

# The evolution of cloud platform engineering: From manual deployments to full automation

Bhanu Prakash Kolli \*

*Jawaharlal Nehru Technological University, Hyderabad, India.*

Global Journal of Engineering and Technology Advances, 2025, 23(01), 187-194

Publication history: Received on 08 March 2025; revised on 14 April 2025; accepted on 16 April 2025

Article DOI: <https://doi.org/10.30574/gjeta.2025.23.1.0108>

## Abstract

Cloud platform engineering has undergone a remarkable transformation over the past decade, evolving from manual infrastructure management to sophisticated autonomous systems. This article represents technological advancement and a fundamental shift in operational methodologies. The journey began with tedious manual provisioning and configuration, marked by high maintenance overhead and configuration drift challenges. Infrastructure as Code introduced version-controlled, repeatable processes, while containerization revolutionized application packaging and runtime consistency. Kubernetes emerged as the dominant orchestration solution, providing declarative deployments and self-healing capabilities. The GitOps movement further refined automation by establishing Git repositories as the single source of truth for infrastructure and applications. Today, cloud platforms increasingly incorporate AI-powered operations, policy-driven automation, and cross-cloud abstraction layers. Looking ahead, the industry is trending toward fully autonomous platforms characterized by intent-based infrastructure, AI-driven architecture, and self-optimizing systems that continuously adapt to changing requirements without human intervention.

**Keywords:** Infrastructure Automation; Containerization; Kubernetes Orchestration; GitOps Methodology; Autonomous Cloud Platforms

## 1. Introduction

The landscape of cloud infrastructure management has undergone a remarkable transformation over the past decade. What once required manual configuration, lengthy deployment windows, and extensive human intervention has evolved into sophisticated, automated systems capable of self-healing and predictive optimization. This evolution represents technological advancement and a fundamental shift in how organizations approach infrastructure, application deployment, and operational excellence in the cloud.

Accelerating cloud adoption across enterprises globally has created increasingly complex environments spanning multiple cloud providers and regions. This complexity has naturally driven significant investment in automation technologies as organizations seek to manage distributed cloud footprints efficiently. As cloud services consume a growing portion of IT budgets, the financial imperative to optimize these investments becomes increasingly critical.

Industry research consistently demonstrates the transformative impact of automation on operational excellence. Organizations classified as high performers in DevOps practices deploy code significantly more frequently than low performers, with dramatically faster lead times for changes. These high-performing teams leverage sophisticated infrastructure automation to achieve faster recovery times from incidents while simultaneously reducing change failure rates. The correlation between automation maturity and business outcomes is clear, with advanced practitioners more likely to meet or exceed their organizational performance goals.

\* Corresponding author: Bhanu Prakash Kolli

The evolution toward automated cloud operations yields measurable benefits across multiple dimensions. Organizations with mature automation practices report substantial reductions in unplanned downtime, decreased time spent on routine maintenance tasks, and improved resource utilization. These efficiency gains translate directly to financial benefits, with fully automated cloud environments typically costing less to operate than manually managed equivalents. Despite these advantages, many enterprises remain in the early to middle stages of their automation journey, suggesting significant untapped potential for further optimization.

Modern cloud platforms increasingly incorporate AI-powered operations, enabling systems to process extraordinary volumes of operational telemetry daily. This wealth of data allows AI systems to identify patterns invisible to human operators, resulting in measurable improvements in incident detection and resolution times. The impact is particularly pronounced for security incidents, where AI-assisted detection has substantially reduced the average time to identify potential breaches compared to traditional manual monitoring approaches.

The shift toward declaration-based infrastructure management represents a crucial evolutionary step in cloud automation. Organizations utilizing infrastructure-as-code approaches typically experience fewer configuration-related incidents and deploy new environments faster than those relying on manual processes. Declarative approaches enable enterprises to achieve consistent configurations across multi-cloud environments, addressing a critical challenge in hybrid deployments. This approach has proven particularly valuable for compliance management, reducing the time required to implement regulatory controls across complex cloud footprints.

