FIRST STRATEGY · CASE STUDY

HVAC and Field Services

A real engagement, anonymized to industry label. The full case study: the story, the deliverables we produced, and the plays that ran it.

The diagnosis everyone agreed on was wrong

The Director of Operations had a whiteboard with three vendor logos on it, and three stories that ended the same way. Each company had pitched AI for scheduling. Each demo had dazzled. Each pilot had died quietly once his dispatchers actually tried to use it. By the time he called us, on a referral, he had stopped asking for a scheduling tool and started asking a better question: where does AI actually help in an operation like ours, and where is it just marketing.

The honest answer was that nobody in the room knew yet, including us. So we asked for a day on his floor before we proposed anything, because the operation that runs in the documents is never the operation that runs.

The morning went the way he expected. Leadership walked us through dispatch, the place every vendor had pointed. Then we left the conference room, and the floor started disagreeing with the diagnosis almost immediately. We watched the lead dispatcher first, certain we were looking at the bottleneck. We were not. She was not scheduling in the dispatch software at all. It sat minimized on her screen while she worked from a spreadsheet she had refined over years, color-coded by hand, green and yellow and red, with a note in nearly every cell. The software knew where her technicians were. It did not know which one runs long on certain days and why, or which customers always add work once the truck is in the driveway, or how a particular route backs up after three in the afternoon. "This is how I schedule," she told us. "The software is for billing." Automating her would have torn out the one part of the operation that worked.

The real friction was downstream, and the billing manager had been naming it for years to anyone who would listen. When we asked to watch her desk, she said, "Please. Someone needs to see what we are dealing with." What we saw: a clerk pulling paper job tickets from a basket, squinting at a technician's handwriting, guessing at a part number, calling a technician who was already on the next job and would not pick up. By late morning, three invoices were stuck waiting for a callback, and two had already gone out with errors that someone would have to chase down later. This was a normal morning. More than a third of invoices came back needing correction. Nearly half of jobs triggered a clarification call. Billing ran three days behind the work. About six hours a day, in that one location, went to cleaning up a handoff that should have taken seconds.

When we traced it back, roughly eighty percent of those invoice errors entered at a single step: the paper form in the technician's hand. Not scheduling. A piece of paper. The thing every vendor had wanted to sell was barely connected to the problem the company actually had. That gap, between the diagnosis leadership inherited and the one the floor revealed, is the whole case. It set the rule for everything after: we would not trust a diagnosis we were handed, including our own, until cheap evidence forced it to be true.

Cheap experiments beat confident builds

Knowing where the friction lived was not the same as knowing what to build, and the fastest way to waste the next six months would have been to forget the difference. A confident team starts building. We ran three experiments first, days each, designed to kill our own ideas before they got expensive.

The first one tested the inherited theory head-on. We put a scheduling visibility board in front of technicians for a week, the exact thing the vendors had always promised. Nobody used it. They get their next job by text and never came back to look at a board. By Friday it had quietly become a place to post a fantasy football league. The scheduling theory was dead, and it had cost a week, where no one could resurrect it later.

The second tested the obvious replacement for paper: a digital form. It was slower than paper. Tapping through dropdowns took longer than scribbling, and it failed the moment a technician lost signal in a basement. They abandoned it within days. The failure taught the thing that mattered most: digital capture could work, but not if it asked technicians to change how they work.

The third tested something lighter. Keep the paper. Photograph each finished ticket. Let AI read it into billing, and when the AI could not read a field, text the technician to confirm. Invoice errors fell from thirty-eight percent to nine, and the technicians did nothing different. It worked, and it opened a bigger question: if a photo plus AI could do this, could a photo plus a spoken word retire the paper entirely. Each experiment narrowed the path by testing, not asserting, and the one that worked did not just validate an approach. It showed the shape of the thing worth building.

The build that worked, and the one that broke

The hard line was fifteen seconds of capture in a customer's driveway, eight the target, because anything slower than paper would be rejected no matter how clever it was underneath. Technician photographs the finished job, records a short voice note, AI extracts what billing needs. The first field test taught a lesson no specification could have. The system parsed a complicated part number without blinking, then returned pure garbage on a commercial rooftop, defeated by wind across the microphone. The hard part was never the vocabulary. It was the noise. So the team stopped chasing perfect voice and gave it a partner: voice as the primary path, a photographed paper form as the fallback, the same AI reading both, and then retrained the model on real field audio, compressors and truck engines and wind, not recordings made in a quiet office.

Then came the pilot, and the person who decided whether it could be trusted was the billing manager. For years she had been the one catching what the paper got wrong. Now she reviewed every submission the AI produced, photo, transcription, every extracted field. One afternoon she stopped on a routine water-heater swap. The technician's note was clear enough, a standard replacement, a job of about two hours. The AI had recorded the duration as eight. She pulled the photo, confirmed it was an ordinary tank, screenshotted the error, and walked it to the developer's desk. That catch was the system working as

designed: not an AI that is never wrong, but an AI whose mistakes a human sees before a customer does. Her error log became the training data that made the next week better and the checklist the next location's reviewer would use.

The pilot held. Invoice errors at nine percent, down from thirty-eight. Clarification calls near zero. Billing same-day instead of three days behind. Capture averaging eleven seconds, faster than paper. That one location recovered about forty-seven thousand dollars in a single month, billing that would otherwise have been disputed, delayed, or quietly written off.

Success nearly caused the next mistake. At the executive review, leadership looked at the result and wanted to rip out the entire back-office system and rebuild it from scratch. We held the line, and the Director of Operations held it with us: replace one component at a time. We had just proven you can fix a single handoff and validate it before touching the next. Tearing out everything at once is how a project becomes a multi-year overrun that still does not work. The decision was to expand the proven piece, location by location.

The second location, residential like the first, went smoothly and proved the result was not a fluke. The third broke on purpose. It was fully commercial, where jobs run across days instead of hours, and the system had quietly learned a false rule from the first two locations: that a capture means a finished job. In commercial work, a technician captures at every phase, so the system started firing invoices for work that was not done. Errors climbed to fifty-five percent, worse than the paper it had replaced. The team pulled it within days and put that location back on paper, which at least was stable. For a stretch, the project's best location and its worst were running the same software.

That break was not a failure. It was the lesson arriving on schedule: a working system dropped into a real context reveals what no specification anticipated. Three days of watching commercial crews showed the difference. Commercial jobs have milestones, and billing should fire on a milestone, not on every capture. Building the fix surfaced a constraint nobody had flagged, a billing system that did not support milestone billing, which forced an intermediary layer and a timeline that slipped. The developer owned the missed estimate directly, to the client, in plain terms. The honesty held the trust through the delay. The commercial version brought that location from fifty-five percent down to eleven, and the rollout finished, residential and commercial, two locations to a wave.

