EXPERT SOFT
software development

When Enterprise AI Architecture Meets Platform-First Strategy

Where platform-first helps, and where AI architecture still needs you.

When enterprises stop experimenting with isolated AI features and start treating AI as part of their core systems, the conversation shifts from “what can we try?” to “what can we safely build on?”

For many organizations, that’s where platform-first naturally appears as a safe choice for complex commerce and ERP landscapes. But whether that choice pays off depends less on the platform itself and more on how it fits into the architecture you already live with: systems, data, workflows, regulations, and all their quirks.

This whitepaper looks at platform-first through that lens. It maps the structural core any enterprise AI system needs, shows where platforms genuinely carry the load and where responsibility quietly shifts back to your teams, and outlines how forward-thinking organizations design pilots and early use cases so they can scale later.

The perspective comes from systems that had to keep working on Monday morning, not from slideware.

Table of Contents

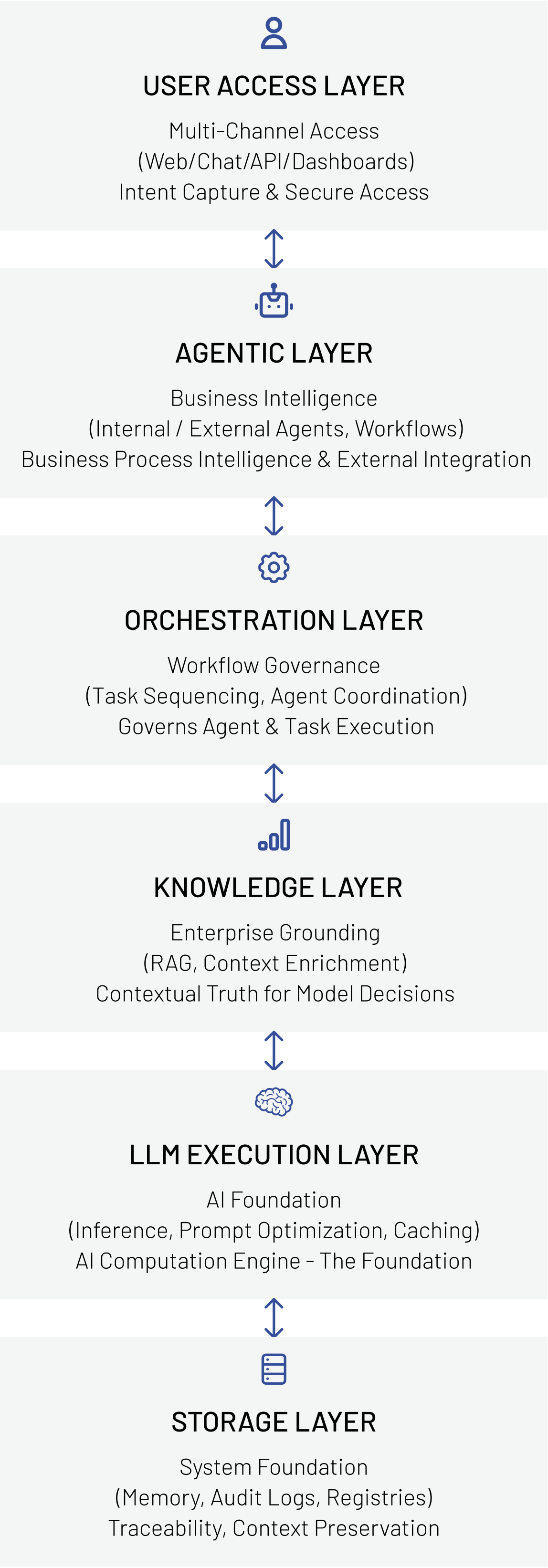
Architectural Core Behind Enterprise AI Solutions

For AI solutions to take root in a complex enterprise landscape — with numerous systems, intricate data flows, and countless interdependencies — **they should be designed for that environment from the start**. Such resilience comes from an AI architecture built to operate within enterprise realities, not beside them.

In this context, architecture no longer focuses on connecting a model to data. **It creates a system that coexists with everything the enterprise already relies on.** A production-grade AI solution is defined not by the sophistication of its models, but by how resilient, observable, and governable the entire system becomes.