The future direction of cloud automation is trending toward autonomous operations. Forward-thinking organizations are implementing self-healing infrastructure capable of automatically responding to incidents without human intervention. These systems leverage sophisticated event-driven architectures where routine operational tasks are increasingly automated. The most advanced practitioners are moving toward intent-based infrastructure models where engineers specify desired business outcomes rather than technical specifications, yielding improvements in implementation time and deployment reliability.

---

## 2. The Era of Manual Infrastructure Management

In the early days of cloud computing, around 2006-2010, infrastructure provisioning closely resembled traditional on-premises approaches. Engineers would manually provision virtual machines through cloud provider consoles, SSH into instances to configure services and dependencies, maintain snowflake environments with unique, often undocumented configurations, and execute deployments through manual checklists and runbooks.

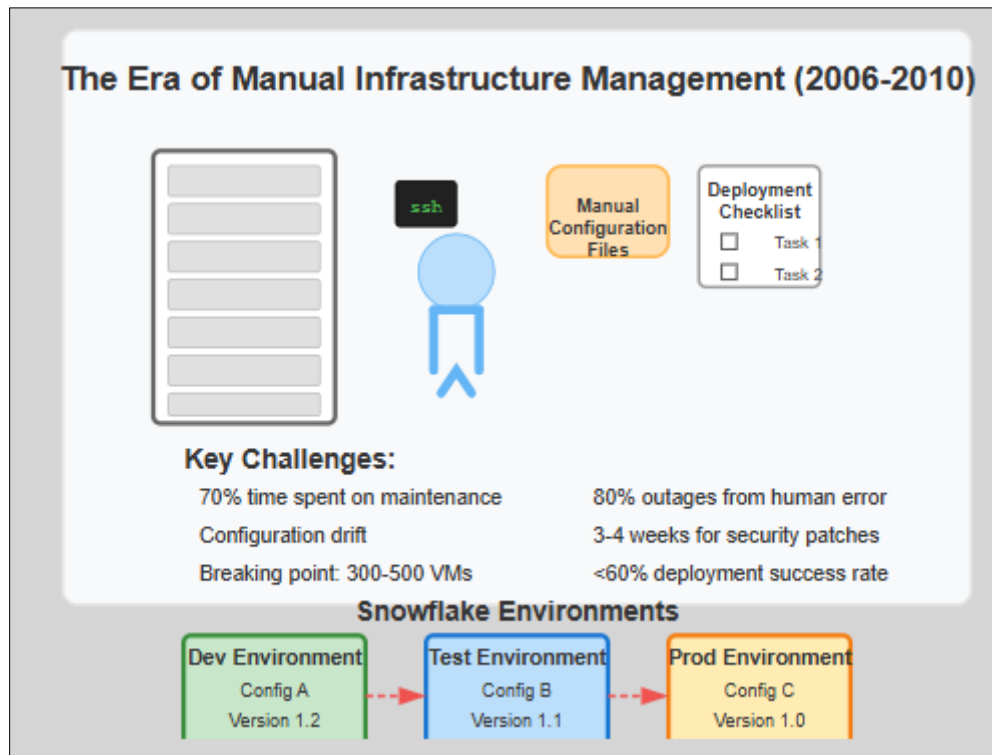
This approach created several significant challenges across the industry. According to an analysis by Gartechology Solutions, organizations in this pre-automation era faced substantial operational inefficiencies, with IT teams spending up to 70% of their time on routine maintenance tasks rather than innovation. The manual configuration of infrastructure not only consumed valuable engineering resources but also introduced significant inconsistencies across environments. These inconsistencies led to what practitioners termed "configuration drift," where production environments would gradually diverge from their intended state, creating unpredictable behavior and making troubleshooting exponentially more complex [3]. "We spent nearly 70% of our time just maintaining existing infrastructure rather than delivering new features," recalls a senior platform engineer from that era. "Configuration drift was inevitable, and troubleshooting production issues meant investigating unique environment quirks that existed nowhere else."

