

Kubernetes and Cloud Computing

SEMINAR REPORT

SUBMITTED IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATIONS ENGINEERING

SUBMITTED BY:

KRISHNA KAPOOR

ROLL NO. UE215051

SEVENTH SEMESTER



DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

UNIVERSITY INSTITUTE OF ENGINEERING AND TECHNOLOGY,

PANJAB UNIVERSITY, CHANDIGARH, INDIA

DEC 2024

CANDIDATE’S DECLARATION

I hereby certify that the work presented in the Seminar entitled “Kubernetes and Cloud Computing” in partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Electronics and Communications Engineering from the University Institute of Engineering and Technology, Panjab University, Chandigarh, is an authentic record of my work carried out under the supervision and guidance of Dr. Garima Joshi.

Date: 03-12-2024

Chandigarh

(Krishna Kapoor)

(UE215051)

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief .

Date: 03-12-2024

Chandigarh

(Dr. Garima Joshi)

ECE, UIET, Panjab University

ABSTRACT

Kubernetes, an open-source container orchestration platform, has emerged as a pivotal technology in the realm of cloud computing, significantly reshaping how applications are deployed, managed, and scaled across diverse computing environments. It provides an automated system for containerized application deployment, scaling, and management, making it essential for modern cloud-native architectures. In combination with cloud computing, Kubernetes enables businesses to unlock the full potential of cloud resources—whether on public, private, or hybrid clouds—by facilitating efficient, reliable, and scalable infrastructure solutions. Cloud computing, characterized by its ability to offer on-demand access to a pool of computing resources like servers, storage, and networking, complements Kubernetes in a symbiotic relationship that enhances the flexibility, cost-efficiency, and scalability of services. With cloud infrastructure providing the underlying resources and Kubernetes managing the applications at scale, organizations are equipped to handle dynamic workloads, ensuring high availability and fault tolerance while minimizing operational overhead. This report explores how Kubernetes and cloud computing together revolutionize application development and infrastructure management, focusing on their impact on the IT landscape. A key feature of Kubernetes is its use of containerization, which isolates applications in lightweight, portable containers that can run consistently across different environments. This eliminates traditional barriers associated with application portability and environment configuration, enabling applications to be easily moved between on-premises data centers, public clouds, and hybrid cloud setups. Kubernetes' ability to abstract away the underlying hardware allows developers to focus on building applications without worrying about the infrastructure details, thereby accelerating development cycles and enhancing productivity. This report further delves into Kubernetes' essential components, such as Pods, Deployments, and Services, and their roles in managing containers effectively. It highlights how Kubernetes leverages these components to provide features like auto-scaling, self-healing (automatically replacing failed containers), and rolling updates, ensuring that applications are always available and resilient to failures. In cloud computing, one of the main benefits Kubernetes offers is its ability to seamlessly scale applications. The dynamic nature of Kubernetes also allows for optimized resource utilization, as workloads are distributed across the cloud infrastructure based on demand and resource availability.

Acknowledgment

I would like to express my deep gratitude to Faculty-in-charge Dr. Garima Joshi for their patient guidance, enthusiastic encouragement, and useful critiques during the completion of my Seminar named “Kubernetes and Cloud Computing”.

I would also like to extend my thanks to the University Institute of Engineering & Technology (UIET), Panjab University for giving me such an opportunity to develop a project in Embedded Systems.

Further, I want to thank my parents for supporting me during my project and helping me procure the required resources for the fulfillment of my project and report.

