

From AI Conversations to Useful Work

Three real projects. One way to pick your first.
The people it takes.

Conversation → Workflow → Agent

Andrew Miller · Rebel Ops · October 6, 2026

WHO IS TALKING

I spend my time inside real organizations, working out where AI and software belong, and where they don't.

Andrew Miller

Founder of Rebel Ops, an AI consulting and product studio in Greenville, SC.

Most of the stories today come from client work, not from a lab.

WHERE THAT WORK HAS HAPPENED

Construction

Loan servicing

Creative services

Sales

Nonprofits

Workforce education

Home services

CLIENT WORK

Rebel Ops

Inside the operation with owners and teams, deciding what to build and what to leave alone.

PRODUCTS

Tools I build

Strata and Presentable, built for problems I kept running into.

COMMUNITY

Builder's Loft

Helping people get real repetitions, through Builder's Loft and local events in Greenville.

Your first AI project probably shouldn't be an AI project.

By the end: one piece of your own work, and what you'd need to try AI on it responsibly.

WHERE ARE YOU STARTING FROM? HANDS UP FOR EVERY LINE THAT FITS.

- 1 I've tried an AI tool

- 2 I use AI weekly for recurring work

- 3 I've connected AI to something that takes action

- 4 Not yet, and that's fine

Who decides what happens next?

Conversation

You work through it with AI.

WHO DECIDES NEXT

You

Scripted workflow

AI helps carry out a process we designed.

WHO DECIDES NEXT

The process

Agent

We delegate a goal and some discretion.

WHO DECIDES NEXT

AI, within limits

Three ways to put AI to work. Not a ladder. A schedule or a tool connection doesn't make something an agent.

Conversation: you hold the pen.



WHO DECIDES NEXT
You

HOW IT RUNS

You ask



AI answers or drafts



You push back, add context

↺ repeat until it's useful



You decide what happens next

GOOD FOR

Thinking through a problem, a first draft, a one-off analysis.

EXAMPLE

Before a hard client call, paste in approved notes and ask for the three objections you're most likely to hear.

WATCH FOR

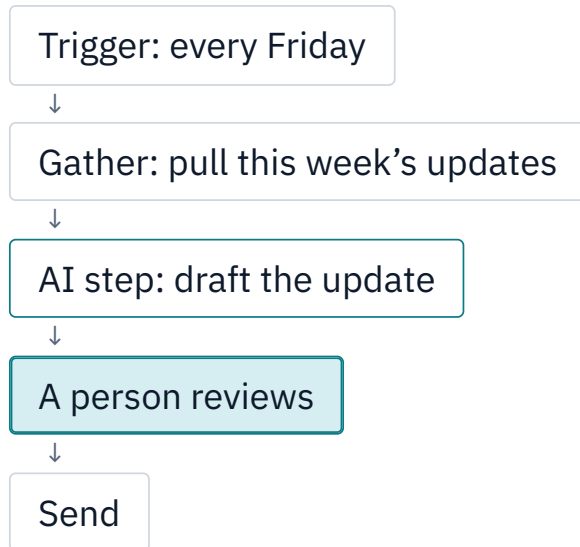
Fluent isn't the same as right. And don't paste in what you wouldn't email to a stranger.

The value is in how well you ask and how hard you check.

Scripted workflow: the process holds the pen.

WHO DECIDES NEXT
The process

HOW IT RUNS



GOOD FOR

Recurring work with a known shape.

EXAMPLE

Every new intake form gets summarized, tagged, and routed to the right person's queue for review.

WATCH FOR

It fails quietly when the input changes shape. Somebody has to notice.