The operational inefficiencies of manual cloud management extended beyond day-to-day maintenance. Gartner's analysis found that human error accounted for approximately 80% of outages in manually managed environments, with misconfigurations representing the single largest cause of security incidents during this period. Without standardized deployment mechanisms, organizations typically experienced deployment success rates below 60%, with the remainder requiring some form of manual intervention or rollback. These challenges were particularly acute during scaling events, where manual processes simply couldn't keep pace with rapidly changing infrastructure requirements [4].

Security and compliance concerns were equally problematic in manually managed environments. Without automated controls, security policies were inconsistently applied, creating significant vulnerabilities that often went undetected until after breaches occurred. Gartechology Solutions identified that organizations using manual processes took an average of 3-4 weeks to implement security patches across their entire infrastructure, compared to just hours for highly automated environments. This extended exposure window dramatically increased risk profiles, particularly for

organizations operating in regulated industries where compliance requirements demanded consistent configuration and regular auditing [3].

The scalability limitations of manual management became increasingly apparent as cloud adoption accelerated. Gartner observed that the breaking point for manual infrastructure management typically occurred when organizations reached approximately 300-500 virtual machines or 20-30 applications. Beyond this threshold, the operational overhead became unsustainable, with incident rates, resolution times, and operational costs all increasing non-linearly with scale. This practical ceiling created significant barriers to cloud expansion and prevented many organizations from fully realizing the potential benefits of cloud computing [4].



**Figure 1** The Era of Manual Infrastructure Management (2006-2010)

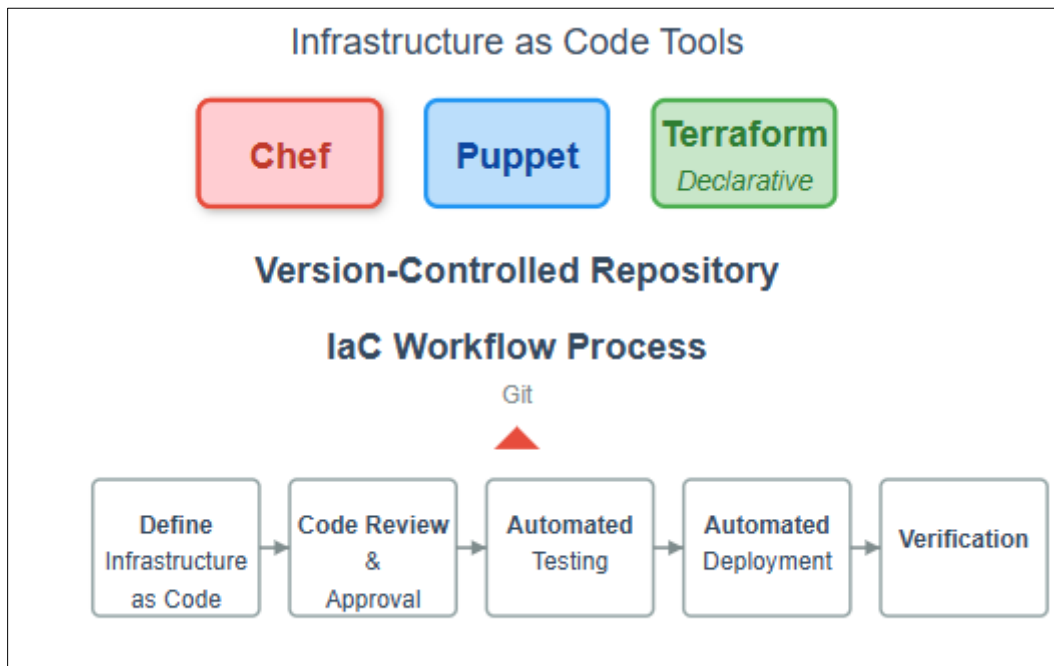
### 3. The Rise of Infrastructure as Code

The first major paradigm shift came with Infrastructure as Code (IaC) tools like Chef, Puppet, and later HashiCorp's Terraform. These technologies addressed fundamental problems by defining infrastructure in version-controlled code repositories, creating repeatable, consistent provisioning processes, enabling infrastructure testing and validation before deployment, and documenting system architecture implicitly through code.

