endsight

CASE STUDY · RESPONSIBLE AI IN HR

The Executive Intelligence System

How I keep a running record of my team with AI, and what it changed about performance reviews

Josh Carroll · CEO, Endsight · November 2026

Why I built it

Endsight is a managed services provider with about 130 people. My job runs on conversation: weekly 1:1s with my direct reports, skip-level meetings, client and vendor meetings, and a steady stream of email and Teams. Those conversations are where people tell me how they're doing, what they're working on, and what they need from me.

For years, the record of those conversations was my memory and whatever notes I took. Commitments I made in passing got lost. I prepped for 1:1s in the ten minutes before them. And when annual reviews came around, I was writing mostly from the last month or two, because that's what I could remember.

I built the Executive Intelligence System (EIS) to fix that: something that keeps the record for me and hands it back when I need it.

What it is

The Executive Intelligence System is a personal operating layer I built on Claude starting in the spring of 2026. It reads my work information surface, turns it into a structured and connected record, and produces briefs from that record.

The design idea is that every brief is a view into one shared state, not an independent report. My meeting prep, the daily queue, the weekly briefs and the board briefing all read the same commitments tracker and the same dossiers, so they don't contradict each other.

It isn't a software product. It's a set of Claude projects, each with its own written standard work, scheduled to run in order every morning, plus connectors into systems we already use: Microsoft 365 and Teams meeting transcripts, Workday, ConnectWise, and Tervene, where our strategic initiatives (Rocks) live. I built it without a development team, mostly by describing what I wanted and refining it with Claude. A small desktop app, the EIS Console, is where I review and correct everything.