Contents

CHAPTER 1	6
<i>Understanding Cloud Computing</i>	6
<i>Types of Cloud Computing</i>	8
Private cloud.....	8
Public cloud	9
<i>Key Benefits of Cloud Computing</i>	11
<i>Challenges of cloud computing</i>	12
CHAPTER 2	14
Kubernetes.....	14
<i>Kubernetes Architecture</i>	14
<i>Kubernetes and Cloud Computing: A Synergistic Relationship</i>	16
<i>Benefits of Kubernetes in Cloud Computing Environments</i>	17
<i>Kubernetes on Leading Cloud Platforms</i>	18
<i>Conclusion</i>	19
The future Scope	20
CONCLUSION.....	22
REFERENCES	23

CHAPTER 1

CLOUD COMPUTING

The rise of cloud computing has significantly reshaped the way businesses deploy and manage applications. One of the most revolutionary technologies that has emerged in the cloud-native ecosystem is **Kubernetes**, an open-source container orchestration platform. Kubernetes has become a fundamental tool for automating the deployment, scaling, and management of containerized applications. This report explores the interplay between Kubernetes and cloud computing, examining their respective roles, benefits, and how they work together to drive modern application infrastructure.

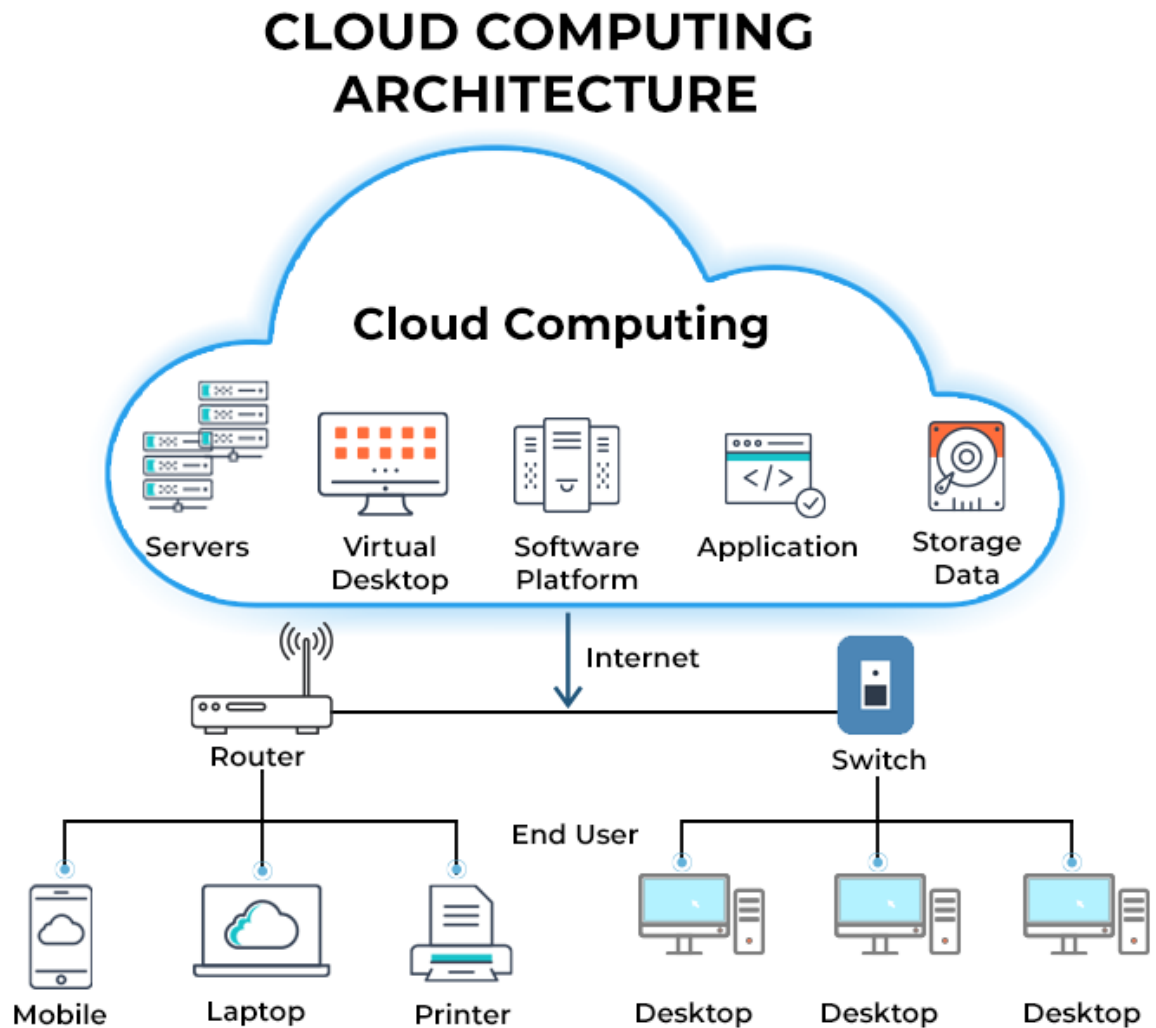
Understanding Cloud Computing

Cloud computing refers to the delivery of computing services—such as storage, processing power, databases, networking, software, and analytics—over the internet. Rather than owning and maintaining physical data centers, businesses can rent these services on-demand from cloud providers like Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).

Key characteristics of cloud computing include :

1. **Scalability:** The ability to scale resources up or down based on demand.
2. **Flexibility:** Cloud services offer a wide range of services that can be customized to meet specific needs.
3. **Cost Efficiency:** Pay-as-you-go models allow organizations to optimize their expenditure by only paying for what they use.
4. **Reliability:** High availability and disaster recovery options are typically built into cloud services.

5. **Global Reach:** Cloud providers offer data centers worldwide, enabling low-latency access and global service delivery.



Instead of storing files on a storage device or hard drive, a user can save them on cloud, making it possible to access the files from anywhere, as long as

they have access to the web. The services hosted on cloud can be broadly divided into infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), and software-as-a-service (SaaS). Based on the deployment model, cloud can also be classified as public, private, and hybrid cloud.

Further, cloud can be divided into two different layers, namely, front-end and back-end. The layer with which users interact is called the front-end layer. This layer enables a user to access the data that has been stored in cloud through cloud computing software.

