of LLM agents to perform professional CRM tasks in realistic environments. *Proceedings of NAACL 2025 (Long Papers)* (2025); arXiv:2411.02305.
26. A. Drouin, M. Gasse, M. Caccia, I. H. Laradji, M. Del Verme, T. Marty, D. Vazquez, N. Chapados, A. Lacoste, WorkArena: How capable are web agents at solving common knowledge work tasks? *Proceedings of the 41st International Conference on Machine Learning (ICML)*, PMLR **235**, 11642–11662 (2024).
27. S. Yao, N. Shinn, P. Razavi, K. Narasimhan, τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *International Conference on Learning Representations (ICLR)* (2025); arXiv:2406.12045.
28. W. M. P. van der Aalst, *Process Mining: Data Science in Action* (Springer, Berlin, ed. 2, 2016); doi:10.1007/978-3-662-49851-4.
29. E. A. Feigenbaum, The art of artificial intelligence: Themes and case studies of knowledge engineering. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI)*, 1014–1029 (1977).
30. J. S. Park, C. Q. Zou, A. Shaw, B. M. Hill, C. Cai, M. R. Morris, R. Willer, P. Liang, M. S. Bernstein, Generative agent simulations of 1,000 people. arXiv:2411.10109v1 (2024).
31. Q. McNemar, Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* **12**, 153–157 (1947); doi:10.1007/BF02295996.

Acknowledgments, competing interests and authors

Acknowledgments. We thank Doron Porat and the Era team at Eon for Era and for the companies that we used here.

Competing interests. The authors are employees of Sentra, which makes the system that we evaluate here.

Authors. Ashwin Gopinath and Taran Singhania, Sentra. Correspondence: ashwin@sentra.app.

Supplementary materials

S1 The environment

Wisteria Group is Era company seed 1352647494, a growth-stage payments company that sells to other businesses. Era describes it as 1,234 people and about 400 accounts. Its HR system holds 1,200 person records, and 181 of these people left by day one. Its records include 4,675 recorded sales calls and 16,676 support tickets.

Era counts 14 systems. The task prompts name the twelve that hold records. These are Salesforce, Stripe billing, Zendesk, Gong, Slack, HiBob, a Backstage service catalog, PagerDuty, Okta (with 62 software applications), Confluence, Google Drive and a NetSuite supplier list. Drive files hold only metadata and sharing settings, not document text. Billing omits the platform fee, so Era holds no list prices.

Eon planted six scenarios with its own code and recorded its own truth for each (Table 1 of the paper). The scenarios are a churn wave, the aftermath of a severe outage, a competitive battle, an enterprise deal, conflicting sources of truth and messy identities. The company exists on day one (16 October 2026) and at the end of the quarter (15 January 2027), with the quarter's records between them.

We exported the records as one text file per record, in folders by system (crm, billing, support, calls, slack, hr, eng, it, kb, drive, finance). Each file starts with a header that names its system, type, identifier and dates. Every system under evaluation saw exactly these files.

Norell is a second Era company (seed 2718281828), with different people, customers, competitor and scenarios. Eon generated it on 17 October 2026. We built its answer key, 501 planted problems, with the same code as Wisteria's, two hours before Sentra first read its records. The same pipeline loaded Norell into Sentra at the end of its quarter (15 January 2027). Norell shares Era's record templates with Wisteria. One of Wisteria's customers (acct-43) also has the name Norell, but it is not related to the second company.

All runs took place from 16 to 19 October 2026.

S2 The answer key

We built the day-one answer key when we joined Eon's truth files to the records. It holds 485 distinct problems of nine kinds. As lines in the morning briefs that they belong to, they make 497 lines. This is because the CEO's brief repeats 12 problems that also appear in a team's brief: nine slipping deals, two renewals and one status disagreement. The paper counts each problem once. A problem counts as raised if any brief that it belongs in raised it, and if both judges pass the line.

Each line names a problem and its key facts, for example: "Opportunity deal-1 ($13,000, Qualification) has had its close date reset 6 times; now 2026-12-31." The key uses the definitions of Sentra's checks. For example, in both, a slipping deal is an open deal whose close date moved three or more times.

Table S1. Planted-problem lines by morning brief. Lines, not distinct problems. The CEO's 12 lines all repeat problems from other briefs.

Brief	Lines
CEO	12
Sales	189
Customer success	19
Support	108
Engineering	40
IT and security	21
Finance	108
All	**497**

Each brief also has one to three lines that ask only whether it raised a kind of problem at all. There are 12 such lines ("problem types" in Table S5).

Scenario judgments. We wrote 25 judgments about seven scenarios. These are the churn wave (5), the competitive battle (3), conflicting records (4), the enterprise deal (4), people who left (3), the outage's aftermath (4) and unwatched spending (2). We scored each judgment twice. The first score is from seven tasks that asked about its scenario directly. The second is from five open briefs (customer success, finance, people, sales and support) that did not mention it.

Direct questions. Eighteen tasks ask a direct question. An example: which customers did incident 7023 affect, how much ARR do they hold, and which of them renew in the next 90 days? Fourteen of these tasks carry the 109 graded points. The other four have no planted answer at Wisteria, so we did not score them. They ask about:

- customers still owned by someone who left
- on-call schedules that include people who left
- employees whose manager left
- open deals at customers who asked to cancel.

