International Journal of Computational Research in Science and Technology
E-ISSN: 3108-0871, Volume 2 Issue 2
Mar-Apr 2026, PP 01-13

Transforming Platform Engineering with Large Language Models: Evidence Mapping and Future Perspectives

Michael Reed[1], Abigail Foster[2], Victoria Stewart[3], Chaitanya Srinivas[4], Rishi Kumar[5]

[1]Professor of Platform Operations and Site Reliability Engineering
[2]Associate Professor of Artificial Intelligence and Cloud Service Engineering
[3]Professor of Intelligent Software Systems and AI-Augmented Platform Management
[4]Senior Java Software Developer, [5]Database Administrator

Abstract. Platform engineering has emerged as a critical discipline for managing the growing complexity of modern cloud-native ecosystems, enabling organizations to standardize infrastructure, streamline software delivery, and improve operational efficiency. The rapid advancement of Large Language Models (LLMs) has introduced new opportunities for transforming platform engineering through intelligent automation, natural language–driven interactions, automated decision support, and enhanced operational intelligence. This study presents an evidence mapping analysis of the application of Large Language Models in platform engineering, with the objective of identifying current research trends, implementation approaches, technological capabilities, benefits, challenges, and future development directions. The study systematically examines existing literature across key domains including infrastructure automation, DevOps, Site Reliability Engineering (SRE), AIOps, cloud operations, incident management, observability, and software delivery pipelines. The findings indicate that LLM-driven solutions significantly enhance platform engineering by automating routine operational tasks, accelerating troubleshooting processes, improving knowledge management, supporting intelligent workflow orchestration, and enabling context-aware operational decision-making. Furthermore, the integration of generative artificial intelligence with cloud-native platforms, infrastructure-as-code frameworks, and autonomous operations is creating new possibilities for self-managing and adaptive digital environments. The evidence mapping also identifies important research challenges related to model reliability, explainability, security, governance, privacy, hallucination mitigation, and responsible AI adoption in enterprise operations. The study concludes that Large Language Models represent a transformative force in the evolution of platform engineering and are expected to play a central role in the development of intelligent, autonomous, and highly resilient operational ecosystems, providing significant opportunities for future research and innovation.

Keywords: Platform Engineering, Intelligent Platform Engineering, Large Language Models (LLMs), Generative Artificial Intelligence, Generative AI, Artificial Intelligence, Machine Learning, Deep Learning, Foundation Models, Transformer Models, AI-Driven Automation, LLM-Driven Automation, Operational Automation, Intelligent Operations, Autonomous Operations, Autonomous Systems, AI-Augmented Engineering, Platform Automation, Enterprise Platform Engineering, Cloud-Native Platforms, Cloud Computing, Cloud-Native Architecture, Distributed Systems, Enterprise Infrastructure, Infrastructure Engineering, Infrastructure Automation, Infrastructure as Code (IaC), Cloud Infrastructure Management, Digital Infrastructure, Platform Operations, Site Reliability Engineering (SRE), Reliability Engineering, DevOps, DevSecOps, GitOps, Continuous Integration, Continuous Delivery, Continuous Deployment, CI/CD Pipelines, Software Delivery Automation, Software Engineering, Software Development Lifecycle, Cloud Operations, IT Operations Management, Intelligent IT Operations, AIOps, IT Automation, Intelligent Service Management, Operational Intelligence, Predictive Analytics, Predictive Operations, Intelligent Decision Support, Context-Aware Automation, Knowledge Management, Enterprise Knowledge Systems, AI Assistants, Conversational AI, Natural Language Processing (NLP), Natural Language Understanding, Prompt Engineering, AI Agents, Autonomous Agents, Agentic AI, Multi-Agent Systems, Workflow Automation, Intelligent Workflow Orchestration, Process Automation, Business Process Automation, Cloud Orchestration, Infrastructure Orchestration, Resource Management, Dynamic Resource Allocation, Cloud Resource Optimization, Observability, Digital Observability, Monitoring and Logging, Distributed Tracing, Telemetry Analytics, Incident Management, Incident Response Automation, Root Cause Analysis, Automated Troubleshooting, Fault Detection, Failure Prediction, Anomaly Detection, Self-Healing Systems, Self-Managing Systems, Autonomous Infrastructure, Intelligent Infrastructure Management, Cloud Reliability, Platform Reliability, Service Reliability, Operational Resilience, Infrastructure Resilience, Enterprise Resilience, High Availability, Scalability Engineering, Performance Optimization, Capacity Planning, Cloud Governance, AI Governance, Responsible AI, Explainable AI (XAI), Trustworthy AI, Ethical AI, AI Security, Security Automation, Cybersecurity Operations, AI Risk Management, Privacy Preservation, Compliance Automation, Human-AI Collaboration, Intelligent Decision Making, Enterprise Digital Transformation, Adaptive Systems, Future of Platform Engineering, AI-Powered DevOps, AI-Driven SRE, Intelligent Cloud Platforms, Cloud-Native Transformation, Enterprise Automation, Emerging Technologies, Technology Innovation, Research Synthesis, Evidence Mapping, Systematic Evidence Mapping, Future Perspectives, Next-Generation Platform Engineering, Intelligent Software Delivery, Autonomous Cloud Operations, LLM-Based Operations Management, AI-Enhanced Productivity, Digital Operations Transformation, Platform Ecosystems, Modern Infrastructure Management, Intelligent Engineering Platforms.

