The Austrian Journal of Technical and Natural Sciences, 2026 No 3–4
ISSN 2310-5607

PREMIER Publishing
ppublishing.org

Section 3. Computer sciences

DOI:10.29013/AJT-26-3.4-98-105

DECOMPOSITION OF MONOLITHIC SOLUTIONS IN CORPORATE PROCUREMENT PLATFORMS: ARCHITECTURAL MIGRATION STRATEGIES AND COMPLEXITY MANAGEMENT

Roilian Mykyta [1]

[1] Engineering Manager, Lean DNA

Cite: *Roilian Mykyta. (2026). Decomposition of Monolithic Solutions in Corporate Procurement Platforms: Architectural Migration Strategies and Complexity Management. Austrian Journal of Technical and Natural Sciences 2026, No 3–4.* https://doi.org/10.29013/AJT-26-3.4-98-105

Abstract

This article examines the process of decomposing monolithic information systems in corporate procurement platforms, with a focus on architectural migration strategies and the management of emerging complexity. The study analyzes principles for transitioning to modular architectures, including the use of facade layers, service isolation, and intermediary adapters that enable phased transformation without disruption of business processes. The importance of managing architectural and operational complexity in the adoption of distributed solutions is emphasized. Practical examples of applying architectural approaches under industrial operating conditions are presented. Typical risks associated with the transition are discussed, along with the potential long-term benefits of modular systems. Particular attention is given to resilience, scalability, and integration compatibility of the resulting architectures.

Keywords: *monolithic architecture, corporate platforms, microservices, modular systems, digital transformation, information systems architecture*

Introduction

In the context of rapid digitalization of corporate processes and increasing demands for the flexibility of information systems, an increasing number of industrial and logistics enterprises are facing the need to transform legacy monolithic solutions. This challenge is particularly acute in corporate procurement and warehouse management platforms, where a high degree of process formalization, strict requirements for fault tolerance, and the need for integration with both internal and external systems necessitate technological solutions capable of ensuring operational stability and adaptability. Under these conditions, the transition from monolithic architectures to modular and distributed systems becomes not merely a technical task, but a strategic decision that directly affects the sustainability of the enterprise's overall operating model.

The purpose of this article is to examine architectural migration strategies from monolithic information systems to modular solutions in the context of corporate procurement platforms. The analysis focuses on decomposition technologies, including the use of façade layers, service isolation mechanisms, and intermediary adapters, which make it possible to minimize the impact of architectural changes on business processes and to ensure a gradual and controlled transition.

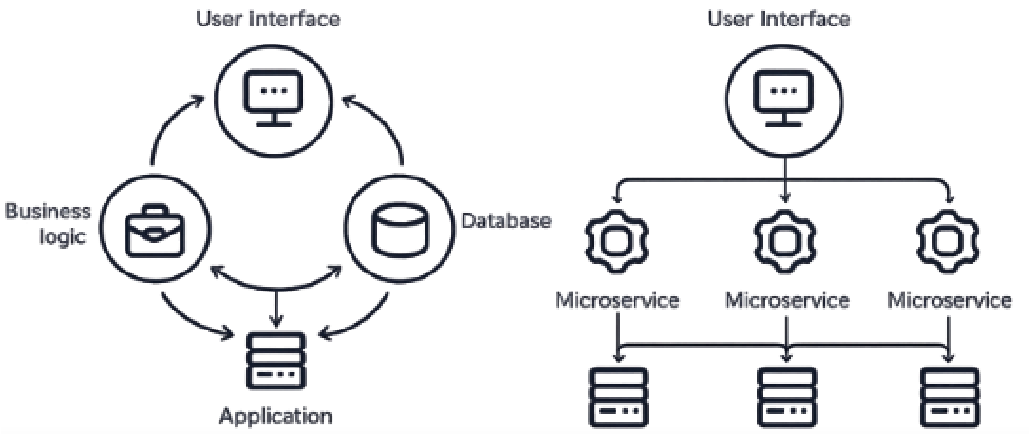
To achieve the stated objective, the study employs a structured analytical and comparative approach based on the synthesis of academic literature and reported industry cases, focusing on architectural migration strategies, integration constraints, and sources of complexity in the decomposition of monolithic procurement platforms.