Stable is not the same as governed

Live everywhere and stable, the system invited the last mistake: assuming the work was done. Stable is not the same as governed, and a system that looks fine in aggregate can be failing quietly underneath. This one was. Extraction accuracy on plumbing jobs drifted downward while the headline dashboard stayed green. The training data had skewed toward the trade that adopted first, so the model had learned that trade deeply and plumbing shallowly. The aggregate hid it. What caught it was watching disputes by service type rather than by total, the same instinct the billing manager had brought to the pilot. The fix had three parts: retrain on balanced data, alert whenever any segment fell more than five percent from

its baseline, and run a standing weekly review of accuracy by segment. Customers billed wrong during the drift window were credited. Plumbing recovered, and the rule went into the record for good: AI optimizes for exactly what you measure, so measure the segments, not just the total.

Governance then became structure instead of vigilance. The team defined three tiers of oversight by job risk. Low-risk routine jobs auto-approve, with random spot-checks and monthly attempts to deliberately break the system. Medium-risk jobs get a one-screen human check on price, parts, and duration. High-risk jobs, the complex repairs and the commercial projects, get full review with no exceptions. A job type earns its way down a tier only on evidence: ninety days above ninety-five percent accuracy, disputes below the old baseline, zero boundary violations. Routine residential earned it. Nothing earned it by decree. Autonomy was grown the way trust is.

Stepping back on purpose

The billing manager's job had inverted. She started as the person who fixed what the paper got wrong. She became the person who catches what the AI misses, and feeds each correction back so it misses less the next time. Nobody was laid off. The clerks who had spent their mornings deciphering handwriting became exception handlers, and the same team now carries more locations than it could before. Capacity grew where headcount would have.

Across the operation, invoice errors settled at nine percent on residential work and twelve on commercial, down from thirty-eight. Billing runs same-day. Recovered billing reached about two hundred forty-seven thousand dollars a month, on the order of three million dollars a year.

Then we did the thing the method is actually for. We grew the client's autonomy the way the build grew the AI's: trust earned, agency transferred, oversight loosened only as reliability was proven. We led the first project and built it ourselves. We handed the second to their team and trained a guide on their side while we co-guided and held the technical calls. We oversaw the third while they led it outright. They now run the process on their own, and we check in monthly on a low advisory retainer. The point was never to be permanently needed. It was to build the muscle and step back. The first problem is solved, and the team knows it will not be the last, because now they know how to find the next one.

Day One Proposal

Day One

Prepared for the operator, on referral.

What this is

One day in your business with your leadership. Real work, not slides.

In the morning, we sit with you and your leadership. What is running. What is stuck. Where AI already shows up in the business, and where it does not. What you have already tried, including the AI scheduling that impressed in a demo and never got used.

The rest of the day, we walk the work with the people who do it. The dispatcher. A technician through a full job. The billing desk. What their day actually looks like, what slows them down, where they would want help. On the floor, not in a side room. Because you run across [several] locations, the day follows the real work end to end rather than sitting in one office. This is the part of the day that does the work.

We come ready to listen and to think on our feet. No prepared deck.

What you walk away with

A playbook, within two weeks. Not a deck. Not a recommendation memo hiding in a PDF. A written read for operators.

The day pulls signal from your floor. The two weeks after are when we test what we heard and decide what would matter most. That work turns the signal into a sequenced plan you can run.

The playbook answers three questions:

- Where AI fits in your business, and where it does not.
- The highest-leverage moves we saw, sequenced so you can act on them in order. A roadmap, not a list.
- What it would take to run the sequence: with your own team, with another firm, or with us.

The playbook is yours. Run it however makes sense.

What we need from you

- Your leadership for the morning, including the Director of Operations who brought us in.
- Access to the people doing the work for the rest of the day: dispatch, the field, and billing.
- A day on the floor, in the business, not on video calls.

The terms

A flat fee of [flat fee] for the day and the playbook. Travel and expenses billed at cost, on top.

No retainer. No commitment beyond the day itself. If we are the right fit for what comes next, we will already have been talking about what that looks like. If we are not, the playbook is still yours to run.

What happens next

After the playbook, you decide. Run it with your own team, hand it to another firm, or build it with us. If the work points to a build we are right for, we will scope it in a separate proposal once the playbook has shown what is worth building.

Day One Audit

The one-line finding

Your AI opportunity is not where you were told to look. Leadership and three prior vendors pointed at scheduling. A day on your floor found the friction somewhere else entirely: the handoff that carries job data from the field to billing. That handoff corrects more than a third of its invoices, loses roughly six hours per day per location to rework, and runs a three-day billing reconciliation lag. The waste is large, contained to one handoff, measurable against a clear baseline, and reversible. It is the right place for the first AI work. Scheduling, where every vendor wanted to sell, is barely involved in the problem.

How we looked, and how we measured

One day in the business, with a short follow-up to quantify what we saw. The morning with leadership, capturing the diagnosis you carried without accepting it. The rest of the day on the floor, shadowing the work as it actually happens: dispatch, a technician through a full job, and the billing desk.

The numbers in this audit are measured, not assumed. We shadowed [a sample of jobs across three of the locations], timed each handoff with a stopwatch, and counted corrections against [a sample of recent invoices pulled from the billing system]. Where a figure is an extrapolation rather than a direct count, this audit says so. We did not optimize the documented process; we found the real one, and the most load-bearing parts of it are nowhere in any process document.

The systems landscape

Four disconnected tools do the work that should belong to one, and none of them is the system you think it is.

System	Official role	Actual role
Dispatch software	The scheduling system of record	A passthrough kept current for billing, not used to schedule
Dispatcher's spreadsheet	Not official	The real scheduling system, refined over years, on one desktop
Paper job tickets	A field formality	The actual source of truth for everything downstream
Accounting and billing system	Billing of record	Fed by hand from paper, the place errors surface

Paper is the connective tissue between all four. Paper cannot be searched, verified, or automated. Every downstream system inherits whatever the paper got wrong.

Stakeholder map

Each role defines the problem differently. None is wrong about their own pain. None sees the whole.

Role	What they own	Where their pain is	Their definition of the problem
Operations leadership	Throughput and dispatch	Manual work everywhere	Scheduling is the bottleneck
Lead dispatcher	The schedule	The software cannot hold her judgment	The software is for billing, not for her
Field technicians	Job capture	Anything slower than paper	Leave my workflow alone
Billing team	Transcription and invoicing	Drowning in corrections and clarification calls	We fix more invoices than we send
Finance	Margin	Quiet revenue leakage from billing errors	The errors are eating margin

The gap between leadership's definition and the billing team's definition is the whole case. Leadership looked where the vendors pointed. The people downstream had been naming the real problem for years. The billing manager put it plainly: we spend more time fixing invoices than sending them.

