International Journal of Global Tech Management (IJGTM)
Volume 1, Issue 1, January-June, 2024
Available online at: https://pgrpublication.com/index.php/ijgtm

Data Migration Strategies for Cloud-Native Applications: A Framework for Large-Scale Enterprises

Pankaj Dixit

Student, MSc. Computer Science

Article history: Received: 5 February 2024, Accepted: 20 February 2024, Published online: 06 March 2024.

ABSTRACT

Cloud-native computing emerges as a result of the growth of cloud computing delivery paradigms. As the most important online application development paradigm, cloud native computing has already garnered more and more interest from both business and academics. A defined research path on this issue is still lacking, despite the enthusiasm in the cloud-native industrial community. Adoption of state-of-the-art methods, technologies, frameworks, tools, and infrastructure is necessary for the creation of contemporary software applications. Although each category offers a wide range of alternatives, the common features include reduced cost, lightweight and scalable applications, and quicker delivery. Furthermore, testing, deployment, and operational automation have grown in importance. Considering how accessible and reasonably priced data is, cloud computing has a lot to offer. Because users often store sensitive data in cloud storage providers, which may not be trustworthy, ensuring cloud data security is an important consideration in the world of cloud computing. There is now a tendency towards "interclouds," also known as "multi-clouds" or "cloud-of-clouds." The organisation will see significant growth and profitability if it can operate cloud-native apps. The impact of cloud native security on business outcomes is significant. The architecture of cloud-native security is also covered in this article. This article highlights the new methods for creating and implementing Security-as-a-Service (SecaaS) applications leveraging cloud-native design paradigms. The immediate threat to computer systems and applications is not adequately addressed by the most recent SecaaS solutions. This issue is resolved by cloud-native design patterns, which combine micro service architectures with cloud-focused interface design to provide features like substantial optimisation and durability.

Keywords: Cloud Native, Design Patterns, (SecaaS) Applications, Inter clouds, Infrastructure, Delivery Models, Delivery Models, Micro Service, Cutting-Edge Techniques.

INTRODUCTION

Software applications of various sizes are implemented for internal and external usage by organisations across all disciplines. Governments, for instance, employ software tools to facilitate internal administration management for thousands of people [1]. Simultaneously, they may put in place an online software system to allow people and residents to access online services for smart governance. As an additional example, a scientific community may put in place a group scoped system for managing scientific large data while allowing visualisation capabilities for outside parties [1, 2]. Building software programs as a single deployable entity, or the monolithic method, was the prevalent paradigm in software development.

This method was useful in the past since software was often smaller and had fewer features and components [2, 3]. This method was also widely used since it was convenient and simple to design. Furthermore, only a small number of customers with somewhat steady and long-term needs used the systems. A few monolithic application examples [2, 3]. But thanks to the Internet, cell phones, cloud computing, big data, and the start-up environment, things have changed [3]. Response to changing regulations and time to market have become crucial. With hundreds of millions of concurrent users using the same service simultaneously, scalability has reached previously unattainable and necessary levels. Furthermore, [3], this wave has also been impacted by the Internet of Things (IoT), which has led to an exponential rise in the number of devices using internet services.

A revolutionary approach to handling and using large data has been made possible by the rise of cloud computing paradigms [3]. Cloud platforms allow businesses to overcome the constraints of on-premise infrastructure by providing scalable, affordable processing and storage capabilities. Data migration, or the act of moving data assets from on-

premise systems to the cloud environment, is a crucial obstacle in the cloud adoption process. Conventional methods of data transfer are often labour-intensive, manual, and prone to errors [4–5].

These approaches usually include laborious activities including data identification, dependency mapping, transformation, and integration, all of which need for a high level of human involvement. These manual procedures are prone to human mistake, which may result in inconsistent data, security flaws, and delays in the adoption of cloud computing [5]. Furthermore, the full potential of cloud-based big data analytics is hampered by the inability of standard data transfer approaches to scale successfully when working with large datasets [5, 6]. The revolutionary potential of artificial intelligence (AI) in simplifying and improving data transfer for cloud settings is examined in this research study. A robust toolset for automating and improving crucial phases of the data transfer process is provided by artificial intelligence (AI), which includes a wide range of machine learning (ML) methods and methodologies [6]. Organisations may harness the full potential of cloud-based data management and analytics by using AI capabilities to accomplish quicker, more secure, and more efficient cloud migrations [6, 7].