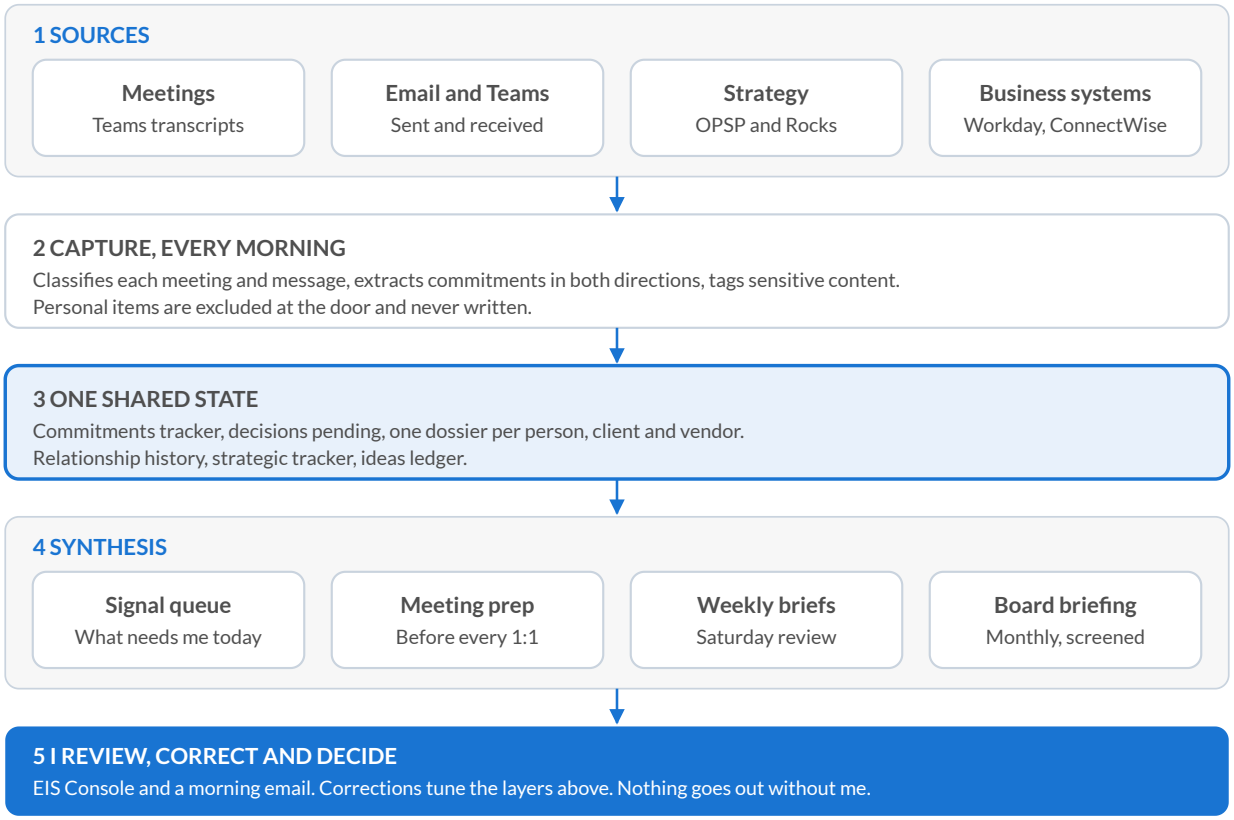


Figure 1. Sources feed capture, capture writes one shared state, and every brief reads from that state. My corrections tune the layers above.

The employee dossier

Each of my direct reports has a dossier: a single running document the EIS updates after every 1:1. Our 1:1s happen on Teams and are transcribed, and the morning after a meeting the system reads the transcript and adds a short entry: the themes we discussed, commitments in both directions, how the person seemed, and anything notable, like a first pushback, a breakthrough, or a missed commitment. Notable moments are kept word for word. Everything else is summarized at the end of each quarter so the file stays readable.

A few other parts of the dossier matter as much as the meeting notes:

- **Development plan.** The one to three development targets we've agreed on, each with a dated log of progress. A target only counts as active when I set it or we both agreed to it. If the system hears something that sounds like a new goal, it marks it as proposed and asks me to confirm.

- **Connection notes.** Birthdays, family, interests and life events, limited to what the person has chosen to share with me. Nothing is inferred or looked up.
- **Coaching notes on me.** Each entry includes a note on how well I held to our own 1:1 standard: whether I listened, followed up on what I said I'd do, or cut someone off. The record holds me accountable too.

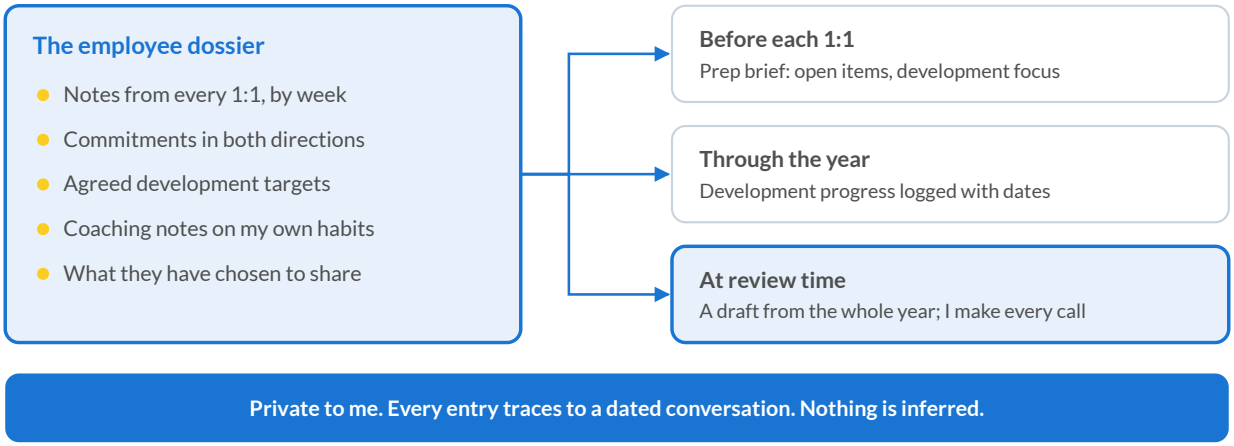


Figure 2. One record per person feeds the 1:1 prep, the development conversation and the annual review, so all three work from the same facts.