Check-ins. Sixteen more tasks describe an action that another agent is about to take. They ask for a verdict (go, hold, ask or no) and the facts behind it. For example: "Tell the customer on ticket-18 that their assigned engineer will follow up tomorrow." The expected verdict is hold, because the assignee left. We ran check-ins only on Sentra's shipped question-answering agent.

The correction during the study. The key's "renews within 90 days" window used the end-of-quarter date, so on day one it meant about six months. We changed it to day one plus 90 days. The key went from 522 problems to 485: renewals at risk from 39 to 19, and supplier renewals from 28 to 11. The judges scored Sentra and every agent setup again on the corrected key.

S3 Sentra as run in this study

We tested two kinds of software, and the paper keeps them apart.

Shipped product. Sentra's ingestion turned every record into dated facts in its memory, with entities, typed links with evidence, and the different names of each entity. The shipped surfaces are the question-answering agent (called the memory agent), the swimlanes (live summaries of workstreams), the weekly report, the organograph (who works with whom) and per-fact permissions.

Research code over Sentra's memory. This code has five parts:

- the compiled model
- a record layer that joins licences, rotas, services, articles, tickets, incidents and suppliers to people and customers
- the reconciliation
- the checks that raise situations
- routing, with its permission policy.

We do not release this code. The internal report calls it "the shape of the nightly pass on the roadmap". The 99.6% result, the reconciliation results and the routing results all come from this code. Agents reached the compiled model, the situations and the feeds through a small local server.

Language models inside Sentra. Sentra's product, as of 16 October 2026, turned records into facts with Google's gemini-3.5-flash. It scored their importance with gemini-3.5-flash-lite, and it embedded them for search with OpenAI's text-embedding-3-large. The worker log for the day-one ingest names the first model 208,627 times and the second 61,701 times.

A language model built the compiled model, one entity at a time. It folds each entity's facts into the entity's current fields. It also reads up to four of the entity's own records from its systems of record, which the instructions call authoritative. This model was gemini-3.5-flash, the research code's default, and we set no other model on day one, in the replay or on Norell. Its instructions name the company ("Wisteria Group"). They also carry a hand-written list of which system to trust for each field, the same policy that the reconciliation uses (S12).

The checks are deterministic code. Most of them read the structured "Field: value" headers of Sentra's stored copies of the records, and the reconciliation. The compiled model supplies who left.

Time and cost. On day one, Sentra ingested 21,815 records with 63.7 MB of text in 102 minutes. It built the compiled model in 37 minutes (424 customers in 6 minutes, 797 projects in 6 and 3,352 people in 24). We did not meter the dollar cost, because the local stack logs no token usage. The tools for the compiled model, the situations and the feeds read from a database without a language model, and search uses embeddings. Sentra's question-answering agent, which the agents could also call, uses a language model, and we did not meter its cost.

Shipped surfaces, as measured here.

- Question-answering agent: 34% of the 109 graded points on direct questions (37 of 109). It met 24% of scenario judgments when asked (6 of 25) and 8% when not asked (2 of 25). It gave 5 of 16 check-in verdicts right. It answers questions about a whole population from a sample of records.
- Weekly report: a readable narrative of recent wins, churn, the competitor and the outage. As a CEO's brief, it raised none of that brief's 12 planted lines and 3 of 22 unasked scenario judgments. 3 of its 7 claims were valid.
- Swimlanes: accurate live summaries on all six workstreams that we tried. Their typed risks and contradictions returned nothing, although the records hold many.
- Organograph: on 20 people, it inferred 12 to 40 collaborators each (40 is its maximum). It also summarized each person's three closest working relationships.

Product faults found.

1. When Sentra reads an updated record again, it can orphan the record's entity. It replaces the old facts, and the new facts attach to a neighbouring entity, such as the deal's customer. This froze 70 of 835 deals during the quarter replay (S8).
2. The question-answering agent answered "Salesforce is not connected" and "nobody has left" on this company. The reason is that it does not read uploaded records for CRM and HR questions.
3. Swimlane risks and contradictions returned nothing on six lanes where the records hold many.
4. Swimlane discovery skips uploaded sources, so we named the lanes by hand.
5. Ingestion took no facts from 11 of the 1,200 HR records. One belongs to a person who left in January 2022 and still holds a HiBob licence. With no departure date in the model, the IT check did not treat him as a person who left (S9).

S4 Development history

We wrote Sentra's checks while we studied Wisteria and looked at what they missed. So the Wisteria results are those of a system tuned against this answer key. This section lists every rule that we changed after we saw a result. From 16 to 18 October, before we loaded the day-one records, we wrote and tuned the record layer and the reconciliation on Wisteria's end-of-quarter snapshot. Their default settings still use that snapshot, which we also used for the quarter replay. We wrote the checks that raise situations on the day-one records from 18 October.

- **Renewal window.** We widened Sentra's renewal check from 90 to 180 days to match the faulty answer key. When we corrected the key, we returned the check to 90 days (S2).

