Review Article *Open Access*

Journal of Pioneering Artificial Intelligence Research

ISSN: 3069-0846

DOI: *doi.org/10.63721/26JPAIR0134*

Towards Autonomous Zero-Downtime Migration: AI-Driven Patterns of Safe Monolith Decomposition

Thomas Paul

Independent Researcher, USA

Citation: *Thomas Paul (2026) Towards Autonomous Zero-Downtime Migration: AI-Driven Patterns of Safe Monolith Decomposition. J of Poin Artf Research 2(3), 1-7. WMJ-JPAIR-134*

Abstract

Autonomous or AI-assisted modernization of legacy monoliths is gaining attention as organizations seek to migrate to microservices without interrupting mission-critical services. However, decomposing a monolith safely remains difficult because the process introduces architectural risk, dependency uncertainty, data-consistency challenges, and operational failure modes that can directly affect availability. This paper presents a multivocal literature review of AI-driven approaches that support safe monolith decomposition and enable zero or near-zero downtime migration. Across the synthesized evidence, recurring patterns include runtime-informed partition discovery with explainable clustering, AI-guided dependency analysis to expose coupling risks, quality-driven decomposition approaches that treat migration as an optimization problem, AI-driven partitioning frameworks for service boundary design, and integrated toolchains that combine static and dynamic signals to recommend candidate microservices. Emerging directions further include large language model representations and contrastive learning to improve boundary quality, generative AI–driven code transformation for modernization, and systematic evidence mapping of AI methods and research gaps. Findings are presented as design propositions rather than universal claims, reflecting heterogeneity in evaluation settings and limited end-to-end migration validation under real operational constraints. The review contributes a structured pattern catalog linking AI capabilities (discovery, ranking, validation support) to safety goals (explainability, incremental rollout, rollback readiness), and it identifies future work needed to operationalize autonomy for continuous migration [1-7].

***Corresponding author**: Thomas Paul, Independent Researcher, USA.

Submitted: 27.04.2026 **Accepted**: 01.05.2026 **Published**: 12.05.2026

Keywords: Autonomous Migration, Zero-Downtime Migration, Monolith Decomposition, Microservices, Artificial Intelligence, Large Language Models, Software Modernization

Introduction

Modern enterprises increasingly modernize legacy monolithic systems into microservices to improve agility, scalability, and operational resilience. Yet, migration introduces significant risk: service boundary errors, hidden dependencies, inconsistent data flows, and coordination issues can cause regressions and downtime if not managed carefully. While traditional migration relies heavily on expert-driven analysis, recent work explores whether AI can reduce migration risk and effort by automating boundary discovery, producing ranked refactoring plans, and supporting validation.

AI-assisted decomposition is motivated by the observation that decomposition is widely considered a core obstacle to migration success and that practical execution requires decision support grounded in program behavior and dependencies. Toolchains such as Mono2Micro collect both static and runtime information and use AI methods to recommend partitions intended to be explainable and actionable for engineers. Dependency-oriented approaches such as CARGO propose AI-guided analysis to reveal dependency structure that drives risky couplings across components. Other work frames decomposition as a quality-driven process with explicit trade-offs between cohesion, coupling, and migration feasibility, while partitioning frameworks aim to provide systematic boundary recommendations. A mapping study synthesizes the AI landscape for migration and indicates clustering as a frequent technique used to generate candidate boundaries from software artifacts [1-5].

This paper investigates the emerging topic of **autonomous zero-downtime migration**, defined as migration where AI performs a meaningful portion of boundary discovery, planning, and validation support, while a human remains accountable for adoption and rollout decisions. The goal is not to claim push-button modernization, but to extract patterns that make migration safer, more explainable, and more reversible.

Methodology

This work conducts a **multivocal literature review (MLR)** to synthesize both peer-reviewed ("white") and high-relevance non-traditional ("grey") sources on AI-driven monolith decomposition and its implications for zero/near-zero downtime migration. The methodology follows a structured workflow consisting of: (1) goals and research questions (Section 2.1), (2) search strategy and study selection (Section 2.2), (3) data extraction and synthesis (Section 2.3), and (4) replicability measures (Section 2.4). The review treats results as design propositions where end-to-end validations are limited.

Goals and Research Questions

The goals of this study are three-fold. First, it identifies AI-based techniques used to recommend microservice boundaries from monoliths. Second, it examines safety-oriented mechanisms that support operational adoption (explainability, risk scoring, staged rollout support). Third, it evaluates the extent to which the literature connects AI-based decomposition to continuous availability requirements.