I. Introduction

Platform engineering has emerged as a foundational discipline for modern software development and cloud operations, providing standardized tools, processes, and infrastructure that enable development teams to build, deploy, and manage applications efficiently. As organizations increasingly adopt cloud-native architectures, microservices, DevOps practices, and distributed computing environments, the complexity of managing operational workflows, infrastructure resources, deployment pipelines, and service reliability has grown substantially. Traditional platform management approaches often require significant manual intervention, creating operational bottlenecks and increasing the risk of human error. Consequently, organizations are seeking intelligent solutions that can automate routine operational tasks, improve decision-making, and enhance platform resilience.

The rapid advancement of Large Language Models (LLMs) has introduced transformative opportunities for platform engineering. LLMs possess advanced natural language understanding, contextual reasoning, code generation, knowledge synthesis, and conversational capabilities that can be integrated into platform operations. These capabilities enable automation of infrastructure management, incident response, troubleshooting, documentation generation, workflow orchestration, and operational decision support. By combining generative artificial intelligence with cloud-native platforms, organizations can develop intelligent operational ecosystems capable of supporting autonomous and adaptive infrastructure management.

This evidence mapping study investigates the role of Large Language Models in transforming platform engineering practices. The study examines existing research, identifies key application areas, analyzes technological advancements, evaluates benefits and challenges, and explores future directions for intelligent platform engineering. The findings contribute to a comprehensive understanding of how LLM-driven automation is reshaping modern platform operations and creating new opportunities for innovation in cloud computing and software engineering.

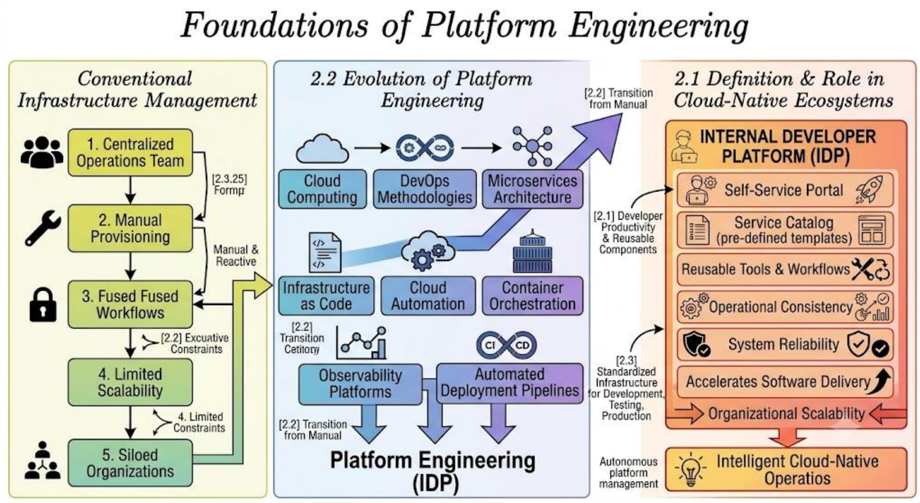
II. Foundations of Platform Engineering

Definition of Platform Engineering
Platform engineering refers to the design, development, and management of internal developer platforms that provide reusable tools, services, infrastructure, and workflows for software development teams. The primary objective is to improve developer productivity, operational consistency, and system reliability.
Platform engineering promotes self-service capabilities that allow development teams to deploy and manage applications without extensive involvement from infrastructure specialists. This approach accelerates software delivery and supports organizational scalability.

