International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 13 Issue: 1
Article Received: *25 November 2024* ***Revised:*** *12 December 2024* ***Accepted:*** *30 January 2025*

A Unified Framework for Digital Delivery: Transition Strategies from Legacy to Cloud-Native Systems

Rajalingam Malaiyalan
Independent Researcher, USA.

Abstract

In this essay, the paradigm changes in cloud migration strategies—from lift-and-shift to full cloud-native transformation—is examined. By examining technology elements, organisational factors, and architectural patterns, the paper offers a comprehensive framework for comprehending and deploying cloud native infrastructure. The demands of contemporary analytics workloads, real-time processing needs, and the exponential expansion of data volumes are becoming more and more difficult for legacy data warehouse systems to handle, despite their decades of dependability. Cloud-native tactics are revolutionising how businesses link heterogeneous systems, manage processes, and provide uniform user experiences. In order to address the particular security needs of cloud-native systems, this paper looks at a range of privacy-enhancing and trust-centric tools and strategies. In particular, a range of solutions are discussed, including cloud-native endpoint security solutions for guaranteeing trust and resilience in dynamic contexts, runtime protection platforms for real-time threat detection and responses, and service mesh technologies for secure service-to-service communication. To improve trust and transparency in cloud-native security, the significance of threat detection and response systems, cloud-native security information and event management (SIEM) solutions, and network security are also discussed. To guarantee comprehensive security in a cloud-native architecture, we also provide an extensive case study that illustrates how security measures are implemented across many levels, including application, network, infrastructure, security, and compliance. Organisations may strengthen the security posture of their cloud-native implementations by looking at these privacy-enhancing techniques and technologies. This will lower risks and guarantee the reliability of their data and apps in the dynamic ecosystem of today's digital world.

Keywords: - Cloud Migration, Cloud-Native Transformation, Legacy Data, Real-Time Processing, Security Requirements, Security Information and Event Management (SIEM), Privacy-Enhancing Methods, Digital Landscape, Network Security.

I. INTRODUCTION

In today's rapidly evolving digital world, characterised by rapid technological advancement and relentless innovation, the idea of cloud-native architecture has become crucial for effectiveness, scalability, and resilience [1]. As businesses struggle to be flexible and competitive, using cloud-native strategies has become essential rather than optional [2]. This research aims to clarify the foundations of cloud-native architecture, emphasise the critical role that cloud-native architecture plays in the domains of artificial intelligence (AI) and cybersecurity, and stress the critical necessity of cloud-native application security.

Organisations are facing serious difficulties with conventional data warehousing infrastructures as the digital revolution of data management reaches a critical point. According to recent studies, legacy systems continue to have disadvantages, such as limited scalability, expensive maintenance, and rigid data processing capabilities [2, 3]. This study attempts to fill important research gaps by investigating the changing field of cloud-native data warehousing [3, 4] and conducting a thorough examination of organisational preparedness, performance standards, and technology migration options.

1.1 Cloud Computing Adoption Trends

In terms of how businesses design, implement, and oversee their IT infrastructure, cloud computing has completely changed the game. As Eleni Karatza points out, cloud technologies have developed across a number of discrete stages, each distinguished by escalating abstraction levels and service delivery methods. Because cloud computing promises more scalability, lower capital costs, and higher operational agility, businesses all over the globe are adopting it at an accelerated rate [3, 4]. This tendency has been especially noticeable in industries

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 13 Issue: 1
Article Received: *25 November 2024* ***Revised:*** *12 December 2024* ***Accepted:*** *30 January 2025*

going through fast digital transition, as pressure from competitors demands more adaptable and responsive IT capabilities.

1.2 Evolution of Cloud Migration Strategies

Similarly, the methods used to migrate to cloud systems have changed over time. At the beginning, a lot of companies used a "lift-and-shift" strategy, which entailed moving current apps to cloud infrastructure with very minor architectural changes [3, 5]. The route of least resistance provided by this strategy enabled companies to quickly move workloads while maintaining current operating paradigms. Nevertheless, as noted by P.A. Selvaraj, M. Jagadeesan, and associates, companies often find that these migrations fall short of using the full potential of cloud platforms [3, 4]. The study they conducted on adoption determinants in private sector organisations reveals that just moving apps without improving their architecture might restrict their potential advantages and perhaps create new inefficiencies.