Infrastructure as Code fundamentally transformed cloud operations by bringing software engineering practices to infrastructure management. According to the "State of Infrastructure as Code 2023" webinar hosted by DevOps.com, organizations adopting IaC reported significant improvements in deployment velocity, with elite performers achieving 80% faster infrastructure provisioning compared to organizations still using manual processes. The research highlighted that 89% of organizations now recognize Infrastructure as Code as a critical element of their cloud strategy, though implementation maturity varies widely across industries. This dramatic improvement stemmed from the elimination of error-prone manual processes, as infrastructure definitions became executable code that could be tested, validated, and reviewed before deployment [5].

Terraform, in particular, changed the game by providing a declarative approach to infrastructure definition. Engineers could specify the desired end state, and the IaC tool would determine how to achieve it. This declarative model represented a fundamental shift from the imperative scripting approaches that preceded it, as it allowed engineers to focus on "what" infrastructure was needed rather than "how" to create it. The DevOps.com webinar identified that 67%

of organizations now prefer declarative IaC tools over imperative alternatives, recognizing the significant operational advantages of this approach, particularly for complex, multi-cloud environments [5].



**Figure 2** The Rise of Infrastructure as Code

The version control aspect of Infrastructure as Code proved transformative for governance and compliance. According to Spiceworks' comprehensive analysis of Infrastructure as Code, placing infrastructure definitions in version-controlled repositories has revolutionized compliance management for regulated industries. By treating infrastructure changes like software changes—with formal code reviews, approvals, and audit trails—organizations have been able to establish clear governance over their cloud environments while simultaneously increasing deployment velocity. This approach enables organizations to validate that infrastructure meets security and compliance requirements before deployment, rather than discovering issues during post-implementation audits [6].

Infrastructure as Code also dramatically improved the consistency of multi-environment deployments, addressing one of the most persistent challenges in cloud computing. Spiceworks' analysis highlights that IaC enables organizations to maintain consistent configurations across development, staging, and production environments, eliminating the "it works on my machine" problem that plagued earlier cloud implementations. This consistency extends to security configurations, with 78% of respondents in Spiceworks' survey reporting improved security posture after adopting IaC. The ability to define security controls as code and apply them consistently across all environments has proven particularly valuable as organizations navigate increasingly complex compliance requirements and threat landscapes [6].

#### 4. The Containerization Revolution

The next transformative wave came with containerization, led by Docker's rise around 2013. Containers addressed a critical gap in cloud infrastructure: application packaging and runtime consistency.

The containerization revolution fundamentally altered how organizations approach application deployment and infrastructure management. According to the Cloud Native Computing Foundation's Annual Survey 2023, container adoption has shown substantial growth year-over-year since Docker's introduction, with the vast majority of organizations now using containers in production environments. The survey highlights how Kubernetes has become the de facto standard for container orchestration, with its adoption continuing to rise across organizations of all sizes. This trend illustrates how containerization has evolved from an experimental technology to an essential component of modern cloud infrastructure, providing organizations with the ability to create isolated, consistent application runtimes across diverse environments [7].

The operational impact of containerization extends far beyond deployment velocity. The CNCF survey reveals that organizations are increasingly deploying containerized applications across multiple clouds and on-premises environments, leveraging containers' inherent portability to avoid vendor lock-in and optimize for cost and performance. This hybrid approach has become particularly prevalent among enterprise organizations seeking to balance existing investments in private infrastructure with the scalability benefits of public cloud services. An engineer who worked through this transition notes, "Containers fundamentally changed our relationship with infrastructure. We stopped thinking about configuring servers and started thinking about application artifacts that could run anywhere."

Container adoption has driven significant changes in infrastructure utilization and cost structures. Red Hat's Enterprise Open Source Report 2022 emphasizes that organizations implementing containerization are achieving higher server utilization and improved cost efficiency compared to traditional virtual machine deployments. The report highlights how containers enable organizations to run applications with minimal overhead, allowing for higher application density on the same hardware. This efficiency gain has proven particularly valuable as organizations seek to optimize their infrastructure investments while maintaining application performance [8].

