> *Nobody in the room has an incentive to tell you the legacy system is fine. That's why almost nobody does.*

The Business Case They Didn't Show You

Every person in that room — the vendor, the CIO, the consulting firm — gets paid when you say yes. You're the only one who gets hurt when the number they omitted turns out to matter.

Paul's garage is keeping your AS/400 running. He just doesn't know it yet.

Honda stopped actively supporting the 2008 Civic years ago. No more factory parts. No dealer service campaigns. Officially, that car is on its own. And yet millions of them are still on the road — maintained by independent mechanics, third-party parts manufacturers, and a mature ecosystem that has commoditized every component. The car didn't become unreliable when Honda moved on. It became cheap to run.

Enterprise software works the same way. When a vendor announces end of support, CFOs reach for the budget approval. They shouldn't. What the vendor is actually announcing is the end of their revenue relationship with you — not the end of the system's useful life. The ecosystem that fills the gap is older, more experienced, and significantly cheaper than the vendor relationship it replaces. The fear is a sales tactic. The decision to replace should be made on capital grounds, not on the vendor's timeline.

> *The vendor support withdrawal announcement is a commercial event, not a technical one. Treat it accordingly.*

WHY THIS MATTERS NOW

CFOs are approving legacy replacement programs based on vendor-generated urgency rather than capital analysis. In most cases, the fully loaded cost of the replacement — licensing, implementation, and the permanent organizational infrastructure required to run it — is a multiple of what the legacy system costs to maintain independently.

The people who know this are not in the room. The people who profit from the upgrade are.

The Upgrade That Wasn't

Consider a European division running a legacy ERP on an older platform. The system was old. The vendor had moved on. By every conventional measure, it was a modernization candidate. The replacement initiative was approved, the consulting engagement began, and the project team grew.

The legacy system had run the division with 1.5 people in IT support. The replacement, once fully staffed, required a team approaching 70 — plus systems costs, cloud infrastructure, licensing, and ongoing maintenance contracts. After millions spent, the CFO finally asked the question that should have been first: what exactly were we buying?

The answer was uncomfortable. The legacy system wasn't inefficient. The process was inefficient. But the old system had been executing that inefficient process at virtually zero cost. The replacement didn't fix the process. It automated the same inefficiency at 70 times the human cost, with a six-figure annual technology stack underneath it.

> *The manual workaround cost almost nothing. The modern system built to eliminate it costs a multiple of everything the legacy system ever cost to run.*

The initiative was unwound. The legacy system was kept. And with it, the majority of the IT department that had been built to support the replacement was also unwound. The savings weren't just the avoided upgrade cost — they were the avoided permanent infrastructure that the modern system would have required forever.

What Vendor End-of-Life Actually Means

The vendor support withdrawal announcement follows a predictable script: security risk, compliance exposure, operational fragility, the looming cliff of unsupported infrastructure. It is designed to create urgency. It works because CFOs hear it without the context that changes its meaning.

What end-of-life actually means is that the vendor is moving their investment elsewhere. Their developers stop working on the platform. Their support contracts expire. Their sales team pivots to the replacement. None of that affects the system itself. The code doesn't degrade. The data doesn't corrupt. The processes the system executes don't stop working.

What fills the gap is the Paul's garage ecosystem. Third-party specialists who built their practices around the platform. Independent consultants who understand it more deeply than the vendor's own support staff, who were reading from documentation while the independents were solving real problems. Parts manufacturers — in software terms, the developers who maintain compatibility layers, integration toolkits, and system extensions — who have no incentive to force an upgrade cycle and every incentive to keep the ecosystem alive.

This ecosystem is not a workaround. It is a mature market. And it prices accordingly — far below what the vendor was charging for the same support level, because competition has arrived where the vendor's monopoly used to be.

> *End of vendor support is often the moment a legacy system becomes genuinely cheap to operate. The vendor's exit is the beginning of a competitive market for everything the vendor used to charge a premium for.*

The Process Conformance Trap

The standard response to the cost argument is process transformation. Modern ERP systems don't just replace legacy infrastructure — they improve the way the business operates. Best practices baked in. Industry-standard workflows. The inefficiencies of the old way finally addressed.

This is the most expensive misunderstanding in enterprise technology. ERP vendors sell process change. What they deliver is process conformance — reshaping the organization to fit the vendor's template. That is not transformation. It is compliance with someone else's design choices, built to serve the median customer across their entire install base.

The manual workarounds, the tribal knowledge, the informal adaptations that grew up around the legacy system — these are often not inefficiencies. They are the organization's institutional response to its specific operating environment. They represent decades of learning about what the business actually needs, encoded in the behavior of the people running it. Replacing them with a vendor's standard workflow doesn't capture that knowledge. It discards it.

> *The process change box gets checked. The ROI never materializes. And nobody connects the dots because the failure is diffuse and slow.*
>
> *This is ROI Theater at its most expensive.*

The genuine process transformation that would have justified the investment — the radical remapping of how work gets done — was never part of the contract. The vendor sold software. Process transformation is a separate engagement, a separate cost, and a separate risk. Most organizations discover this after go-live, when the new system is running the old process at ten times the cost.

