



Global Orchestration

What ERP Makes Possible.



Executive Summary

Consider what happens when a supply chain team in Munich makes a routine inventory decision on a Tuesday morning. That decision, the kind made a hundred times a day, ripples outward. It affects fulfillment commitments made to customers in London. It touches a procurement contract negotiated in Singapore. It has tax implications in three jurisdictions before lunchtime. The business is already operating as one system whether the ERP is designed that way or not.

An organization's ERP is the nervous system of that reality; ERP has to hold together things that are intentionally different and make them work as one. Finance is global, but tax is local. Supply chain is global, but execution is regional. Data is centralized, but ownership is federated. The goal of ERP is global orchestration: the ability to align decisions, data, and processes across an enterprise in real time, across borders, without losing coherence.

The Starting Premise

There is still a prevailing misconception baked into how most organizations think about global ERP from the onset. The assumption is that ERP is global because the company operates globally; multiple geographies, multiple currencies, multiple legal entities, so naturally, a global system. That assumption is exactly why so many global ERP programs, even well-resourced ones with strong vendors and experienced SIs, still struggle to deliver. If the premise is wrong, everything that follows is just expensive execution.

ERP is global because value, risk, and decisions are now globally interconnected. What used to be regional decisions, pricing, supply chain, tax, working capital, are now tightly coupled across borders. A pricing call made in one market affects margin in another. A supply chain disruption in one region hits fulfillment in three others within hours. A tax structure in one jurisdiction flows into reporting obligations somewhere else entirely.

The goal isn't to make everything the same everywhere.

It's to make everything work together; even when it's different.





The Three Layer Approach to Global ERP

Getting to a point of global orchestration requires designing the right architecture; knowing what to standardize, what to harmonize, and what to leave deliberately local. And then protecting that architecture through execution; because even the right design can be destroyed in the handoffs, the translations, and the gaps between work streams. Global ERP needs to be designed as a three-level layer. Each layer requires a different level of control and confusing them is where programs break.

1

The Global Core: Standardize Without Exception

This is the non-negotiable layer. Finance structures, chart of accounts, ledgers, master data governance, and the core process flows, such as record-to-report, procure-to-pay, order-to-cash, must be consistent everywhere, without exception, or the whole architecture will fall apart.

This is also where orchestration begins. A core that is genuinely standardized is what makes it possible to see the whole enterprise clearly; to move data across borders without translation, to consolidate financial reporting without reconciliation difficulties, to make decisions with confidence that the numbers mean the same thing in every market.

The discipline required here is protecting this layer from exception pressure. Every country team will have reasons why their situation is different. Some of those reasons are legitimate, and they belong in the next layer. Many of them aren't, and they need to be held to the global standard regardless.

2 Regional Execution: Standardize the core. Control variation at the edge

Tax, statutory reporting, supply chain execution nuances, customer-specific processes; these vary significantly across markets and for good reason. They cannot be standardized without either breaking compliance or failing operationally. But they can be harmonized. That means defining the permitted range of variation, setting guardrails, and ensuring localization decisions are made deliberately and with authority; not picked up ad hoc during rollout when they're expensive to fix.

This layer is where orchestration gets tested. The global core sets the rules. The regional execution layer is where those rules meet the complexity of real markets, and where the design either holds or begins to fracture.

Europe is the clearest example of why this matters. There are so many regulatory environments stacked on top of each other; different VAT regimes, different statutory reporting requirements, different labor protections, different data privacy obligations; that a template built without European input from the start will crack when it arrives there. The regional execution layer is the mechanism that makes global orchestration actually work in practice.





3

The Innovation Layer

The third layer is where the technology is moving fastest: AI, advanced analytics, automation, hyperscale capabilities, industry-specific differentiation, customer experience. It is also where organizations have the most opportunity to build genuine competitive advantage on top of their ERP foundation.

But this layer is only accessible if the two layers beneath it are working. AI is only as good as the underlying structure it sits on. It can be activated in any operating model and will technically function. But if master data is inconsistent, if the global core wasn't held to standard, if the regional execution layer is full of ungoverned local exceptions — AI will simply amplify bad decisions faster. The innovation layer is where orchestration pays off. The global core is what makes it possible.

When the three layers are designed correctly and kept distinct, the result is an ERP program that can genuinely orchestrate; that can hold global coherence and local relevance simultaneously, and adapt to change in without rebuilding from the ground up. That is what best practice is at the design stage.

The Translation from Blueprint to Build

Across multinational implementations, the design is rarely what fails. What fails is what happens to an ERP program after the design phase ends; in the translation from blueprint to build, in the pressure of testing cycles, in the gap between a system that is technically live and a business that can operate on it. The design determines what is being built. Execution determines whether it actually gets built.

Every global ERP program has the same work streams: design, build, test, cutover, go-live, stabilization. The lanes are defined, resourced, and managed. What gets far less attention is what happens between them. Orchestration breaks down in the seams; the handoffs and transitions between work streams where the logic of the design meets the complexities of delivery. No one owns those seams, not the PMO, not the SI, and not the business. And this is where programs are won or lost.

Every ERP program is well structured. Most are well resourced.
They still fail because **no one owns the seams.**



From Blueprint Design to Building the Program

The first seam is where the orchestration logic of the design gets lost in translation.