1.3 The Paradigm Shift to Cloud-Native Approaches

An industry-wide paradigm shift from lift-and-shift migrations to cloud-native transformations is examined in this article [5]. Cloud-native methods, which include declarative APIs, serverless computing, microservices, and containerisation, constitute a fundamental rethinking of infrastructure management and application design. This change is not only technological; it also affects development processes, operational procedures, and organisational structures.

II. THE LIMITATIONS OF TRADITIONAL LIFT-AND-SHIFT APPROACHES

2.1 Definition and Characteristics of Lift-and-Shift Migrations

The simplest method for moving to the cloud is the lift-and-shift technique, commonly known as rehosting. With little change to the underlying architecture or coding, this technique transfers apps from on-premises infrastructure to cloud settings. Lift-and-shift migrations, as defined by C. Ward, N. Aravamudan, and associates [3, 5], basically duplicate current workloads in cloud environments while preserving the same operational features, dependencies, and architectural patterns as the source environment.

2.2 Historical Context and Initial Value Proposition

Lift-and-shift became a popular migration technique at the same time as business cloud use was just getting started. During this time, businesses aimed to minimise interference with current systems and procedures while taking advantage of cloud platforms' elastic resource allocation and operational expense model [6, 7]. Rapid infrastructure cost reduction and data centre consolidation were given precedence over architectural optimisation in many early cloud migration projects.

Table 1 Lift-and-Shift vs. Cloud-Native Methods Comparison. [4, 5]

Aspect	Lift-and-Shift Approach	Cloud-Native Approach
Architecture	Monolithic, unchanged	Microservices, containerized
Infrastructure	Utilization Static, often overprovisioned	Dynamic, demand-based
Scalability	Limited by original design	Horizontal, automatic
Deployment	Infrequent, scheduled	Continuous, independent
Operations	Manual, traditional IT	Automated, DevOps-oriented
Cost Model	Capital expense reduction	Consumption-based optimization
Technical Debt	Preserved or increased	Reduced through redesign
Organizational Impact	Minimal process change	Significant cultural shift

2.3 Technical Constraints and Operational Challenges

Lift-and-shift migration has a number of technological limitations that restrict the potential advantages of cloud adoption, despite its seeming simplicity. List the main difficulties they encountered while analysing real-time data analytics workloads in their case study [10]. The architectural features required to take use of cloud-native features like auto-scaling, self-healing, and distributed processing are usually absent from applications built for static, on-premises infrastructure [4, 6].

2.4 Cost Inefficiencies in the Long Term

Lift-and-shift migrations often result in long-term cost inefficiencies, even if they may provide short-term cost gains via data centre consolidation and lower capital investment [6, 7]. Applications that have been moved

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 13 Issue: 1
Article Received: *25 November 2024* ***Revised:*** *12 December 2024* ***Accepted:*** *30 January 2025*

without undergoing architectural changes are usually unable to benefit from cloud-native cost optimisation features like spot instances, serverless computing, and fine-grained resource allocation.

2.5 Case Examples of Lift-and-Shift Implementations and Their Outcomes

Many recorded case studies from many sectors demonstrate the limits of lift-and-shift techniques. Numerous instances of businesses that first used lift-and-shift tactics but later experienced declining performance, increased operational complexity, and rising expenses.

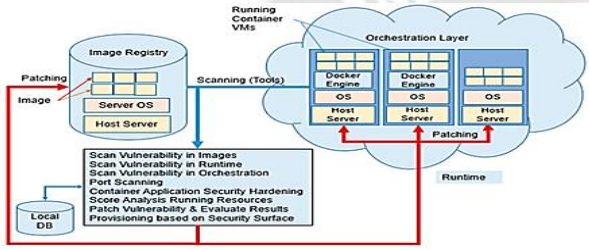

Fig. 1 An organised summary of the main features of cloud-native security, with particular attention on blockchain-based systems, container runtime security, DevSecOps, AI-driven threat detection, and compliance automation. [7, 8]

Additionally, studies have looked at the advantages and challenges of safeguarding certain cloud-native technologies, such as serverless computing and edge computing. The author's study, which looked at the security implications for serverless architectures, found that serverless applications need robust authentication, authorisation, and data protection protocols [25]. In a similar vein, studies conducted by authors studied security concerns and best practices in edge computing settings [7, 8], emphasising the need of protecting edge devices, data transit, and app deployment in decentralised edge environments. Regarding the present state of cloud-native security practices and trends, the examination of vendor landscapes, market trends, and adoption patterns for security solutions in cloud-native settings has also provided valuable insights [8]. These studies provide helpful advice to businesses seeking to strengthen their cloud-native security posture and effectively handle the ever-evolving threat environment, as shown in Table 2 [22].