Evolution of Platform Engineering
The evolution of platform engineering is closely linked to the growth of cloud computing, DevOps methodologies, and microservices architectures. Early infrastructure management relied heavily on manual processes and centralized operations teams.
Modern platform engineering incorporates Infrastructure as Code, cloud automation, container orchestration, observability platforms, and automated deployment pipelines. These advancements have established the foundation for intelligent and autonomous platform management.

Role in Cloud-Native Ecosystems
Cloud-native environments require highly automated and scalable operational models. Platform engineering provides standardized infrastructure services that enable organizations to maintain consistency across development, testing, and production environments.

The integration of intelligent automation technologies further enhances the ability of platform engineering teams to manage complex cloud-native ecosystems efficiently.



III. Large Language Models and Intelligent Automation

Overview of Large Language Models

Large Language Models are advanced artificial intelligence systems trained on vast collections of textual and programming data. These models can understand natural language, generate human-like responses, write code, summarize information, and assist with complex reasoning tasks.

Recent developments in generative AI have significantly improved the accuracy and applicability of LLMs in enterprise technology environments.

Key Capabilities of LLMs

LLMs provide several capabilities relevant to platform engineering, including natural language interaction, automated documentation, code generation, operational analytics, knowledge retrieval, and intelligent recommendation systems.
These capabilities enable organizations to automate complex workflows and improve operational efficiency through AI-assisted decision-making.

LLMs as Operational Assistants

In platform engineering environments, LLMs function as intelligent operational assistants capable of answering technical queries, generating configuration files, interpreting logs, troubleshooting incidents, and recommending remediation actions.
Such capabilities reduce operational workload and accelerate problem resolution processes.

IV. LLM-Driven Platform Engineering

Intelligent Platform Operations

LLM-driven platform operations leverage artificial intelligence to automate routine administrative tasks and support operational decision-making. These systems continuously analyze infrastructure data and provide actionable insights to platform teams.

Intelligent operations improve productivity while reducing the complexity associated with managing large-scale cloud environments.

Natural Language Infrastructure Management

One of the most significant innovations introduced by LLMs is the ability to manage infrastructure through natural language commands. Engineers can interact with systems conversationally to retrieve information, generate deployment configurations, and automate operational workflows.

This approach simplifies infrastructure management and improves accessibility for users with varying technical expertise.

AI-Augmented Developer Platforms

Modern developer platforms increasingly incorporate AI-powered assistants that provide contextual guidance, automate repetitive tasks, and enhance software development workflows.

These platforms enable developers to focus on innovation while reducing time spent on operational and administrative activities.

V. Infrastructure Automation with Large Language Models

Infrastructure as Code Generation

LLMs can generate Infrastructure as Code templates for cloud resources, networking configurations, security policies, and deployment environments. Automated code generation improves consistency and reduces manual configuration errors.
This capability accelerates infrastructure provisioning and simplifies cloud resource management.