The standardization introduced by containers has also transformed the software development lifecycle. According to Red Hat's research, organizations utilizing containerization report substantial improvements in developer productivity and deployment reliability. This acceleration stems from containers' ability to create a clear separation between application and infrastructure concerns, allowing development teams to focus on application logic while operations teams manage the underlying infrastructure. The container format effectively became a standardized interface between development and operations, resolving many of the traditional conflicts between these groups and enabling more effective DevOps practices across the organization [8].

---

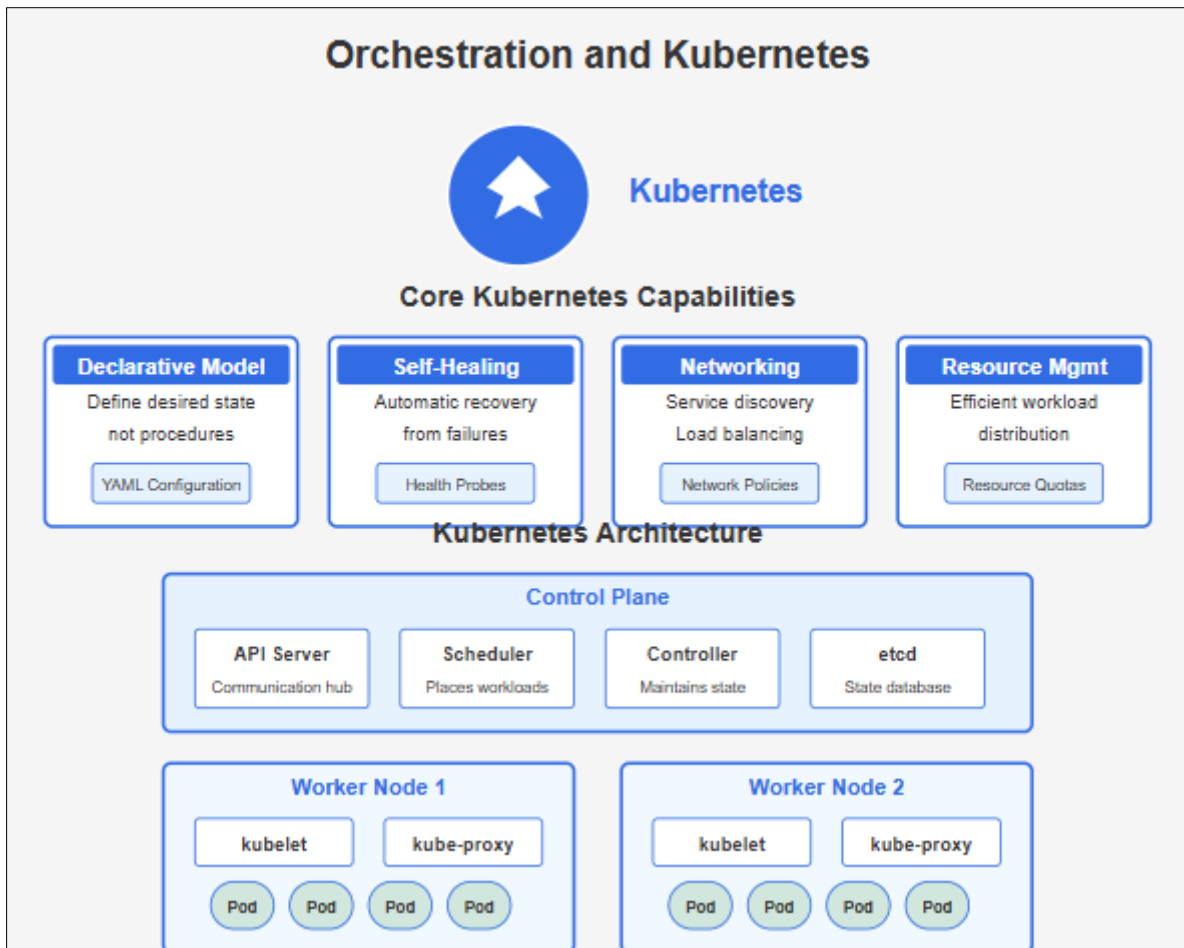
## 5. Orchestration and Kubernetes

While containers solved the packaging and consistency problem, they introduced new challenges around scheduling, networking, and lifecycle management at scale. Kubernetes emerged as the dominant solution by providing:

The rise of container orchestration platforms, particularly Kubernetes, represents one of the most significant technological shifts in modern cloud computing. According to Red Hat's Kubernetes Adoption, Security and Market Trends Report, Kubernetes has rapidly transitioned from an emerging technology to an essential platform for enterprise operations. The report highlights how organizations are expanding their Kubernetes footprints beyond initial pilot deployments to run mission-critical workloads, with a growing percentage of production applications being deployed on Kubernetes. This widespread adoption reflects Kubernetes' unique ability to address the operational challenges that emerged as containerization scaled beyond simple use cases. The platform's declarative deployment model has proven particularly valuable, allowing organizations to describe desired application states rather than specifying step-by-step procedures [9].

Kubernetes' self-healing capabilities have fundamentally changed how organizations approach application reliability. The platform's ability to automatically detect and respond to failures—restarting failed containers, rescheduling pods when nodes become unhealthy, and maintaining desired replica counts—has created a new paradigm for application resilience. According to the Voice of Kubernetes Experts Report by Portworx, organizations implementing Kubernetes at scale are experiencing significant improvements in operational resilience. The report emphasizes how Kubernetes' automated recovery mechanisms have transformed incident response, allowing teams to focus on systemic improvements rather than reactive firefighting. This shift in operational approach has proven particularly valuable as organizations scale their container deployments across multiple environments and cloud providers [10].

The networking capabilities provided by Kubernetes have addressed one of the most complex aspects of containerized application deployment. By abstracting service discovery, load balancing, and network policy management, Kubernetes enables organizations to implement sophisticated networking topologies without managing underlying infrastructure components. Red Hat's analysis highlights how Kubernetes' networking abstractions have proven essential for organizations implementing microservice architectures, where service-to-service communication patterns grow exponentially with scale. These abstractions enable development teams to focus on business logic rather than the complexities of network configuration and security [9].



**Figure 3** Orchestration and Kubernetes

Resource management across distributed infrastructure represents another area where Kubernetes has delivered significant value. By providing an abstraction layer above individual nodes, Kubernetes enables efficient resource allocation and workload placement across clusters that may span thousands of machines. The Portworx report indicates that organizations are increasingly leveraging Kubernetes' resource management capabilities to improve infrastructure utilization and application performance. The platform's ability to distribute workloads based on resource requirements, node availability, and affinity rules has proven particularly valuable for organizations seeking to optimize their infrastructure investments while maintaining application service levels. This capability has fundamentally changed how organizations approach capacity planning and resource allocation in cloud environments [10].

## 6. GitOps: Infrastructure and Application Convergence

Around 2018, the GitOps movement began gaining momentum, with tools like Flux and ArgoCD leading the charge. GitOps built upon the declarative nature of both IaC and Kubernetes by:

The GitOps methodology emerged as a natural evolution of DevOps practices, extending version control principles to encompass all aspects of infrastructure and application deployment. According to Octopus Deploy's State of GitOps Report, organizations are increasingly adopting GitOps as a core practice for managing cloud-native environments, with adoption growing steadily across organizations of all sizes. The report highlights how GitOps practices enable organizations to improve deployment consistency and reliability by establishing Git repositories as the single source of truth for all infrastructure and application configurations. This approach eliminates confusion about the intended state of environments and significantly reduces the risk of configuration drift that plagued earlier deployment methodologies [11].