The work, end to end

We followed one residential job from dispatch to invoice. The path crosses six handoffs and four systems, with paper in the middle of all of them.

1. Dispatcher assigns the job from her spreadsheet.
2. Dispatch software updated afterward, for billing's benefit.
3. Technician arrives, diagnoses, completes the work.
4. Technician records the job on a paper ticket by hand, gets a signature on the carbon copy.
5. Paper tickets pile up and are dropped at dispatch at end of day.
6. The next morning, a billing clerk re-keys the paper into the billing system, squinting at handwriting, guessing at part numbers, calling technicians who are back in the field.

We watched this play out at the billing desk in real time. By late morning, on a normal day, [three invoices were stuck waiting for a technician callback, and two had already gone out with errors that would need

correction later]. The invoice goes out days after the work. A large share comes back disputed.

Friction quantified

Measured on the floor, by timing handoffs and counting errors across the jobs and invoices we sampled:

Friction point	Frequency	Time impact	Error rate	How measured
Paper ticket completion	Every job	5 to 10 min per job	25% illegible or incomplete	Timed across shadowed jobs; legibility judged at the billing desk
Re-keying paper into billing	Every job	8 min per job	15% data-entry errors	Timed at the clerk's desk; errors counted against the source ticket
Billing clarification calls	~40% of jobs	12 min per call	Blocks the invoice 24+ hours	Counted against the sampled jobs
Invoice corrections	~38% of invoices	20 min per correction	Customer trust erodes	Counted against sampled recent invoices

Total handoff overhead: roughly six hours per day per location. That is the sum of the per-job times above across a location's daily job volume, and it is rework time only. It does not count the invoices that are never corrected and the margin that leaks with them.

How 38 percent compares

A field-services back office in good shape keeps invoice rework in [the low single digits to low teens of percent]. Correcting 38 percent of invoices is several times a normal rate. This is not a back office that needs tightening. It is a structural defect in how data reaches billing.

The error stack and root cause

Errors compound as the job moves downstream.

Stage	Error introduced	Type
Dispatch to technician	~2%	Minor scheduling conflicts
Job capture on paper	~25%	Illegible, incomplete, wrong codes
Paper re-keyed into billing	~15%	Transcription errors
Cumulative at invoice	~38%	Combination of the above

About 80 percent of invoice errors originate at the paper-capture step. The paper form at the point of service is the root cause. Everything downstream is inheriting bad data and paying to clean it up. A new billing system would not fix this. Better data capture would.

The scheduling myth

Every vendor pointed at scheduling because that is where they had a product. The floor says scheduling is not broken. The dispatcher schedules from judgment the software cannot hold: which technician runs long on which days and why, which customers always add work on arrival, how traffic moves on a given route at a given hour. Automating that would damage a part of the business that works.

Her knowledge splits roughly into two parts: about 70 percent is pattern-based and could be supported by data over time, and about 30 percent is judgment that should stay human. That split is a later opportunity, not the first move.

Opportunity, sized

Directional, to be validated cheaply before any build, but sized with arithmetic rather than adjectives. Two buckets.

Rework cost, the floor. Six hours per day per location of pure rework, at [a loaded labor cost of about 30 dollars an hour], over [about 22 working days a month], is roughly [4,000 dollars] per location per month in labor alone, before a single dispute. Across the operation that is in the low tens of thousands per month. This bucket is real but it is not the prize.

Recovered revenue, the prize. The larger loss is billing that is disputed, delayed past collection, or written off because the invoice was wrong. At [a representative volume of roughly 1,500 invoices per location per month] and [an average invoice around 450 dollars], a 38 percent error rate touches a large share of revenue, and even a few points of that revenue leaking through billing errors is tens of thousands of dollars per location per month.

Put together, these put credible per-location recovery in the tens of thousands of dollars per month, compounding across the operation into seven figures a year. We flagged this as directional and said it must be proven cheaply before any build. It was. The build later recovered about 47,000 dollars in a single location in a single month, and reached about 247,000 dollars per month across the operation, roughly 2.96 million dollars a year. The pre-build arithmetic and the measured result agree, which is the point: this number was earned, not promised.

Where AI fits, and where it does not

- **Fits:** the field-to-billing data-capture handoff. Contained, measurable, reversible, and carrying the most waste.
- **Does not fit yet:** scheduling. Judgment-heavy and working. Touching it first would automate the wrong thing and lose the field's trust.

Risks and constraints we observed

- The field will reject anything slower than paper. Whatever gets built must beat paper on speed, not match it.
- Connectivity is unreliable in parts of the service area. Any field tool needs an offline-tolerant path.
- Critical knowledge sits in a few experienced heads. That is a resilience risk as well as an opportunity.
- The running systems are load-bearing. They must not be ripped out. They must be improved one handoff at a time.

The signal we leave with

The first AI move is the field-to-billing data capture. Before committing to a build, three assumptions need cheap tests: that scheduling visibility is not what technicians want, that the handoff is genuinely the source of the errors, and that the dispatcher's knowledge is replicable enough to support later. Those tests are where the work goes next. The plan, sized by impact, is the Playbook and Delivery Proposal.

Playbook and Delivery Proposal

The playbook and the delivery proposal are one document because they are one act. The playbook says where AI fits and sizes the moves in order. The delivery proposal scopes the build for the move you choose to start with. The first earns the second. Nothing past the first move is committed until the first move proves the approach in your business.

Part One: The Playbook

A written read for operators, not a deck. It answers three questions: where AI fits in your business and where it does not, the highest-leverage moves in sequence, and what it takes to run them.

Where AI fits, and where it does not

It fits the field-to-billing data-capture handoff, where roughly 80 percent of invoice errors originate and where six hours a day per location are lost to rework. It does not fit scheduling, which runs on judgment your dispatcher holds better than software would. The evidence is in the Day One Audit. The short version: the paper handoff from the field to billing is the most contained, most measurable, most reversible waste in the operation, and it is where the first AI work belongs.

How to read the roadmap

Two of these moves we diagnosed on your floor and can size with measured numbers. The rest we saw the shape of but did not diagnose, and we say so rather than dress them up. Honesty about what is proven and what is a candidate is the difference between a roadmap and a sales sheet.

Each move is read across six dimensions: time, accuracy and quality, cost and recovered revenue, growth, employee experience, and risk. The first move earns the right to the next.

The roadmap at a glance

#	Move	Status	Leverage	Containment	Why it sits here
1	Field-to-billing data capture	Diagnosed, sized	Highest	One handoff	Most waste, most reversible. Start here.
2	Dispatch decision support	Diagnosed, sized	High	One role	Needs the clean data move 1 produces.
3	Inventory and truck stock	Candidate, not yet diagnosed	Medium-high	One workflow	Builds on the parts data move 1 captures.
4	Technician routing density	Candidate, not yet diagnosed	Medium	One decision loop	Compounds with moves 2 and 3.
5	Parts procurement	Candidate, not yet diagnosed	Medium	One supplier loop	Last, once demand data is trustworthy.