The layer made up of software and hardware, i.e., the computers, servers, central servers, and databases, is the back-end layer. This layer is the primary component of cloud and is entirely responsible for storing information securely. To ensure seamless connectivity between devices linked via cloud computing, the central servers use a software called middleware that acts as a bridge between the database and applications.

Types of Cloud Computing

Cloud Computing can be classified based on the deployment or the type of services offered.

Private cloud

In a private cloud, the computing services are offered over a private IT network for the dedicated use of a single organization, usually managed via internal resources and is not accessible to, anyone outside the organization. Private cloud computing provides all the benefits of a public

cloud, such as self-service, scalability, and elasticity, along with additional control, security, and customization.

Private clouds provide a higher level of security through company firewalls and internal hosting to ensure that an organization's sensitive data is not accessible to third-party providers. The drawback of private cloud, however, is that the organization becomes responsible for all the management and maintenance of the data centers, which can prove to be quite resource-intensive.

Public cloud

Public cloud refers to computing services offered by third-party providers over the internet. Unlike private cloud, the services on public cloud are available to anyone who wants to use or purchase them. These services could be free or sold on-demand, where users only have to pay per usage for the CPU cycles, storage, or bandwidth they consume.

Public clouds can help businesses save on purchasing, managing, and maintaining on-premises infrastructure since the cloud service provider is responsible for managing the system. They also offer scalable RAM and flexible bandwidth, making it easier for businesses to scale their storage needs.

Hybrid cloud

Hybrid cloud uses a combination of public and private cloud features. The “best of both worlds” cloud model allows a shift of workloads between private and public clouds as the computing and cost requirements change. When the demand for computing and processing fluctuates, hybrid cloud opens a new window allows businesses to scale their on-premises infrastructure up to the public cloud to handle the overflow while ensuring that no third-party data centers have access to their data. In a hybrid cloud model, companies only pay for the resources they use temporarily instead of purchasing and maintaining resources that may not be used for an extended period. In short, a hybrid cloud offers the benefits of a public cloud without its security risks.