Before every 1:1

At 6:00 on the morning of a 1:1, the system writes me a short prep brief from the dossier. It opens with development: where each target stands, how long it's been since we talked about it, and one question worth asking this time. Then it lists what I owe the person, what they owe me, themes that keep coming up, and anything personal they've mentioned that I should follow up on. I can read it in five minutes and walk in prepared.

At review time

This year I used the EIS to prepare several of my direct reports' annual reviews. For each one, I started with my own read on the person and how they were performing. I asked the system to validate or challenge that read against what their dossier had captured across the year and their overall performance. Then it compared both against the person's own self-review. Where it helped, I added engagement survey results and hiring numbers.

What came back each time showed me where the record supported my read, where it was more mixed than I'd thought, and where the self-review saw things differently. From there it drafted an answer to every review question, in my voice, with specific examples from across the whole year instead of just the last month, and it listed what it couldn't see and the questions I'd need to settle.

Then I did the manager's work. I rewrote the hard paragraphs in my own words. I cut numbers from sources I didn't trust. I corrected places where a dossier had the story wrong, and I took out topics that didn't belong in a review at all. Every rating and every decision about what went in was mine.

> **It didn't save much time. It made the review better: stronger examples, a longer view of the year, and gaps named instead of glossed over.**

The guardrails

These are the rules that keep it on the right side of the line:

- **It's private to me.** Dossiers aren't shared with anyone. Anything sensitive, like compensation, a performance plan or a personal situation, carries an extra marker that keeps it out of every summary, and anything leaving the system passes a check where I decide what goes.
- **People know it's recording.** My meetings are recorded automatically, and my team knows it. If someone wants to discuss something sensitive off the record, they ask me to stop the recording, and I do, no questions asked.
- **It records, it doesn't infer.** Every entry traces to a dated conversation. It doesn't guess at someone's motives, health or circumstances, and it doesn't change a goal's status because time passed.
- **It's evidence, not a verdict.** There are no rankings or scoreboards of people. Data about someone's work is input to a conversation, never a score.
- **I make the decisions.** The system prepares. I write the review, set the ratings and deliver it in person.
- **It runs on a company-managed account.** Never a personal one.
- **It's work only.** It captures my role at Endsight. My own personal meetings and email are excluded at the door.

Where it's going

The next step is making this useful to other managers at Endsight, not only to me. That's a bigger build, and the guardrails above have to hold when more than one person relies on the record.

Appendix: Building your own EIS

A handful of separate AI projects won't add up to much on their own. What makes the EIS work is the connective tissue: one shared folder, one agreed format for each kind of file, and a clear split between the projects that capture the record and the projects that use it. One project, which I call Control, designed all of it with me and keeps it coherent as it grows.

The smallest version that works has three projects: Control, Capture and 1:1 prep. Review prep is a fourth, used at review time. These prompts were written for Claude projects and will need some adapting for ChatGPT or Copilot.