Main Part

Monolithic IT Systems in the Corporate Architecture of Procurement Platforms

Monolithic IT systems of corporate architecture are defined as a class of enterprise information systems where functional modules, business logic, mechanisms of data processing, and user interface are implemented as a monolithic software application, which is deployed as an integral artifact (fig. 1).

Figure 1. *Comparison of Monolithic and Microservice Architectures*



Historically, monolithic architectures have served as the foundation of corporate information systems, including platforms supporting procurement, warehouse management, and logistics processes. Such solutions are characterized by a high degree of component integration: all system modules operate within a single application, sharing a common codebase, data storage, and execution infrastructure. This organization provides initial development efficiency, particularly under conditions of fixed business processes typical of large industrial enterprises with strictly regulated document management, logistics, and warehouse operations.

However, over time, monolithic architectures tend to exhibit limited adaptability to changing external conditions and internal organizational transformations (table 1).

Table 1. *Technical Limitations of Monolithic Architecture in Corporate IT Systems (Saucedo A. M. et al., 2025; Katal A. et al., 2025)*

Limitation	Description
Limited scalability	Scaling is performed by deploying the entire application as a whole, leading to inefficient resource utilization.
Change complexity	Modifications to a single functional module require rebuilding and redeploying the entire system.
High component coupling	Tight dependencies between modules complicate refactoring and increase the risk of side effects.

Limitation	Description
Limited fault tolerance	Failure of one component can disrupt the operation of the entire system.
Integration rigidity	Integrating new external services and platforms requires significant architectural changes.

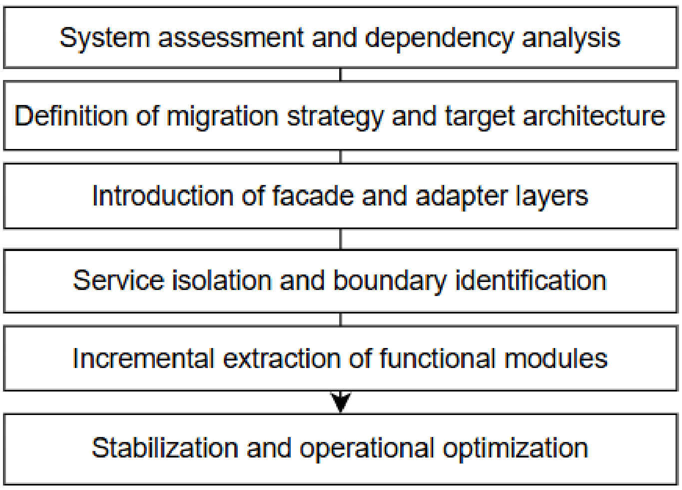
Under conditions that require rapid responses to the demands of new suppliers, partners, or regulatory bodies, monolithic systems can become a barrier to the flexible adaptation of procurement and logistics processes. This limitation is particularly evident in industries experiencing rapid digital development, such as intelligent logistics, procurement automation, and the application of artificial intelligence for supply audit and control. These innovations require the ability to introduce new modules without compromising the stability of the entire system (Miloserdov A., 2025).

In addition to technological constraints, monolithic solutions impose substantial managerial and economic overhead, as any modification requires full-system testing and risky rollbacks affecting multiple functional areas. Centralized architectures also limit the adoption of distributed teams, DevOps practices, and incremental delivery, thereby motivating a strategic shift toward modular and isolated architectures that support flexibility and effective governance in the digital transformation of supply chains.