1.1 Cloud Migration Landscape: Challenges and Opportunities

The process of moving IT infrastructure, data, and apps from on-premise data centres to a cloud environment is known as cloud migration [6, 7]. The advantages of cloud computing, like as scalability, agility, cost-effectiveness, [7], and access to cutting-edge services, are made possible by this shift. There are several cloud migration strategies, each with unique benefits and factors to take into account:

- **Lift-and-shift:** With little change, this method entails moving current apps and data "as-is" to the cloud [7, 9]. For applications that are previously virtualised, it is a quick and economical solution, although it may not fully use cloud native architectures.

- **Re-platforming:** Re-architecting programs to use cloud-native capabilities and services is the goal of this strategy [8, 9]. Re-platforming may be more time- and resource-intensive than lift-and-shift, even if it offers better scalability and performance [9, 10].

- **Cloud-native development:** This approach entails creating new apps that are tailored for the cloud environment by using cloud-native concepts such as micro services and containerisation [8, 9]. Although it demands a substantial amount of development work, it provides the best degree of scalability and adaptability.

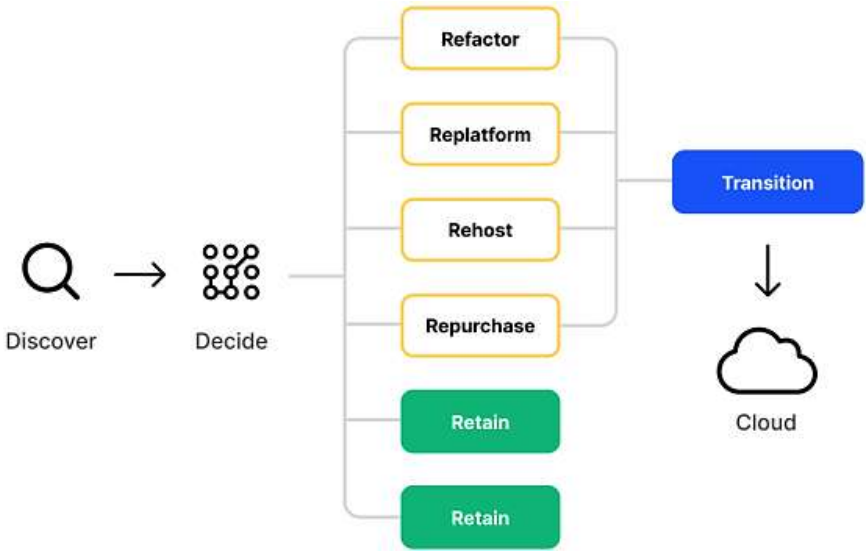

Fig. 1 Cloud Migration Landscape: Challenges and Opportunities. [9, 10]

The open source, vendor-neutral Cloud Native Computing Foundation (CNCF) [9, 10] defines cloud-native computing as a group of technologies that divide applications into micro services and package them in lightweight containers for deployment and orchestration across multiple servers [9, 10]. The following terms are used to describe cloud-native in addition to the micro services architecture:

- **Containerization:** Using the Linux kernel to isolate resources and create containers as distinct processes inside the host operating system, containerisation is a function isolation technique. With a decade of development, Docker [9, 10] is the most widely used containerisation solution. When containerisation and micro services architecture are

combined, every component of an application—processes, libraries, [10], etc.—is bundled into a separate container. This makes resource separation, transparency, and repeatability easier.

- **Orchestration:** The automatic setup, administration, and synchronisation of interconnected micro services to provide scalable and elastic features is known as orchestration. Orchestration reduces to the automation of the operational effort associated with handling the containers' life-cycle, which includes resource provisioning, deployment, scheduling, growing (up and down), social networking, load balancing, etc., in order to execute the applications' workflows or processes. This is because micro services are deployed in the same manner as containers [11]. Kubernetes is the most widely used open-source container orchestration software, and it was inspired by Google's Borg cluster manager.