The Permanent Organizational Footprint Nobody Models

Modern enterprise systems don't just require implementation investment. They require a permanent organizational infrastructure to operate. This is the number that almost never appears in the business case, and it is the number that matters most.

A mature, stable legacy system running on institutional knowledge requires minimal IT. The people who understand it have been running it for years. The processes are documented in behavior, if not always on paper. Interventions are rare because the system is stable. When something breaks, the 1.5 people who know it can fix it.

A modern replacement arrives with a staffing requirement baked in that doesn't go away. It requires administrators, developers, integration specialists, vendor managers, and a helpdesk function to support business units that will encounter problems the vendor's implementation team didn't anticipate. It requires people to manage the cloud infrastructure, the licensing renewals, the version upgrades, the security patches. That team doesn't shrink after go-live. It grows, because every integration added increases the surface area that needs to be managed.

The 1.5 versus 70 comparison is not a legacy system problem. It is a business case problem. The 70-person team was always the cost of the modern system. It was simply never put in the business case, because putting it there would have killed the initiative before it started.

> *The question is not whether to modernize. It is whether the organization is prepared to permanently fund the infrastructure the modern system requires — and if the answer is no, the initiative should not be approved.*

The Real Risks — And How to Weigh Them Honestly

The case for keeping a legacy system is not without limits. Two risks are real and should be modeled honestly rather than used as sales arguments by vendors.

Security exposure is genuine. Unsupported systems do not receive vendor security patches. For systems with external network exposure or sensitive data, this is a real vulnerability. The question is not whether the risk exists — it is whether the risk is being accurately sized. An isolated system running internal processes behind a firewall faces a different threat profile than a customer-facing platform. The security argument should be evaluated on specifics, not wielded as a blanket justification for replacement.

Integration brittleness compounds over time. Legacy systems that sit at the center of a growing integration ecosystem accumulate fragility with every connection added. Each integration is a dependency. Each dependency is a failure point. This risk is real — but it is also a reason to limit integration sprawl, not necessarily to replace the core system. The architectural question is whether the legacy system should be insulated from the modern ecosystem around it, rather than whether it should be replaced.

Neither of these risks negates the capital argument. They are factors to weigh in the analysis, not reasons to skip the analysis.

Three Questions Before Any Legacy Modernization Is Approved

Most legacy modernization decisions are made backwards — the initiative is approved, the vendor is selected, and the business case is built to justify the decision already made. The framework below reverses that sequence. These three questions, asked honestly before a dollar is committed, will determine whether the initiative is a capital decision or ROI Theater.

One: Are we fixing the system or the process? If the answer is the system, stop. A system replacement that leaves the underlying process unchanged will reproduce the same inefficiency at a higher cost. The investment required to fix the process is a separate initiative, a separate governance decision, and a separate risk. If the organization is not prepared to fund and execute genuine process transformation, the system replacement will not deliver the promised return. It will deliver process conformance to a vendor template — and the inefficiency will remain, now running on a modern platform with a 70-person team serving it.

Two: What is the fully loaded cost of the current state — modeled honestly? This means the legacy system cost, the manual workarounds, the tribal knowledge risk, and the third-party ecosystem cost at current market rates. Not the vendor's assessment of your risk. The actual cost of keeping what you have, maintained by Paul's garage rather than the original manufacturer. That number, modeled honestly, is frequently far lower than the business case for replacement assumes.

Three: What is the fully loaded permanent cost of the future state? Not just the implementation. The team required to run it. The cloud infrastructure. The licensing renewals. The integration management. The version upgrade cycle. The vendor support contracts. Add it up across five years and compare it to the legacy carrying cost. If the replacement costs ten times more to operate permanently than the current state, the burden of proof is on the upgrade — not on the decision to stay.

> *The CFO who asks these three questions before approving the initiative is doing their job. The CFO who approves the initiative and asks them afterward is managing a crisis.*

KEY TAKEAWAYS

→ Vendor end-of-support is a commercial announcement, not a technical failure. The ecosystem that replaces vendor support is often cheaper, more experienced, and more competitive — the Paul's garage market that makes the 2008 Civic still the right car for millions of drivers.

→ Legacy systems that outlive vendor support frequently become the most capital-efficient infrastructure in the portfolio. The risk is theoretical. The savings are real and compounding.

→ ERP modernization does not deliver process transformation. It delivers process conformance to a vendor template. The genuine process change that would justify the ROI is a separate initiative that is almost never funded or executed.

→ The permanent organizational footprint required to run a modern system is the number that matters most — and the number most consistently absent from the business case. Model it across five years before approving any replacement.

→ Three questions before any legacy modernization: Are we fixing the system or the process? What does the current state actually cost, modeled honestly? What does the future state cost permanently — not just to implement, but to operate forever?

PLATEAU STRATEGY
Capital Efficacy Series

Authored by
Allan Salek

Four continents. Multiple capital cycles. I've watched brilliant CFOs get blindsided — by vendors selling certainty, by organizations caught in politics, by leaders advancing careers ahead of the balance sheet.

We work alongside CFOs so they walk into every room ready for the conversation that defines whether they keep the seat.

GET IN TOUCH
allan.salek@plateaustrategy.com

Capital Efficacy · IT Balance Sheet · External Risk

plateaustrategy.com