Move 1: Field-to-billing data capture (start here)

Capture job data at the point of service in a form that flows straight to billing, with AI doing the transcription and a human reviewing it while trust is earned. Faster than paper for the technician, or they will not use it.

- **Time:** removes about 8 minutes of re-keying per job, 12-minute clarification calls on roughly 40 percent of jobs, and 20-minute corrections on roughly 38 percent of invoices. Reclaims on the order of six hours per day per location, all of it measured in the audit.
- **Accuracy and quality:** invoice error rate from about 38 percent toward single digits. The handful of errors that remain get caught by a human before the invoice goes out, not by the customer after.
- **Cost and recovered revenue:** recovers billing that is currently disputed, delayed, or written off. Sized in the audit at tens of thousands of dollars per location per month, compounding across the operation into seven figures a year. This is the one move where the estimate can be checked against a result: the build later recovered about 47,000 dollars in one location in one month and about 247,000 dollars per month across the operation.
- **Growth:** same-day billing instead of a three-day lag improves cash flow and frees capacity without adding headcount. The billing team carries more locations than it could before.
- **Employee experience:** technicians keep their fast workflow. Billing clerks move from chasing errors to handling exceptions, which is higher-value work and a better job.
- **Risk:** lowest on the board. Contained to one handoff, reversible to paper at any point. The right place to prove AI in this business.

Move 2: Dispatch decision support

Once capture holds, support the dispatcher with the pattern-based part of her work while keeping judgment human. We spent a day documenting her decision logic: about 70 percent is pattern-based and replicable with data, about 30 percent is judgment that stays with her. This move needs the clean, structured job data that move 1 produces, which is why it comes second.

- **Time:** targets the routine assignment work, [an estimated two to three hours a day per dispatcher] of pattern-based decisions a support tool could propose for her approval.
- **Accuracy and quality:** more consistent ETAs and technician matching, drawn from the durations and job data the capture system now records rather than from memory.
- **Cost and recovered revenue:** more jobs completed per technician per day from tighter assignment and fewer return trips. Even [one additional completed job per technician per day] across the fleet is a large gain, because the marginal job is close to pure margin.
- **Growth:** capacity to take on more volume without proportional dispatch headcount, and resilience if the lead dispatcher is out, which today is a single point of failure.
- **Employee experience:** removes the resilience risk of knowledge held in one head, and frees the dispatcher for the 30 percent only she can do.
- **Risk:** medium. Judgment stays human. The system proposes, the dispatcher approves. The 30 percent is never automated.

The later moves: named, not yet diagnosed

The audit surfaced three more candidates. We name them so the roadmap is honest about where this goes, but we did not diagnose them on the floor, so we will not pretend to size them to the dollar. Each gets its own cheap validation before any build, the same way move 1 did.

- **Move 3: Inventory and truck stock.** The parts data that move 1 captures on every job becomes a forecast of what each truck should carry. The waste to chase: return trips for a missing part, and holding cost from stocking everything instead of the right things. The number to beat is first-visit completion rate.
- **Move 4: Technician routing density.** Tighter clustering of jobs across the day, to complete more per truck without longer hours. The waste to chase is windshield time. The number to beat is completed jobs per truck per day.
- **Move 5: Parts procurement.** Once demand data is trustworthy across moves 1 and 3, buy from forecast and price history rather than job by job. The waste to chase is expedite premiums and emergency orders. The number to beat is price variance.

Each of these is a contained handoff or decision with its own measurable waste. None is committed now. They earn their turn only after the moves ahead of them prove out.

What it takes to run the moves

The discipline matters more than the technology.

- **Test cheap before building.** Days, not months. Rule out the inherited scheduling theory directly, then test capture approaches against what technicians actually use in the field.
- **Build for real conditions.** Faster than paper, works in a noisy driveway, tolerant of lost signal. If it is slower than paper, it fails.
- **Keep a human in the loop.** Every AI output reviewed while trust is earned. Billing already knows what correct job data looks like. That becomes the validation layer.
- **Expand by evidence, one location at a time.** Prove it, prove it is not a fluke, then test a genuinely different context before scaling everywhere.
- **Govern what gets built.** Grow autonomy against documented thresholds, with a human accountable for every billing decision.

The plays that run each canon come from our reusable plays library. The ones selected for this engagement are instantiated in the Charter. ## Who runs it

This can run with your own team, with another firm, or with us. It needs a few clear accountabilities: someone who owns the objective and clears the way, someone who does close observation and design, a builder who moves fast and discards what fails, a respected field voice whose adoption signals whether a change holds, and someone who watches the numbers for drift. You have most of these people. The build capability is the piece you would bring in.

The recommended first move and the 90-day frame

Start with the data-capture handoff. It is the highest-leverage, most contained, most measurable move on the board. The first 90 days: cheap experiments to validate the approach and rule out the wrong paths, a working build piloted in one location with a human reviewing every submission, and a measured result against the baseline that decides whether to expand. Prove AI works at one handoff and the operation gains not just a fixed handoff but the capability to fix the next one.

Part Two: The Delivery Proposal

The proposal to build the playbook's first move: the field-to-billing data-capture system. Scoped only after the playbook showed what is worth building.

What we understand

Technicians capture job data on paper. Humans re-key it into billing. That handoff produces a 38 percent invoice correction rate, clarification calls on nearly half of jobs, and a three-day billing reconciliation lag. Compounded, it runs to roughly six hours of rework per day per location. The scheduling software you were told to fix was never the problem. This handoff is.

What we will build

A field data-capture system that moves job data from the technician to billing without a human re-keying paper. AI does the transcription. A human reviews it while trust is earned. Technicians spend less time on capture than they do on paper, not more, or they will not use it.

How we will work

Four phases, mapped to the WISER canons. Each phase is independently valuable, priced on its own, and earns the next. The engagement can stop at any phase boundary with value already in hand.

Phase 1: Interrogate

Cheap experiments before any production build. Rule out the scheduling theory directly. Test capture approaches against what technicians will actually use in the field. Days, not months.

- Validate where the friction really compounds.
- Test capture options and measure them against paper on speed and accuracy.
- End of phase: a validated approach, with the wrong paths ruled out for the cost of a few days.

Phase 2: Solve

Build the working system and pilot it in one location with a human in the loop on every submission.

- Build the capture system for real field conditions, with a fallback path for when the primary fails.
- Pilot in one location. Billing validates every submission and logs what the AI misses.
- End of phase: the system proven in one location against the baseline, with the numbers to show it.

Phase 3: Expand

Roll out region by region. Residential first, then commercial, adapting to context rather than assuming it transfers.

- Prove it in a second similar location. Then test it where the context is genuinely different.
- Adapt the system to contexts the first locations did not have.