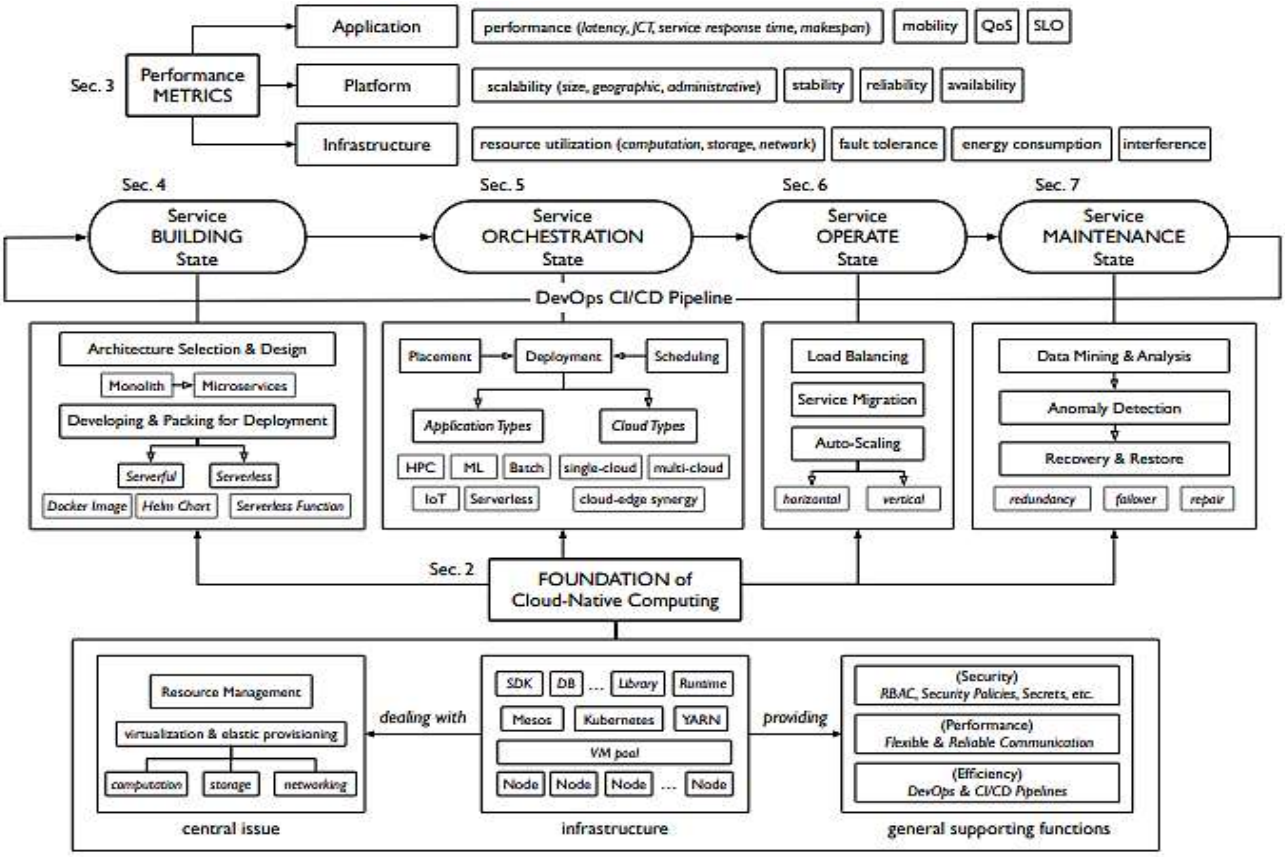

Fig. 2 The cloud-native computing research plan, which was created with services in mind. [11]

In this study, we attempt to examine the history and present of cloud-native apps with regard to the main issues over their life-cycle from a research viewpoint, given that cloud-native is more well-known in the market [11, 12]. We try to combine the popularity of open-source software and platforms, which are extensively used in industry, with the most recent theoretical and systematic research from the standpoint of services computing [12, 13]. Building, orchestration, operation, and maintenance are the four stages of a cloud-native application's life-cycle [13], which is seen as a service in the context of service-oriented computing (SOC). Various phases emphasise distinct important challenges, as seen in Fig. 2 [14]. Furthermore, we gather the performance indicators that are often stated while developing cloud-native apps and examine them from three perspectives: software, platform, and infrastructure. In addition to the performance measures and service life-cycle shown in Fig. 2 [15].