Architectural Migration Strategies in Corporate Procurement Systems: Principles of Decomposition and Transition to Modular Architectures

Migration from monolithic to modular architecture in corporate procurement platforms represents a complex engineering and organizational process that requires a clearly defined strategy and a detailed assessment of the existing system landscape. The primary objective is to ensure a controlled and incremental transition that preserves functional continuity and avoids disruption of mission-critical business processes. In this context, migration is best understood as a sequence of interrelated stages, each addressing specific architectural, technological, and organizational challenges (fig. 2).

Figure 2. *Stages of Migration from Monolithic to Modular Architecture in Corporate Procurement Platforms*



The process of migration starts with a thorough evaluation of the current system, including its dependencies such as functional modules, data flow, integration links, and critical components affecting the stability of procurement processes, followed by the determination of a target architecture and a migration strategy, including decomposition priorities, acceptable risks, and requirements for system resiliency, as well as organizational constraints. The process then continues with the introduction of a facade and an adapter layer, enabling the decoupling of internal system structure from external consumers while ensuring interoperability between new and legacy system components, followed by the isolation of functional domains into logically independent units with defined interaction contracts. The last step in the process involves the incremental extraction of selected modules as independent services with minimal impact on operational processes, including measures for system stabilization and optimization.

The selection of a migration strategy depends on factors such as maturity level of the existing architecture, availability of resources, degree of process formalization, and level of risk. It is also essential to consider the architectural scalability of the target solutions. As noted in recent studies, the design of scalable distributed systems requires a balanced approach that integrates modularity, observability, and maintainability (Berezhnoy A., 2025).

Thus, successful migration requires not only technical implementation but also strategic architectural governance. Engineering teams must make decisions based on architectural patterns, business priorities, and operational constraints, which positions migration processes as an object of scientific analysis and project-oriented modeling within the broader context of digital transformation of corporate IT landscapes.

Methods for Ensuring Business Process Continuity during Architectural Migration of Corporate Procurement Platforms

The transition from monolithic architecture to modular solutions in corporate procurement systems is not feasible without ensuring the continuity of mission-critical business processes. Since such platforms typically support operations with strict temporal and logistical dependencies – from order intake and allocation to warehouse management and shipment – any disruption may lead to significant operational and financial consequences. This necessitates the application of architectural and organizational–technical methods that enable phased migration within coordinated scenarios while maintaining the stable operation of all involved systems (table 2).

Table 2. *Architectural and Organizational–Technical Methods for Ensuring Business Process Continuity during Migration from Monolithic Systems (Pittu R., 2025; Ogunwole O. et al., 2023)*

Method	Technical description
Strangler pattern strategy	Gradual encapsulation of monolithic functionality by external services with progressive request redirection.
Facade and adapter layers	Abstraction of internal monolith interfaces from external consumers.
Parallel execution (dual run)	Simultaneous operation of legacy and new implementations of the same functional block.
Interface versioning	Support for multiple API and interaction contract versions.
Incremental deployment	Step-by-step rollout of new components with continuous metric monitoring.

The practical applicability of the methods presented in table 2 is illustrated by the case of the U.S.-based company 3M, which carried out a large-scale phased migration of corporate applications without disrupting mission-critical functions. According to AWS, **3M** migrated 2,200 applications over 24 months in 51 migration waves with minimal downtime (AWS, cited: 20.01.2026). This case is particularly relevant to the decomposition of monolithic procurement platforms, where tightly integrated ERP and supply chain components require gradual functional extraction and strict compatibility control. Fast and controlled cutover activities, such as the migration of hundreds of applications within hours and critical ERP workloads within less than 20 hours, are also a reflection of the success of incremental deployment, blast radius reduction, and central monitoring.

It is important to note that the continuity of the process is largely dependent on the maturity of the monitoring, tracing, and alerting systems. The use of centralized monitoring platforms allows for the monitoring of the newly introduced components. This is particularly critical in procurement systems, where a high level of availability and execution accuracy is required. Recent studies emphasize the importance of implementing intelligent automation and predictive control mechanisms in such environments, especially through the use of artificial intelligence for assessing operational efficiency and analyzing incidents (Korostin O. et al., 2025).