Each layer in a modern AI system exists for that reason: to create clarity of responsibility and ensure the system can be trusted as it grows.

From experience designing and operating AI within enterprise environments, seven foundational layers consistently define what "production-grade" AI truly means.



USER ACCESS LAYER

Multi-Channel Access
(Web/Chat/API/Dashboards)
Intent Capture & Secure Access

AGENTIC LAYER

Business Intelligence
(Internal / External Agents, Workflows)
Business Process Intelligence & External Integration

ORCHESTRATION LAYER

Workflow Governance
(Task Sequencing, Agent Coordination)
Governs Agent & Task Execution

KNOWLEDGE LAYER

Enterprise Grounding
(RAG, Context Enrichment)
Contextual Truth for Model Decisions

LLM EXECUTION LAYER

AI Foundation
(Inference, Prompt Optimization, Caching)
AI Computation Engine - The Foundation

STORAGE LAYER

System Foundation
(Memory, Audit Logs, Registries)
Traceability, Context Preservation

USER LAYER

The user layer acts as the controlled entry point where user intent is captured, translated, and routed into the organization's AI workflows. In mature environments, this layer extends across web, mobile, chat, and internal tools, supporting natural language, file uploads, and structured inputs consistently. Real-time interaction, contextual continuity, and secure identity handling turn the interface into part of the system's reliability, not just its surface.

Key Capabilities: multi-channel access, real-time interaction and contextual workflows, rich outputs such as tables, citations, and action buttons.

Role in Enterprise: establishes a secure, consistent interface for AI interactions across teams and systems, forming the first point of governance and observability.

If Weak or Missing: AI access becomes fragmented and inconsistent, context is lost between channels, and user feedback can't flow back into improvement cycles.

There's more than one way to build a scalable AI core. Explore their trade-offs, strengths, and where each makes the most sense to choose the one that suits you best.

AGENTIC LAYER

The agentic layer is the collaboration fabric for AI in the enterprise. Inside the organization, specialized agents handle focused tasks or domains and coordinate via shared orchestration, message buses, and short-term working memory. Outside, the same layer controls how these agents interact with cloud models, partner APIs, and third-party service. So external intelligence extends the system without weakening security or governance.

- **Key Capabilities:** supervisor/worker agent roles, short-term task and context memory, shared coordination policies, secure API and partner integrations, access and data governance; throttling and encryption.
- **Role in Enterprise:** enables scalable cooperation between internal AI functions and external services, supporting parallel execution, controlled knowledge exchange, and safe use of outside capabilities.
- **If Weak or Missing:** coordination logic fragments, responsibilities blur, integrations become brittle or risky, and workflows are harder to evolve and operate reliably at scale.

ORCHESTRATION LAYER

The orchestration layer gives structure to how AI systems act. It coordinates intent, data, and tools, deciding which components to activate, in what sequence, and under what conditions. This layer keeps complexity manageable by translating individual model calls into governed, repeatable workflows. It’s the difference between a collection of AI features and a functioning, observable system.

- **Key Capabilities:** intent routing, workflow management, agent coordination, human-in-the-loop escalation.
- **Role in Enterprise:** acts as the operational brain that ensures AI logic runs consistently across teams, channels, and use cases with defined entry points, safeguards, and recovery paths.
- **If Weak or Missing:** AI remains fragmented into isolated automations, logic duplicates across solutions, and reliability declines as integrations multiply.

RETRIEVAL & KNOWLEDGE LAYER

The retrieval and knowledge layer anchors AI in the organization’s factual reality. It connects models to verified enterprise data, from product catalogs and documents to transactional and historical records, ensuring every response is grounded in truth, not assumption.

This layer enables retrieval-augmented generation (RAG) pipelines, vector and graph databases, and semantic enrichment workflows that give AI access to the right information with the right context.

- **Key Capabilities:** retrieval-augmented generation (RAG), metadata and semantic enrichment, contextual grounding and relevance scoring, knowledge synchronization across systems.
- **Role in Enterprise:** serves as the single source of truth for AI-driven decisions and content generation, ensuring that outputs align with approved enterprise knowledge.
- **If Weak or Missing:** AI responses drift from factual accuracy, duplication and data silos emerge, and every new use case rebuilds its own version of “truth,” making governance and consistency impossible.