The Concept of Multi-Cloud Computing

Combining two or more different cloud computing services from different cloud suppliers is known as multi-cloud. A multi-cloud area might be all-public, all-private, or a combination of both [16, 17]. Businesses use multi-cloud systems to disperse computational resources and lower the possibility of data loss or disruption [16, 17]. They could also enhance an organization's processing and storage capabilities. Recent advancements in cloud computing have led to a shift from private, single-user clouds to multi-tenant public clouds and community clouds, often referred to as hybrid clouds, which are centralised networks that use many computing environments, including public and private clouds [18]. The terms "inter-cloud" and "fog-of-clouds" are comparable to "multi-cloud."

We can see right now from the fermentation patterns in the IT sector that cloud awareness and adoption will rapidly increase. One improvement is that businesses all over the globe are connecting their own IT departments and traditional storage arrays to one or more public clouds in order to benefit from the distinct and obvious benefits of the cloud paradigm [18, 19]. As a result, the business model Cloud primarily defines the intimate relationship that is created and maintained between new IT and traditional IT. Hybrid IT is the name given to this dimension [19, 20]. The agility, adaptability, and inventiveness required to advance any business are provided by this novel and integrated operations approach. A hybrid IT strategy enables IT to accomplish significant business goals:

- Boost market shares, customer happiness, trust, and engagement.
- Create new, distinct, and productive production domains [20].
- Increasing risk and reducing manufacturing costs via increased performance and speed.

Another exciting development is the hybrid cloud, which combines private clouds with one or more public clouds [20]. There are certain technically solid advantages to this contemporary tendency [20, 21].

- **Challenges in Multi-cloud computing:** The terms "multi-clouds" and "inter-clouds" or "cloud-of-clouds" are synonymous. Typically, cloud computing computations are not intended to terminate in a single cloud [21]. According to their illustration, a foggy sky blends different cloud hues and patterns, resulting in particular operational and implementation jobs. Recent research has concentrated on the multi-cloud environment, which manages many clouds and avoids excessive dependence on almost any cloud.

- **Data Governance and Compliance:** Excellent flexibility and conformity may be obtained in many global capitals via a variety of clouds and cloud service conditions, albeit this will not happen immediately [21, 22]. The primary challenge is identifying the realistic location of the data; small and mid-sized enterprises may find this issue more difficult. If the multi-cloud environment is usable, it might be easy to make errors and attempt to execute an implementation in an unauthorised environment.

- **Security issues:** Numerous security issues with cloud computing have been noted by experts and professionals in the field, including project assurance, data disclosure and privacy, virtual OS protection, separation management, confidence and enforcement, and [23].8. New security issues are surfaced as a result of decentralised administration and interchange across many clouds. Particular concerns in multi-cloud computing setups include those of privacy, legislation, and trust. Stabilising coordination and settling confidence. As with other IT architectures, building trusting connections between the people involved is crucial to cloud protection. When a client gives up the cloud services provider's (CSP) exclusive confidentiality and property protection, trust is required.

- **Multi-cloud Brokering:** In the era of connected and federalised clouds, broking goods and technologies are crucial [23]. Connectivity, facilitation, and other improvement and facilitation functions are carried out by these cloud service providers. There are several adapters, connectors, operators, and other options available to establish an instantaneous connection between public and private clouds. Bridge options are intended to provide superior connectivity between public clouds [23].

- **Policy heterogeneity and conflicts:** When proxies enable active cooperation between many CSPs, security risks may arise due to disparate security standards. VPs need to assess these infractions and take precautions. Security breaches may occur simply via integration, even though established regulatory review systems can validate personal domain projects. Through the use of proxy servers, services may continue to drive dynamic, transient, [23], and expensive contact between various application components in multi-cloud collaborative activities.