- End of phase: all locations live, residential and commercial, monitored through each wave.

Phase 4: Refine

Grow autonomy as reliability is proven, under documented governance.

- Define tiers of human oversight by job risk.
- Set the evidence thresholds that let a job type move to less oversight.
- Stand up the monitoring and review cadence that catches drift before it compounds.
- End of phase: a governed system where autonomy has grown where it was earned, and a human is accountable for every billing decision.

What we need from you

- A field validator. A respected technician who will test honestly and whose adoption signals whether the field will follow.
- A billing validator. Someone who knows what correct job data looks like and will review the AI's output during the pilot.
- Leadership air cover and weekly check-ins.
- Access to the systems the capture touches.

Infrastructure

You provide the field devices, the billing system access, and the software licenses for the AI services used. We provide the build, the AI architecture, and the implementation.

Who is working on this

A senior practitioner who leads the engagement and owns the objective with your team, and a builder who does the development. Your people fill the field, billing, and operations seats. Small team, close to the work.

Investment

Phased. Each phase is priced on its own so the engagement can stop at any phase boundary with value already delivered. The fee basis and amounts are held in `../../../Clients/`. We did not fabricate figures for this anonymized record.

Charter

What a Charter is

Not a project plan. Not a requirements document that executes once and collects dust. A Charter is the memory that survives the chaos. Its value is the decision log: when someone asks six months later why photo capture won over voice-only, or why we did not rip out the ERP, the answer is here, with the alternatives that were weighed and the evidence that settled it. The Architect keeps it current, same-day.

Metadata

Field	Value
Project	Field-to-billing data capture
Client	The HVAC and field services company (anonymized)
Charter Keeper	The Architect
Dates	Held in `../../../Clients/`; relative markers used here
Current canon	Refine. The system is live across locations and governed.
Version	End-of-first-project state

Positions

The work was held together by clear accountabilities, not an org chart.

Position	Who held it	Tension owned
Sponsor	Operations leadership	Authority. Owned the objective and cleared the way.
Guide	First Strategy senior practitioner	Translation. Carried the method and kept the Charter honest.
Architect	First Strategy	Curiosity and stewardship. Close observation, design, and Charter Keeper.
Sage	A long-tenured insider	Context. Opened doors and supplied history.
Scout	A respected field technician	Empathy. Validated whether a change would actually be adopted.
Builder	First Strategy developer	Execution. Moved fast and discarded what failed.
Finance lead	Client finance	Safety. Watched the numbers and caught the drift.
Billing validator	The billing manager	Integrity. Became the human validation layer.

On a small team one person can hold several Positions. As the system proved reliable, a Position could be augmented by an AI agent inside documented constraints, with the human shifting from doing to directing and reviewing.

Objectives and constraints

The build specification: what the project set out to do and the lines it would not cross.

Scope

In scope: capture job data at the point of service and get it to billing without a human re-keying paper. Out of scope: scheduling, inventory, routing, procurement. Those are later moves on the roadmap and were not touched in this build.

Objective and success criteria

Capture accurate job data at the point of service with under fifteen seconds of technician effort, and prove it against the baseline.

Measure	Baseline	Target	Result
Invoice error rate	~38%	Single digits	Residential 9%, commercial 12%
Billing reconciliation	3 days	Same-day	Same-day on the large majority of jobs
Technician capture time	5 to 10 min on paper	Under 15 sec, 8 the goal	~11 sec average
Paper usage	100%	Down 80%, rest a fallback	Down ~80%, remainder the fallback path

Constraints

- Faster than paper or the field rejects it. Under fifteen seconds is the hard line.
- Works in real field conditions: background noise, weather, gloves, a customer waiting.
- Tolerant of lost connectivity in parts of the service area.
- Does not disturb the running systems. It replaces one handoff, nothing else.

Architecture and human-in-the-loop design

A capture client in the technician's hand. An extraction layer that turns photo and voice into structured fields, trained on the operation's own historical invoices so it knows the local vocabulary and voices. A human-validation step. A flow into the billing system. A dual-path design: voice primary, photographed paper fallback, both handled by the same extraction layer. Every failed transcription becomes training data.

During the pilot, the billing reviewer validated every submission: photo, transcription, and each extracted field, checked against the job record. Misses were logged with the pattern that caused them, part names confused, phrases misread, unclear photos, and fed back to improve the model. That log became the checklist for the next location's reviewer. The validation grip loosened only later, under the governance tiers in the Hierarchy of Agency, never before the evidence supported it. A human stays accountable for every billing decision.

Current state at the start

Carried from the Day One Audit. The dispatcher schedules from judgment the software cannot hold. The real friction is the paper handoff from field to billing. The baseline: a 38 percent invoice correction rate, six hours of daily rework per location, and a three-day reconciliation lag. Roughly 80 percent of invoice errors originate at the paper-capture step.

Decision log

The decisions that shaped the build, each with the alternatives weighed and the evidence that settled it. This is the part of the Charter that answers "why did we do it this way."

When	Decision	Alternatives rejected	Rationale	Evidence
Interrogate, wk 1	Rule scheduling out as the first move	Build the AI scheduling the vendors pitched	The thing leadership and three vendors believed was never tested on the floor	Experiment 1: the visibility board went all but unused
Interrogate, wk 2	Target the field-to-billing handoff	Keep looking; replace the billing system	The errors cluster at one step, not across the system	Experiment 2: error tracing put ~80% at paper capture
Interrogate, wk 2	Treat the dispatcher as a later move, not the first	Automate dispatch now	A third of her work is judgment that should stay human	Experiment 3: a day mapping her decision logic, ~70/30
Interrogate, wk 3	Kill the digital form	Roll a digital form out to the field	Slower than paper and failed on low signal	Experiment 4: technicians abandoned it within days
Solve, wk 1	Build dual-path capture: voice primary, photo-of-paper fallback	Voice only; chase noise cancellation or better microphones	Voice fails in field noise, but the field needs a path that always works	First field test returned garbage on a noisy rooftop
Solve, wk 3	Human reviews every submission during the pilot	Trust AI output and spot-check	AI hallucinated on a meaningful share of submissions; humans must direct AI	Pilot review caught duration and part-name errors before billing
Solve, exec review	Replace one component at a time	Rip out the ERP and rebuild from scratch	Clean-slate replacements overrun for years and still fail	Two prior failures cited; the pilot proved a single handoff can be replaced and validated
Expand	Build a commercial version with milestone billing	Force the same-day model onto commercial work	Commercial jobs span phases; firing an invoice on every capture billed for unfinished work	The third location broke to a 55% error rate
Refine	Monitor accuracy by segment, alert at a 5% drop	Trust the aggregate dashboard	Aggregates hide a single failing segment	The drift incident: one service line fell while the headline held
Refine	Graduate routine residential to Tier 1 oversight	Keep full human review everywhere	Autonomy is earned on evidence, not granted	90 days above 95% accuracy, disputes below baseline, zero boundary violations

The decision and experiment record

The supporting narrative behind the log. The project ran the full WISER method. Witness had already found where AI fit; the project picked up at Interrogate and ran through Refine.

