

The Process-Before-Platform Framework

Why optimization has to come before your next system — and why AI raises the stakes

By Cedric Thomas, CEO · Strategy Planning Execution · cthomas@spxltd.com

Every few months a mid-market leadership team calls us with the same plan and the same conviction. They have picked a new platform — an ERP, a workflow tool, increasingly an AI agent — and they are certain it will fix a process that has frustrated them for years. The contract is nearly signed and the implementation partner is lined up. They want help with change management.

Our first question is rarely welcome: what does the process actually look like today?

Most of the time, nobody can say. There is a procedure document somewhere that describes how the work is supposed to happen, and there is the way it actually happens, and the gap between the two is exactly where the budget goes to die. A new system does not close that gap. It encodes whichever version you hand it — and if you hand it the real one, with all its workarounds and exceptions, you have just paid to make your dysfunction permanent.

This is the single most expensive mistake we see in technology implementation, and it has a simple cause: teams treat the platform as the solution, when the platform is only ever an accelerant. What it accelerates is up to you.

Why “platform-first” feels safe — and isn’t

Choosing the platform first feels like progress. It is concrete. It has a price tag, a demo, a go-live date, and a vendor whose entire job is to make you feel confident. Re-engineering a process, by contrast, is ambiguous, political, and unglamorous. So teams reach for the concrete thing and quietly hope the software will impose the discipline they have not.

It never does. Software is infinitely configurable, which means it will happily reproduce a fourteen-step approval chain if that is what you map into it. Six months later the symptoms are familiar: adoption stalls because the new system is as confusing as the old one; the customization bill balloons because you bent the platform to fit broken steps instead of fixing the steps; and someone proposes a second implementation to fix the first. The technology was never the problem. The process you poured into it was.

A new system does not fix a broken process. It makes the breakage faster, permanent, and harder to see.

The AI amplifier

For years, “process before platform” was good discipline. In 2026 it is closer to a survival skill, and the reason is AI.

Embedded and agentic AI has crossed from novelty to baseline. The major mid-market ERPs now ship intelligent agents that can read an invoice, match it to a purchase order, route an approval, and post the entry with little or no human touch. That is genuinely powerful — and it is precisely why getting the process right first matters more, not less.

A human following a bad process introduces friction that, perversely, acts as a brake. People hesitate, ask questions, escalate the thing that looks wrong. An autonomous agent has no such instinct. Hand it a broken process and it will execute that process flawlessly, tirelessly, and at a scale you will not notice until the cleanup bill arrives. AI is an amplifier. Give it a clean process and it makes the work faster, cheaper, and more reliable. Give it a broken one and it industrializes the breakage.

So the order of operations has never mattered more. Fix the process, then automate it. Never the reverse.

The framework

We take every optimization engagement through five stages in a fixed order. Each stage has a single gate question; you do not advance until you can answer it honestly. The platform decision lives at the very end — by design.

THE PROCESS-BEFORE-PLATFORM FRAMEWORK



Software is the last 20%. The first 80% is deciding what the process should be.

1. Map. Document the process as it really runs, not as the procedure claims. Sit with the people doing the work and trace a real transaction end to end — every handoff, every approval, every workaround. The map you want is the embarrassing one. Gate question: does this match how work really flows?

2. Measure. Put numbers on it before you change anything. Baseline cycle time, cost per transaction, exception rate, and the number of human touch points. Without a baseline you cannot prove improvement, justify investment, or tell whether the platform actually helped. Gate question: what is the baseline — time, cost, exceptions?

3. Eliminate. Remove the steps that should not exist. This is where the real value is created, and almost none of it requires software. Most processes are carrying controls added years ago for problems that no longer exist, duplicate approvals, and rework loops nobody owns. Cut them. Gate question: why does this step exist?

4. Standardize. Define the one right way before you encode it. If three regions run the process three ways, you have three problems, and automating them gives you three automations to maintain. Agree the standard path and the legitimate exceptions to it. Gate question: is there one agreed way?

5. Select. Now — and only now — choose and configure the platform. Let the requirements fall out of the optimized process, not the vendor's feature list. You will buy less, configure faster, and adopt more, because the system is shaped to a process you have already made sound. Gate question: what must the platform actually do?

The framework in motion: a hypothetical

The sequence is easier to see in motion. The scenario below is hypothetical — a composite of situations we see often, with illustrative numbers — but it shows how the five stages change the outcome.

Imagine a mid-market distributor whose accounts payable team is drowning. Vendors wait weeks to be paid, the team spends its days chasing approvals, and leadership has decided the fix is an AI agent that will read invoices, match them to purchase orders, and route approvals on its own. The instinct is to buy the agent and point it at the workflow that already exists. Run the same situation through the framework instead, and it goes somewhere very different.

Map. You trace a single real invoice from receipt to payment and find seven approval steps — not the four the policy describes, but seven, thanks to workarounds that have accreted over the years.

Measure. You baseline it: roughly three weeks from invoice to payment, a large share of invoices kicked back at least once, and a cost per invoice high enough to make the CFO wince. Now there is a number to beat.

Eliminate. You put the gate question to every step — why does this exist? Several do not survive it: an approval added years ago for a risk a newer system already covers, a duplicate sign-off, a review that waves through everything it sees. On paper, three of the seven steps add no control at all.

Standardize. With the redundant steps gone, you define the one clean path the remaining four steps follow every time, plus a short, deliberate list of exceptions — so every branch runs it the same way rather than inventing its own.

Select. Only now do you bring in the agent, and you point it at a streamlined four-step process instead of a tangled seven-step one. It has less to do, less to get wrong, and a clean baseline to be measured against.

The payoff is twofold. The process is faster and cheaper before a single line of code is configured — and the automation, when it arrives, actually delivers, because it is not faithfully reproducing steps that never should have existed. Skip the first four stages and the agent would have automated all seven, locking the dysfunction in at machine speed and calling it transformation.

Automate first and you build a machine to reproduce your worst habits perfectly.

How to know you're ready to choose the platform

Before you sign anything, your team should be able to answer yes to each of these. If you cannot, the gap is in the process, and no platform — however intelligent — will close it for you.

- We have mapped the process as it actually runs, validated by the people who do the work.
- We have a measured baseline: cycle time, cost, exception rate, and human touch points.
- We have eliminated every step we could not justify, not just the obvious ones.
- We have a single standard path and a defined, deliberate list of exceptions.
- Our platform requirements come from the optimized process, not the vendor's demo.
- We can state, in one sentence, what the system must do that the improved process cannot do on its own.

The order is the strategy

There is nothing wrong with ambition about technology. AI genuinely is changing what mid-market operations can do, and the firms that move well will pull ahead of the ones that hesitate. But moving well is not the same as moving first into a purchase. The advantage goes to the teams that do the unglamorous work first — that decide what the process should be before they decide what to buy to run it.

Software is the last twenty percent of the job. The first eighty percent is judgment. Get the order right and the technology pays off. Get it backward and you will pay for the technology twice.

Work with SPX

SPX helps founder-led and PE-backed mid-market companies optimize the process before they implement the platform — and put AI where it actually earns its place.

Cedric Thomas, CEO · cthomas@spxltd.com