Based on the service model, cloud can be categorized into IaaS (Infrastructure-as-a-Service), PaaS (Platform-as-a-Service), and SaaS (Software-as-a-Service). Let's take a look at each one.

Infrastructure as a service (IaaS)

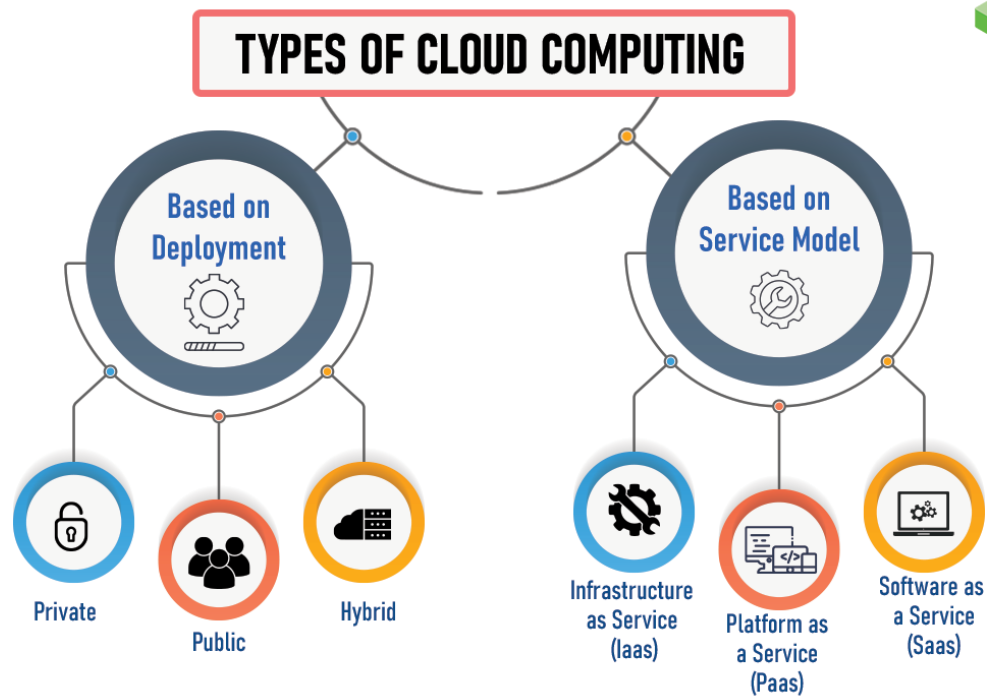
Infrastructure as a service or IaaS is a type of cloud computing in which a service provider is responsible for providing servers, storage, and networking over a virtual interface. In this service, the user doesn't need to manage the cloud infrastructure but has control over the storage, operating systems, and deployed applications. Instead of the user, a third-party vendor hosts the hardware, software, servers, storage, and other infrastructure components. The vendor also hosts the user's applications and maintains a backup.

Platform as a service (PaaS)

Platform as a service or PaaS is a type of cloud computing that provides a development and deployment environment in cloud that allows users to develop and run applications without the complexity of building or maintaining the infrastructure. It provides users with resources to develop cloud-based applications. In this type of service, a user purchases the resources from a vendor on a pay-as-you-go basis and can access them over a secure connection. PaaS doesn't require users to manage the underlying infrastructure, i.e., the network, servers, operating systems, or storage, but gives them control over the deployed applications. This allows organizations to focus on the deployment and management of their applications by freeing them of the responsibility of software maintenance, planning, and resource procurement.

Software as a service (SaaS)

SaaS or software as a service allows users to access a vendor's software on cloud on a subscription basis. In this type of cloud computing, users don't need to install or download applications on their local devices. Instead, the applications are located on a remote cloud network that can be directly accessed through the web or an API.