LLM EXECUTION LAYER

At the center of the system, the LLM execution layer turns intent into computation. It manages how models are invoked, balanced, and observed, keeping performance stable as workloads grow. In enterprise settings, this layer transforms AI from an experimental capability into a predictable operational service, maintaining both efficiency and transparency under enterprise workloads.

- **Key Capabilities:** prompt routing and execution, caching and response validation, latency and cost monitoring, quality and performance observability.
- **Role in Enterprise:** creates a controlled environment for inference, where performance can be tuned, budgets forecasted, and results audited with confidence.
- **If Weak or Missing:** behavior under load becomes erratic, costs expand silently, and AI output drifts beyond measurable standards.

STORAGE & REGISTRY LAYER

This layer keeps AI systems coherent over time. It maintains stateful, multi-turn conversations and short-term agent memory, so context is never lost between requests or workflows. It acts as the single source of truth for the orchestrator's discovery and routing decisions, ensuring every task is directed to the most competent and authorized resource.

Beyond continuity, it provides traceability, building unalterable audit trails that support debugging, quality assurance, and regulatory standards such as ISO 9001 or SOX.

- **Key Capabilities:** sateful memory and context preservation, routing metadata and discovery index, audit logs for compliance and traceability.
- **Role in Enterprise:** anchors the system's operational memory and compliance posture, allowing every decision, interaction, and agent action to be verified or reproduced when needed.
- **If Weak or Missing:** routing becomes unreliable, context disappears mid-conversation, and compliance collapses, leaving no verifiable record of how or why AI decisions were made.

When these layers work together, they create a living system that can adapt, scale, and remain accountable within enterprise boundaries.

Some of these capabilities can be accelerated through modern AI platforms, while others demand engineering ownership. The balance between the two defines how well an enterprise can turn architectural discipline into lasting agility.

The Architecture Platform-First Solutions Build

The platform-first approach turns architecture into a layered partnership. The platform stabilizes what should be universal, while enterprises retain control over what defines them.

PLATFORM-FIRST AI APPROACH IN ENTERPRISE REALITY

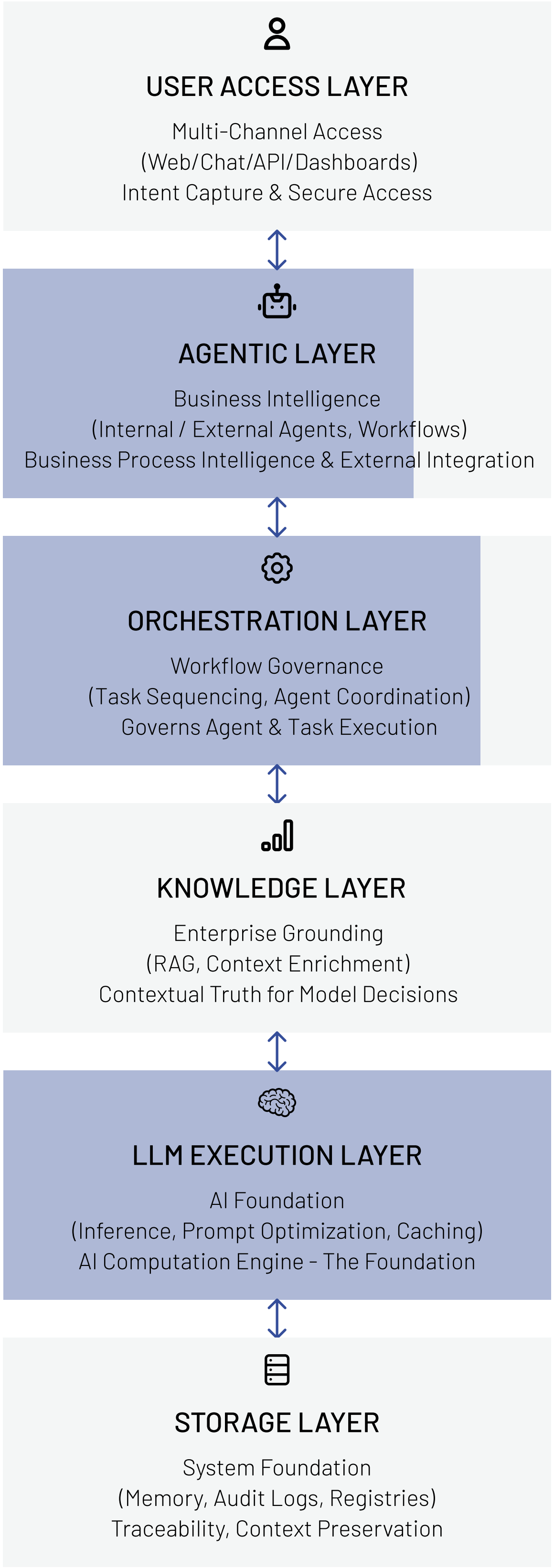
For enterprises running large digital and commerce platforms, the platform-first approach offers a practical way to introduce AI at scale without destabilizing existing systems. It focuses on shared, repeatable components — execution, orchestration, and governance — that every use case depends on. Platform-first works because it provides:

- **Unified foundation:** shared runtime, orchestration, and monitoring replace fragmented, ad-hoc tooling.
- **Faster scaling:** teams can deploy and iterate across multiple AI use cases without rebuilding execution layers each time.
- **Governed consistency:** security, access control, and audit hooks come built in, reducing operational risk.
- **Extensibility:** enterprises can extend platform primitives with domain logic rather than reinventing infrastructure.

Besides, this approach fits naturally with the architecture of modern enterprise AI systems. It covers several foundational layers and takes over the operational load they usually create, giving enterprises stability without adding another system to manage.

HOW IT BRINGS STRENGTH

If you look back at the architecture diagram from the previous section, the picture changes once a platform-first approach comes into play: several layers light up in blue as the platform takes over their underlying complexity.



USER ACCESS LAYER

Multi-Channel Access
(Web/Chat/API/Dashboards)
Intent Capture & Secure Access

AGENTIC LAYER

Business Intelligence
(Internal / External Agents, Workflows)
Business Process Intelligence & External Integration

ORCHESTRATION LAYER

Workflow Governance
(Task Sequencing, Agent Coordination)
Governs Agent & Task Execution

KNOWLEDGE LAYER

Enterprise Grounding
(RAG, Context Enrichment)
Contextual Truth for Model Decisions

LLM EXECUTION LAYER

AI Foundation
(Inference, Prompt Optimization, Caching)
AI Computation Engine - The Foundation

STORAGE LAYER

System Foundation
(Memory, Audit Logs, Registries)
Traceability, Context Preservation

The following view summarizes where this shift typically occurs and how it strengthens the overall system design, turning recurring engineering effort into reusable infrastructure.