- **Distributed Denial of Service (DDoS):** In the cloud context, Distributed Denial of Service (DDoS) is a well-known security issue that might have serious consequences for decentralised clouds [26]. Attackers and malicious users seek to take advantage of flaws in cloud infrastructure supported by virtualisation software, provisioning procedures, and multi-tenancy. Other customers often get greater financial benefit with less optimal architecture since the attack frees them from paying for finances and processing power on the cloud [26].

- **Monitoring:** It is challenging to identify security and performance problems while evaluating many clouds. You may avoid this issue by finding a reliable cloud monitoring program. Numerous more clouds may be supported by numerous APM solutions. This provides the organisation with a wide range of choices for selecting the appropriate measuring system. However, [22] effective organisational performance for multi-clouds requires that the tool recognise how the cloud's volume of work operates.

International Journal of Global Tech Management (IJGTM)
Volume 1, Issue 1, January-June, 2024
Available online at: https://pgrpublication.com/index.php/ijgtm

CLOUD-NATIVE

"Web-native" computing is a methodology for designing and running applications that integrates the advantages of the computing delivery method. It makes full use of a number of contemporary technologies, including micro-services, PaaS, agile methodology, multi-coloured containers, CI/CD, and DevOps. "Cloud-native" refers to how, not where, applications are created and deployed. In contrast to an on-premises data centre, it shows that the apps are available anywhere in the public cloud [22, 26]. Developing cloud-based apps specifically for the cloud is a matter of fundamental decency [23]. However, there isn't a precise definition that explains what it entails [24].

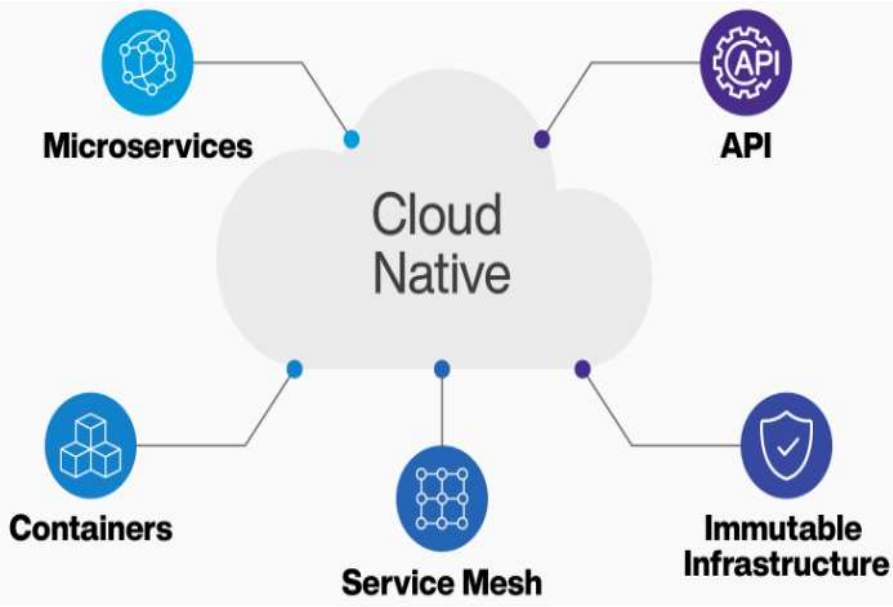

Fig. 3 Cloud Native. [23]

- **Developed with best-of-breed languages and frameworks:** The programming language used to create each cloud-native application service is specifically tailored to the application. A variety of terms, third-party tools, and frameworks are used by service providers in cloud implementations, which are polyglot [23]. For example, developers may use Node.js to create a Web Socket-based real-time streaming service and use Flask and Python to expose the API. They may choose the optimal framework for a given task by attempting to design micro service architecture in a granular manner [25].

- **Designed as loosely coupled micro services:** During the implementation's runtime, services from the same implementation discover one another. Regardless of other services, they do exist. Stretchy software architectures and infrastructure may be accurately tuned out for increased productivity when completely implemented [24].

- **Isolated from the server and operating system dependencies:** Applications that are cloud-native are not dependent on any particular machine or operating system. They operate at a higher abstraction level [26]. The one exception is when a micro service needs additional hardware, such as graphics processing units (GPUs) and solid-state drives (SSDs), which a collection of computers may provide [19, 22].