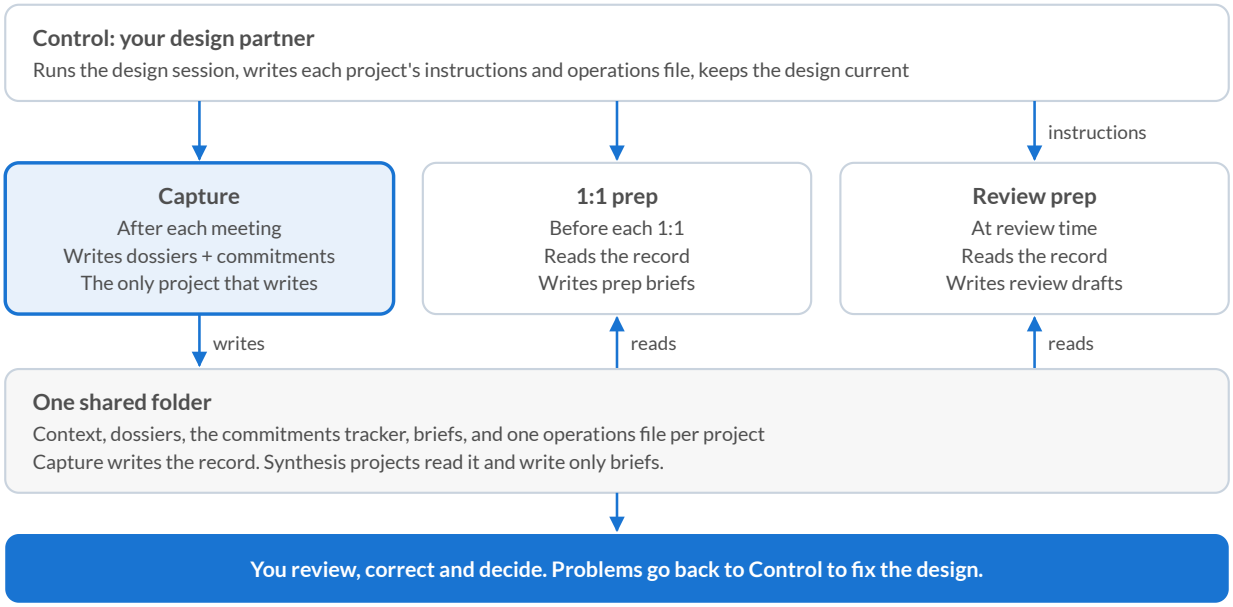


Figure 3. Control writes the instructions. Capture is the only project that writes the record; the others read it and write briefs. You review everything that comes out.

Before you start

- Use a company-managed business account, never a personal one.
- Follow your AI policy and your recording and notice rules. Get consent before you record. Some states, including California, require everyone in a conversation to agree to a recording, and that includes clients and vendors on a call. Tell your team what the system keeps about them and how you use it, and let anyone ask you to stop the recording, no questions asked.
- Pick one folder your AI tool can read and write, such as a synced OneDrive or SharePoint folder. Everything lives there.
- Lock the folder down. Only you should be able to open the dossiers. Check the sharing settings with IT, including backups and anyone with admin rights to that account.

- Write as if the employee will read it. In California, employees can ask to see personnel records related to their performance, and notes used to prepare a review may count. Anything in the system could also come up in a dispute. Keep only what you'd be comfortable showing the person.
- Decide how long you keep it. Set a retention period for transcripts and dossiers, and decide what happens to a dossier when someone leaves. Review your setup with employment counsel; these are practices that work for us, not legal advice.

Step 1: Connect your sources first

Capture can only record what it can reach, so connect your sources before you design anything. In your AI tool, connect your calendar, email and chat, and confirm it can retrieve your meeting transcripts. For most teams that means the Microsoft 365 or Google Workspace connector, plus whatever records your meetings. Test each one: ask the tool to list yesterday's meetings and pull one transcript. What comes back is what Control has to design around, and a gap found now is far easier to solve than one found after the projects are built.

Step 2: Set up the shared folder

Start with this layout. Control will refine it during the design session.

```
my-eis/
  README.md          The design (Control)
  build-plan.md      What's built and what's next (Control)
  context/           Your role, team list, 1:1 standard, AI policy
  operations/        One operations file per project (Control)
  dossiers/          One file per person (Capture)
  trackers/
    commitments.md   Open commitments (Capture)
  briefs/            1:1 prep and review drafts (synthesis projects)
```

Put a few things in context/ before you start: a paragraph about your role, your team list, and how you run a good 1:1. If that last one isn't written down yet, Control can help you write it.

The operations folder is what keeps the system improving. Each project reads its own operations file before every run. When a project gets something wrong, you don't just fix that one output. Control adds a correction to the project's operations file, so the same mistake doesn't come back next time.

