

BEYOND RE-PLATFORMING

ERP in the Age of AI — Why the Great Re-Platforming Wave Arrived at the Worst Possible Moment

The case for migrating anyway — under a fundamentally different design philosophy

Giovanni Leonardi

December 2025



ERP in the Age of AI — Why the Great Re-Platforming Wave Arrived at the Worst Possible Moment

Transformation · No. 428
December 2025

"It will deliver a modern system. It will not deliver a modern architecture."

1. The Moment the Business Case Went Quiet

The slide had not changed in eighteen months. Total cost of ownership over a decade. Reduced customisation burden. Modern architecture. Cloud-native operations. The ERP re-platforming business case was, by any reasonable measure, sound — the kind of investment thesis that boards have been approving for thirty years, adjusted for inflation and acronyms.

Then someone at the steering committee asked the question that had been circling the corridors for six months: what happens if the processes we are migrating are the ones the agents will replace?

The room went quiet — not because the question was hostile, but because nobody had a defensible answer. That silence, replicated in boardrooms and programme offices across every sector running a major platform migration, captures the defining tension of enterprise technology in late 2025. A generation of ERP estates has hit end-of-life at precisely the moment when artificial intelligence has begun to redefine what enterprise software might become. The timing is not merely unfortunate. It is structurally the worst collision these programmes could face.

2. The False Binary

The instinct is to frame this as a choice between two camps: migrate now, or wait for clarity. Both positions are internally coherent. Both are wrong.

The case for waiting sounds prudent. If AI agents are going to reshape procurement, order management, financial close, and planning — the bread and butter of the ERP core — then committing hundreds of millions to re-platforming those processes onto a new system is, at best, premature. Why build a cathedral for a congregation that may never arrive?

The case for migrating regardless sounds equally rational. The current estate is ageing. The skills to maintain it are retiring. Vendor support windows are contracting — SAP's ECC maintenance timeline is narrowing toward its end-of-decade horizon, Oracle is pressing its installed base firmly toward Cloud ERP — and no CIO would choose to fly through a closing window under duress. Meanwhile, the data locked inside these legacy systems is paradoxically the very asset that new AI capabilities need to function. Waiting does not preserve optionality; it preserves fragility.

The trap is that both camps are arguing about the wrong thing. The debate about whether to migrate has obscured the far more consequential question of what the migration is for. And it is here that most programmes are failing — not in execution, but in conception.

I wrote about exactly this pattern in 2007, when service-oriented architectures collided with the tail end of the great ERP consolidation wave. Organisations that had just finished rationalising onto a single instance discovered that the monolith they had built was precisely the wrong shape for the integration model that followed. Different decade. Same trap: a legitimate modernisation need, met by a technology shift that changed what "modern" meant, answered by programmes that solved last decade's problem with last decade's design philosophy.

3. What AI Actually Changed

The disruption is not, as some vendor keynotes suggest, a matter of bolting intelligent features onto the existing application model. It is a question about where intelligence sits in the enterprise architecture — and that question has profound implications for what the ERP core should be.

For two decades, the ERP system was the process engine. Business logic lived inside it. Workflow ran through it. Decision-making, to the extent it was automated, was encoded in its configuration. The system was the process, and customisation was the mechanism by which organisations made the process their own.

The AI agent model — still nascent, still largely unproven at enterprise scale, but directionally unmistakable — inverts this relationship. Intelligence moves to the edge. An agent observes, reasons, and acts across systems; the value sits not in the transactional engine but in the data it produces and the interfaces it exposes. The ERP system, in this model, becomes infrastructure: a system of record and a transactional backbone, not a process orchestrator.

Consider what this means for something as fundamental as procurement. Today the process runs end-to-end inside the ERP platform: requisition, approval, purchase order, goods receipt, invoice matching, payment — each step configured, constrained, and executed by the system. In an agent model, the intelligence that selects the supplier, negotiates terms, and resolves the three-way match operates across the procurement engine, the supplier portal, the contract repository, and market data — using the ERP's transactional core for recording and compliance, not for orchestration. The system of record remains. The process engine migrates outward.

