PROCESS > STRUCTURED CAPTURE > AI

Process Data for AI Training.

Why the next generation of enterprise AI will be trained on procedures, not paragraphs.

BY CARRV.AI

8 MIN READ · MAY 2026

From language to procedure.

For a decade, enterprise AI meant chatbots. Customer service bots, FAQ bots, sentiment classifiers. They worked on text, were trained on text, produced text. The substrate was language.

That era is ending. The substrate for the next era of enterprise AI isn't language. It's process.

This is why every serious conversation about AI agents — the kind that can take on the procedural weight of enterprise work, so the people doing it spend their time on the parts that actually need them — keeps arriving at the same bottleneck: the agent doesn't know how the work is done.

It can read English. It can write SQL. It can summarize a PDF. It cannot, without explicit instruction, do what your AP clerk does at 11 a.m. on a Tuesday with an invoice in her inbox. Not because the model is weak. Because nothing in the training set told it.

That's the gap. And the data that fills it isn't text. It's process.

What the models have eaten.

Open up any frontier foundation model and look at what it has eaten.

- Text from the open web
- Books, academic papers
- Code from open repositories
- Public datasets, conversations
- Image–caption pairs
- Video, audio (recently)
- **Your enterprise's process** THE GAP

This isn't a failing of the model. There's no public corpus for that data. It doesn't sit in a Common Crawl bucket. It exists only inside your SaaS sessions, your authenticated apps, your team's local context. The models couldn't have trained on it even if they wanted to.

So when you connect an AI agent to a real workflow — process this insurance claim — the agent has to be told. Step one, step two, step three, with screenshots, with edge cases, with branching. The agent isn't learning from your business; it's being programmed by it, instruction by instruction.

That programming is the cost. And it scales linearly with every workflow you want to automate.

What process data actually is.

A structured representation of one complete way of doing one task. At minimum, it contains five things.

01 **Ordered sequence of steps**

Each tagged with the actor, the application, the action type (click / type / read), and the target element.

02 **Screen state**

Image, OCR-extracted text, or structured UI element data — what was on the screen at each step.

03 **Actor input**

What the user typed, selected, or pasted in. The values that mattered.

04 **Decision points**

Where the procedure could have branched, and which branch was taken.

05 **Handoffs**

Who passed the work to whom, when the procedure moves through multiple actors.

From this single source, downstream tools generate prose SOPs, simulations, training data, eval sets. Notice what isn't on the list: no written description. No bullet-point procedure. Those are outputs of process data. The canonical artifact is the structured capture itself.

Why text isn't a substitute.

This is where most enterprise AI initiatives quietly fail. The team says: we have all this documentation. Confluence, SharePoint, the SOP library. Surely we can train on that.

Then they try, and they discover that text describes procedures the way a recipe describes cooking — accurately, perhaps, but without any of the actual texture. The Confluence page says click Submit. It doesn't say which of the four Submit buttons on the screen. It doesn't say what to do when the system returns a validation error. It doesn't say what the experienced reviewer notices and the new one misses.

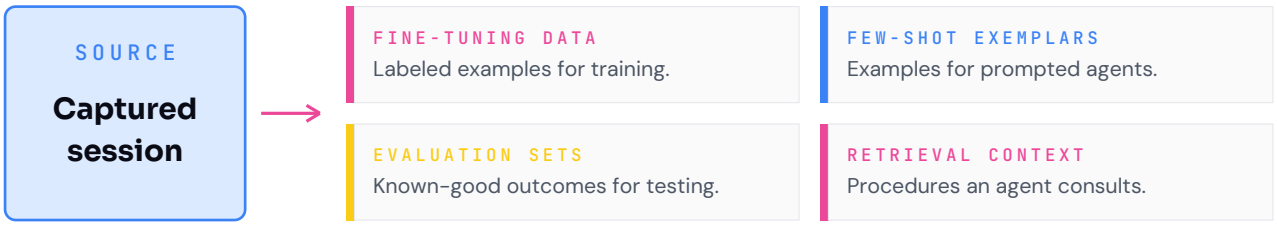
Text-trained AI knows the vocabulary of invoicing. It can talk fluently about purchase orders, three-way match, GL coding. It cannot do invoicing — because doing requires the screen, the click, the recovery from the error, and the choice between Submit and Send for Approval.

Process data carries all of that. Text strips it out.

> What process data doesn't carry — and shouldn't be asked to — is the judgment the human brought to the procedure. The procedure is the part that repeats. The judgment is the part that doesn't. Process data captures the first. Humans keep the second.

One capture, four outputs.

A discovery tool captures a session. From that single source, four AI training–grade outputs are generated.



ONE CAPTURE > FOUR REGENERABLE TRAINING OUTPUTS

All four outputs regenerate from the same captured artifact. When the underlying software changes, you recapture. The training data updates with it. You capture once. The downstream regenerates.

When an enterprise has it.

Four shifts, in order of impact.

01

Deployment cost collapses.

Today, deploying an AI agent against a real workflow is an integration project — engineer reads SOP, writes prompts, hand-codes edge cases, tests against the live system. Weeks per workflow. With process data as input, the captured artifact is the instruction set.

02

Coverage expands.

When workflows are expensive to teach an agent, you only teach it the top ten. With process data flowing automatically from discovery, you teach it the long tail. The cumulative time-savings from the next two hundred workflows is often larger than from the top ten.

03

Trust improves.

An AI agent deployed against a documented procedure is auditable: you can show exactly which captured procedure it learned from, which branch it chose, which step it deviated on. Auditability is the gating factor for compliance-sensitive workflows.

04

The human role sharpens.

When the procedural layer is handled by an agent, the human's job stops being execution and starts being supervision — catching edge cases, applying judgment, deciding when the rule should be broken. The work the human keeps is the work that needed a human all along.

Two paths from here.

Enterprises that already invested in process discovery have been collecting the training data of the next AI cycle without knowing it. They built capture engines to produce simulations and SOPs; they ended up with the most valuable corpus in enterprise AI.

WITHOUT PROCESS DATA

Prettier chatbots.

Hand-written documentation. Prompt-engineered agents. Useful, but bounded by the prose they were taught from.

WITH PROCESS DATA

Augmented teams.

Captured procedures as training corpus. Agents that handle the procedural layer. Humans focused on judgment, edge cases, and the work that requires them.

The same shift that happened to natural language with the open web — vast, structured corpora unlocking general-purpose models — is about to happen to enterprise work. And the corpus for enterprise work is process data.

AI agents need to know how work is done.

Text doesn't carry that. **Process data does.**

THE COEXISTENCE

But process data captures the procedure, not the judgment. AI takes on the part that repeats. The human keeps the part that doesn't. That coexistence is the model — not a transition, not a phase, the actual end state.

Whoever captures procedure with structure today is collecting the training data of the next cycle.

Whoever doesn't will be writing prompts forever.

CARRV.AI

ENTERPRISE INTELLIGENCE, CARVED AT THE SOURCE.

Process Data for AI Training

AN OUTLOOK ESSAY · MAY 2026