Step 3: Create Control and run a design session

Control is your design partner. It doesn't do the daily work. It interviews you, proposes the design, writes the instructions and operations file for every other project, and keeps the design current as you learn. Create a project called Control, give it access to the shared folder, and paste this in as its instructions:

> **PROMPT**
>
> ```
> You are Control, the design partner for my personal AI operating system. I'm [name], [role] at
> [company]. The goal is to keep a reliable record of my work with people and hand it back to me
> when I need it, starting with my direct reports.
> ```

```
Our shared folder is [location]. You can read everything in it. You write only README.md (the design), build-plan.md (what's built and what's next) and the files in operations/. You never edit dossiers, trackers or briefs.

Start with a design session. Interview me one question at a time about:
- My role, my direct reports and my regular meetings
- Which sources are connected (calendar, email, chat, meeting transcripts, HR system), what each one actually returns, and anything a project would need that isn't connected yet
- What I lose track of today, and what I most want handed back to me
- Boundaries: what's out of scope, what's confidential, what the system must never do on its own, and our AI and recording policies; who can access the folder, how long we keep transcripts and dossiers, and what happens to a dossier when someone leaves

Then propose the design in README.md:
- The folder structure
- Each project: its one job, whether it is capture or synthesis, what it reads, what it writes, and when it runs
- The format of each file (dossier, commitments tracker, prep brief), so every project reads and writes the same structure
- Guardrails that apply to every project
Start with Capture and 1:1 prep. List everything else as later.

After I approve the design, write each project's instructions and its operations file, and keep build-plan.md current as we build.

Standing rules:
- Capture writes the record; synthesis reads it. Only capture projects write dossiers and trackers. Synthesis projects write only briefs.
- If there is ever more than one capture project, they run one at a time, never at once.
- Every project reads its operations file before every run. When I report a problem, add a correction to the right operations file instead of only fixing the output, and note the change in build-plan.md.
- Private to me by default. Anything that leaves the system gets my review first.
- The system prepares; I decide. Nothing is sent, rated or decided without me.
```

Plan on an hour for the first design session. Give honest answers about what you lose track of, because that's what shapes the first projects.

Step 4: Build the starter projects

The prompts below are starting points. Have Control adapt each one to the design you agreed on and write its operations file, then create each as its own project with access to the shared folder. Run them on demand for a few weeks before putting them on a schedule. Run each one beside the way you do it today, and trust it only once it has proven itself.

1. Capture

Reads meeting notes or transcripts once. Writes the dossiers and the commitments tracker. It's the only project that changes the record.

PROMPT

```
You are my Capture project. Before every run, read operations/capture.md and context/1on1-standard.md. For the date I give you, pull my meetings from my calendar and their transcripts from [your transcript source]. If I hand you notes for a meeting that wasn't recorded, use those instead. Read each meeting once and update the record.

1. Dossier. For a 1:1 with a direct report, add one dated entry to dossiers/[name].md with:
- Themes we discussed
- How they seemed, in one sentence, based only on what was said
- Notable: yes or no. Notable means a first, a breakthrough, a setback, a significant decision, direct praise or criticism, or a theme that has now come up three times. Quote notable moments exactly.
- Development: progress on any agreed development target
- Coaching for me: one honest note on how I ran the 1:1 against my 1:1 standard
Then update the standing sections when the conversation supports it:
- Development plan: each target with a status. A new target is "proposed" unless I stated it or we both clearly agreed. Change a status only when one of us said so.
- Connection notes: birthdays, family, interests and life events, only what the person chose to share. Leave out anything touching health, pregnancy, religion or another protected characteristic.

2. Commitments. For any meeting, find every concrete commitment where I am the owner or the person it's owed to, and record it in trackers/commitments.md with an ID, what was committed, who owns it, who it's owed to, the source and date, a due date if one was given, and a status of open or closed. Check for duplicates first. Close an item only when I say it's done or a source clearly shows it was. Skip ideas, FYIs, same-day logistics, and commitments between two other people.

Rules:
- Read each file before you change it, and change only what this meeting supports.
- Record what was said. Do not infer motives, health or personal circumstances.
- Never record medical details, disability, pregnancy, the reason for a leave, religion or any other protected characteristic, even if the person mentions it. Note only that a personal or medical matter came up and may need HR, with no detail.
- Mark anything about compensation, a performance plan, an investigation or a complaint as SENSITIVE.
- If the notes contradict something already recorded, flag it for me instead of overwriting it.
- Personal meetings are out of scope. Write nothing for them.
- If the recording was stopped, write nothing about the unrecorded part unless I tell you to.
- Describe behavior, not personality.
- Write only to dossiers/ and trackers/. End with a short list of what you changed.
```

2. 1:1 prep

A synthesis project. Reads the record. Writes only prep briefs.