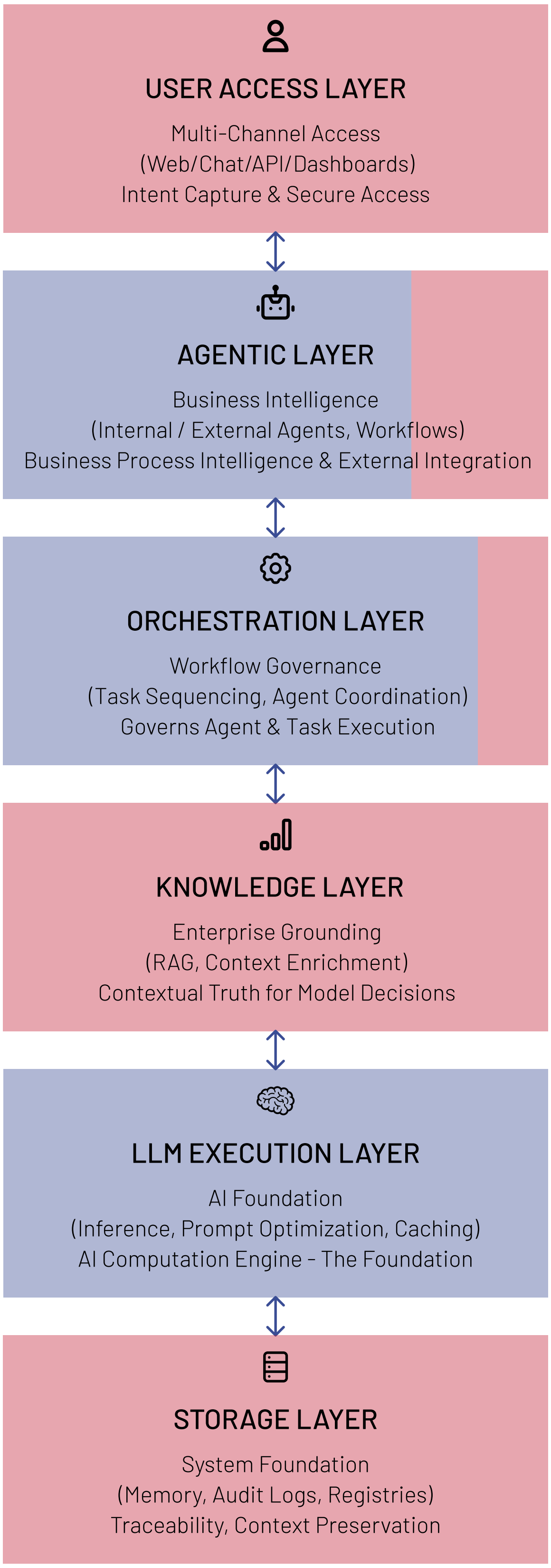
LAYER	PLATFORM COVERAGE (CAPABILITIES)	IMPACT FOR ENTERPRISE
Retrieval & Knowledge Layer	Retrieval-augmented generation (RAG) framework, embedding pipelines, vector and hybrid database integrations, unified query interface	Standardizes data access and retrieval workflows, enabling consistent grounding across domains while keeping data ownership within enterprise boundaries
LLM Execution Layer	Unified model runtime with provider-agnostic APIs, prompt routing, caching, cost and latency monitoring, and execution isolation	Removes the need to operate model infrastructure, while ensuring predictable performance and auditability across use cases — a foundation for scaling without performance drift
Orchestration Layer	Workflow and task orchestration primitives, stateful execution, retry and timeout handling, async and streaming support	Provides a resilient coordination layer that abstracts operational complexity and ensures workflow reliability under enterprise-level load
Storage & Registry Layer	Logging and tracing frameworks, audit trail foundations, model and agent registries, access control primitives	Establishes a baseline for observability and traceability, supporting internal governance and external audit requirements without additional tooling
Agentic Layers	Agent runtime environment, discovery registry, lifecycle management, secure connectors, and controlled API gateways	Enables secure collaboration between agents and external systems through standardized communication and governance, reducing integration friction and compliance risk

The strength of a platform-first approach appears once its boundaries are clear. Platforms anchor the system by handling what's universal, but lasting success depends on how well enterprises recognize where that coverage stops and where their own architectural responsibility begins.

The Enterprise Side of Platform-First Approach

Platform-first does not remove complexity, but moves it. Instead of managing execution or infrastructure, you focus on the layers where data, orchestration, and business context live.

Let's once again look at our diagram, which is extended to show the full picture of how this responsibility is distributed. The red color is added to show the responsibility of the enterprise.



USER ACCESS LAYER

Multi-Channel Access
(Web/Chat/API/Dashboards)
Intent Capture & Secure Access

AGENTIC LAYER

Business Intelligence
(Internal / External Agents, Workflows)
Business Process Intelligence & External Integration

ORCHESTRATION LAYER

Workflow Governance
(Task Sequencing, Agent Coordination)
Governs Agent & Task Execution

KNOWLEDGE LAYER

Enterprise Grounding
(RAG, Context Enrichment)
Contextual Truth for Model Decisions

LLM EXECUTION LAYER

AI Foundation
(Inference, Prompt Optimization, Caching)
AI Computation Engine - The Foundation

STORAGE LAYER

System Foundation
(Memory, Audit Logs, Registries)
Traceability, Context Preservation

Each of these layers holds the complexity that keeps AI aligned with real enterprise operations. Look at the table.