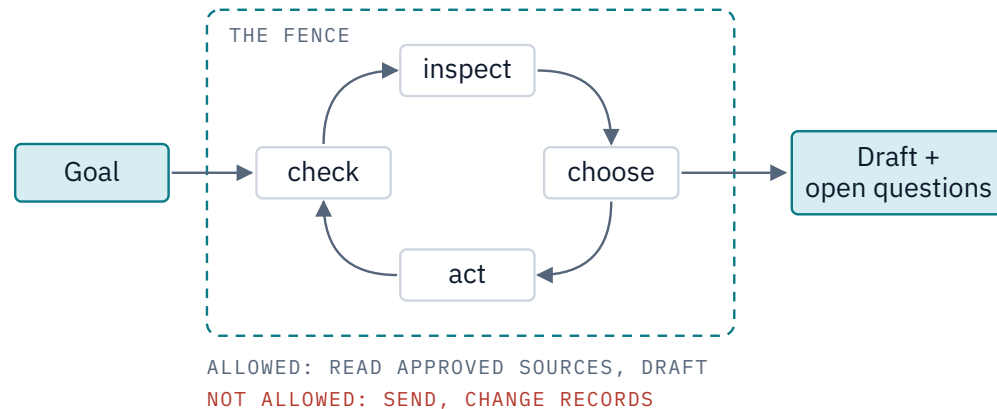
A workflow is only as good as the process you wrote down.

Agent: AI holds the pen, inside a fence.



WHO DECIDES NEXT
AI, within limits

HOW IT RUNS



GOOD FOR

Work where the next step depends on what it finds.

EXAMPLE

Prep a renewal brief. It checks the contract, the support history, and recent emails, decides what matters, and flags what it couldn't confirm.

WATCH FOR

The fence is the permissions, not the prompt. "Please don't send email" isn't a boundary.

Give it a goal, a fence, and a way to show its work.

One task, three ways.

CONVERSATION

I paste in approved notes and work on a draft with AI. I decide what to ask and what to send.

WORKFLOW

Every Friday the system pulls the updates, drafts in our format, and sends the draft to me for review.

AGENT

“Prepare an accurate update for this project.” It checks approved sources, notices two conflicting dates, decides where to look next, and returns a draft with open questions.

inspect → choose → act → check

**Friday isn't what makes it an agent.
Deciding what to look at next is.**

What would you tell a new colleague before asking for this?

BEFORE

“Write a project update.”

AFTER

Using these approved project notes, draft a short client update.
Separate completed work, blockers, and decisions needed. Use the latest dated source when information changed. Flag conflicts instead of guessing. Don't invent commitments or send anything.

goal

facts

what good looks like

limits

AI doesn't automatically know what your team takes for granted.

The right information, not all the information.

Three questions to ask about any AI setup.

ONE

Where does the information go?

TWO

Who can access it?

THREE

What can the system do?

WHERE PROCESSING CAN HAPPEN

Public tools

→

Enterprise services

→

Private infrastructure

→

Models on your own hardware

Local is not automatically secure. Storage, access, and connected tools still matter.

Boundaries decide what **can happen.
Evidence decides what you **trust**.**

Sometimes the first fix isn't AI.

THE OPERATION



A handoff nobody clearly owned.

The question everyone asked: “Can’t AI just do this?” Sure, eventually. First, nobody could tell me who owned the handoff.

THE QUESTIONS THAT MATTERED FIRST

Who owns the handoff?

What does “done” mean?

When does something escalate?

What does the next person need to know?

Bad process plus AI usually gives you faster bad process.

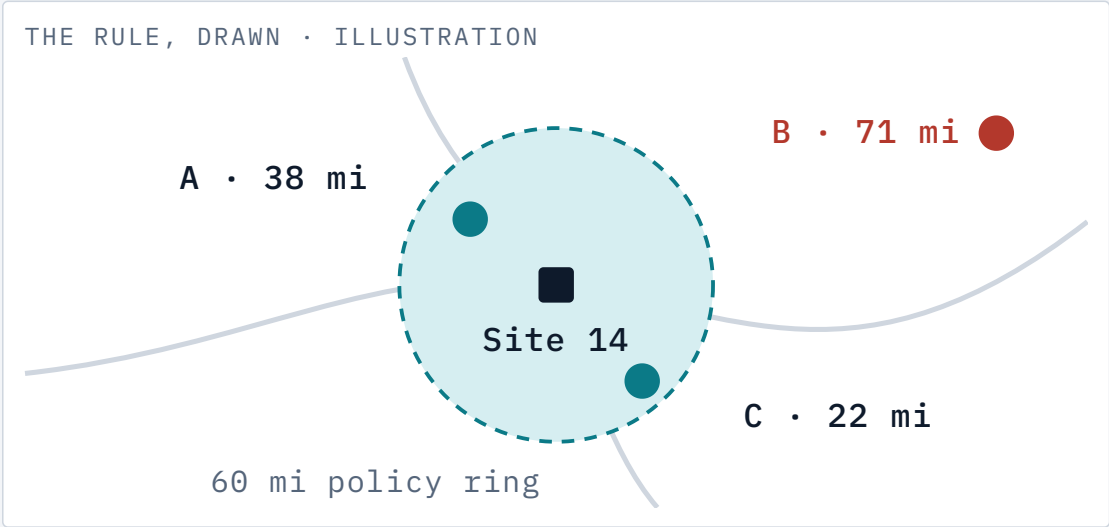
Start with the decision.