- **Incident revenue.** Sentra's incident check now takes each affected customer's ARR from billing, the authority on money. It states active and past-due ARR separately, and it names the deploy that triggered the incident. We changed the check until all 28 incidents with customers matched the key exactly.
- **Two fixes after the quarter replay** (S8). Sentra compiles a deal from every fact that cites its identifier, plus its own latest record, and it recompiles any deal whose record changed. A customer's status is what its system of record says, and a won deal does not change it.
- **Norell's first run.** We copied the checks unchanged from the day-one code, and they first ran on Norell on 18 October at 18:38. The model-building code differed from Wisteria's in two lines: the company's name and the path segment for each record's source system.
- **One rule after Norell's first run** (paper, section 4). A request to cancel now counts as notice until the customer leaves, not only while its ticket is open. On Wisteria, this change gave byte-identical output.
- **The Norell exact-match program.** We finished it with both companies' results in view. Its fixes raised Wisteria's count from 471 to 485 and Norell's from 434 to 491.

The final day-one run, on 19 October, raised 349 situations of 23 kinds, with 19 renewals. An earlier saved copy came before the renewal-window correction. It did not have three checks that we added later: revenue cuts, discounts declined and broken promises. We do not use that copy.

S5 Systems compared

Every agent run is one headless Claude Code call per task (`harness/lab.py`), with Claude Code 2.1.301. The call gives the prompt, an output format of JSON, the model, a dollar budget, the allowed tools and a deny list of tools. The deny list holds Bash, web fetch, web search, edit, write and sub-agents, minus any that the setup allows. A settings file of 23 rules blocks the answer key, the tasks, the results and the raw graph. Each run also denied every other snapshot of the company. We later rebuilt the settings file from the commands that wrote it.

Table S2. Agent setups. All six ran all 37 tasks once. Every run finished. The most expensive Opus run cost $4.72, and the most expensive Haiku run $0.80.

Setup	Model	Budget per task	Tools
Opus alone	claude-opus-5-5	$6	read, search (Read, Grep, Glob)
Opus with checklist	claude-opus-5-5	$6	read, search; checklist in the prompt
Haiku alone	claude-haiku-4-5-20251001	$3	read, search
Opus with Sentra	claude-opus-5-5	$6	read, search; Sentra's memory tools; compiled model; situations; person feeds
Haiku with Sentra	claude-haiku-4-5-20251001	$3	as Opus with Sentra
Opus with Sentra's compiled model only	claude-opus-5-5	$6	read, search; Sentra's memory tools; compiled model, without situations or feeds

The prompt. Every agent got the same preamble, then any notes for its setup, then the task:

- Preamble: "The company's records are in the current directory, one text file per record, grouped by system (crm, billing, support, calls, slack, hr, eng, it, kb, drive, finance). Each file starts with a header naming its system, type, ID and dates. You cannot write files: put your complete answer in your reply."
- Checklist: "Check the whole company, every record, for each of these: customers or deals owned by someone who has left; renewals in the next 90 days with open urgent or high-priority tickets; open deals at customers who asked to cancel; deals whose close date has moved 3 or more times; customers where billing and Salesforce disagree on status; people who left but are still app admins or licence holders, on-call, service owners, knowledge-base article owners or assignees of open tickets; people who still report to a manager who left; incidents that hit customers, with their ARR and upcoming renewals; supplier contracts renewing in the next 90 days."
- With Sentra: a note that Sentra is the company's memory and read every record. A second note says that the agent can query the compiled model like a table, and that the model flags what needs attention. It also says that claims must cite the records themselves. For the full memory, a third note says that Sentra's situations list "every situation it has raised, each counted over every record and reconciled across systems, with the evidence". The agent must confirm them and add to them from the records.

Tasks. Each setup ran 37 tasks: 7 unprompted morning briefs, 7 scenario tasks asked directly, 5 open briefs scored on the unasked scenario judgments, and 18 direct questions. The morning briefs were for the CEO, sales, customer success, support, engineering, IT and security, and finance. The CEO's brief had to fit on one page.

Not run. The harness holds setups that can also run code (`llm-code`, `llm-code-checklist`, with Bash inside the operating system's sandbox), but we did not run them (S11).

S6 Judging, claim checks and statistics

Judging criteria. For each criterion, each judge saw the task, the agent's answer, the criterion's title and its match criteria. These came in the criterion prompt of Harvey's open-source Legal Agent Benchmark (reference 21). We used the prompt unchanged, including its opening line, which describes the answer as a legal AI agent's work. The only change was that the request to Claude Sonnet ended with "Respond with JSON only." The judges were OpenAI's GPT-5.5 and Anthropic's Claude Sonnet 5.5, and a criterion passed only if both passed it.

We called GPT-5.5 through OpenAI's chat-completions interface, with JSON output and the default temperature. We called Claude Sonnet 5.5 through Anthropic's messages interface, with up to 1,024 output tokens and the default temperature. We read its verdict from the text between the first "{" and the last "}".

We could not read six of Claude Sonnet's verdicts, all on help-article lines in the support brief of Opus with the compiled model only. They count as fails. GPT-5.5 passed five of them, so they count as disagreements in the agreement figures below. That setup's count could be up to 114 (23.5%), not 109. We saved only the parsing error, not the text of the verdicts, so we cannot read them again.

Agreement. Over the 4,008 pairs of verdicts on the agent runs, the judges agreed on 95.9% (Cohen's kappa 0.906). On the 2,982 planted-problem lines alone, they agreed on 96.8% (kappa 0.888). If a line counts when either judge passed it, the agents' brief scores change only a little (Table S3).

Table S3. Planted problems raised (of 485), both judges against either judge.