The study addresses the following research questions (RQs):

- **RQ1:** What AI-driven techniques are used to identify candidate microservice boundaries from monoliths?
- **RQ2:** What mechanisms are proposed to make AI-assisted decomposition safe and operationally usable?
- **RQ3:** What evidence exists regarding effectiveness and what gaps remain for autonomous zero-downtime execution?

Search Strategy and Study Selection

Search String

The search string combines (i) monolith-to-microservices migration terms, (ii) AI/automation terms, and (iii) safety/availability terms. The migration-term group uses wildcard stems to capture common variants (e.g., migration/migrating; modernization/modernise variants), consistent with systematic review practice.

Search String

(microservice* OR micro-service* OR "micro service*") AND (monolith* OR "legacy system" OR "monolithic application" OR "monolithic architecture") AND (migrat* OR moderni* OR refactor* OR rearchitect* OR decompos* OR transform*) AND (AI OR "artificial intelligence" OR "machine learning" OR "deep learning" OR "large language model" OR "language model" OR

"generative AI" OR "graph neural network" OR clustering OR "representation learning") AND ("zero downtime" OR "near-zero downtime" OR "continuous availability" OR "live migration" OR rollout OR rollback OR "risk-aware" OR "safe migration")

White Literature

Sources. Peer-reviewed literature was searched in IEEE Xplore (https://ieeexplore.ieee.org/), ACM Digital Library (https://dl.acm.org/), and SpringerLink (https://link.springer.com/) (all accessed on 21 February 2026). Scopus (https://www.scopus.com/) and Web of Science (https://www.webofscience.com/) were additionally consulted as citation indexes to validate coverage; however, no relevant records were retrieved from these platforms under the defined query and screening criteria.

- Scopus
- IEEE Xplore
- ACM Digital Library
- SpringerLink
- Web of Science

Inclusion and Exclusion Criteria

Inclusion: studies proposing or evaluating AI-assisted decomposition/refactoring for monolith-to-microservices migration, with technical detail sufficient for extraction (inputs, method, outputs, evaluation).

Exclusion: out-of-scope papers; duplicates; inaccessible full text; opinion-only sources without method.

Search and Selection Process: Searches were executed using advanced search features, restricting to Title/Abstract/Keywords when supported. Results were consolidated into a single working set and duplicates were removed based on title, authors, venue, year, and DOI/URL. Screening was then performed by title/abstract, followed by full-text eligibility checks and snowballing.

Table 1: Initial Literature Search by Library (White Literature)

Library (White Literature)	Records
Scopus	0
IEEE Xplore	45
ACM Digital Library	10
SpringerLink	32
Web of Science	0
Total	87
Non-duplicates	83

Final selection (white vs grey): After full-text screening, five peer-reviewed studies were retained for synthesis, and three grey sources were retained as complementary evidence. (These correspond to the seven attached reference materials used for analysis [1-7].)

Grey Literature

Sources: The same search string was executed in Google Scholar (https://scholar.google.com/) and arXiv (https://arxiv.org/) (both accessed on 21 February 2026) to identify high-relevance technical reports, practitioner articles, and preprints. Broad forum threads and general web search results were not used as primary sources to reduce low-verifiability evidence.

Grey literature was included because AI-driven modernization advances quickly and some detailed technical methods are published first as preprints, practitioner reports, or tool-focused manuscripts. Grey sources were included only when authorship and provenance were identifiable and the document

sufficient technical depth (architecture description, method, evaluation or experimental plan). Sources lacking verifiable provenance were excluded.

Data Extraction and Synthesis

For each included study, the following data were extracted:

- **AI technique** (clustering, dependency reasoning, graph learning, language-model embeddings, generative transformation)
- **Inputs** (static graphs, runtime traces, code artifacts, dependency graphs, textual representations)
- **Outputs** (service partitions, ranked recommendations, refactoring plans, toolchain pipelines)
- **Safety mechanisms** (explainability, risk scoring, staged planning, validation hooks)
- **Evaluation evidence** (metrics, datasets, baselines, limitations)

Synthesis was conducted using narrative thematic analysis aligned to RQ1–RQ3. Where evidence was incomplete or not end-to-end validated, findings are framed as propositions rather than definitive conclusions.

Replicability

Replicability is supported by documenting the search strings, the selection criteria, screening workflow, and extracted categories. Future researchers can re-run the search using the same stem-based query groups and update the evidence base as new AI and migration studies appear.

Data Collection

Data for this review consisted of the final included corpus of seven studies. Peer-reviewed evidence includes toolchain and dependency-analysis approaches that provide explicit mechanisms for partitioning and boundary recommendation. Additional peer-reviewed or formally published research describes decomposition as a quality-driven optimization or framework-based partitioning approach. Grey evidence includes emerging LLM-driven approaches and generative AI modernization proposals, as well as systematic mapping of AI migration research. The combined evidence base enables synthesis across boundary discovery, plan generation, and safety-related constraints [1-4, 6-7].