LAYER	ENTERPRISE RESPONSIBILITY	WHY IT MATTERS	FOCUS AREAS
User Layer	User interfaces, interaction channels (chat, dashboard, API), and role-based UX logic integrated into corporate systems.	Defines how users interact with AI and ensures outcomes are transparent, explainable, and aligned with enterprise workflows.	Focus on seamless integration into existing tools, accessibility across roles, and clarity of user feedback loops.
Retrieval & Knowledge Layer	Data source selection, ingestion pipelines, cleansing and enrichment logic, business semantic model, and graph schema of enterprise entities.	Keeps AI grounded in accurate and governed enterprise knowledge, avoiding hallucinations and data drift.	Prioritize data lineage, enrichment quality, and semantic alignment with operational systems.
Orchestration Layer	Business workflows, process sequences, approval and escalation paths, and exception handling logic beyond system-level orchestration.	Aligns AI processes with real decision structures and compliance requirements.	Focus on workflow flexibility, observability, and audit readiness for regulated or high-impact operations.
Storage & Registry Layer	Configuration of data retention policies, audit scope, privacy controls, and compliance with regional and industry standards.	Guarantees that stored outputs and logs meet enterprise governance and audit requirements.	Focus on balancing storage optimization with regulatory depth and secure access control.
Agentic Layers	Domain-specific agents for HR, Sales, or Operations; reasoning logic; fallback and escalation rules; human-in-the-loop configurations.	Translates enterprise knowledge and process logic into autonomous, accountable actions.	Focus on the reliability of agent reasoning, transparency of decision chains, and escalation handling

Ownership of these layers lies in interpretation: how data connects, how workflows adapt, how agents reason inside a business context. These choices define whether an AI system stays alive within the enterprise or turns rigid over time. The strongest teams treat this space not as what's left after the platform, but as where design discipline begins.

Designing Platform-First AI with Enterprise Reality in Mind

Platform-first gives structure, but it doesn't guarantee stability. Once the platform takes over the basics, success depends on how the enterprise designs around it — how teams handle what stays theirs, what grows with the platform, and what may one day need to move beyond it.

The real advantage appears when the platform becomes part of the enterprise mindset, not just its stack. The principles outlined in this section are precisely about this.

THINK IN SYSTEMS, NOT STAGES

You can start small, with a PoC, a use case, or a single agent. But these early steps silently shape the long-term system. Thinking in systems means treating even first experiments as connected pieces of a shared architecture — same data principles, same observability, same security boundaries. When that continuity exists, scale stops being a rebuild and becomes a rollout.

Architect's perspective: trace how each "temporary" solution connects to a shared dependency, because it already does.

KEEP WHAT'S YOURS INDEPENDENT

Business logic, reasoning flows, and decision frameworks should live apart from platform abstractions. The thinner the dependency, the easier it becomes to evolve without friction. Enterprises that preserve this separation can refactor logic or replace components without breaking operational consistency.

Architect's perspective: treat coupling as a cost, even when it feels convenient.

DESIGN FOR VISIBILITY FROM DAY ONE

Systems that lack traceability at the start often compensate with over-control later. Embedding observability and auditability early allows performance, reasoning, and compliance to evolve together. Transparency becomes part of system health, not a constraint.

> **Architect's perspective:** visibility is never retrofitted, by the time you need it, you've already lost context.

BUILD FOR ALIGNMENT, NOT DEPENDENCE

The best enterprise systems work with their platforms, not against them. Alignment means using platform primitives as stable foundations while keeping intent and intelligence in enterprise-owned layers. Dependence, on the other hand, locks evolution to someone else's roadmap. Architecture grows stronger when the platform is a partner, not a boundary.

> **Architect's perspective:** real alignment is measured by how easy it is to disagree without breaking the system.

KEEP COHERENCE AS SYSTEMS MULTIPLY

Enterprise AI rarely grows linearly, it branches by function, geography, and integration depth. Without common orchestration patterns and data semantics, every new case becomes a custom island.

Coherence means giving those branches a shared language — so they can grow apart without falling out of sync.

> **Architect's perspective:** coherence erodes fastest when every team optimizes locally, and no one maintains the shared rules.