Thus, methods for ensuring business process continuity during architectural migration represent a synthesis of technical solutions, quality control processes, and governance mechanisms. Their appropriate combination enables organizations to minimize operational risks and achieve sustainable digital transformation without disrupting the core functions of corporate platforms.

Management of Architectural Complexity in Distributed Corporate Procurement Platforms

The decomposition of monolithic solutions in corporate procurement platforms shifts architectural complexity from a single application to interacting services and infrastructure layers. Unlike centrally controlled monoliths, modular architectures introduce distributed complexity related to dependencies, data consistency, interface evolution, and operational control, which in procurement platforms is further intensified by regulated processes, material flow constraints, and integration with external partners and governmental systems (table 3).

Table 3. *Technical Aspects of Architectural Complexity Management in Distributed Procurement Platforms (Rusum G. P. & Anasuri S., 2023; Saari T. et al., 2024)*

Type of complexity	Technical source
Component coupling	Shared business logic and data models.
Interface complexity	Growth in the number of APIs and interaction protocols.
Transactional complexity	Distributed operations and asynchronous communication.
Operational complexity	Network latency and partial failures.
Infrastructure complexity	Heterogeneous runtime and orchestration environments.

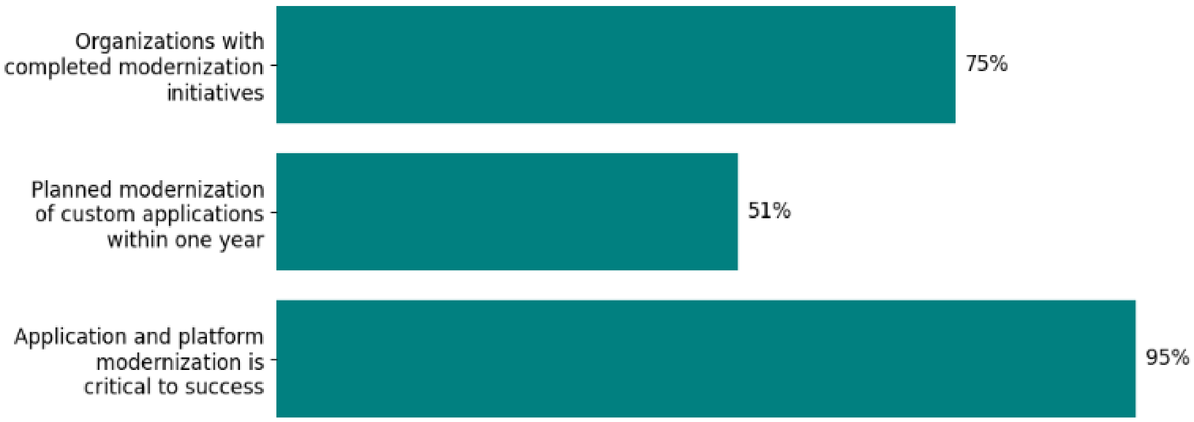
Architectural complexity is strongly influenced by operational and development lifecycle layers, including monitoring, tracing, quality control, and coordinated delivery practices. In distributed systems, increased risks of hard-to-reproduce failures caused by latency, asynchronous interactions, and partial outages make centralized observability and automated analysis essential for reliability. The adoption of DevOps and DevSecOps practices supports consistent architectural evolution and reduces cognitive load on engineering teams, while insufficient governance risks the emergence of a “distributed monolith,” highlighting the need for disciplined, system-level complexity management in corporate procurement platforms.