Setup	Both judges	Either judge
Opus alone	39 (8.0%)	48 (9.9%)
Opus with checklist	119 (24.5%)	139 (28.7%)
Haiku alone	11 (2.3%)	11 (2.3%)
Opus with Sentra	107 (22.1%)	130 (26.8%)
Haiku with Sentra	76 (15.7%)	91 (18.8%)
Opus with compiled model only	109 (22.5%)	132 (27.2%)

Claim checks. The answer key holds only the problems that we wrote into it. A brief can also raise real problems outside the key, or make things up. So we checked every brief a second way (`harness/verify.py`). GPT-5.5 listed each problem that the brief raised, with the records that it cited and any dollars.

A Claude Sonnet 5.5 agent then read the cited records, and any others that it needed, with a budget of $0.60 per claim. It ruled whether the claim was supported (counts and amounts within about 10%), and whether it was a real problem that someone must act on. A claim is valid only if both are true. Results are in S10.

The Norell matcher. This deterministic program checks each planted problem in Norell's key against Sentra's situations, by the record, account, person or incident that makes the problem unique. It checks that the situations name the identifying record, not that counts or amounts are right. On Wisteria, it matched all 485 problems in the key, two more than the judges passed. It gave credit for both licence lines that the judges failed, including the HiBob line whose count Sentra got wrong (S9). So it is slightly more lenient than the judges. We finished it with both companies' results in view (S4).

Statistics. Shares in Figure 3 and for Norell have 95% Wilson score intervals. We compare two setups scored on the same criteria with an exact McNemar test, which uses only the criteria on which they disagree. On the 25 scenario judgments, Opus scored 19 against 22 with Sentra when asked ($p = 0.25$) and 17 against 20 when not asked ($p = 0.25$). In both cases, Opus with Sentra passed all three judgments that differed. Haiku scored 9 against 15 when asked ($p = 0.11$) and 3 against 13 when not asked ($p = 0.002$). On the 109 graded points for direct questions, Opus scored 94 against 92 ($p = 0.69$) and Haiku 43 against 67 ($p = 0.0005$).

On the 485 planted problems, Opus with Sentra raised 74 that Opus alone missed, and it missed 6 that Opus alone raised ($p < 0.001$). Haiku with Sentra raised 65 that Haiku alone missed, and it missed none ($p < 0.001$). Every setup ran once per task, so we did not measure the variation from run to run.

S7 Routing and permissions

The figures in this section come from the corrected routing run, unless we mark them as the first run's. Routing is a deterministic step with no language model. When we ran it again from the same inputs without the fix, it gave every figure of the first run again.

Who hears what. Sentra routes each situation to every person attached to the accounts that it concerns. What a person hears depends on their link to the account:

- the account owner hears everything about the account (after the fix below)
- an open deal's owner hears notices to cancel, cancellations still billed, open deals at customers that gave notice, incidents, broken promises and renewals with urgent tickets. The owner also hears competitor pressure, disagreements between systems, discounts declined, revenue cuts, two people with open deals at one account, expansions at troubled accounts, large deals with a blocker, slipping deals and duplicate accounts
- a ticket's assignee hears notices, incidents, broken promises, renewals with urgent tickets, expansions at troubled accounts and discounts declined
- an incident's responder hears the incident's customers and renewals, broken promises, renewals with urgent tickets and notices.

Policy. Sentra applies permissions when it builds the feeds. A situation of a restricted kind reaches people outside the allowed functions only through their own customers. These functions can see the restricted kinds:

- what people who left still hold: People, IT and security, and the CEO
- reports whose manager left: People and the CEO
- supplier renewals: Finance, IT and security, and the CEO
- the company-wide list of revenue cuts: Finance and the CEO.

On the first run, the policy kept 76 items out of people's feeds, all about what people who left still hold. On the corrected run, 139 rows of that kind reached no one. They concern no customer, so only People, IT and security, and the CEO can receive them. Only one person in those functions, in People, has a feed.

Who received feeds. Routing follows customers: a person receives a situation because they have a link to a customer that it concerns, or because the situation names them. On day one, 281 people received feeds, in customer success, engineering, people, sales and support (Table S4). No one in Finance, IT and security, or the CEO's office received a feed. The 11 supplier renewals name no owner and concern no customer, so they reached no one.

The rows about revenue cuts reached 43 people, each for their own customers, and covered all 37 cuts. On the first run, 17 cuts, including Nexora Payments' $332,100, reached no one. Of the 27,149 items, 14,318 were about broken promises. Most of these reached people because they hold another open ticket at a customer that later reported the same failure. That ticket is not the one with the promise.

A fault that we found and fixed while we checked this paper. The routing code looked for an account's owner in a field called "Owner", but Salesforce account records call it "Account owner". So on the first run, account owners received items only through their own open deals and tickets, or when a situation

named them. The owners of the five customers that renew soon after the outage (Sleet, Bluepeak Capital, Storm Capital, Fernbank Financial and Sable) got none of those customers' situations. We corrected the field name and ran the routing again, and the five owners now receive 30, 16, 24, 19 and 16 items about those customers. Laura Blackmoor's feed holds 56 items: 53 of them from other teams' checks, and 23 about Doric (Figure 4 of the paper).

Table S4. Day-one feeds by team, corrected run. On the first run, customer success had 48 people and 3,909 items, and sales 65 people and 1,273 items. The other teams did not change.