- **Managed through agile DevOps processes:** An adaptive DevOps approach manages the individual development cycle that connects cloud-native technologies [26]. A cloud-based application may be installed and managed using a variety of configuration management/continuous deployment pipelines (CI/CD).

- **Automated capabilities:** Cloud-based applications may be completely automated. As code, they align nicely with the technological principle [26]. However, a significant degree of optimisation is necessary to handle such large and complex systems [26, 27].

- **Cloud-Native architecture:** When dispersed systems are functioning in the cloud, the redundant duplication problem may be resolved in a number of ways. In general, because the cloud service provider controls such resources, we are unable to control redundancy in the cloud service. As a result, we may modify the way we replicate the application stage [27].

Cloud-Native Design Patterns to Security-As-A-Service Applications

- **Cloud-Native Architectures Design Principles and Motivations for Migration:** The advantages that the cloud offers are usually what motivate the use of cloud technology. However, some implementations make use of techniques specifically created for conventional data centres [27, 28]. Bundling application hosting stacks into cloud virtual machines (VMs), such as a three-tiered web application, is a typical example. The software is rigid in terms of capacity, and scaling is possible but limited by certain factors, such as the virtual machine's RAM capacity [27, 28]. On the other hand, the application may grow horizontally, but this is expensive since each new case requires VM replication. Even while these strategies were popular, companies adopted more contemporary ones, such as Netflix, which is more effective.

- **Use Case of a Cloud Native SecaaS - Security Integration in DevOps:** The goal of the DevOps technique is to improve collaboration between the operations and enterprise development teams [28]. In essence, DevOps generates energy for shorter software development stages, which results in better coordination and collaborative decision-making across various groups.

- **Security requirements for migration to CNA and CAVAS system model:** We have used CAVAS as the objective framework to investigate SecaaS migrations for CNA. In order to conduct security tests for Dynamic User Acceptance Testing (DAST), CAVAS was first developed as a monolithic framework [28]. It was created for earlier studies on cloud risk analysis. CANVAS was selected as the source application for this task due to our present proficiency and working knowledge of the program [29, 30].

- **Justification for Migration to CNA and Redesign Efforts:** Because CAVAS needed to successfully handle intermittent spikes in scan requests, efficient auto-scaling was a necessary [30, 31]. Horizontal auto-scaling of the overloaded modules is a workable solution for handling overload demands. However, monolithic systems are now semi-fault tolerant and inflexible, and they are often scaled up vertically [32]. Conversely, CNA offers opportunities for horizontal scalability, each web service.

- **System Model: CAVAS can be implemented in 2 ways:** either as an auto-testing process or as an integrated software pipeline module. In a later style, a user supplies the hostname or IP address of a selected tool for vulnerability scanning. Through the API Gateway [32], the test request is sent to the UI Server, where the user has the necessary authorisation. Information is added to the real-time process if verification of the intended resource ownership is needed.