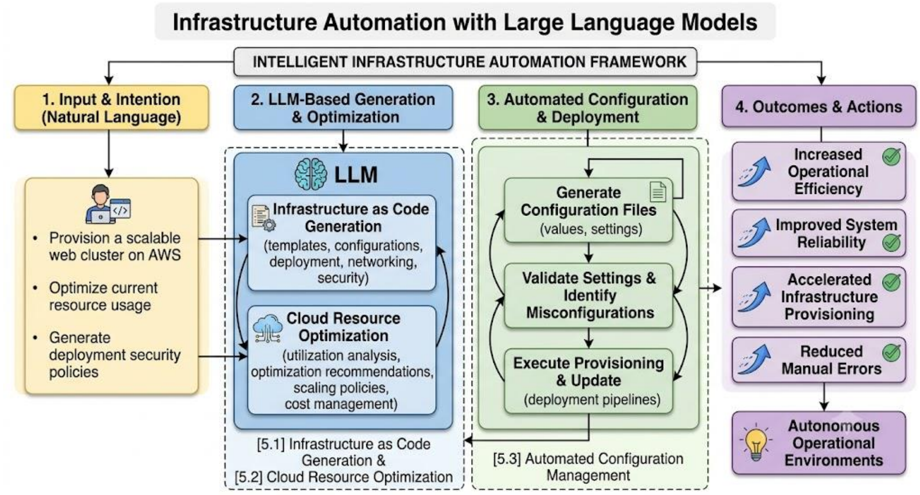
Cloud Resource Optimization

Intelligent automation systems analyze infrastructure utilization patterns and recommend optimization strategies for resource allocation, scaling policies, and cost management.

Resource optimization improves operational efficiency and supports sustainable cloud adoption.

Automated Configuration Management

Configuration management is essential for maintaining infrastructure consistency across environments. LLMs assist by generating configuration files, validating deployment settings, and identifying potential misconfigurations.
These capabilities contribute to improved system reliability and operational stability.



VI. DevOps and Site Reliability Engineering Transformation

LLMs in DevOps Workflows

DevOps practices emphasize collaboration, automation, and continuous delivery. LLMs enhance DevOps workflows by automating documentation, generating deployment scripts, analyzing build failures, and assisting with pipeline management.
These improvements streamline software delivery processes and reduce operational complexity.

Intelligent Incident Management

Incident management is a critical component of Site Reliability Engineering. LLMs analyze operational data, correlate events, identify probable root causes, and recommend remediation actions.

This capability accelerates incident resolution and improves service reliability.

Automated Root Cause Analysis

Root cause analysis traditionally requires extensive manual investigation. LLM-driven analytics systems process logs, metrics, and event data to identify underlying causes of operational issues.

Automated analysis reduces troubleshooting time and enhances operational resilience.

VII. AIOps and Operational Intelligence

Integration with AIOps Platforms

Artificial Intelligence for IT Operations combines machine learning, analytics, and automation technologies to improve infrastructure management. LLMs enhance AIOps platforms by enabling contextual understanding and natural language interaction.
This integration improves the effectiveness of operational analytics and automation frameworks.

Predictive Operations
Predictive operations utilize historical and real-time data to forecast infrastructure issues before they occur. LLM-powered systems contribute by interpreting operational patterns and supporting proactive decision-making.

Predictive capabilities help organizations reduce downtime and improve service continuity.

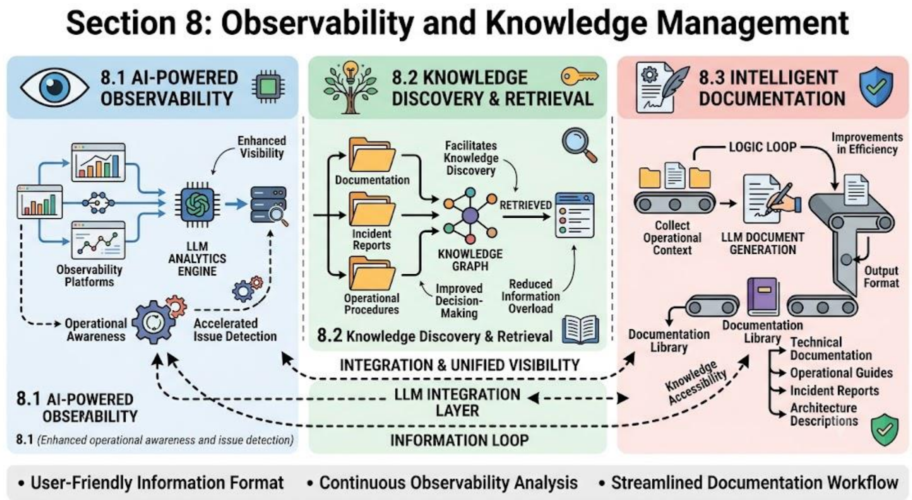
Intelligent Workflow Orchestration
Workflow orchestration coordinates operational activities across complex infrastructure environments. LLMs facilitate orchestration by automating task execution, generating workflow logic, and adapting processes based on changing operational conditions. These capabilities support autonomous operational ecosystems.