Key Benefits of Cloud Computing

Let's look at all the benefits cloud offers:

1. **Reduced costs:** Maintaining IT systems requires big outlays of capital, something that cloud helps reduce. By using the resources provided by the cloud provider, businesses avoid the need to purchase expensive infrastructure, substantially reducing their expenditure. Cloud providers work on the pay-as-you-go model, which means businesses only pay for the services they use, further reducing costs.

2. **Scalability:** Cloud allows organizations to grow their users from merely a few to thousands in a very short time. Depending on the need, a business can scale their storage needs up or down, allowing organizations to be flexible.
3. **Flexibility and collaboration:** Since the data on cloud can be accessed directly via the internet, it gives employees the ability to work from anywhere, anytime. Cloud gives you the freedom to set up your virtual office anywhere you are. It also allows teams to work on a project across locations by giving them access to the same files as third-party vendors.
4. **Business continuity:** Cloud safely stores and protects your data in the event of an outage or crisis. This makes it easier to resume work once the systems are up and running again
5. **Competitive edge:** Cloud takes care of various business aspects, such as maintaining the IT infrastructure, licensing software, or training personnel to manage your data. It, therefore, gives you an edge over your competitors since the time and resources you invest are minimal.

Challenges of cloud computing.

The biggest challenge of cloud computing is the security concerns associated with the technology. Although cloud service providers assure you of implementing the best security standards and industry certifications, there's always a risk while storing your data on cloud.

1. **Downtime:** Almost every cloud user will tell you that outages tops their list of cloud computing challenges. At times, cloud service providers may get overwhelmed due to the huge number of clients they provide services to each day. This may lead to technical outages, due to which your applications may temporarily experience some downtime.

2. **Internet connection dependency:** A user may not be able to access the data on cloud without a good internet connection and a compatible device. Moreover, using public Wi-Fi to access your files could pose a threat if the right security measures are not taken.
3. **Financial commitment:** Cloud providers use a pay-as-you-go pricing model. However, businesses need to give a monthly or annual financial commitment for most subscription plans. This needs to be factored into their operating costs.
4. **Security risks:** Even if your cloud service provider promises you that they have the most reliable security certifications, there's always a chance of losing your data. With hackers increasingly targeting cloud storage to gain access to sensitive business data, this might be an even greater concern, for which the appropriate measures need to be taken.
5. **Limited access:** A user may have minimal control since the cloud service provider owns and manages the infrastructure. The user would only be able to manage applications and not the backend infrastructure. Crucial tasks, such as firmware management, may not be passed to the user at all. You always have to trust a third-party vendor to ensure security and take care of your data.

CHAPTER 2

Kubernetes

Kubernetes, often abbreviated as K8s, is a container orchestration platform designed to automate the deployment, scaling, and management of containerized applications. Initially developed by Google, Kubernetes has become the industry-standard tool for managing containerized workloads and services.

Containers are lightweight, portable environments for running applications, and they ensure that applications run consistently across different computing environments. Kubernetes enhances containers by managing their lifecycle, making it easier to deploy, scale, and maintain applications in a microservices-based architecture.

Key components of Kubernetes include:

1. **Pods:** The smallest deployable units, which contain one or more containers.
2. **Nodes:** Physical or virtual machines that run the Kubernetes cluster and host the pods.
3. **Control Plane:** The brain of Kubernetes, responsible for managing the cluster and ensuring its desired state.
4. **Deployments:** Objects that define how many replicas of a pod should be running at any time.
5. **Services:** Abstracts access to a set of pods, providing a stable endpoint for communication.
6. **Namespaces:** A way to divide cluster resources between multiple users or applications.

Kubernetes Architecture

A Kubernetes cluster consists of a control plane plus a set of worker machines, called nodes, that run containerized applications. Every cluster needs at least one worker node in order to run Pods.