THE DECISION

Who gets assigned to which job site. It depends on travel distance and company policy.

WHERE IT STARTED

A distance-checking spreadsheet that hit API limits.



WHAT WE BUILT

- Import employee and job data
- Calculate driving distances
- Show threshold information
- Export for the people who decide
- A person decides

Delivered: distance app, some fixes open Selected direction: daily paperwork visibility Proposed: job startup

SHOW CASE 1 THE ARTIFACT AND THE PEOPLE ILLUSTRATION, NOT CLIENT DATA

BEFORE · THE SPREADSHEET

EMPLOYEE	JOB SITE	MILES
Crew member A	Site 14	#ERROR
Crew member B	Site 14	#ERROR
Crew member C	Site 9	#ERROR

Distance lookups hit the API limit.

→

AFTER · THE APP

EMPLOYEE	SITE	DRIVE	POLICY
Crew member A	Site 14	38 mi	Within
Crew member B	Site 14	71 mi	Over threshold
Crew member C	Site 9	22 mi	Within

Exported to the project manager, who decides.

DATA AND ACCESS

Operations lead

ASSIGNMENT DECISION

Project manager

BOOKING

Administrator

The app does not assign people, book hotels, or send notifications. The distance math and thresholds are ordinary software.

Figure out the decision first. Pick the tool second.

One field, one decision.

Prototype

Fabricated test data

Document

 →

Proposed value

 →

Source evidence

 →

Reviewer decision

FIELD

Maturity date

PROPOSED

June 1, 2031

SOURCE EVIDENCE · NOTE, P. 2

“...all unpaid principal shall be due and payable on June 1, 2031 (the Maturity Date).”

CONFLICT FOUND

Loan modification, p. 1: “...the Maturity Date is extended to June 1, 2034.”

Accept

Correct

Reject

Escalate

Does the work today: the reviewer

Builder: outside partner

Data: fabricated for the prototype

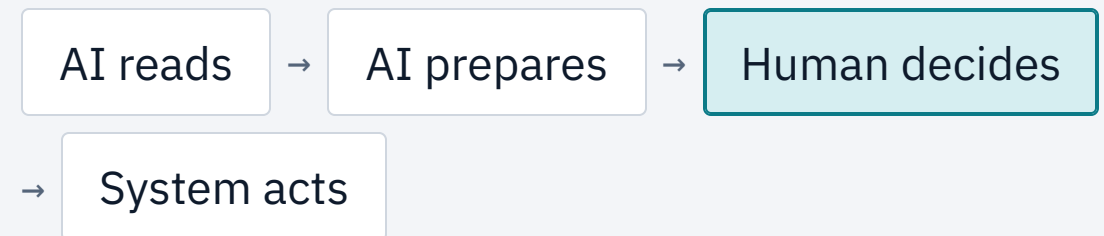
A polished answer hides a conflict. A review card shows it.

Let AI propose. Don't let it commit.

WOULD YOU LET A SYSTEM...

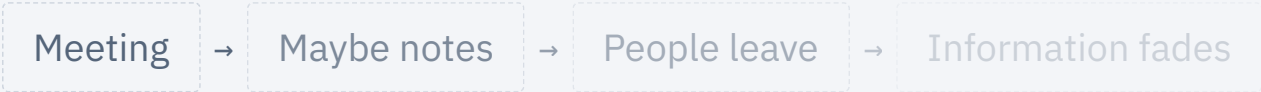
- 1 Extract the value?
- 2 Propose a correction?
- 3 Commit the correction without review?

What evidence would change your answer?