The implementation of reconciliation controllers represents one of GitOps' most transformative innovations. These controllers continuously monitor the actual state of deployed infrastructure and applications, automatically reconciling

any deviations from the desired state defined in Git. The CNCF GitOps Microsurvey reveals that GitOps practices are increasingly moving into mainstream adoption, with most practitioners learning and implementing these approaches directly on the job. The survey highlights how this automated reconciliation capability has fundamentally changed how organizations approach deployment, with GitOps becoming a critical element of modern Kubernetes management. As the technology matures, organizations are reporting that GitOps practices are significantly improving their ability to maintain consistent configurations across increasingly complex environments [12].

GitOps has transformed the deployment pipeline by creating a clear, auditable path from code commit to production deployment. Octopus Deploy's research shows that organizations implementing GitOps are achieving more consistent and reliable deployments across environments. This improvement stems from the comprehensive audit trail inherent in Git-based workflows, where every change is documented, reviewed, and versioned. "GitOps closed the loop on automation," explains a platform engineering director quoted in the report. "By making Git the central control point, we eliminated manual interventions and created a completely auditable system where any change could be traced back to a commit." [11]

The ability to roll back changes quickly and reliably represents another significant advantage of the GitOps approach. The CNCF microsurvey indicates that organizations implementing GitOps are experiencing improved recovery capabilities for deployment-related incidents compared to those using traditional deployment methods. The survey highlights how GitOps' immutable infrastructure approach makes reverting to a previous known-good state as simple as reverting to a previous Git commit. This capability has proven particularly valuable for organizations operating in highly regulated industries, where demonstrating precise control over environment changes is essential for compliance requirements [12].

---

## 7. The Future: Toward Fully Autonomous Cloud Platforms

### 7.1. Looking ahead, the industry is moving toward increasingly autonomous cloud platforms characterized by:

The evolution toward fully autonomous cloud platforms represents the next major frontier in cloud computing. According to AWS Executive Insights on Cloud Strategy, organizations are increasingly seeking to implement intelligent automation that goes beyond basic scripting to create truly autonomous operations. The transition to intent-based infrastructure, where engineers specify business objectives rather than technical details, is enabling a new paradigm in cloud management that reduces operational overhead while improving responsiveness to changing business requirements. AWS notes that leading organizations are implementing platforms that can interpret high-level business intent and automatically translate it into appropriate technical implementations, bridging the gap between business objectives and infrastructure configurations [13].

AI-driven architecture that evolves based on changing application needs is rapidly moving from research to practical implementation. According to Jessup University's analysis on the future of cloud computing, artificial intelligence and machine learning are becoming integral components of modern cloud platforms. These technologies enable systems to analyze vast amounts of operational data, identify patterns and trends, and make intelligent decisions about resource allocation and configuration. The report highlights how AI-driven architectures can adapt to changing workload characteristics without human intervention, enabling a level of responsiveness and efficiency that would be impossible with traditional management approaches. This capability is becoming increasingly critical as application complexity and scale continue to grow [14].

Self-optimizing systems represent a logical evolution of current automation practices, extending beyond simple rule-based scaling to encompass comprehensive optimization across multiple dimensions. AWS's cloud strategy insights emphasize that organizations implementing advanced automation are achieving substantial improvements in both operational efficiency and business agility. These self-optimizing systems continuously monitor application behavior, user experience metrics, and infrastructure utilization, automatically adjusting configurations to maintain an optimal balance between performance, reliability, and cost. As these capabilities mature, they enable organizations to maintain optimal operations even as business requirements and technical environments evolve [13].

Cross-cloud abstraction layers that manage workloads across providers transparently have emerged as a critical requirement for organizations seeking to avoid vendor lock-in while maintaining operational efficiency. Jessup University's analysis notes that multi-cloud and hybrid cloud strategies are becoming the norm rather than the exception, with organizations deliberately spreading workloads across multiple providers to optimize for cost, performance, and resilience. The research indicates that platforms providing consistent management interfaces across cloud providers will be essential for managing this complexity. This capability will be particularly valuable as

organizations seek to leverage the unique strengths of different cloud providers while maintaining operational consistency and avoiding provider-specific dependencies [14].