The worker node(s) host the Pods that are the components of the application workload. The control plane manages the worker nodes and the Pods in the cluster. In production environments, the control plane usually runs across multiple computers and a cluster usually runs multiple nodes, providing fault-tolerance and high availability.

1. Control plane components

The control plane's components make global decisions about the cluster (for example, scheduling), as well as detecting and responding to cluster events (for example, starting up a new pod when a Deployment's replicas field is unsatisfied). Control plane components can be run on any machine in the cluster. However, for simplicity, setup scripts typically start all control plane components on the same machine, and do not run user containers on this machine..

2. kube-apiserver

The API server is a component of the Kubernetes control plane that exposes the Kubernetes API. The API server is the front end for the Kubernetes control plane. The main implementation of a Kubernetes API server is kube-apiserver. kube-apiserver is designed to scale horizontally—that is, it scales by deploying more instances. You can run several instances of kube-apiserver and balance traffic between those instances.

3. etcd

Consistent and highly-available key value store used as Kubernetes' backing store for all cluster data. If your Kubernetes cluster uses etcd as its backing store, make sure you have a **back up** plan for the data.

4. kube-scheduler

Control plane component that watches for newly created Pods with no assigned node, and selects a node for them to run on. Factors taken into account for scheduling decisions include: individual and collective resource requirements, hardware/software/policy constraints, affinity and anti-affinity specifications, data locality, inter-workload interference, and deadlines.

5. kube-controller-manager

Control plane component that runs controller processes. Logically, each controller is a separate process, but to reduce complexity, they are all compiled into a single binary and run in a single process.

There are many different types of controllers. Some examples of them are:

- Node controller: Responsible for noticing and responding when nodes go down.
- Job controller: Watches for Job objects that represent one-off tasks, then creates Pods to run those tasks to completion.
- EndpointSlice controller: Populates EndpointSlice objects (to provide a link between Services and Pods).
- ServiceAccount controller: Create default ServiceAccounts for new namespaces.

Kubernetes and Cloud Computing: A Synergistic Relationship

While cloud computing provides the infrastructure to run applications, Kubernetes provides the platform to manage and orchestrate those applications. Together, they form the backbone of modern, scalable, and highly available cloud-native application architectures.

1. Scalability and Resource Management

Kubernetes automates the scaling of applications based on demand. This aligns perfectly with the scalability inherent in cloud computing, where resources can be adjusted dynamically based on real-time needs. For example, when an application experiences a traffic spike, Kubernetes can automatically scale out the number of pods, and cloud providers can allocate additional resources like CPU, memory, or storage.

2. Microservices Architecture

Kubernetes thrives in environments where applications are broken down into microservices. Cloud providers, with their flexible infrastructure and high availability, complement Kubernetes

by enabling the deployment of these microservices across various regions. The combination of Kubernetes and cloud computing offers a highly resilient and fault-tolerant system, where services can automatically reroute traffic if one region or microservice fails.

3. Cost Efficiency and Resource Optimization

Cloud platforms allow users to pay only for the resources they use, while Kubernetes ensures that applications only consume the resources they need. Kubernetes automatically manages resources by scaling workloads up and down, thus optimizing the use of compute and storage resources, which can help businesses minimize costs.

4. High Availability and Disaster Recovery

Cloud providers typically offer multiple availability zones (AZs) to ensure high availability. Kubernetes can be configured to distribute applications across these zones, ensuring that if one AZ fails, the application continues to run in another. Kubernetes' self-healing capabilities, such as auto-restarting failed pods and rescheduling them, complement the cloud's infrastructure, providing continuous service uptime.