Interrogate

Three cheap experiments tested the inherited diagnosis, then three more tested how to capture data without slowing the technician. The scheduling theory died when a visibility board went unused and became, by the end of the week, a place to post a fantasy football league. Error tracing put roughly 80 percent of invoice errors at the paper-capture step. A day mapping the dispatcher's logic showed about 70 percent pattern-based and 30 percent judgment. Then: a digital form, killed for being slower than paper; a photo of the paper plus AI transcription, which dropped errors without changing technician behavior and was kept as the fallback path; and photo plus voice, which raised the question of removing paper entirely and set the Solve target.

Solve

The build target was photo plus voice, under fifteen seconds, eight the goal. The first field test exposed the jagged frontier of AI: it parsed complex part numbers without trouble and choked on wind noise on a commercial rooftop. The hard part was the noise, not the vocabulary. The team added the dual path and retrained the voice model on real field audio, compressors and truck engines and wind. Voice recognition in noisy conditions climbed [from about 60 to about 85 percent], with the paper-photo fallback catching the rest.

The pilot ran in one location: twelve technicians, four weeks, a human in the loop on every submission. The billing manager reviewed each one and logged every miss. The misses were not random; the model struggled with one trade's vocabulary and with certain spoken numbers, which is the same weakness that would later resurface as drift. The pilot held: invoice errors at 9 percent down from 38, clarification calls near zero, same-day reconciliation, eleven-second average capture, paper down 80 percent. One location recovered about 47,000 dollars in a single month.

At the executive review, leadership wanted to rip out the whole back-office system. The recommendation held: replace one component at a time. Ripping out everything at once is how a project becomes a multi-year overrun that still does not work.

Expand

The second location, residential, matched the first and proved it was not a fluke. The third was fully commercial and broke. The system had learned that a capture means a finished job, because every job in the first two locations was same-day. Commercial work captures at every phase, so the system fired invoices for unfinished jobs and the error rate climbed to 55 percent, worse than paper. The team pulled it within days, returned that location to paper, observed commercial work for three days, and rebuilt for milestone billing. Building it surfaced a constraint: the billing system did not support milestone billing,

which forced an intermediary layer and a slipped timeline that the Builder owned directly to the client. The commercial version recovered the location from 55 to 11 percent. All locations went live, two to a wave, residential at 9 percent and commercial at 12.

Refine

Going live was not the end. Stable is not governed. The team moved from building to watching.

Hierarchy of Agency

Three tiers of human oversight by job risk. The technician's field confirmation is the first gate; the tier governs what happens after that confirmation.

Tier	Oversight	Applies to
1: Auto-approve with spot-checks	No review of individual invoices. Random spot-checks on a small share, plus monthly red-team testing	Low-risk routine jobs the AI has proven reliable on
2: Oversight on key details	A one-screen summary highlights price, parts, and duration. One click to approve if the highlighted fields look right	Medium-risk jobs where specific fields need a check
3: Full review	Every invoice gets human eyes before it goes out. No exceptions	High-risk jobs: complex repairs, commercial projects

A job type moves to less oversight only on evidence: accuracy above 95 percent for 90 consecutive days, disputes below the pre-system baseline, zero boundary violations. Routine residential graduated to Tier 1 on this evidence. If a tier drifts, it falls back to more oversight. A human is accountable for every billing decision, at every tier. Autonomy is grown, not granted.

Risk register

Risk	Mitigation	Status
Field rejects anything slower than paper	Hard fifteen-second constraint; validated with a respected technician before building	Held; capture ran ~11 sec
Voice fails in noise	Dual-path fallback to photographed paper; retrained on real field audio	Resolved; noise accuracy climbed and the fallback catches the rest
Lost connectivity	Offline-tolerant capture with deferred sync	Designed in
AI hallucination on edge cases	Human review on every submission during pilot; hallucination log; segment monitoring later	Active control
A new context breaks assumptions	Sequence the rollout; test a genuinely different context before scaling	Realized at the commercial location; recovered
Drift hidden by aggregate metrics	Monitor by segment, alert on any type dropping >5% from baseline, weekly drift review	Added after the drift incident

Drift and incident record

After routine residential graduated to Tier 1, the system drifted. Extraction accuracy for one service line quietly dropped while the headline accuracy still looked fine. The training data had been heavier on the trade that adopted first, so the model learned that trade deeply and the other shallowly, and disputes for the weaker line rose before anyone saw it in the aggregate. It was caught by watching dispute trends by service type, not by the headline metric.

The response:

Action	Detail
Retrain	Balanced data for the under-served line, including fresh sample voice notes
Segment monitoring	Track accuracy by service type; alert if any type drops more than 5% from baseline
Weekly drift review	A standing 30-minute review of segment accuracy, dispute trends, and anomalies, led by the Finance lead
Remediation	Credits issued to customers billed wrong during the drift window

The weaker line recovered. The lesson logged: AI optimizes for what you measure, so measure segments, not just aggregates, from the start. Any drift past threshold triggers a fall-back to more oversight under the Hierarchy of Agency until resolved.

Evolution history

How the oversight posture changed over time, and why.

When	Change	Trigger
Pilot	100% human review on every submission	Trust not yet earned
Post-pilot rollout	Same posture carried location to location; each wave monitored	Expansion by evidence
After Tier 1 graduation	Routine residential auto-approved with spot-checks and monthly red-team	Graduation criteria met
After the drift incident	Added segment monitoring, 5% alerts, weekly drift review, and the fall-back rule	One service line drifted under the aggregate

Current status and the autonomy transfer

Three projects delivered. The client now runs the method largely on their own, and we are on a low monthly advisory retainer. The relationship moved through tiers the same way a job type graduates to less oversight: on evidence.

Tier	Project	We held	Client held
Lead	Project one: the data-capture build	The Guide role and the build	Watched the process work, learned the method
Co-guide	Project two (unnamed)	Co-guiding, technical oversight, trained their Guide	Led the doing
Oversight	Project three	Oversight and judgment calls	Led the project
Advisory	Now	A monthly check-in and a second set of eyes	Owns the capability

The goal was never to be permanently needed. It was to build the muscle and step back. The client continues to apply the same process to new components. If a problem arises that is worth more than oversight, we scope it the same way we scope any engagement.

Outcomes

- Invoice errors: residential 9 percent and commercial 12 percent, down from 38.
- Billing reconciliation: same-day, down from three days.