Analysis of Risks and Effects of Architectural Decomposition in Corporate Procurement Platforms

Architectural decomposition of monolithic information systems represents not only a technical but also a managerial challenge that requires a comprehensive assessment of potential benefits and risks. Among the most **important benefits** of decomposition are increased architectural flexibility, easier scaling of individual components, faster times-to-market for new functionalities, and faster integration with external services. It is also possible to leverage heterogeneous technological strategies such as polyglot persistence, event-driven interaction, and serverless components within a single platform through the decomposition of functional domains, thus speeding up innovation without requiring a redesign of the system. Decomposition also allows for system robustness, since the failure of a single service does not equate to the failure of the entire system – a factor that is very important in supply chains since system availability equates to business success and meeting contractual obligations.

The analytical studies of recent years show that decomposition and modernization of the application landscape have become a common practice, increasingly realized as a long-term transformation program rather than a project. According to Red Hat (2024), 95% of respondents consider the modernization of applications and corporate platforms critical for organizational success (fig. 3).

Figure 3. *Prevalence and Prioritization of Application and Corporate Platform Modernization Based on Red Hat Survey Data (Red Hat, cited: 22.01.2026)*



Infrastructure prerequisites for modular architectures are also strengthening: the CNCF Annual Survey 2024 reports that the adoption of cloud-native techniques has reached 89%, while 91% of organizations use containers in production environments (CNCF Annual Survey, cited: 22.01.2026). At the same time, the benefits of decomposition become evident only when supported by mature observability and change management practices. According to the Grafana Observability Survey 2024, 76% of respondents have centralized observability, and among them 79% report time or cost savings; however, only 26% use SLOs in production, indicating a persistent gap between tool-level visibility and formalized reliability management (Grafana, cited: 23.01.2026).

On the other hand, the process of decomposition is fraught with considerable risks. First, the architectural and operational complexity of the system increases. In the early stages of development, this can lead to increased maintenance costs, diagnostics, and the workload of engineering teams. Second, the process of decomposition is fraught with organizational risks. It often necessitates changes in the responsibility boundaries, interaction processes between development, operation, and business teams, as well as the implementation of a disciplined interface/data governance. Third, in the absence of sufficient maturity in DevOps culture, SLO practices, and observability tooling, distributed systems become more vulnerable to inter-service failures, data inconsistencies, and violations of transactional integrity – factors that are particularly critical in highly regulated industries, where incidents may result in regulatory non-compliance and financial penalties.

For example, one scientific study analyzes 107 incident reports collected over the period from June 2020 to February 2025 and correlates them with different stages of the transition from a monolithic architecture to microservices (Eldenk D. & Çetin H. A., 2025). Notably, a substantial share of the incidents had high user impact: 65 incidents were classified as high severity, 29 as medium severity, and 13 as low severity. In terms of recovery time, 57 incidents were resolved within 0–1 hour, 26 within 1–3 hours, 14 within 3–24 hours, and 8 incidents required more than

24 hours to restore service. The authors also grouped root causes into ten categories and demonstrated that, during decomposition, inter-service and "transitional" failure sources become particularly critical. In addition to database-related issues (30 incidents), a significant proportion was associated with refactoring or migration of business logic (23), over-fetching (23), infrastructure-related problems (21), and breaking changes (17), that is, compatibility violations between services and their clients or dependencies. These findings directly support the need for proactive planning of operational risks and strengthened control over the inter-service interaction layer.

Overall, evidence from academic studies and industry cases suggests that architectural transformations can improve change velocity and system controllability when supported by mature delivery and observability practices, typically through phased implementation and prior validation of key decisions. Accordingly, the assessment of risks and benefits of architectural decomposition should form an integral part of architectural strategy for the digital transformation of supply chains, incorporating scenario analysis, control metrics, and operational risk management.

Conclusion

The process of decomposition of monolithic IT systems in corporate procurement platforms is a multidimensional process that includes architectural, organizational, and managerial dimensions. The transition to a modular architecture is a process that increases the flexibility, scalability, and robustness of the system, allowing for the seamless integration of external services and the adoption of innovation. However, the process of transition to a modular architecture necessitates the application of strategic planning, incremental migration techniques, the management of the complexity of the architecture, and the analysis of technological and operational risks. A comprehensive approach that integrates engineering practices with managerial modeling allows organizations to carry out transformation without disrupting mission-critical business processes, thereby ensuring sustainable development in the context of the digital economy.