VIII. Observability and Knowledge Management

AI-Powered Observability
Observability platforms collect metrics, logs, traces, and events to provide visibility into system performance. LLMs analyze this data and generate meaningful insights that assist engineers in understanding infrastructure behavior.
Enhanced observability improves operational awareness and accelerates issue detection.

Knowledge Discovery and Retrieval
Platform teams often manage large volumes of documentation, incident reports, and operational procedures. LLMs facilitate knowledge discovery by retrieving relevant information and presenting it in a user-friendly format.
Knowledge retrieval systems improve decision-making and reduce information overload.

Intelligent Documentation



Documentation is essential for maintaining operational consistency and knowledge sharing. LLMs automatically generate technical documentation, operational guides, incident reports, and architecture descriptions.
Automated documentation improves efficiency and knowledge accessibility.

IX. Challenges and Limitations

Reliability and Accuracy
Although LLMs offer substantial benefits, they may occasionally generate inaccurate or misleading information. Ensuring reliability is essential when deploying AI systems in operational environments.
Organizations must implement validation mechanisms to maintain trust and operational safety.

Security and Privacy Concerns
The integration of LLMs into enterprise platforms raises concerns regarding data privacy, access control, and information security. Sensitive operational data must be protected through robust governance frameworks.
Security considerations remain a key challenge for AI adoption.

Explainability and Governance
Enterprise environments require transparent and explainable decision-making processes. Understanding how LLMs generate recommendations is essential for ensuring accountability and compliance.

Effective governance frameworks are necessary to support responsible AI deployment.

X.Future Perspectives of Intelligent Platform Engineering

Autonomous Platform Operations
Future platform engineering ecosystems are expected to incorporate autonomous operational capabilities that require minimal human intervention. AI systems will continuously monitor, optimize, and manage infrastructure resources independently.
This evolution will improve scalability and operational efficiency.

Agentic AI for Platform Engineering
Agentic AI systems capable of autonomous reasoning and task execution are emerging as a significant advancement in intelligent operations. These systems can coordinate multiple workflows and make context-aware decisions.

Agentic architectures represent the next stage in platform engineering automation.

Human-AI Collaborative Engineering
The future of platform engineering will likely involve close collaboration between human experts and intelligent AI systems. While AI will automate routine tasks, human engineers will provide strategic oversight, governance, and innovation.
This collaborative model will maximize the benefits of both human expertise and artificial intelligence capabilities.

XI. Conclusion

The rapid advancement of cloud-native technologies, distributed computing, and digital transformation initiatives has elevated platform engineering to a critical role within modern enterprise technology ecosystems. As organizations continue to manage increasingly complex infrastructures, software delivery pipelines, and operational environments, traditional approaches based on manual administration and reactive management are becoming insufficient. Large Language Models (LLMs) have emerged as a transformative technology capable of enhancing platform engineering through intelligent automation, natural language interaction, contextual reasoning, and operational decision support. Their ability to process vast amounts of information, generate actionable insights, and automate repetitive tasks has created new opportunities for improving efficiency, reliability, and scalability across platform operations.

This evidence mapping study systematically examined the application of Large Language Models in platform engineering and identified key areas where LLM-driven automation is reshaping operational practices. The analysis revealed that LLMs are increasingly being integrated into infrastructure management, DevOps workflows, Site Reliability Engineering (SRE), observability platforms, incident management systems, knowledge repositories, and cloud operations. These technologies enable organizations to automate infrastructure provisioning, generate Infrastructure as Code templates, accelerate troubleshooting, support root cause analysis, enhance documentation processes, and improve operational intelligence. As a result, platform teams can reduce operational overhead, increase productivity, and focus on higher-value engineering activities.

The study further highlighted the growing convergence of LLMs with AIOps, predictive analytics, autonomous operations, and cloud-native platform technologies. This convergence is driving the development of intelligent operational ecosystems capable of proactive monitoring, adaptive decision-making, workflow orchestration, and self-healing capabilities. Emerging concepts such as AI agents, autonomous platform management, and intelligent service operations demonstrate the potential for future platform engineering environments that can operate with greater autonomy while maintaining high levels of reliability and resilience.
Despite these advancements, several challenges remain. Issues related to model accuracy, hallucination risks, explainability, governance, security, privacy, compliance, and ethical AI deployment must be carefully addressed before fully autonomous operational systems can be widely adopted. Organizations must establish robust governance frameworks, validation mechanisms, and human oversight processes to ensure that AI-driven recommendations and automated actions align with business objectives, regulatory requirements, and operational best practices.