## 8. Conclusion

The evolution from manual infrastructure management to automated, policy-driven cloud platforms represents one of the most significant transformations in modern IT. Each advancement—from Infrastructure as Code to containerization, Kubernetes, GitOps, and AI-powered automation—has fundamentally changed how engineers build and operate cloud systems. For organizations embarking on cloud transformation journeys, understanding this evolutionary path provides valuable context for building modern platforms. The future promises even greater automation and intelligence, further reducing operational overhead and allowing engineers to focus on delivering business value rather than managing infrastructure. As we continue this journey, the most successful organizations will be those that embrace automation not just as a technical capability but as a core organizational principle that reshapes how cloud infrastructure is conceptualized, built, and operated.

## References

- [1] Flexera, "2024 State of the Cloud Report," Flexera, 2024. [Online]. Available: <https://resources.flexera.com/web/pdf/Flexera-State-of-the-Cloud-Report-2024.pdf>
- [2] Google Cloud and DORA, "Accelerate State of DevOps 2023," Google, 2023. [Online]. Available: [https://services.google.com/fh/files/misc/2023\\_final\\_report\\_sodr.pdf](https://services.google.com/fh/files/misc/2023_final_report_sodr.pdf)
- [3] Roman Burdiuzha, "The Rise of Infrastructure Automation: Streamlining Operations and Boosting Efficiency," Gartech Solutions, 2024. [Online]. Available: <https://gartsolutions.medium.com/the-rise-of-infrastructure-automation-streamlining-operations-and-boosting-efficiency-2a4419400d84>
- [4] Gartner, "Summary Translation: Market Guide for Infrastructure Automation and Orchestration Tools," Gartner Research, 2024. [Online]. Available: <https://www.gartner.com/en/documents/5464295>
- [5] Cindy Blake, "The State of Infrastructure as Code (IaC) 2023," DevOps.com Webinar. [Online]. Available: <https://webinars.devops.com/the-state-of-infrastructure-as-code-iac-2023>
- [6] Chiradeep BasuMallick, "What Is Infrastructure as Code? Meaning, Working, and Benefits," Spiceworks, 2025. [Online]. Available: <https://www.spiceworks.com/tech/cloud/articles/what-is-infrastructure-as-code/>
- [7] Cloud Native Computing Foundation, "CNCf Annual Survey 2023," Cloud Native Computing Foundation, 2023. [Online]. Available: <https://www.cncf.io/reports/cncf-annual-survey-2023/>
- [8] Red Hat, "The State of Enterprise Open Source" 2022. [Online]. Available: <https://www.redhat.com/en/enterprise-open-source-report/2022>
- [9] Red Hat, "Kubernetes adoption, security, and market trends report 2024," 2024. [Online]. Available: <https://www.redhat.com/en/resources/kubernetes-adoption-security-market-trends-overview>
- [10] Portworx, "The Voice of Kubernetes Experts Report 2024," 2024. [Online]. Available: <https://portworx.com/wp-content/uploads/2024/06/The-Voice-of-Kubernetes-Experts-Report-2024.pdf>
- [11] Steve Fenton, "The state of GitOps report: Insights into the adoption and challenges of real-world GitOps," Octopus Deploy. [Online]. Available: <https://octopus.com/publications/state-of-gitops-report>
- [12] Cloud Native Computing Foundation, "CNCf GitOps microsurvey: learning on the job as GitOps goes mainstream," CNCf, 2023. [Online]. Available: <https://www.cncf.io/blog/2023/11/07/cncf-gitops-microsurvey-learning-on-the-job-as-gitops-goes-mainstream/>
- [13] AWS, "Cloud Strategy," Amazon Web Services. [Online]. Available: <https://aws.amazon.com/executive-insights/cloud-strategy/>
- [14] Jessup University, "What is the Future of Cloud Computing in 2024 and Beyond," Jessup University. [Online]. Available: <https://jessup.edu/blog/engineering-technology/what-is-the-future-of-cloud-computing/>