Table 2 Cloud-native security issues and applications of current review articles. [6, 9]

Focus Area	Issues	Description	Tools and Techniques
Cloud-native security's dynamic nature	Dealing with rapid changes in containerised systems	Investigated the impact of dynamic cloud-native environments on security, emphasising flexible security solutions.	Adjustable security methods and modular threat modelling
The impact of microservice architecture on security	lowering the dangers associated with unauthorised access and decentralised patterns	Investigated the impact of microservice architectures on security in cloud-native environments.	Strong authentication methods in decentralised access control systems
Pipelines for DevSecOps	Encouraging a development approach that puts security first	Explored how DevSecOps pipelines may automate security testing along the software development lifecycle.	Vulnerability scanning and automated security testing
Making use of threat intelligence	Overcoming challenges in proactive cyberattack defence	looked at combining DevSecOps pipelines with cloud-native threat intelligence technologies.	Gathering and assessing threat data
Container security at runtime	Avoiding runtime mistakes in containerised programs	Investigated the challenges of preventing runtime vulnerabilities in containerised programs.	Container isolation and lowering host operating system vulnerabilities
Improved security at runtime	Putting rules in place to secure cloud-native settings	Examined the use of advanced runtime security features and sandboxing in cloud-native settings.	Methods for runtime protection and sandboxing

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 13 Issue: 1
Article Received: *25 November 2024* ***Revised:*** *12 December 2024* ***Accepted:*** *30 January 2025*

III. CLOUD-NATIVE APPLICATIONS

The idea of cloud-native applications, which use all facets of cloud computing to provide unparalleled efficiency, scalability, and flexibility, signifies a paradigm change in software development, deployment, and operation. Let's now examine the different types of cloud-native apps in further depth, as shown in Figure 2, to examine their real-world applications.

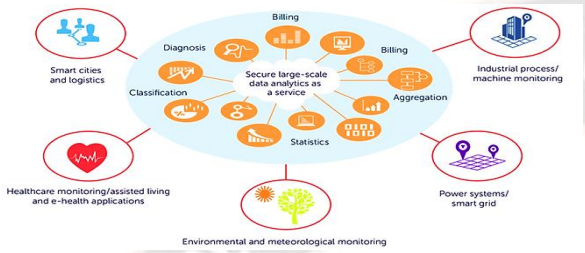

Fig. 2 The classification of popular cloud-native application types in a variety of fields, including blockchain, AI, big data, IoT, and healthcare systems. [9, 10]

3.1 Overview of Security Techniques for Cloud-Native Applications

In cloud-native architectures, service mesh technologies, such as Istio, have gained popularity as a way to manage the complexity of communication between services. In the framework of inter-service communication, they provide a specialised infrastructure layer that controls traffic routing, load balancing, and service discovery [24]. One of the main security benefits of service mesh is Mutual Transport Layer Security (mTLS) encryption, which ensures secure communication between services by encrypting traffic and enables both client-side and server-side authentication .

Runtime security monitoring and protection are necessary for loud-native environments. Examples of runtime application self-protection (RASP) techniques that satisfy this need include Trivy and Sysdig Secure from Aqua Security. Unlike traditional application security technologies that focus on perimeter protection, RASP operates inside the application runtime, allowing it to monitor application behaviour in real time and respond to security threats dynamically.

Serverless computing platforms, such as AWS Lambda and Azure Functions, may facilitate the development and deployment of cloud-native applications since they don't need infrastructure administration . However, serverless systems introduce several security challenges, including function separation, runtime monitoring, and access control. Serverless security solutions like AWS Lambda Layers and Azure Functions Proxies, which provide several features for safeguarding serverless programs, solve these problems.

Cloud-native firewall solutions such as Aqua's Cloud-Native Security Posture Management (CSPM) and Palo Alto Networks' Prisma Cloud provide centralised security management for cloud-native infrastructure and apps. These solutions provide a wide range of features to protect cloud-native environments against various threats, including malware, network intrusions, and data breaches [10, 25].

3.2 Cloud-Native Identity and Access Management (IAM)

IAM solutions such as AWS IAM, Role-based Access Control (RBAC) IAM, and Attribute-based Access Control (ABAC) IAM services are crucial for safeguarding user access to cloud resources in modern application architectures [13]. These systems, which provide centralised management of user identities, access limitations, and permissions across dispersed environments, enable organisations to consistently enforce security requirements.