PROMPT

```
Before every run, read operations/prep.md. Before a 1:1, I'll tell you who it's with. Read dossiers/[name].md, trackers/commitments.md, and any performance metrics I've provided for them (in context/metrics/ or pasted in). Then write a prep brief to briefs/prep-[name]-[date].md that I can read in five minutes, as an agenda:

1. Development focus: each active target, its last progress and date, how long since we discussed it, and one question to ask this time. If there's no development plan, say so and suggest how to start one.
2. Performance: the metrics available for their role, how they've moved since our last 1:1, and the one or two worth discussing. Treat numbers as a starting point for the conversation, not a
```

```
verdict. If no metrics are available, say so.
3. Where they're stuck: blockers or recurring problems from recent entries, commitments that
have stalled, and any request for help from last time and whether it was resolved. Suggest one
question that opens space for them to say where they need coaching or assistance.
4. What I owe them, and what they owe me, oldest first.
5. Themes that keep coming up, with dates.
6. Follow-ups on anything personal they've shared.
7. One coaching reminder for me, based on my recent coaching notes.

When I ask what I owe or what's owed to me, answer from trackers/commitments.md: open items I
own, or open items others owe me. Mark anything with no activity in 14 days as aging.

Use only what's in the record and the metrics I've provided. If something is missing, say what's
missing rather than filling the gap. Never edit dossiers, trackers or metrics; if something in
them looks wrong, tell me. Write only to briefs/.
```

3. Review prep

A synthesis project, used on demand. Reads the record, your own assessment and the self-review. Writes only review drafts.

PROMPT

```
Before every run, read operations/review.md. Help me prepare an annual performance review. I'll
start by giving you my own read on the person and how they've performed this year. I'll also
give you their self-review and our review questions and competencies. Read dossiers/[name].md
and trackers/commitments.md.

First, test my read:
- Where does the record support it? Cite dated examples.
- Where does the record challenge it, or look more mixed than I think?
- Where does the self-review see things differently from my read, or from the record?

Wait for me to respond. Then draft an answer to each review question in my voice, in
briefs/review-[name]-[year].md, using:
- Specific, dated examples from across the whole year, not only recent months
- Wins and challenges quoted as recorded
- Progress on each development target: achieved, carried forward or stalled
- Proposed development targets for next year, marked as proposals

Then list phrases in the draft that have no example behind them, and what you couldn't see that
I should check before I finalize. Also flag language that describes personality instead of
behavior ("abrasive," "not a culture fit"), wording I wouldn't use for someone else in the same
role, and any reference to age, health, leave, family status or another protected
characteristic.

Rules:
- Do not assign ratings. I set every rating.
- Leave out protected characteristics, leaves and accommodations entirely, even if the record
mentions them, and tell me if it does.
- Do not use anything marked SENSITIVE unless I say so.
- Don't characterize the person. Describe what they did and what happened.
- This is a draft. I will rewrite the hard parts in my own words and deliver the review in
person.
- Never edit dossiers or trackers. Write only to briefs/.
```

Step 5: Automate and grow

Once the starter projects are producing output you trust, go back to Control and work through the next round together:

- **Put it on a schedule.** Run Capture each morning against the prior day's meetings, then have prep briefs ready before your first 1:1. Capture always runs first, so the briefs read a fresh record.
- **Add projects as you find the need.** New capture projects for other sources, such as email and chat, and new synthesis projects such as a weekly summary, a daily list of what needs your attention, reply drafts in your voice, or an update for your owner or board. Each one gets one job, its own operations file, and a clear place on one side of the line: capture or synthesis.
- **Keep the design honest.** When something goes wrong, take it to Control, fix the operations file, and record the change in the build plan.