In conclusion, Large Language Models represent a significant technological milestone in the evolution of platform engineering. The evidence synthesized in this study confirms that LLM-driven operational automation has the potential to transform how organizations design, manage, and optimize digital platforms. As research and technological innovation continue to advance, intelligent platform engineering will increasingly move toward autonomous, adaptive, and highly resilient operational models. Future developments in generative AI, agentic systems, and intelligent automation are expected to redefine platform engineering practices, enabling organizations to achieve greater agility, operational excellence, and sustainable innovation in complex cloud-native environments.

Refernces

1. Chen, L. (2015). Continuous delivery: Huge benefits, but challenges too. IEEE Software, 32(2), 50–54. https://doi.org/10.1109/MS.2015.27
2. Yamsani, N. (2024). Designing trustworthy enterprise AI systems through a zero trust intelligence fabric with a unified approach to identity centric governance, adaptive policy automation, and end-to-end data security. ESP Journal of Engineering & Technology Advancements, 4(4), 210–222. https://doi.org/10.5281/zenodo.20092686
3. Thota MR. Extending NLIDB Paradigms to Cloud Infrastructure: A Framework for Natural Language-Driven Operations. J Artif Intell Mach Learn & Data Sci 2025 8(4), 3401-3409. DOI: doi.org/10.51219/JAIMLD/Madhava-rao-thota/678
4. Seetala, S. R. (2025). Architecting autonomous data platforms: Integrating AI-driven governance, metadata intelligence, and data mesh principles. International Journal of Science, Engineering and Technology, 13(1). https://doi.org/10.5281/zenodo.19208729
5. Kubam, C. S., Vollem, S., Akkenapally, G. C., & Fnu, H. (2026). AI-enabled cloud accounting for predictive financial management in modern businesses. In 2026 Innovations in Machine, Engineering, and Digital Conference (IMED) (pp. 1–5). IEEE. https://doi.org/10.1109/IMED68921.2026.11484120
6. Leite, L., Rocha, C., Kon, F., Milojicic, D., & Meirelles, P. (2019). A survey of DevOps concepts and challenges. ACM Computing Surveys, 52(6), 1–35. https://doi.org/10.1145/3359981
7. Ghanta, S. (2025). Learning-driven control loops for self-improving microservice platforms: Autonomic architectures and adaptive policy optimization. International Journal of Research Publications in Engineering, Technology and Management (IJRPETM), 8(1), 11827–11835. https://doi.org/10.15662/IJRPETM.2025.0801012
8. Teegala, R. (2024). AI-assisted code generation for enterprise Java services: Productivity, reliability, and governance considerations. International Journal of Scientific Research in Science, Engineering and Technology, 11(8), 312–329. https://doi.org/10.32628/IJSRSET24118043
9. BasiReddy, S. R. (2025). A multilayer governance framework for trustworthy AI-augmented CRM ecosystems. European Journal of Advances in Engineering and Technology, 12(7), 80–86. https://doi.org/10.5281/zenodo.17949783
10. Reed, J., Carter, E., Thompson, M., Reynolds, S., & Richard, A. (2022). A systematic review of explainable artificial intelligence techniques for trustworthy machine learning systems. Zenodo. https://doi.org/10.5281/zenodo.20065858
11. Boddupally, H. L. (2025). Self-improving enterprise platforms using learning loops and AI-driven orchestration. FMDB Transactions on Sustainable Intelligent Networks, 2(3), 164–175. https://doi.org/10.69888/FTSIN.2025.000538
12. Menda, J. R. (2025). Hybrid reasoning developer assistants leveraging LLMs for enterprise grade Java and Node financial systems. AVE Trends in Intelligent Management Letters, 1(3), 150–164. https://doi.org/10.64091/ATIML.2025.000164