3.3 Comparative Analysis of Cloud-Native Security Techniques

Although each cloud-native security method has a unique purpose, adoption of these methods often depends on trade-offs between scalability, performance, and integration simplicity. Companies that want to customise their security stack according to operational needs and architectural specifications must have a comparative perspective [18]. Service mesh systems like Istio and Linkerd, for example, provide good observability and scalability, but they are less suitable for workloads that are sensitive to latency because of their sidecar proxy approach, which adds a modest performance penalty. However, runtime protection solutions like as Sysdig Secure provide highly compatible, lightweight, and real-time protection, which makes them simple to include into container orchestration settings.

3.4 Theoretical Justification of Security Techniques in Cloud-Native Environments

Cloud-native systems have distinct operational and architectural features that are intrinsically linked to the efficacy of certain security measures [14, 15]. In dynamic, distributed microservice architectures, where workloads are transient and components are loosely

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 13 Issue: 1
Article Received: *25 November 2024* ***Revised:*** *12 December 2024* ***Accepted:*** *30 January 2025*

connected, traditional perimeter-based security methods are inadequate. Because cloud-native apps often run in containers that instantiate and terminate quickly, for instance, static scanning is inadequate, making runtime protection systems especially useful.

IV. CASE SCENARIO: ENSURING SECURITY IN CLOUD-NATIVE APPLICATIONS

Applications in contemporary cloud-native settings are often made up of many microservices operating on dispersed infrastructures. Although this design has many benefits in terms of agility and scalability, it also presents difficult security issues. To safeguard private information, preserve service integrity, and stop unwanted access, every microservice, container, and communication channel has to be protected [16, 17]. Figure 3 illustrates how the security architecture combines many levels, with the security and compliance layer acting as the topmost layer that protects every other element.

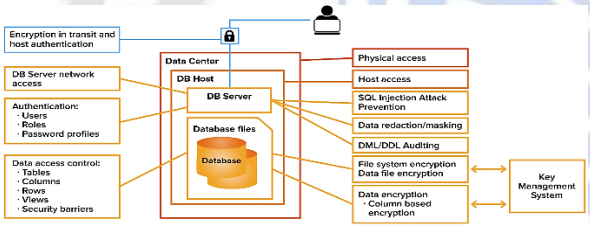

Fig. 3 Cloud-native apps with a multi-layered security architecture that shows safeguards implemented at the network, infrastructure, application, and compliance levels. [16, 17]

V. CHALLENGES AND SOLUTIONS

The dynamic nature of cloud environments and the unique features of containerised architectures provide a number of challenges when creating safe cloud-native apps [22]. One of the primary concerns with cloud-native applications is the security of managing and keeping sensitive data.

It is challenging to maintain stability and security in cloud-native settings due to their dynamic design, which is defined by microservices and containers that grow rapidly in response to demand [17, 18].

Another challenge is the potential difficulty of detecting sophisticated malware designed for containerised programs with conventional techniques in cloud-native systems. Using advanced software that avoids detection by conventional methods, such as malware designed to mine bitcoins, cybercriminals exploit this [19].

Microservices need secure inter-container connectivity and robust access control measures to prevent unauthorised access and data leaks [23]. In the absence of adequate security measures, attackers may be able to intercept or change communication across microservices, endangering the system's security and integrity.

Table 3 Privacy issues and fixes for protecting cloud-native apps. [20]

Concerns	Restrictions and Limitations	Challenges and Issues	Solutions
Security issues	Because cloud-native settings are dynamic, it can be difficult to maintain security and stability.	Security risks include things like handling sensitive data, following privacy regulations, and scalability problems.	Development of advanced malware detection techniques, enhanced container runtime security, and AI/ML-based security solutions.
Malware identification	Advanced malware designed for containerised apps could be invisible to conventional detection techniques.	By exploiting vulnerabilities in the host OS kernels, cybercriminals might cause damage to the whole system.	By exploiting vulnerabilities in the host OS kernels, cybercriminals might cause damage to the whole system.
Access management	Inadequate access control procedures may lead to	Inter-container communications need to be	Reducing the likelihood of unauthorised access and single points of failure by strengthening access

	unauthorised access and data breaches.	safeguarded to prevent security breaches.	control procedures and developing decentralised systems.
Examining conditions	Realistic testbeds for evaluating security solutions in microservices-based settings are challenging to construct.	Inadequate testing circumstances may result in security measure errors.	Development of advanced modelling tools and testing frameworks to accurately simulate real-world cloud-native environments.
Scalability consequences	Scalability provides benefits, but there are security dangers as well that need to be properly evaluated.	The dynamic nature of cloud-native settings may give rise to runtime vulnerabilities.	Preventive measures, enhanced container runtime security, and scalable DDoS mitigation technologies should be used to decrease the impact of DDoS attacks on cloud containers.