- Capture time: about eleven seconds per job, faster than paper.
- Recovered billing: about 247,000 dollars per month, roughly 2.96 million annually.
- No layoffs. Billing clerks who chased data-entry errors became exception handlers who catch what the AI misses and feed corrections back. The same team now carries more locations than before. Capacity grew instead of headcount.

Plays

The WISER plays this engagement ran, instantiated with the client's specifics. This is the index and what each produced. The high-value plays are held as standalone documents; the rest were applied inline in this Charter. | Canon | Play | What it produced | Source | |-------|------|------------------|--------| | Witness | Friction Mapping | The friction table and root-cause read at the field-to-billing handoff | Standalone play | | Witness | Field Observation, User Flow Mapping, Documenting Current State | The end-to-end job trace and systems landscape | Inline in the Day One Audit | | Interrogate | Assumption Auditing | The register of inherited beliefs to test, scheduling theory first | Standalone play | | Interrogate | Experiment Selection, Logging, Rapid Prototyping | The six experiments and what each ruled in or out | Standalone play | | Solve | Human-in-the-Loop Design | The billing-reviewer validation layer and miss log | Standalone play | | Solve | Quality Objective Setting, Pilot Planning, Value Validation | The success criteria, the one-location pilot, the measured recovery | Inline above | | Expand | Readiness Check, Sequencing, Context Fit, Deployment Gating | The wave plan and the commercial-context pivot | Inline above | | Refine | Hierarchy of Agency Design | The three oversight tiers and graduation evidence | Standalone play | | Refine | Drift Monitoring, Incident Response | The segment-monitoring plan and the drift fix | Standalone play | | Refine | Graduation Decision Making, Red Team Testing | The evidence thresholds and tier-one spot-checks | Inline above |

The first problem is solved, and it will not be the last. The operation now has a list of candidates for the same treatment: inventory, scheduling support, technician routing, parts procurement. The difference is that the team knows how to do this now. They built the muscle.

The plays

The WISER plays this engagement ran, instantiated with the client's specifics, ordered by canon.

WITNESS PLAY

Friction Map

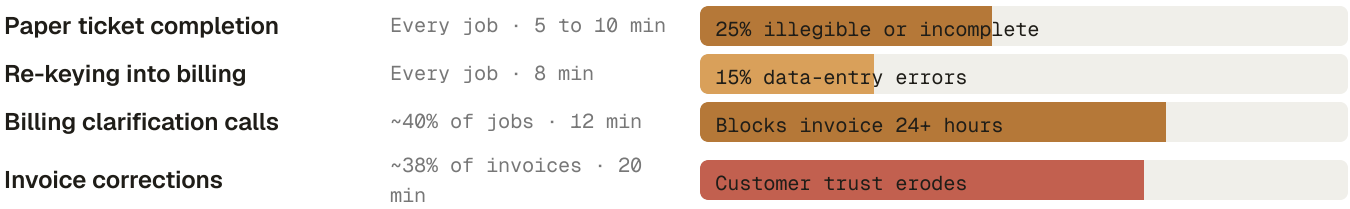
Witness play, instantiated for the HVAC services engagement. Purpose: locate and quantify where the work breaks, so the build targets a real problem rather than a theoretical one.

The handoffs

One job, dispatch to invoice. Six handoffs, four systems, paper in the middle.

1. Dispatcher assigns from her spreadsheet.
2. Dispatch software updated afterward, for billing.
3. Technician completes the work.
4. Technician records the job on paper, customer signs the carbon copy.
5. Paper dropped at dispatch at end of day.
6. Billing clerk re-keys paper into the billing system the next morning.

The friction, quantified



Friction point	Frequency	Time impact	Error rate
Paper ticket completion	Every job	5 to 10 min	25% illegible or incomplete
Re-keying into billing	Every job	8 min	15% data-entry errors
Billing clarification calls	~40% of jobs	12 min	Blocks invoice 24+ hours
Invoice corrections	~38% of invoices	20 min	Customer trust erodes

Aggregate: roughly six hours per day per location. Across eight locations, on the order of forty-eight hours of daily waste.

The error stack

Stage	Error introduced	Type
Dispatch to technician	~2%	Scheduling conflicts
Job capture on paper	~25%	Illegible, incomplete, wrong codes
Re-keyed into billing	~15%	Transcription
Cumulative at invoice	~38%	Combination

Root cause

About 80 percent of invoice errors originate at the paper-capture step. The paper form at the point of service is the root cause. The hottest point on the map is the field-to-billing handoff, not scheduling.

INTERROGATE PLAY

Assumption Register

Interrogate play, instantiated for the HVAC services engagement. Purpose: surface the assumptions to test before committing to a build, including the diagnosis the client arrived with. Each gets a cheap test and a clear bar for proof.

KILLED

Technicians need better scheduling visibility

Believed by leadership and three prior vendors. A visibility board ran for a week and went unused.

CONFIRMED

The field-to-billing handoff causes the errors

An end-to-end shadow of one day of jobs put roughly 80 percent of invoice errors at the paper-capture step.

REFRAMED

The dispatcher's knowledge is a single point of failure

About 70 percent pattern-based and replicable with data; about 30 percent judgment that stays human. A later move, not the first.

#	Assumption	Source	Cheap test	What proves or kills it
1	Technicians need better scheduling visibility	Leadership and prior vendors	A simple visibility board in front of technicians for a week	Proven if they use it; killed if they ignore it
2	The field-to-billing handoff causes the errors	The audit's friction map	A manual end-to-end shadow of one day of jobs, tracking where errors enter	Proven if errors cluster at the paper step; killed if they cluster elsewhere
3	The dispatcher's knowledge is a single point of failure	Observation of the spreadsheet	A day documenting her decision logic, sorting it into pattern-based versus judgment	Confirmed if most is replicable with data; reframed if it is mostly judgment

Why assumption 1 is tested at all: leadership and three prior vendors believed it for years. Ruling it out directly, cheaply, means nobody second-guesses the real finding later.

Results are recorded in the Experiment Log.

INTERROGATE PLAY

Experiment Log

Interrogate play, instantiated for the HVAC services engagement. Purpose: record each experiment, its result, and what it changed. Cheap experiments, days each, before any production build.

1 **Scheduling visibility board** INVALIDATED

Almost no use. Scheduling ruled out for good.

2 **End-to-end error shadow** CONFIRMED

~80% of invoice errors originate at paper capture. Root cause fixed.

3 **Dispatcher decision logic** REFRAMED

~70% pattern-based, ~30% judgment. A later opportunity.

4 **Digital form to replace paper** REJECTED

Slower than paper; failed in low signal. Killed the form approach.

5 **Photo of paper plus AI** WORKED

Errors dropped sharply with no behavior change. Kept as the fallback path.

6 **Photo plus voice** PROMISING

Set the Solve target: under 15 seconds, paper retired.