13. Srujana, P. (2025). Architecting LLM powered support bots for data engineering operations: Controlled reasoning, evidence grounding, and audit ready incident workflows. Journal of Scientific and Engineering Research, 12(11), 211–225. https://doi.org/10.5281/zenodo.20201082
14. Kephart, J. O., & Chess, D. M. (2003). The vision of autonomic computing. Computer, 36(1), 41–50. https://doi.org/10.1109/MC.2003.1160055
15. Srikanth CV. A Unified Agentic AI Framework for Self-Adaptive Quality Engineering in Cloud-Native Financial Architectures. J Artif Intell Mach Learn & Data Sci 2026 9(1), 3419-3425. DOI: doi.org/10.51219/JAIMLD/Srikanth-chakravarthyvankayala/680
16. Seetala, S. R. (2024). Architecting trustworthy AI: Governance frameworks for responsible artificial intelligence in enterprise data ecosystems. International Journal of Science, Engineering and Technology, 12(1). https://doi.org/10.5281/zenodo.19208753
17. Huebscher, M. C., & McCann, J. A. (2008). A survey of autonomic computing. ACM Computing Surveys, 40(3), 1–42. https://doi.org/10.1145/1380584.1380585
18. Ghanta, S. (2024). Embedding governance controls into enterprise language model workflow architectures. International Journal of Core Engineering & Management, 7(11). https://doi.org/10.5281/zenodo.18921552
19. Thota, M. R. (2025). Autonomous policy-driven cloud platforms: Integrating declarative governance, distributed data systems, and AI-driven control loops for intelligent enterprise modernization. International Journal of Research and Applied Innovations, 8(4), 12642–12657. https://doi.org/10.15662/IJRAI.2025.0804013
20. Hayes, A., Brooks, N., Cooper, R., Simmons, V., & Richard, A. (2024). Optimization techniques for large-scale deep neural networks: A performance and efficiency analysis. International Journal of Scientific Research & Engineering Trends, 10(1). https://doi.org/10.5281/zenodo.20052353
21. Salehie, M., & Tahvildari, L. (2009). Self-adaptive software: Landscape and research challenges. ACM Transactions on Autonomous and Adaptive Systems, 4(2), 1–42. https://doi.org/10.1145/1516533.1516538
22. Boddupally, H. L. (2024). Cognitive decision automation framework integrating LLMs with SQL datastores and enterprise rule engines. Journal of Artificial Intelligence, Machine Learning & Data Science, 2(3), 3154–3163. https://doi.org/10.51219/JAIMLD/hema-latha-boddupally/644
23. Vollem, S. (2025). Hybrid cloud deployment models for enterprise modernization: Architectural strategies, integration patterns, and governance frameworks. International Journal of Scientific Research in Science and Technology, 12(16), 515–527. https://doi.org/10.32628/IJSRST25121660
24. Yamsani, N. (2023). Context-aware metadata enrichment in enterprise master data management: A natural language processing approach for EBX repositories. International Journal of Sustainable Development in Computing Science, 5(1), 1–28. Retrieved from https://www.ijsdcs.com/index.php/ijsdcs/article/view/707/270
25. Menda, J. R. (2024). Transforming Java and Node banking applications through generative AI-centric code engineering. Journal of Scientific and Engineering Research, 11(4), 394–408. https://doi.org/10.5281/zenodo.18085354
26. Garlan, D., Cheng, S. W., Huang, A. C., Schmerl, B., & Steenkiste, P. (2004). Rainbow: Architecture-based self-adaptation. Computer, 37(10), 46–54. https://doi.org/10.1109/MC.2004.175