The development of more complex security solutions tailored to dynamic cloud-native systems is thus a promising avenue. For these systems to provide real-time threat identification and response, contemporary technologies like artificial intelligence and machine learning should be included.

Taking proactive measures will also be essential to reducing the potential damage that DDoS assaults might do to cloud containers. Scalable DDoS mitigation systems that can adapt to changing traffic loads and attack patterns on their own may be necessary to achieve this [20, 21]. Testing environments that are more realistic and detailed should eventually be created in order to reflect the complexity of cloud-native deployments. Potential avenues for future research include developing advanced testing frameworks and simulation tools that provide a more accurate representation of cloud-native environments in the real world, assisting businesses in swiftly identifying and fixing security vulnerabilities.

VI. CONCLUSION

The transition from lift-and-shift to cloud-native transformation signifies a fundamental change in the way businesses plan, build, and manage their infrastructure. This paper has shown that harmonisation across many dimensions—technical architecture, organisational structure, operational practices, and governance frameworks—is necessary for the effective deployment of cloud-native technology.

However, achieving their full potential requires a corresponding transformation in organisational culture, team structures, and development approaches. Organisations that take a comprehensive strategy to cloud-native transformation reap much more advantages than those that just pursue technical migrations, as shown by a variety of industry case studies.

Since the introduction of cloud-native architecture, application development, deployment, and administration have undergone a radical transformation. In cloud-native systems, there are specific security and trust-related issues that need to be resolved to guarantee data and application availability, integrity, and privacy. This research has looked at recent advancements in cloud-native application security techniques and technologies, including DevSecOps pipelines, cloud-native intelligence on threats platforms, and runtime protection solutions.

In addition to showing practical implementation, this layered model highlights how each approach complies with cloud-native design principles. Future studies will probably concentrate on creating ever more complex security systems that take use of developments in artificial intelligence and machine learning. In order to lessen the impact of DDoS assaults, these technologies are anticipated to transform container runtime security, improve privacy-aware access-control methods, and fortify trust-centric security models. Furthermore, the use of blockchain technology may enable fresh approaches to identity management and verification in cloud-native settings, resulting in increased security and transparency. Addressing the challenges of safeguarding cloud-native infrastructures will need a shift towards more intelligent and integrated security solutions, with findings based on architectural fit and trade-off analysis rather on actual data.

VII. REFERENCES

[1] C. Chen and J. Liu, "Research of Cloud-Native AS/RS Warehouse Management and Control Platform Architecture," *2024 5th International Conference on Computer Engineering and Application (ICCEA)*, Hangzhou, China, 2024, pp. 277-285, https://doi.org/10.1109/ICCEA62105.2024.10604148

[2] Ramakrishna Manchana, "Operationalizing Batch Workloads in the Cloud with Case Studies," volume 9 issue 7, July 2020, International Journal of Science and Research (IJSR), ISSN: 2319-7064, 2018. https://www.ijsr.net/getabstract.php?paperid=SR24820052154

[3] C. Ward, N. Aravamudan, et al., "Workload Migration into Clouds: Challenges, Experiences, Opportunities," in 2010 IEEE 3rd International Conference on Cloud Computing, 26 August 2010. https://www.computer.org/csdl/proceedings-article/cloud/2010/4130a164/12OmNwNOaKu

[4] Kimball R, Ross M. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. 3rd ed. Wiley; 2013 ISBN: 978-1118530801.. https://www.wiley.com/en-us/The+Data+Warehouse+Toolkit%3A+The+Definitive+Guide+to+Dimensional+Modeling%2C+3rd+Edition-p-9781118530801

[5] **Dageville, B., et al.** The Snowflake Elastic Data Warehouse. In Proceedings of the 2016 International Conference on Management of Data (SIGMOD), June. https://dl.acm.org/doi/10.1145/2882903.2903741