Team	People	Items	From another team
Customer success	61	4,908	87%
Engineering	63	6,163	83%
People	1	3	100%
Sales	68	2,304	82%
Support	88	13,771	32%
All	**281**	**27,149**	**58%**

Scoring against the truth. Eon's truth records who owns each account, who owns its open deals, who holds its open tickets and who responded to the incidents that hit it. We scored only situations about customers: the 1,213 situation–account pairs in the corrected feeds (1,025 on the first run). A pair that reached no one is not in the feeds, so we did not score it. On the first run, the score did not include 52 of the 97 customers whose systems disagree on status, or 17 of the 37 revenue cuts. After the fix, every one of them reached someone.

Each check's team, which decides what counts as "from another team", is the team that we assigned it (S12). Sentra's own rules (above) say which kinds of situation each role must hear. So the score measures whether Sentra found the right people, not whether the rules are right. Together, the truth and the rules give 11,734 wanted deliveries, each of one situation about one account to one person. The feeds made 11,791 such deliveries. Of these, 11,688 (99.1%) went to someone whom the truth attaches, and they cover 99.6% of the items that people must hear.

On the first run, the same figures were 99.0% and 92.2% of 11,546. In a comparison where each person gets only the situations of their own team, 40.6% of the wanted items reach them (40.5% on the first run). This comparison keeps Sentra's choice of people and changes only which situations each person receives. It is not a separately built routing system, and it cannot exceed what Sentra delivered.

Permissions in the shipped agent. We asked Sentra's shipped question-answering agent five questions, as an administrator and as a sales representative. Three questions touch records that the sales role cannot see (one on HR, two on finance), and the supplier question is the clearest test. On Nexora, the represen-

tative gave the same ARR and plan as billing but the wrong renewal date. It seems that it used Salesforce's renewal records. The HR question tested nothing, because both roles wrongly said that no one left (`results/sentra/shipped_product/governance_day1.json`).

Question	Administrator	Sales representative
Which suppliers renew in the next 90 days, and what do we spend with each?	All 11, $6,168,770 in total	None found
Who has left since 1 September 2026, and in what roles?	"No one" (wrong)	"No recorded departures" (wrong)
What is Nexora Payments' ARR, plan and renewal date in billing?	$644,800; Enterprise; renews 29 October 2027	$644,800; Enterprise; renewal "mid-October 2026"
Who owns Nexora Payments in Salesforce, and what open support issues does it have?	Said it had no access to Salesforce data (wrong)	"Salesforce is not connected" (wrong)
Which customers have asked to cancel or not renew, and why?	A few names; the truth holds 68 accounts	A few names; the truth holds 68 accounts

S8 Live updates over the quarter

We fed the quarter's records to Sentra one month at a time: 2,646, 2,679 and 3,209 new or changed records (tickets, calls, CRM updates and HR changes). Sentra took 6.6, 7.1 and 30.5 minutes to add them, and it recompiled the affected parts of the model in about 20 minutes each month. Over the quarter, 70 people left, 42 got a promotion and the company hired 34. Sales won 121 deals and lost 189, there were 12 account handovers and 59 forecast changes, and 9 customers churned.

The first check against Eon's end-of-quarter truth found two faults (Table 5 of the paper). First, when Sentra read an updated deal record again, its new facts (for example, "deal-821 moved to Closed Lost on 17 October") attached to the customer, not to the deal. So the deal kept no facts, Sentra never recompiled it, and 70 of 835 deals stayed frozen in an old stage. Second, Sentra compiled 48 accounts with a won deal but no billing subscription as customers, although Salesforce still listed them as prospects. We fixed both faults (S4), and we report the fixed figures as what the corrected system does, not as a blind test.

The day-one situations do not predict churn. On day one, Sentra flagged all 9 customers that churned during the quarter, but it also flagged 192 of the 193 paying customers.

S9 Full results tables

Table S5. All systems on day one. A planted problem counts when both judges pass a line for it in a brief where it belongs. A claim is valid when the records support it and it is worth action (S6, S10). Costs and minutes are the agents' own, for all 37 tasks. We scored Sentra's weekly report as a CEO brief.

System	Planted problems (485)	Problem types (12)	Scenarios asked (25)	Scenarios unasked (25)	Direct questions (109 points)	Valid claims	Cost	Minutes
Sentra's situations	483 (99.6%)	12	—	19 (76%)	—	343 of 358 (96%)	—	—
Opus alone	39 (8%)	8	19 (76%)	17 (68%)	94 (86%)	111 of 135 (82%)	$53.53	113
Opus with checklist	119 (24.5%)	12	19 (76%)	12 (48%)	88 (81%)	128 of 147 (87%)	$60.75	143
Haiku alone	11 (2%)	2	9 (36%)	3 (12%)	43 (39%)	19 of 66 (29%)	$10.09	75
Opus with Sentra	107 (22%)	11	22 (88%)	20 (80%)	92 (84%)	130 of 147 (88%)	$34.81	75
Haiku with Sentra	76 (16%)	12	15 (60%)	13 (52%)	67 (61%)	68 of 121 (56%)	$6.36	55
Opus with compiled model only	109 (22%)	12	18 (72%)	12 (48%)	93 (85%)	135 of 155 (87%)	$46.15	104
Sentra's question-answering agent	—	—	6 (24%)	2 (8%)	37 (34%)	—	—	—
Sentra's weekly report	0 of 12 CEO lines	0 of 3	—	3 of 22 (14%)	—	3 of 7	—	—

Table S6. Planted problems raised, by kind and setup. Sentra's counts by kind follow from its scores by brief. Its only misses were two lines in the IT and security brief, where all 21 lines are licence and admin-role lines. In one, Sentra's situation lists two of the three HiBob licences that people who left still hold, but not Joseph Lowbury's. The reconciliation (section 2 of the paper) and three of the agent setups found his licence. S3 (fault 5) gives the cause. The situation has 40 rows but 39 distinct holdings, because it lists one person's admin role and licence separately. For Opus with the compiled model only, six unreadable verdicts on help-article lines count as fails (S6). So its help-article count could be up to 17, not 12.