DESIGN WITH AN EXIT IN MIND

Platform-first is often the best way to start, but it shouldn't become the only way to continue. Some systems will eventually outgrow the limits of standard frameworks, needing custom orchestration, deeper integration, or domain-specific logic that the platform wasn't built to host. Keeping an exit in mind means designing interfaces, data pipelines, and agent logic so they can evolve outside the platform if needed, without rewriting the foundation.

> **Architect's perspective:** exits fail when design intent lives only inside the platform and nowhere else.

When design thinking stays disciplined — when logic, data, and ownership remain transparent — the platform becomes an enabler, not a limiter. That discipline is what keeps innovation and system costs aligned over time

The Word for Teams Planning AI Pilots Today

For enterprise AI, architecture is where most of the real risk and leverage sit: it decides whether new capabilities stay understandable, controllable, and affordable as they spread across the organization. The platform-first approach can be a very effective way to anchor that architecture, as long as it's seen as shared infrastructure, not as a substitute for design decisions.

The platform can reliably cover the shared core — execution, orchestration primitives, integrations, observability hooks — but it won't decide how your data is modeled, how workflows really run, or where business logic should live.

For teams planning AI pilots today, this means a pilot is already an architectural move, whether it's labeled that way or not. Early decisions about how agents interact with data, how results flow into existing systems, and which parts depend directly on platform-specific features all have a long tail.

Teams that approach pilots with this awareness use platform-first to move faster, while designing with clear boundaries, future options, and system-level coherence in mind.

About Expert Soft

Expert Soft is a targeted ecommerce software delivery company, partnering with Fortune 500 companies and global corporations across the US and EU. With SAP Commerce Cloud and Java as our backbone, we know how to ensure scalable and high-performing solutions that can handle 1 mln requests per second, delivering a smooth customer experience.

Developing a payment engine that saved our client about $100 million in operational expenses, ensuring multi-country platform support, adapting solutions for new market entry with tailored enhancements — these are just a few of the challenges our specialists tackle.

We aim to deliver more than a software system. We aim to deliver tailored solutions that maximize profitability within available resources. Our success is driven by:

TEAM STRENGTHS

- All our engineers have a university background
- Specialists excel their skills in our training LABs
- Perfect English skills
- Ready to help 24/7

CLIENTS

We work with corporations around the world with revenue of over $20 billion and 150K+ employees.

APPROVALS BY AUDITS

Our ongoing work with corporations is consistently validated through rigorous audits, both by internal teams and Big 4 consulting firms.

HIGH-LEVEL SECURITY

Approved by assessments from global companies, who are leaders in their respective industries.

BUDGET EFFICIENCY

By carefully aligning technology investments with your business goals, we ensure optimal value and cost-effectiveness.

PROFESSIONAL TEAM

No offshore outsourcing and our team's average tenure of 4+ years means you get seasoned problem-solvers, not just coders.

EXPERT SOFT EXCELS IN

- PAYMENT ENGINE
- MICROSERVICES ARCHITECTURE
- CONTENT MANAGEMENT
- REDESIGN
- E-COMMERCE PLATFORM
- HEADLESS COMMERCE
- MICRO FRONTENDS
- MIGRATION&INTEGRATION

OUR TECH CORE

FRONT-END

HTML, CSS, JavaScript (Angular, React, Vue, Next, TypeScript, Jquery), Spartacus

BACK-END

Java EE, Spring, SAP Commerce (Cloud), Node.JS.

DEVOPS

Docker, Kubernetes, CI/CD

UX/UI DESIGN

UX Research, UI Design, Figma, Adobe, Sketch

QUALITY ASSURANCE

Manual Testing, Test Automation

TARGETED DOMAINS

RETAIL

TELECOM

HEALTHCARE

FINTECH

WHOLESALE

MANUFACTURING

SHARED PATHS, LASTING ECOM VICTORIES

LET'S TALK SOLUTIONS!

EKATERINA LAPCHANKA
Chief Operating Officer
kate.lapsenco@expert-soft.com

+1 585 4997879

+371 25 893 015

PAVEL TSARYKAU
CEO & Founder
of Expert Soft

Let’s connect

expert-soft.com

LinkedIn