No one standing in late 2025 can say with certainty which processes agents will genuinely absorb, which will prove stubbornly resistant, or how fast the transition will move. The

enterprise vendors are placing large bets — embedded copilots, agent frameworks, process mining married to generative AI — but the gap between demonstration and production remains wide. The history of enterprise technology is littered with capabilities that worked in the keynote and stalled at rollout.

What can be said with confidence is that the design philosophy under which most re-platforming programmes were conceived — clean the data, standardise the processes, lift everything into the vendor's cloud, and call it transformation — was articulated in 2018 and 2019, before the current AI wave was visible. Programmes still executing against that original conception are building to a brief that the world has already outgrown. Not because AI has arrived in its full form, but because it has arrived enough to change what a rational target architecture looks like.

4. Migrate — But for Different Reasons

The position I would argue is not that migration is wrong, but that most programmes have the justification backwards. The re-platforming is warranted — as a data-foundation exercise, not a process-transformation one.

Three principles distinguish a migration designed for the AI era from one designed for 2018:

Clean core as data foundation, not process purity. The drive toward a clean, standard core has been framed as a process discipline — eliminate customisation, adopt the vendor's reference model, reduce complexity. Under an AI-era design philosophy, the clean core matters for a different reason: it creates the structured, consistent, trustworthy data layer that intelligence needs to operate. The motivation shifts from "we should run the vendor's processes" to "we need a data estate that agents can read, trust, and act upon." The practical difference is real. Process standardisation still matters, but the standard is set by data quality and semantic consistency, not by the vendor's default configuration.

Process standardisation as agent-readiness. Standardising processes is not an end in itself — it is the precondition for intelligent automation to attach. A process that is documented, measured, and consistently executed is one that an agent can observe, learn from, and eventually absorb. A process buried in custom code and local workarounds is opaque to any form of intelligence, human or artificial. The programme that standardises its order-to-cash cycle should be doing so not because the vendor's reference model is inherently superior, but because a standardised process is a legible process — and legibility is what the next generation of capability requires.

Extensions outside the core, precisely so intelligence can attach at the edges. The most consequential architectural decision in an AI-era migration is where to stop the core. Every process pulled inside the monolith — embedded in the vendor's proprietary layer, locked behind a closed extension framework — is a process that becomes harder to reach, harder to orchestrate across systems, and harder to hand to an agent operating at the integration layer. Keeping extensions, decision logic, and orchestration outside the transactional core is not a concession to complexity. It is the architectural principle that preserves the space where intelligence will need to attach.

The migration is justified — but only under an AI-era design philosophy, not a 2018 one. Clean core for data integrity. Process standardisation for agent-readiness. Extensions outside the core so intelligence can reach the edges.

None of this requires predicting which AI capabilities will mature or which vendor's agent framework will prevail. It requires only the recognition that the programme's design philosophy — its assumptions about what the core is for — determines whether the resulting architecture is ready for intelligence or sealed against it.

5. The Test

There is a simple diagnostic for any programme currently in flight. Ask the transformation lead to describe the target architecture in terms of what sits outside the ERP core and why. If the answer is thin — if the programme has been designed to maximise what lives inside the vendor's platform — then the migration is solving the 2018 problem with the 2018 design philosophy. It will deliver a modern system. It will not deliver a modern architecture.

The organisations that navigate this moment will be those that recognise the migration as necessary and the original business case as incomplete. They will fund the programme, but they will redesign its purpose: not a process transformation, but a data-foundation and architectural-readiness investment that happens to require a platform change. The vendors will not frame it this way — their commercial interest lies in maximising what lives inside their core — but the practitioners running these programmes know the difference, even when the steering committee slide has not yet caught up.

The great ERP re-platforming wave arrived at the worst possible moment. But the answer is not to stop. It is to be honest about what the programme is building towards — and to ensure it is building towards the architecture that intelligence will need, not the one that the 2018 business case described.