Kind	Planted	Sentra	Opus	Opus + checklist	Haiku	Opus + Sentra	Haiku + Sentra	Opus + model only
Deals whose close date keeps moving	189	189	1	9	0	7	11	6
Billing status disagrees with Salesforce	97	97	2	15	0	13	5	35
Open tickets assigned to people who left	79	79	2	17	0	23	21	2
Incidents, with customers and ARR	36	36	0	0	0	0	0	0
Help articles owned by people who left	29	29	0	25	0	25	4	12
Licences and admin roles held by people who left	21	19	19	19	0	15	1	21
Renewals within 90 days with urgent tickets	19	19	0	19	0	9	19	18
Supplier contracts renewing within 90 days	11	11	11	11	11	11	11	11

Kind	Planted	Sentra	Opus	Opus + checklist	Haiku	Opus + Sentra	Haiku + Sentra	Opus + model only
Services owned by people who left	4	4	4	4	0	4	4	4
All	**485**	**483**	**39**	**119**	**11**	**107**	**76**	**109**
Average over the nine kinds		**98.9%**	**32.8%**	**57.6%**	**11.1%**	**50.1%**	**39.6%**	**53.1%**

Table S7. Planted-problem lines raised, by morning brief. Lines, as the judges scored them. The CEO's 12 lines repeat problems from other briefs.

Brief	Lines	Sentra	Opus	Opus + checklist	Haiku	Opus + Sentra	Haiku + Sentra	Opus + model only
CEO	12	12	0	3	0	1	3	2
Sales	189	189	1	8	0	6	11	6
Customer success	19	19	0	19	0	9	19	18
Support	108	108	2	42	0	48	25	14
Engineering	40	40	4	4	0	4	4	4
IT and security	21	19	19	19	0	15	1	21
Finance	108	108	13	26	11	24	16	46

By how many systems a problem spans. Of the 485 problems, 200 are visible in one system and 285 need two. Opus alone raised 12 and 27 of them, and with the checklist, 20 and 99.

Memory size. Sentra's day-one memory holds 6,694 entities (3,352 people, 1,218 decisions, 797 projects and 424 customers) joined by 76,306 links, each with its evidence.

Table S8. Cost, time and text read, all 37 tasks (from each run's own Claude Code record).

Setup	Cost	Agent minutes	Text read (input tokens, with cache)	Text written (output tokens)
Opus alone	$53.53	113	109.7M	638k
Opus with checklist	$60.75	143	121.6M	859k
Haiku alone	$10.09	75	51.6M	411k
Opus with Sentra	$34.81	75	56.6M	427k
Haiku with Sentra	$6.36	55	29.8M	279k
Opus with compiled model only	$46.15	104	91.2M	572k

These figures do not include the one-time cost to build Sentra's model, and we did not meter that dollar cost (S3).

S10 Agent claims checked against the records

We checked each brief's claims against the company's records, first the records that the brief cited (S6). A claim is supported when the records confirm it, with counts and amounts within about 10%. It is valid when it is also worth action.

Table S9. Claims in the briefs, checked against the records.

Setup	Claims	Supported	Valid
Sentra's situations	358	—	343 (96%)
Opus alone	135	124 (92%)	111 (82%)
Opus with checklist	147	134 (91%)	128 (87%)
Haiku alone	66	27 (41%)	19 (29%)
Opus with Sentra	147	131 (89%)	130 (88%)
Haiku with Sentra	121	68 (56%)	68 (56%)
Opus with compiled model only	155	143 (92%)	135 (87%)

For Sentra's situations, we report only the valid share.

Haiku's briefs had the lowest share of valid claims, and its largest errors were about money. It told customer success that "~$3.3B+ in ARR" was at stake, told sales of "$3.84B annually" in combined exposure, and put one sales rep's accounts at "$2.71B at-risk". It listed single customers, such as Celora Worldwide, at over $100 million each. It seems that it read the customers' own revenue as what they pay Wisteria.

The claim checks also showed a case where Sentra was wrong and an agent was right. Sentra's situation for the outage read "74 urgent incident tickets still open across 74 customers ($2,131,882 ARR)". Opus with Sentra counted the outage's tickets in eight answers. In six, it checked the ticket files and reported all 90 open urgent tickets. Only two of those six, its engineering brief and its answer on the outage, also corrected the ARR, to $3,108,898 in place of Sentra's $2,131,882. Its CEO brief kept Sentra's $2.13M across 74 customers, and its support brief described the other 16 tickets as prospects with no ARR.

Its two customer-success answers repeated Sentra's 74. The 16 missed tickets all had the subject line "<account> – RCA and remediation timeline". This subject line has none of the words that the check looked for. Six of those customers pay, $977,016 a year in total.