Results

The synthesized results are organized into two thematic clusters: (i) AI-driven boundary discovery and decomposition, and (ii) safety mechanisms required to operationalize AI-assisted migration under zero/near-zero downtime constraints.

AI-Driven Boundary Discovery and Decomposition

Runtime- and Trace-Informed Partition Discovery

A recurring pattern is the use of runtime traces and behavior-derived signals to inform service boundaries. Mono2Micro exemplifies this approach by combining static information with runtime traces and applying AI techniques to generate partition recommendations intended to be explainable and aligned with observed execution behavior [1]. This pattern is valuable for safe migration because it reduces reliance on purely static coupling assumptions and better reflects operational workloads.

AI-Guided Dependency Analysis for Risk Visibility

Another recurring result is AI-guided dependency analysis to expose hidden coupling and quantify cross-component dependency risks that can undermine decomposition. CARGO proposes AI-guided dependency analysis to support migration decisions by identifying dependency structure relevant to extraction planning [2]. This improves safety because unsafe cuts often arise from unrecognized shared dependencies and implicit transactional coupling.

Quality-Driven and Framework-Based Partitioning

Several approaches treat decomposition as an optimization or quality-driven design activity. A quality-driven decomposition tool frames boundary decisions using software quality objectives, typically trading off cohesion, coupling, and refactoring effort. Partitioning frameworks propose structured steps to recommend service splits and guide decomposition planning. These approaches are important in autonomous migration because they provide explicit decision criteria rather than purely heuristic grouping [3-4].

Representation Learning and Language-Model-Assisted Decomposition

Emerging approaches use large language models to generate representations of monolith components and cluster them into candidate microservices. Contrastive-learning-enhanced LLM decomposition suggests that

fine-tuning can improve separation and cohesion signals for boundary discovery [6]. These methods broaden the evidence base beyond dependency graphs by incorporating semantic information from code and text-like artifacts, but most remain limited in real-world operational validation.

Generative AI for Transformation and Modernization

Another emerging stream focuses on generative AI to assist transformation of legacy code into modern forms. Generative AI–driven modernization work proposes code transformation pipelines that can reduce manual effort, but it often reports limited end-to-end validation in real migration programs [7]. For autonomous zero-downtime migration, this suggests a future path where AI supports not only partition discovery but also implementation acceleration—provided safety gates and validation are integrated.

Macro-Level Evidence from Secondary Research

A systematic mapping study summarizes the AI landscape for migration, indicating clustering and code-derived signals as common foundations for decomposition research [5]. This supports the observation that most AI migration research currently focuses on boundary discovery rather than fully autonomous execution across production release phases.

Safety Mechanisms for Zero-/Near-Zero Downtime Execution

Explainability and Human-in-the-Loop Review

Across the evidence base, AI recommendations require interpretability for adoption. Toolchains that provide rationales and visible artifacts (e.g., trace-derived graphs and explainable partitions) better support operational safety because engineers can validate and adjust recommendations [1-2].

Risk Scoring and Ranked Migration Planning

A repeated theme is producing ranked options and risk-aware plans rather than single deterministic decompositions. Dependency-analysis approaches help prioritize extraction sequencing by highlighting coupling and potential breakpoints. Quality-driven approaches frame boundaries with explicit trade-offs and can support staged extraction planning [2-3].

Linking AI Recommendations to Incremental Rollout and Rollback Readiness

Although few studies provide full "zero-downtime" execution pipelines, multiple approaches implicitly support it by enabling incremental extraction: stable boundary recommendations, prioritized sequencing, and identification of high-risk couplings that should be handled with conservative rollout and rollback planning [1-3]. This indicates that AI currently functions best as a decision-support layer feeding established progressive delivery practices.

Evidence Strength and Reporting Limits

The final evidence base remains limited in the number of end-to-end validations that measure operational outcomes such as downtime avoided, rollback speed, or production incident reduction. Therefore, the findings should be interpreted as design propositions: AI can meaningfully improve boundary discovery and planning, but autonomous zero-downtime migration requires stronger empirical validation in production-like environments.

Discussion

The results suggest that AI can support safer migration primarily by reducing uncertainty in boundary discovery and by increasing visibility into dependencies and risk. Runtime-informed partitioning approaches align boundaries with observed behavior and can reduce the likelihood of "incorrect cuts" that cause cascading failures during extraction. Dependency-driven approaches expose hidden couplings that often cause transactional or consistency risks, enabling staged migration planning and conservative rollout sequencing. Quality-driven and framework-based methods contribute explicit criteria that can be turned into policy gates for autonomy [1-4].