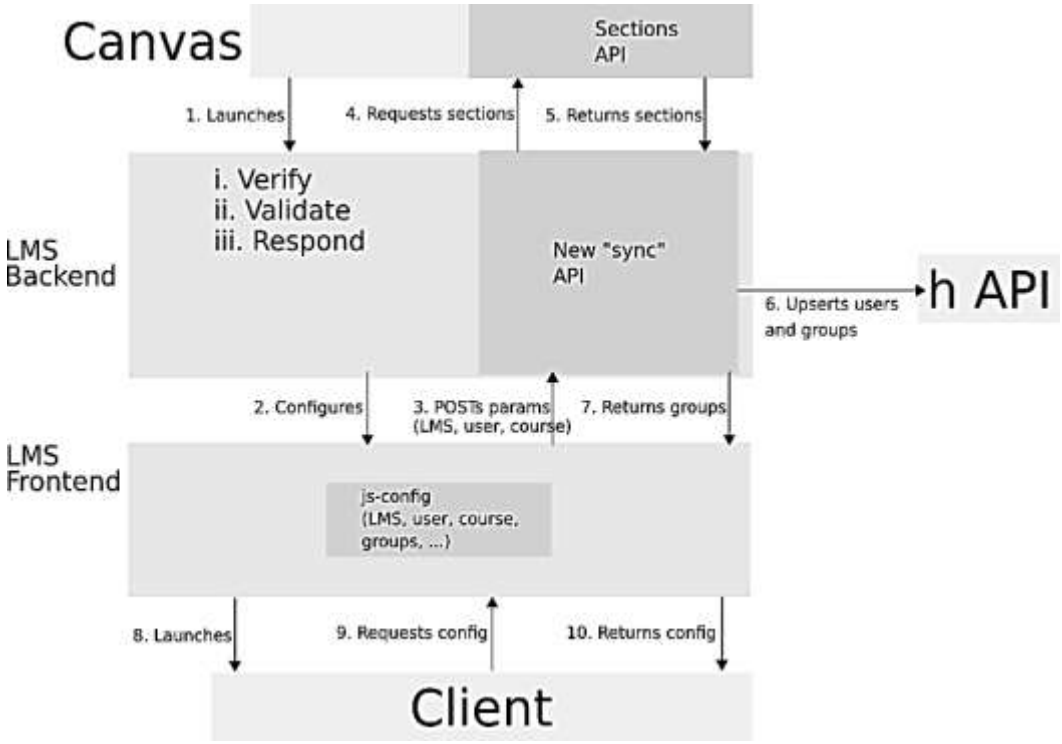

Fig. 4 High-level architecture of CAVAS. [32]

Using the whole spectrum of cloud computing, such as isolation, an approach component or cloudy transforms a framework into the design of microservices; parallelisation and permeability may be carried out in containers [32, 33]. It is necessary to completely rethink access and construct applications from the beginning in order to make the decision to utilise the business cloud platform. When the program is developed in the optimisation stage transitioned for threat reduction, institutions should benefit from the lowest risk re-hosting techniques, [33], re-platform, repurchase, and process restoration refactor. One definition of cloud transformation is the process of sharing an organization's digital properties, goods, IT assets, or cloud applications, either expressly or implicitly [34, 35]. Supply interruptions will occur as a result of the cloud computing platform migration. Therefore, it is necessary to use well-developed methods for both migration legislation and decision-making.

CONCLUSION

As the most prominent concept for online applications, cloud-native has drawn an increasing number of firms and academics to investigate and use it. Through a clear and efficient categorisation, this survey aims to provide potential study options. The revolutionary potential of artificial intelligence (AI) in transforming data transfer procedures has been examined in this research. AI enables businesses to automate data discovery, transformation, and cleaning processes by using unsupervised learning techniques. This drastically cuts down on the time and resources needed in comparison to manual approaches.

The size of data, regulatory concerns, business application cloud readiness, leisure time costs and SLA requirements, and the movement of data and applications in the event of cloud provider transformation should all be considered when creating customised cloud solutions and integration plans for cloud adoption. To put it another way, companies that want to migrate to the cloud need to have a unified strategy, application architecture, and governance model. A cloud-native framework of the highest grade is presented in this study along with a quick introduction to cloud computing. Even though the use of cloud computing has grown rapidly, cloud computing security is still considered to be the biggest issue in the field.

Automating CNA migrations with suitable pre-migration security evaluations is an intriguing area for future research. To deploy the cloud, a tailored cloud strategy and migration plan should be in place, accounting for factors including data volume, regulatory issues, business application cloud readiness, and downtime costs. An overview of cloud computing and cloud-native platforms at the highest level is given in this article. Prominent providers of corporate technology have constructed cloud infrastructures, provide their products the finest migration techniques and solutions, and provide extra assistance to outside suppliers.