Hypothesis experiments

#	Experiment	Result	What it changed
1	Scheduling visibility board in front of technicians for a week	Invalidated. Almost no use; technicians get their next job by text and want less input, not more output	Scheduling ruled out for good
2	Manual end-to-end shadow of one day of jobs, tracking where errors enter	Confirmed. About 80 percent of invoice errors originate at the paper-capture step	Root cause fixed: data capture at point of service
3	A day documenting the dispatcher's decision logic	Partially confirmed. Roughly 70 percent pattern-based and replicable with data, 30 percent judgment that stays human	Reframed the dispatcher as a later opportunity, not the first move

Capture-approach experiments

Having located the handoff as the target, we tested how to capture data without slowing the technician.

#	Experiment	Result	What it changed
4	A digital form to replace paper	Rejected. Slower than paper; too many taps in a customer's driveway; failed in low-signal areas	Killed the form approach
5	Photo of the paper form plus AI transcription	Worked. Errors dropped sharply without asking technicians to change behavior	Confirmed AI transcription; kept as the fallback path
6	Photo plus voice to eliminate paper entirely	Promising. Raised the hypothesis that paper could be removed, not just transcribed	Set the Solve target: under 15 seconds, photo plus voice

Outcome

Two of three inherited hypotheses overturned or reframed. The build target set with the wrong paths ruled out for the cost of a few days. Next: Solve. The build it set up is recorded in the Charter.

SOLVE PLAY

Human-in-the-Loop Design

Solve play, instantiated for the HVAC services engagement. Purpose: define who reviews AI output, how, and what gets logged, so a human directs the AI rather than the reverse. A person stays accountable for every billing decision.

During the pilot

Every miss is logged with its pattern and fed back: retraining data for the model, a checklist for the next location's reviewer.

The billing reviewer validates every submission: the photo, the voice transcription, and each extracted field. They already know what correct job data looks like from years of catching errors.

Element	Design
Who reviews	The billing reviewer, on every submission
What they check	Photo, transcription, and each extracted field against the job record
What gets logged	Every miss, with the pattern that caused it: part names confused, phrases misread, unclear photos
What the log feeds	Retraining the model and a checklist for the next location's reviewer
Accountability	A named human owns each billing decision; the AI never sends unreviewed during pilot

How the grip loosens

Validation does not stay at 100 percent forever, and it does not loosen by decree. It loosens only under the agency tiers in the Hierarchy of Agency, and only when the evidence thresholds are met. Until then, every output gets human eyes.

Why this matters

The validation log is not overhead. It is the training data and the early-warning system. When the system later drifted, the discipline of watching outputs by segment is what made the catch possible. See Drift Monitoring.

REFINE PLAY

Drift Monitoring

Refine play, instantiated for the HVAC services engagement. Includes the incident response for the drift this engagement caught. Purpose: watch for drift, including drift that looks stable in aggregate but is failing a segment, and define how a failure is caught, contained, and fixed.

The incident

- **Routine residential graduates to Tier 1**
  90 days above 95% accuracy, disputes below baseline, zero violations.
- **One service line drifts, quietly**
  Training data skewed to the trade that adopted first. The aggregate dashboard stays green while segment disputes rise.
- **Caught by segment, not by the headline**
  Dispute trends watched by service type expose what the aggregate hid.
- **The three-part fix, plus remediation**
  Retrain on balanced data; segment alerts at a 5% drop; a standing weekly drift review. Customers billed wrong are credited.
- **Recovered, with the lesson logged**
  AI optimizes for what you measure. Measure segments, not just aggregates, from the start.

After routine residential graduated to Tier 1, the system drifted. Extraction accuracy for one service line quietly dropped while the headline accuracy still looked fine. The cause: the training data was heavier on the service line that adopted first, so the AI learned that line deeply and the other shallowly. Disputes for the weaker line rose before anyone saw it in the aggregate number.

It was caught by watching dispute trends by service type, not by the headline metric. The aggregate dashboard was measuring the wrong thing.

The fix

Action	Detail
Retrain	Feed the model balanced data for the under-served service line, including fresh sample voice notes
Segment monitoring	Track accuracy by service type; alert if any type drops more than 5 percent from baseline
Weekly drift review	A standing 30-minute review of accuracy by segment, dispute trends, and anomalies, led by the person who watches the numbers
Remediation	Credits issued to customers affected during the drift window

The weaker line recovered. The deeper lesson was logged: AI optimizes for what you measure. Measure segments, not just aggregates, from the start.

Standing monitoring

- Weekly drift review, by segment.
- Segment accuracy alerts at the 5 percent threshold.
- Red-team testing: deliberately try to break extraction with edge cases and ambiguous inputs.
- Any drift past threshold triggers a fall-back to more oversight under the Hierarchy of Agency until resolved.

Incident response pattern

Catch by segment, not aggregate. Contain by falling back a tier. Fix the root cause, not the symptom. Document the cause, detection, and fix in the Charter so the next person does not relearn it.

REFINE PLAY

Hierarchy of Agency

Refine play, instantiated for the HVAC services engagement. Purpose: define tiers of human oversight by job risk, so autonomy grows where it is earned and a human stays accountable everywhere.

The tiers

TIER 1 **Auto-approve with spot-checks**

Low-risk routine jobs the AI has proven reliable on. Random spot-checks plus monthly red-team testing.

TIER 2 **Oversight on key details**

Medium-risk jobs. A one-screen check on price, parts, and duration; one click to approve.

TIER 3 **Full review, no exceptions**

High-risk jobs: complex repairs, commercial projects. Every invoice gets human eyes.

A job type moves up only on evidence: 90 days above 95% accuracy, disputes below baseline, zero boundary violations. Drift falls it back down.

Tier	Oversight	Applies to
1: Auto-approve with spot-checks	No review of individual invoices. Random spot-checks on a small share, plus monthly red-team testing	Low-risk routine jobs the AI has proven reliable on
2: Oversight on key details	A summary screen highlights price, parts, and duration. One click to approve if the highlighted fields look right	Medium-risk jobs where specific fields need a check
3: Full review	Every invoice gets human eyes before it goes out. No exceptions	High-risk jobs: complex repairs, commercial projects

Job-type mapping

- Routine residential maintenance: Tier 1, after graduation.
- Standard residential repairs: Tier 2.
- Complex repairs and commercial projects: Tier 3.

Graduation criteria

A job type moves to less oversight only on evidence:

- Accuracy above 95 percent for 90 consecutive days.
- Disputes below the pre-system baseline.
- Zero boundary violations.

Routine residential graduated to Tier 1 on this evidence. The technician's field confirmation is the first gate; the tier governs what happens after that confirmation.

The rule that does not move

A human is accountable for every billing decision, at every tier. Autonomy is grown, not granted. If a tier drifts, it falls back to more oversight. See Drift Monitoring.