However, the evidence also indicates that most AI work remains focused on the decomposition decision itself (where to cut) rather than full operationalization (how to migrate live with continuous validation and automated rollback). LLM-based decomposition and generative AI transformation approaches are promising, but current reporting often lacks standardized benchmarks and end-to-end validation under real workload and failure conditions. Secondary synthesis highlights the same gap: research concentrates on boundary identification more than controlled execution and governance [5-7].

Therefore, autonomy should be framed as progressive automation: AI proposes and ranks candidate boundaries and plans, while humans validate and enforce safety requirements. The most actionable near-term contribution is combining AI outputs with established reliability controls (progressive rollout, monitoring, rollback readiness), and the most important research gap is building shared datasets and evaluation baselines that connect decomposition quality to operational outcomes.

Threats to Validity

Study selection validity: Terminology varies (modernization, refactoring, migration, decomposition), so relevant work may be missed despite wildcard stems. **Data validity:** Many studies evaluate partition quality via structural metrics or synthetic datasets, which may not fully represent operational constraints. **Research validity:** Limited end-to-end validations constrain generalization; conclusions are presented as propositions rather than universal findings. **Grey literature bias:** Grey sources may contain promising methods without peer-reviewed validation; they are included only when provenance and technical depth are sufficient.

Related Work

Toolchains that combine static and dynamic signals for boundary recommendation provide strong evidence of AI-assisted decomposition feasibility. Dependency analysis for migration planning contributes risk visibility and coupling insights that support safer staged extraction. Quality-driven decomposition and structured partitioning frameworks contribute formal decision logic and optimization framing for boundary design. Emerging approaches extend the field with LLM-based representations and generative AI for code transformation, though validation is still maturing. A systematic mapping study provides a macro-level view of AI techniques and gaps in migration research [1-7].

Conclusions

This multivocal literature review synthesized evidence on AI-driven patterns for safe monolith decomposition as a foundation for autonomous or semi-autonomous zero/near-zero downtime migration. Across the included studies, recurring patterns include runtime-informed and explainable partitioning, AI-guided dependency analysis for coupling and risk visibility, quality-driven decomposition and structured partitioning methods, and emerging representation-learning and generative transformation approaches that may reduce manual refactoring effort. The synthesis indicates that AI is currently most mature as a decision-support mechanism—helping engineers discover boundaries, rank options, and identify risks—rather than as an end-to-end autonomous migration executor [1-4, 6-7].

Because the evidence base contains limited end-to-end operational validation, the findings are expressed as design propositions. The most practical path to autonomy is combining AI-generated boundary and plan recommendations with established reliability practices such as progressive rollout, continuous observability, and rollback readiness. Future work should focus on shared datasets, standardized benchmarks, and evaluation designs that measure migration outcomes directly (availability continuity, rollback speed, incident reduction, and correctness under failure). Such advances are required before autonomy can reliably support true zero-downtime modernization programs.

References

1. Kalia AK, Xiao J, Lin C, Sinha S, Rofrano J, et al. (2021) Mono2Micro: An AI-Based Toolchain for Evolving Monolithic Enterprise Applications to a Microservice Architecture 11.
2. Nitin V, Asthana S, Ray B, Krishna R (2022) CARGO: AI-Guided Dependency Analysis for Migrating Monolithic Applications to Microservices Architecture https://arxiv.org/abs/2207.11784.
3. Muhammad Hafiz Hasan, Mohd Hafeez Osman, Novia Indriaty Admodisastro, Muhamad Sufri Muhammad (2023) AI-based Quality-driven Decomposition Tool for Monolith to Microservice Migration 181-191.
4. Ramamoorthi V (2023) AI-Driven Partitioning Framework for Migrating Monolithic Applications to Microservices 8: 63-72.
5. Martínez Saucedo A, Rodríguez G (2024) Migration of Monolithic Systems to Microservices using AI: A Systematic Mapping Study https://www.researchgate.net/publication/381055498_Migration_of_Monolithic_Systems_to_Microservices_using_AI_A_Systematic_

Mapping_Study.

6. Sellami K, Saied MA (2025) Contrastive Learning-Enhanced Large Language Models for Monolith-to-Microservice Decomposition https://arxiv.org/abs/2502.04604.
7. Sriram Ghanta (2024) From Monoliths to Intelligence: Generative AI-Driven Code Transformation and Modernization of Legacy Systems 11: 1306-1316