REFERENCES

[1]. VK Kamboj, A Bhardwaj, HS Bhullar, K Arora, K Kaur, Mathematical model of reliability assessment for generation system, Power Engineering and Optimization Conference (PEOCO) Melaka, Malaysia, 2012 IEEE.
[2]. Sharma, P. K., Ryu, J. H., Park, K. Y., Park, J. H., & Park, J. H. (2020). Li-Fi based on security cloud framework for future IT environment. Human-centric Computing and Information Sciences, 10(1), 1-13.
[3]. Wang, T., Li, P., Wang, X., Wang, Y., & Guo, S. (2019). A comprehensive survey on mobile data offloading in heterogeneous network. Wireless Networks, 25(2), 573-584.
[4]. EA Bhardwaj, RK Sharma, EA Bhadoria, A Case Study of Various Constraints Affecting Unit Commitment in Power System Planning, International Journal of Enhanced Research in Science Technology & Engineering, 2013.
[5]. Wang, Y., Shi, W., & Zhang, N. (2018). Energy-efficient task offloading in mobile edge computing: A learning-based approach. IEEE Journal on Selected Areas in Communications, 37(6), 1293-1306.
[6]. Patel, M., Parikh, H., & Dave, G. (2023). Chitosan flakes-mediated diatom harvesting from natural water sources. Water Science & Technology, 87(7), 1732-1746.
[7]. Chen, Y., Wang, T., & Zhang, K. (2021). Adaptive data aggregation for energy-efficient wireless sensor networks. IEEE Transactions on Wireless Communications, 20(3), 1540-1553.
[8]. Gupta, A., Christie, R., & Manjula, P. R. (2017). Scalability in Internet of Things: Features, techniques and research challenges. International Journal of Computational Intelligence Research, 13(7), 1617-1627.
[9]. Vivek Singh, Neha Yadav. (2023). Optimizing Resource Allocation in Containerized Environments with AI-driven Performance Engineering. International Journal of Research Radicals in Multidisciplinary Fields, ISSN: 2960-043X, 2(2), 58–69. Retrieved from https://www.researchradicals.com/index.php/rr/article/view/83

[10]. Guo, J., Song, Z., Cui, Y., Liu, Z., & Ji, Y. (2018). Energy-efficient resource allocation for multi-user mobile edge computing. IEEE Global Communications Conference (GLOBECOM), 1-7.

[11]. Bhardwaj, A., Kamboj, V. K., Shukla, V. K., Singh, B., & Khurana, P. (2012, June). Unit commitment in electrical power system-a literature review. In Power Engineering and Optimization Conference (PEOCO) Melaka, Malaysia, 2012 IEEE International (pp. 275-280). IEEE.

[12]. Patel, D., Narmawala, Z., Tanwar, S., & Kumar, N. (2020). A systematic review on scheduling and load balancing techniques in cloud computing. Sustainable Computing: Informatics and Systems, 27, 100385.

[13]. Raza, U., Kulkarni, P., &Sooriyabandara, M. (2020). Low power wide area networks: An overview. IEEE Communications Surveys & Tutorials, 19(2), 855-873.

[14]. Neha Yadav, Vivek Singh, "Probabilistic Modeling of Workload Patterns for Capacity Planning in Data Center Environments" (2022). International Journal of Business Management and Visuals, ISSN: 3006-2705, 5(1), 42-48. https://ijbmv.com/index.php/home/article/view/73

[15]. Parikh, H., Patel, M., Patel, H., & Dave, G. (2023). Assessing diatom distribution in Cambay Basin, Western Arabian Sea: impacts of oil spillage and chemical variables. Environmental Monitoring and Assessment, 195(8), 993

[16]. Parikh, H. (2021). Diatom Biosilica as a source of Nanomaterials. International Journal of All Research Education and Scientific Methods (IJARESM), 9(11).

[17]. Kumar, P., Gurtov, A., Sain, M., Martin, A., & Ha, P. H. (2018). Lightweight data compression in wireless sensor networks: A survey. Journal of Network and Computer Applications, 116, 37-59.

[18]. Li, S., Xu, L. D., & Zhao, S. (2019). 5G Internet of Things: A survey. Journal of Industrial Information Integration, 10, 1-9.

[19]. Amit Bharadwaj, Vikram Kumar Kamboj, Dynamic programming approach in power system unit commitment, International Journal of Advanced Research and Technology, Issue 2, 2012.

[20]. Vivek Singh, Neha Yadav,"Deep Learning Techniques for Predicting System Performance Degradation and Proactive Mitigation" (2024). International Journal of Open Publication and Exploration, ISSN: 3006-2853, 12(1), 14-21. https://ijope.com/index.php/home/article/view/136