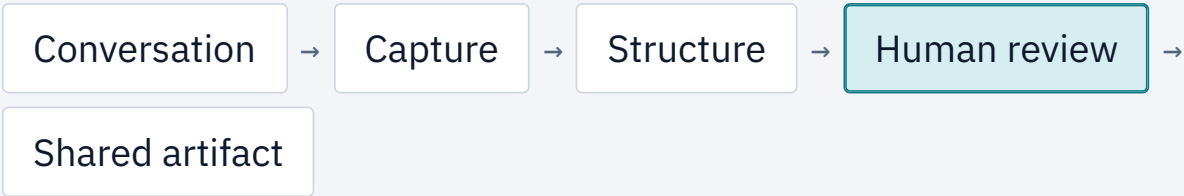


What does the conversation leave behind?

THE OLD PATTERN



DESIGNED AROUND AI



SHARED ARTIFACT · ILLUSTRATION

Community event: what we covered

Key ideas	reviewed
Resources mentioned	reviewed
Open questions	reviewed
Action items	needs an owner

Captured: the system

Reviewed and approved: me

Published: the event host

Faster notes aren't the point. What the conversation leaves behind is.

SOMEONE WHO...	...ANSWERS THIS
Cares if it works	Why are we doing this, what counts as success, and should we continue?
Does the work today	How does this work really happen, including the exceptions?
Builds it	How do we make it work, connect it safely, and fix it?
Signs off	Is this output acceptable, and can the action proceed?

TWO MORE QUESTIONS

- Who authorizes access to the data and systems?
- Who supports it after the demo?

None of the three cases started by hiring an AI department. In a small company, two people wear all four hats.

You can outsource the building. You can't outsource ownership of the problem.

How you get it and who does it are two different decisions.

	BUY	CONFIGURE	INTEGRATE	BUILD
Internal team	Pick a standard product	Set up a tool you already pay for	Connect two systems you own	Build a small workflow in-house
Outside partner	Recommend and install a product	Tune an existing product	Connect your systems	Build custom software

Illustrative examples. Start by testing what you already have before commissioning custom software.

When this breaks on a Wednesday morning, who notices, who can turn it off, and who fixes it?

Problem	Assembling the update from several sources takes time. Measure today’s time first.
Owner	Whoever prepares and checks it now.
Experiment	Five sanitized past input sets, including one with missing information and one with conflicting information.
Boundary	Approved tools and data. Drafts only. No sending, no invented dates or commitments. A person checks each one.
Evidence	Time against the current method, plus errors, omissions, and corrections. Then revise, stop, or test further.

Try it **Wednesday, Oct 7. Review Friday, Oct 9.**

Four quiet minutes: write your own problem, owner, boundary, and first action. No company access? Use your own non-sensitive notes.

OLDER NOTE · SEPT 28

Superseded

Delivery is October 16.

NEWER APPROVED DECISION · OCT 2

Delivery moved to October 23.

AN ADVANCED EXPERIMENT

Write a one-page brief: goal, approved facts, open questions, decision history, dates.

Run a task with it.

Mark one fact as superseded. Run it again.

Does it use the new fact, flag the conflict, or blend both?

When the next AI system prepares an update, how does it know which one governs?

Two versions of the same problem. One for software, one for people.

STRATA

Carrying context between AI systems

Keeping track of where a fact came from, whether it still holds, and who should see it.

BUILDER'S LOFT

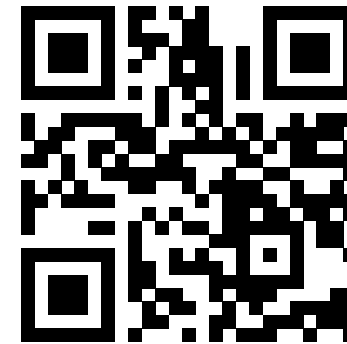
Getting real repetitions

A place where people build this kind of thing together, so the lessons today become practice.

Neither one is finished. Both come straight out of the problems in this talk.

You don't need an AI strategy. You need one good experiment.

Pick one problem.
Give it an owner.
Set a boundary.
Try something small.
Check what actually happened.



SCAN TO JOIN
Builder's Loft
Two minutes.
Tell me what
you're working
on.
hvtdp2qhft.zite.so

Bring me a workflow where the work gets stuck.