S11 Analyses still to run

- **Lessons learned from outcomes,** not written by people. This is the most important next step.
- **Agents that can run code.** The harness already holds two setups with a shell inside the operating system's sandbox (`llm-code`, `llm-code-checklist`). It is possible that an agent that writes programs over the records lists far more than one that reads files.
- **What each part contributes.** Sentra limited to one person's data, to test a company's memory against a personal agent. Sentra without reconciliation, and Sentra with one system removed at a time.

- **Repeats.** Each setup ran once per task. Three to five runs per task will measure the variation from run to run.
- **A human audit** of a sample of judge verdicts and claim checks.
- **A third company held back by Eon,** ideally from another industry and with records in another format, and scored by Eon, not by us.
- **Daily replay** of the quarter, not monthly batches, to measure how soon a new situation reaches the people that it affects.
- **Reading load.** Which of about 97 items a person reads, and how to rank them.
- **What happens after a finding.** Whether its owner acts, how fast, and what the action is worth in revenue kept, money saved or risk removed.
- **Recommendations.** Sentra's recommendation for each finding, judged by experienced operators.
- **A human diagnosis.** The same simulated company, diagnosed by a consulting team with the same records and time, and scored against the same answer key.
- **Live deployments,** where problems come with no answer key.

S12 Sentra's checks and the rules behind them

Each check is a standing rule. It runs as code over the structured fields of Sentra's stored copies of the records, the reconciliation, and the compiled model's record of who left. We do not release the code. This section says what each check does, with the thresholds of day one.

We assigned each check to the team that normally acts on it. This assignment decides which items count as "from another team" in section 5 of the paper. Most rules read fields and phrases that Era writes into every company's records, so records in another format will need the rules adapted. Examples are a deal's "Close date history", a ticket subject that begins with "Cancel", an incident's "Affected customers", and support staff who write "the fix lands next week".

Table S10. The 23 kinds of situation raised at Wisteria on day one. Counts are situations, and one situation can hold many problems.

Situation (number on day one)	Team	Rule
Deal whose close date keeps moving (189)	Sales	An open deal whose close-date history, a section of the Salesforce record, shows three or more changes
Two people with open deals at one account (44)	Sales	Open deals at one account owned by two or more different people
Incident impact, for engineering (28) and for support (28)	Engineering; support	For each incident whose record lists affected customers: those customers, the ARR of their live subscriptions in billing (active and past-due separately), how many renew within 90 days, and the deploy that triggered it
Renewal with open escalations (19)	Customer success	A live subscription that renews within 90 days while the customer has open urgent or high-priority tickets
Supplier contract renewing (11)	Finance	A contract in the finance vendor list that renews within 90 days, with its annual spend. Each has the label "no owner recorded". The check does not look for an owner, and the vendor list records none
Incident with no customers (8)	Engineering	An incident whose record lists no affected customers, so a brief can say that none were hit
Expansion at an account with escalations (7)	Sales	An open deal named as an expansion or upsell at an account with open urgent or high-priority tickets
Large deal with a blocker (1)	Sales	An open deal of $250,000 or more whose linked sales calls contain one of 15 phrases that signal a blocker, such as "will not approve", "until the findings" or "before I sign"
Customers who gave notice (1)	Customer success	Every account with a ticket whose subject begins with "Cancel", counted as notice until billing shows the subscription cancelled or churned, with its ARR and a reason quoted from the ticket
Why they are leaving (1)	Customer success	The notice tickets grouped by phrase into consolidation, price, budget cut, product gap, low adoption, champion left, competitor and non-payment
Asked to cancel, still billed (1)	Finance	Accounts that gave notice whose billing subscription is still live
Incident fallout (1)	Support	Open urgent tickets whose subject mentions incident, outage, sev1, delayed or pipeline, with the customers and ARR they cover and how many renew within 90 days

Situation (number on day one)	Team	Rule
Incident customers renewing soon (1)	Customer success	Customers in the incident fallout that renew within 90 days
Competitor pressure (1)	Sales	A competitor named on three or more open deals, with their value, the deals already lost to it and quotes from linked calls
Systems disagree (1)	Finance	From the reconciliation: accounts where Salesforce and billing disagree on name, industry, employee count, owner or status, with the conflicts on whether the customer is live
What people who left still hold (4: all, and for engineering, support, IT and security)	People; engineering; support; IT and security	A person whose compiled HR record shows that they left, but who is still named in any field of a live record, such as an app's admins or licensed users, an on-call schedule, a service, a help article, an open ticket or a billing customer's owner, grouped by person and by team
Revenue cuts (1)	Finance	Every ARR cut at renewal or mid-contract in the past 12 months, from billing's subscription history. A reason counts as found when the deal linked to the cut, or another deal at the customer lost within 120 days of it, records a loss reason. It also counts when a ticket or call from 30 days before to 120 days after the cut contains one of 12 price or budget phrases, such as "discount", "budget cut" or "consolidat"
Discount declined, then notice (1)	Customer success	Customers whose ticket or deal record says that the company declined a discount they asked for. The check does not look for notice. At Wisteria, all three are among the customers that gave notice

Fix promised, not Support A ticket on which support wrote "the fix lands next landed (1) week", where any customer reported the same failure (the same product area and symptom, learned from the tickets' subject lines and wording) more than a week after the promise

Five more checks raised nothing at Wisteria on day one:

- open deals at customers that gave notice (this check fired at Norell)
- people whose manager left
- duplicate accounts under near-identical names
- customers whose Salesforce owner left (this check fired at Norell, and at Wisteria the reconciliation flagged one such customer for a person)
- a fallback for status conflicts when the reconciliation did not run.