References

Saucedo, A. M., Rodriguez, G., Rocha, F. G. and dos Santos, R. P. (2025) Migration of monolithic systems to microservices: A systematic mapping study. *Information and Software Technology*, – 177. – 107590 p. URL: https://doi.org/10.1016/j.infsof.2024.107590

Katal, A., Prasanna, P. and Birla, R. (2025) Evolution from Monolithic to Microservices Architecture: A New Era in Software Architecture. In: *Advancements in Optimization and Nature-Inspired Computing for Solutions in Contemporary Engineering Challenges*. Singapore: Springer Nature Singapore, – P. 235–279.

Miloserdov, A. (2025) Intelligent automation of supply audit using machine learning in B2B SaaS platforms. *Journal of Economics, Finance And Management Studies*, – 8(12). – P. 7965–7970. URL: https://doi.org/10.47191/jefms/v8-i12-40

Berezhnoy, A. (2025) Architectural approaches to designing scalable information systems in e-commerce. *International Journal of Advanced Trends in Computer Science and Engineering*, – 14(6). – P. 222–226. URL: https://doi.org/10.30534/ijatcse/2025/011462025

Pittu, R. (2025) From Monoliths to Microservices: A Comprehensive Framework for Enterprise Cloud-Native Transformation. *Journal of Multidisciplinary*, – 5(7). – P. 436–441.

Ogunwole, O., Onukwulu, E. C., Joel, M. O., Adaga, E. M. and Ibeh, A. I. (2023) Modernizing legacy systems: A scalable approach to next-generation data architectures and seamless integration. *International Journal of Multidisciplinary Research and Growth Evaluation*, – 4(1). – P. 901–909. URL: https://doi.org/10.54660/ijmrge.2023.4.1.901-909

AWS (2026) Accelerating Migration at Scale Using AWS Application Migration Service with 3M Company. [cited: 20.01.2026]. Available from: https://aws.amazon.com/ru/solutions/case-studies/3m-accelerating-migration-case-study

Korostin, O., Blazhkovskii, A., Tretiakov, I. and Stepanov, M. (2025) Artificial intelligence as a key to improving the efficiency of logistics operations. *Transport Technician: Education and Practice*, – 6(2). – P. 188–195. URL: https://doi.org/10.46684/2687-1033.2025.2.188-195

Rusum, G. P. & Anasuri, S. (2023) Composable Enterprise Architecture: A New Paradigm for Modular Software Design. *International Journal of Emerging Research in Engineering and Technology*, – 4(1). – P. 99–111.

Saari, T., Savolainen, K., Tyllinen, M., Viitanen, J. and Nieminen, M. (2024) Addressing Usability and Complexity in Procurement of IT Systems. *Journal of User Experience*, – 20(1). – P. 27–45.

Red Hat (2026) The state of application modernization. [cited: 22.01.2026]. Available from: https://www.redhat.com/en/resources/app-modernization-report

CNCF Annual Survey (2024) Cloud Native 2024. [cited: 22.01.2026]. Available from: https://www.cncf.io/wp-content/uploads/2025/04/cncf_annual_survey24_031225a.pdf

Grafana (2024) Observability Survey. [cited: 23.01.2026]. Available from: https://grafana.com/media/observability-survey/Obs-Survey-24-final.pdf

Eldenk, D. & Çetin, H. A. (2025) Incidents During Microservice Decomposition: A Case Study. Proceedings of the 29th International Conference on Evaluation and Assessment in Software Engineering (EASE '25), – P. 805–808. URL: https://doi.org/10.1145/3756681.3757005

submitted 14.04.2026;
accepted for publication 28.04.2026;
published 30.04.2026

Contact: n.roylyan@outlook.com