The blueprint is designed onshore, in workshops, with senior business stakeholders. It's well-documented and signed off. The three layers are defined. The guardrails are set. Then the build begins, often off shore. And somewhere in that transition, the program starts to come apart.

Business requirements that were clear in a workshop context become ambiguous when handed to a team as written specifications. The nuances, the things understood in the room but never fully captured on paper, don't make the crossing. What comes back is a build that technically meets the spec but doesn't reflect what the business actually needs. The global core loses its integrity. The regional execution logic gets genericized. By the time the gap surfaces in testing, the schedule is tight and the cost of fixing it is multiples of what it would have been at design stage.



A large European public sector organization initiated an Oracle Fusion implementation with a £19 million budget. The system went live on schedule, but significant functionality and compliance issues surfaced almost immediately. Independent audit reports cited insufficient internal knowledge of the platform, inadequate business readiness validation, and governance gaps between technical delivery and operational requirements. By 2024, total costs had reached approximately £90 million. Projections through 2026 put the final figure close to £216 million — more than eleven times the original estimate. The blueprint had been built. The business requirements that should have shaped it had never been adequately captured or validated.

System Integration Testing

The second seam is where the gaps left by the first one become visible; and where the pressure to ignore them is highest.

Testing is where design failures surface. By the time a program reaches SIT and UAT, the schedule is fixed, the budget is largely spent, and the go-live date has been communicated. The temptation is to move through testing quickly, treat failures as punch list items, and push forward. SIs are often so focused on hitting their testing milestones that the phase becomes about passing the gate, not validating that the orchestration actually works. Is the global core producing consistent data? Is the regional execution layer handling local compliance correctly? Will the system and the business be able to function together under real conditions?

A major confectionery manufacturer compressed its testing phase to fit a rushed timeline and went live at its peak demand season. The result: \$100 million in orders that couldn't be processed even though the inventory was physically in the warehouse; the ERP system couldn't see it. Stock fell 8%.

A global sportswear company's \$400 million supply chain and ERP overhaul went live without adequate integration testing. The system ordered low-demand product lines in excess and couldn't fulfil orders for high-demand ones. The result was \$100 million in lost sales, a 20% stock price drop, and class-action lawsuits. The company then spent a further five years and another \$400 million fixing it. The orchestration between the supply chain system and the ERP was never properly validated before go-live.

Assessing Business Readiness

The third seam is the most consequential.

Business readiness is the validation that the orchestration architecture actually works for the people who have to use it, in the markets where they have to use it, under real operating conditions. It requires operational simulation and it requires someone at executive level who is personally accountable for it.

A system can be technically live and still unable to orchestrate. If the global core wasn't properly standardized, finance teams in different markets will be producing numbers that can't be consolidated. If the regional execution layer wasn't properly harmonized, a compliance process that worked in one market won't work in another. These failures don't announce themselves in testing; they announce themselves when real people try to do real work on the system for the first time.

After multiple go-live delays spanning more than eighteen months, a global medical device manufacturer went live on a new ERP platform. Following go-live, the organization reported significant difficulties with order fulfilment, shipment processing, invoicing, and basic sales reporting, attributing roughly \$75 million in annual revenue impact to implementation-related disruption. The technical go-live happened, but the orchestration at a business level didn't.

A contrasting example illustrates what getting this right looks like. A US-based medical device manufacturer undertook a two-year Oracle rollout across the US and Canada, then Europe. For six months at executive level, the leadership team debated the European rollout. The debate kept circling around timelines and budget until the COO became directly and personally accountable for the decision. Business readiness was elevated to a formal work stream with its own governance. An honest readiness assessment was conducted. Once it was, the deployment model decision became straightforward, and the European go-live succeeded.



Orchestration Is a Discipline

ERP is hard for reasons that don't change much. The technology improves. The vendors ship faster. The cloud changes the options. But the fundamental challenge of getting a large, complex organization to align decisions, data, and processes across borders, functions, and cultures is what remains hard. Because it is, at its essence, an orchestration problem. And orchestration is not a feature of ERP software; it is a discipline of an organization.

The organizations that succeed are the ones that treat orchestration as the standard against which every decision in the program is measured. They design the three layers correctly and hold the line on each of them; standardizing the global core without exception, harmonizing regional execution within clear guardrails, and building the innovation layer on a foundation that can actually support it. And they understand that the design is only the beginning.

The seams are where orchestration gets destroyed. The blueprint-to-build handoff is where the design logic loses its integrity. The testing phase is where gaps get buried rather than fixed. The go-live transition is where a technically live system meets a business that isn't ready to use it. These are not random failure points; they are the predictable places where delivery reality can override design intent.

Knowing where those seams are, addressing them before they become crises, and having the experience to hold the orchestration logic together from design through delivery is what makes a global ERP program successful.

About RGP

RGP is a global professional services firm with nearly three decades of experience helping the world's top organizations—from Fortune 50 to fast-moving startups—solve today's complex business problems. A trusted partner to CFOs and finance leaders, we deliver the talent, consulting, and outsourced services solutions you need to grow faster, work smarter, and keep up with change—all through a flexible model and global network of experts.