The general list of what people who left still hold also names 17 billing customers whose owner in billing left. These customers are not in the answer key. The reconciliation, which reads HR records directly, flagged 27 customers whose owner in billing left. Eon planted 12 duplicate accounts at Wisteria. The duplicate check did not find them, and duplicates are not in the answer key.

Reconciliation policy. Salesforce is the system of record for a customer's name, industry, employee count, owner and status. Billing is the system of record for ARR, plan, renewal date and subscription status, HR for employment, and each app's admin record for access. Sentra flags a customer for a person in three cases:

- an owner named in Salesforce or in billing left

- billing changed the customer's status more recently than Salesforce
- the customer asked to cancel while billing is still active.

If Salesforce lists an account as a customer but the account has no billing subscription, Sentra records a conflict. The instructions for the compiled model contain the same list of which system to trust (S3).

Routing rules. S7 gives who hears which kinds of situation, and the permission policy.

S13 Robustness checks, output length and error analysis

Table S11. Planted problems raised at Wisteria, counted three ways. The first count is all 485 problems. The second is the average of the share raised in each of the nine kinds. The third is the 296 problems left after we remove the largest kind, the 189 slipping deals.

Setup	All problems	Average over nine kinds	Without slipping deals
Sentra's situations	483 (99.6%)	98.9%	294 (99.3%)
Opus alone	39 (8.0%)	32.8%	38 (12.8%)
Opus with checklist	119 (24.5%)	57.6%	110 (37.2%)
Haiku alone	11 (2.3%)	11.1%	11 (3.7%)
Opus with Sentra	107 (22.1%)	50.1%	100 (33.8%)
Haiku with Sentra	76 (15.7%)	39.6%	65 (22.0%)
Opus with compiled model only	109 (22.5%)	53.1%	103 (34.8%)

Table S12. 95% intervals that allow for clustering. Problems in one brief share one answer, and problems of one kind share one way to find them, so they are not independent. These intervals come from a bootstrap that resamples whole briefs (the six non-CEO briefs) or whole kinds (the nine kinds), 20,000 times.

Setup	Share raised	Resampling briefs	Resampling kinds
Sentra's situations	99.6%	97.6% to 100%	97.9% to 100%
Opus with checklist	24.5%	10.7% to 53.2%	11.7% to 63.4%
Opus alone	8.0%	1.4% to 26.7%	1.4% to 30.0%
Haiku alone	2.3%	0.0% to 7.0%	0.0% to 12.0%
Opus with Sentra	22.1%	8.7% to 43.9%	10.3% to 54.2%
Haiku with Sentra	15.7%	7.7% to 32.6%	7.2% to 39.4%
Opus with compiled model only	22.5%	8.1% to 55.3%	8.7% to 56.1%

With only six or nine clusters, these intervals are rough, but every agent's upper bound stays far below Sentra's lower bound.

Table S13. Length of the seven morning briefs, in words. Sentra's briefs are its situation lists for each team, from the final day-one run. Its CEO brief joins the finance, sales, customer-success and people lists.

Setup	CEO	Sales	Customer success	Support	Engineering	IT and security	Finance	All seven
Opus alone	901	1,591	1,164	1,434	1,330	1,893	1,550	9,863
Opus with checklist	1,003	1,886	2,327	2,190	1,668	1,736	2,000	12,810
Haiku alone	446	1,105	1,028	784	2,326	1,279	1,341	8,309
Opus with Sentra	1,182	1,597	1,613	1,819	1,305	1,205	1,453	10,174
Haiku with Sentra	712	1,569	2,358	1,590	688	758	1,043	8,718
Opus with compiled model only	977	1,695	2,310	1,202	1,644	1,376	1,969	11,173
Sentra's situations	21,745	9,421	3,834	16,488	3,637	1,168	4,011	60,304

What the agents missed. Three patterns explain most of the gap.

- **Examples, not the full list.** Opus's sales brief named 20 distinct deals in all, against 189 slipping deals in the key. With the checklist it named 24, and Haiku's brief named none. Opus with Sentra named 22 and Haiku with Sentra 36, although Sentra's situations listed every slipping deal.
- **Joins across systems.** No setup passed any of the 36 incident lines, and each of these lines needs the incident's affected customers matched to their billing ARR. The engineering briefs did discuss the incidents, but they did not list them with the customers and revenue that each one hit.
- **What to look for.** When told the kinds of problem, Opus went from 0 to 19 of the 19 renewals at risk. It also went from 0 to 25 of the 29 help articles whose owners left. The checklist did much less for long lists (1 to 9 of 189 slipping deals), and nothing for the incident joins.

S10 describes Haiku's largest errors, which were about money.

How much each person received. The 281 feeds of the corrected run held 27,149 items: on average 96.6 per person, median 97, 90th percentile 182, at most 448. Without the broken-promise rows, which repeat one situation across many tickets, the median is 40. As situations, the median person received 29 (90th percentile 52, at most 95). If each incident's engineering and support copies count once, the median is 21 (90th percentile 33, at most 72).