[6] Kleppmann M. Designing Data-Intensive Applications. O'Reilly Media; 2017. https://www.oreilly.com/library/view/designing-data-intensive-applications/9781491903063/

[7] Gadde, H. (2020). *AI-Enhanced Data Warehousing: Optimizing ETL Processes for Real-Time Analytics*. Revista de Inteligencia Artificial en Medicina, 11(1), 300–327. Retrieved from http://redcrevistas.com/index.php/Revista/article/view/178/201

[8] Dennis, B., & Rajab, H. (2024, November). *Utilizing Edge Computing in Cloud Environments for Warehouse Efficiency*. *International Journal of Advanced Computing Research, 12*(3), 145–162. https://www.researchgate.net/publication/385495675_Utilizing_Edge_Computing_in_Cloud_Environments_for_Warehouse_Efficiency

[9] S. Chintala, "Talent acquisition & transformation — A new paradigm," 2013 1st International Conference on Emerging Trends and Applications in Computer Science, Shillong, India, 2013, pp. vi-vii, doi: 10.1109/ICETACS.2013.6691383.

[10] Bosi, F., Corradi, A., Di Modica, G., Foschini, L., Montanari, R., Patera, L., & Solimando, M. (2020, September). Enabling smart manufacturing by empowering data integration with industrial IoT support. In *2020 International Conference on Technology and Entrepreneurship (ICTE)* (pp. 1-8). IEEE.Davenport T, Harris J. Competing on Analytics: The New Science of Winning. Updated ed. Harvard Business Review Press; 2017. https://ieeexplore.ieee.org/abstract/document/9215538/

[11] Bradley, T. (2017, February 24). ASOS Streamlines Fashion with Microsoft Azure. Forbes. https://www.forbes.com/sites/tonybradley/2017/02/24/asos-streamlines-fashion-with-microsoft-azure/

[12] Ramalingam, C., & Mohan, P. (2021). Addressing semantics standards for cloud portability and interoperability in multi cloud environment. Symmetry, 13(2), 317. https://www.mdpi.com/2073-8994/13/2/317

[13] Wang, Y. (2019, September). Towards service discovery and autonomic version management in self-healing microservices architecture. In Proceedings of the 13th European Conference on Software Architecture-Volume 2 (pp. 63-66). https://hal.science/hal-02445701/file/article.pdf

[14] N. Mendonca, P. Jamshidi, D. Garlan, and C. Pahl, "Developing self-adaptive microservice systems: challenges and directions," IEEE Software, vol. 38, no. 2, pp. 70-79, 2021. https://ieeexplore.ieee.org/document/8913688

[15] J. Alonso, L. Orue-Echevarria, E. Osaba, J. Lobo, I. Martinez, J. Díaz-de-Arcaya, and I. Etxaniz, "Optimization and prediction techniques for self-healing and self-learning applications in a trustworthy cloud continuum," Information, vol. 12, no. 8, p. 308, 2021. https://www.mdpi.com/2078-2489/12/8/308

[16] S. S. Gill and R. Buyya, "Failure Management for Reliable Cloud Computing: A Taxonomy, Model, and Future Directions," in Computing in Science & Engineering, vol. 22, no. 3, pp. 52-63, 1 May-June 2020. https://ieeexplore.ieee.org/document/8487042

[17] J. Kosińska and K. Zieliński, "Autonomic management framework for cloud-native applications," Journal of Grid Computing, vol. 18, no. 4, pp. 779-796, 2020. https://link.springer.com/content/pdf/10.1007/s10723-020-09532-0.pdf

[18] F. Samea, F. Azam, M. Rashid, M. Anwar, W. Butt, and A. Muzaffar, "A model-driven framework for data-driven applications in serverless cloud computing," PLOS ONE, vol. 15,no.8,e0237317,2020. https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0237317&type=printable

[19] A. Megargel, V. Shankararaman, and D. Walker, "Migrating from monoliths to cloud-based microservices: a banking industry example," Lecture Notes in Business Information Processing,vol.370,pp.85-108,2020. https://ink.library.smu.edu.sg/cgi/viewcontent.cgi?article=5728&context=sis_research

[20] J. Han, Y. Hong, and J. Kim, "Refining microservices placement employing workload profiling over multiple Kubernetes clusters," IEEE Access, vol. 8, pp. 192543-192556, 2020. https://ieeexplore.ieee.org/iel7/6287639/8948470/09235316.pdf