27. Parepalli, S. (2025). Self-improving data pipelines using continuous learning models: Adaptive execution, feedback-driven optimization, and governance-aware automation in next-generation data ecosystems. Journal of Scientific and Engineering Research, 12(3), 169–184. https://doi.org/10.5281/zenodo.20200996
28. Vankayala, S. C. (2024). Intelligent quality assurance in cloud-native systems: A deep learning and reinforcement learning approach to adaptive test coverage optimization. International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), 11(6), 546–553. https://doi.org/10.32628/IJSRSET2512555
29. Ghanta, S. (2024). From monoliths to intelligence: Generative AI-driven code transformation and modernization of legacy Spring systems. International Journal of Scientific Research in Science, Engineering and Technology, 11(8), 290–299. https://doi.org/10.32628/IJSRSET24118041
30. Thompson, D., Bennett, O., Walker, J., Collins, H., & Richard, A. (2021). AI-driven autonomic control with machine learning for self-healing distributed systems. Zenodo. https://doi.org/10.5281/zenodo.20064011
31. Buyya, R., Broberg, J., & Goscinski, A. (2011). Cloud Computing: Principles and Paradigms. Wiley. https://doi.org/10.1002/9780470940105
32. Teegala, R. (2024). A governance oriented study of fine-tuning domain specific large language models with transaction and operations data. KOS Journal of AIML, Data Science, and Robotics, 1(1), 1–10. https://doi.org/10.5281/zenodo.18712442
33. BasiReddy, S. R. (2025). Reinforcement learning for self-optimizing customer relationship management platforms: From contextual bandits to deep sequential decision systems. International Journal of Science, Engineering and Technology, 13(1). Zenodo. https://doi.org/10.5281/zenodo.18185140
34. Boddupally, H. L. (2021). Toward intelligent root cause analysis in multi-layer enterprise systems: Tracing, statistical inference, and dependency-graph reasoning. International Journal of Core Engineering & Management, 6(12). https://doi.org/10.5281/zenodo.18902004
35. Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. Journal of Internet Services and Applications, 1(1), 7–18. https://doi.org/10.1007/s13174-010-0007-6
36. Seetala SR. Real-Time Data Monitoring Using Cloud Observability Tools: Architectures, Techniques and Emerging Practices. J Artif Intell Mach Learn & Data Sci 2023 6(4), 3367-3374. DOI: doi.org/10.51219/JAIMLD/srinivasa-rao-seetala/673
37. Thota MR. Autonomous Cloud Infrastructure: Leveraging Generative AI for Intelligent Blueprinting and Continuous Architecture Optimization. J Artif Intell Mach Learn & Data Sci 2024 7(1), 3410-3418. DOI: https://doi.org/10.51219/JAIMLD/Madhava-rao-thota/679
38. Menda, J. R. (2022). Grounded generation for enterprise knowledge: Automated documentation and knowledge extraction using GenAI agents. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 8(3), 857–866. https://doi.org/10.32628/CSEIT2215512
39. Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A. F., & Buyya, R. (2011). CloudSim: A toolkit for cloud computing environments. Software: Practice and Experience, 41(1), 23–50. https://doi.org/10.1002/spe.995

40. Burns, B., Grant, B., Oppenheimer, D., Brewer, E., & Wilkes, J. (2016). Borg, Omega, and Kubernetes. Communications of the ACM, 59(5), 50–57. https://doi.org/10.1145/2890784
41. Vollem, S. (2025). Serverless deployment strategies for high-availability cloud platforms: Architectural patterns, distributed reliability, and event-driven scalability. International Journal of Scientific Research & Engineering Trends, 11(1). https://doi.org/10.5281/zenodo.19219568
42. Parepalli, S. (2024). Architecting multi cloud data engineering models for high resilience and low latency patterns for active pipelines, consistent governance, and operational automation. International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), 10(7), 309–328. https://doi.org/10.32628/CSEIT24107452
43. Vankayala, S. C. (2023). Reinforcement learning–driven cognitive testing for scalable and resilient financial systems. ESP Journal of Engineering & Technology Advancements, 3(4), 209–217. https://doi.org/10.5281/zenodo.20092735
44. Nagender, Y. (2025). Regulatory intelligence engineering for global enterprises: Governance architectures for AI-enabled compliance operations. European Journal of Advances in Engineering and Technology, 12(4), 101–124. https://doi.org/10.5281/zenodo.19327897
45. Teegala, R. (2023). Secure prompt engineering for banking and payment applications: Design principles, threat models, and governance controls for generative AI in regulated financial systems. KOS Journal of AIML, Data Science, and Robotics, 1(1), 1–9. https://doi.org/10.5281/zenodo.18712372
46. BasiReddy, S. R. (2024). Predictive customer journey intelligence: AI-driven orchestration with LLMs, semantic retrieval, and zero trust governance. Journal of Artificial Intelligence, Machine Learning & Data Science, 2(4), 2994–2999. https://doi.org/10.51219/JAIMLD/santhoshreddy-basireddy/621