5. Continuous Integration and Continuous Deployment (CI/CD)

Cloud computing platforms and Kubernetes both play an essential role in modern CI/CD workflows. Cloud providers offer tools to build, test, and deploy applications, while Kubernetes facilitates the seamless deployment of microservices. Kubernetes ensures that updates are deployed efficiently, minimizing downtime and promoting agile software development practices. Cloud computing also provides integration with CI/CD tools like Jenkins, GitLab, and CircleCI, automating testing and deployment pipelines.

Benefits of Kubernetes in Cloud Computing Environments

1. **Platform Independence:** Kubernetes abstracts away the underlying cloud infrastructure, allowing applications to run consistently across different cloud providers (AWS, Azure,

GCP) or even on-premises environments. This provides a layer of flexibility that enables hybrid cloud deployments.

2. **Simplified Management:** Kubernetes automates routine operational tasks such as load balancing, self-healing, rolling updates, and service discovery. This reduces the complexity of managing large-scale applications and improves the efficiency of operations teams.
3. **Developer Productivity:** Kubernetes provides a unified platform for developers to deploy and manage applications, enhancing collaboration between development and operations (DevOps). Kubernetes supports declarative configuration, meaning developers can define the desired state of their application, and Kubernetes will ensure it's achieved and maintained.
4. **Security:** Kubernetes enables secure application deployment by offering features such as role-based access control (RBAC), network policies, and secrets management. Cloud providers also offer integrated security features, like encryption, firewalls, and DDoS protection, which work in conjunction with Kubernetes' security mechanisms.

Kubernetes on Leading Cloud Platforms

Most major cloud platforms offer fully managed Kubernetes services, which abstract away the complexity of setting up and managing a Kubernetes cluster. These services allow users to focus on application development rather than infrastructure management:

- **Amazon EKS (Elastic Kubernetes Service):** AWS offers a managed Kubernetes service that integrates seamlessly with its ecosystem, including services like RDS, S3, and IAM for identity management.
- **Google Kubernetes Engine (GKE):** Google, the original creator of Kubernetes, offers GKE, which is tightly integrated with Google Cloud's storage, networking, and analytics services.
- **Azure Kubernetes Service (AKS):** Microsoft's AKS simplifies Kubernetes management and integrates with Azure's ecosystem, including Azure Active Directory, security, and monitoring services.

These managed services provide built-in scaling, security, and monitoring, further streamlining the process of managing Kubernetes clusters on the cloud.

Conclusion

Kubernetes and cloud computing have become essential components of modern IT infrastructure. Kubernetes enables the efficient management of containerized applications, while cloud computing offers scalable and flexible resources to run these applications. Together, they form a powerful foundation for building, deploying, and managing cloud-native applications, driving efficiency, cost savings, and resilience. As businesses continue to adopt microservices and containerized architectures, the synergy between Kubernetes and cloud computing will only become more pronounced, unlocking new possibilities for digital transformation in the years ahead.

The future Scope

The future of Kubernetes and cloud computing is poised to be deeply intertwined, as both technologies continue to evolve to meet the increasing demands for scalability, flexibility, and efficiency in IT operations. The combination of Kubernetes' container orchestration capabilities and the inherent benefits of cloud computing has transformed how organizations design, deploy, and manage applications. As enterprises move towards cloud-native architectures, the future scope of Kubernetes and cloud computing will be shaped by several key trends and technological advancements.

1. Expansion of Multi-Cloud and Hybrid Cloud Architectures

One of the most significant trends in the cloud computing space is the shift towards multi-cloud and hybrid cloud environments. Organizations are increasingly seeking to avoid vendor lock-in, optimize resource usage, and ensure higher availability by distributing their workloads across multiple cloud providers (public, private, and on-premise). Kubernetes plays a pivotal role in facilitating these multi-cloud and hybrid strategies by providing a consistent platform for managing applications across diverse environments. In the future, Kubernetes will become even more adept at managing applications across multiple cloud providers and hybrid infrastructures. Cloud-native technologies like service meshes, distributed storage solutions, and advanced networking capabilities will make it easier for organizations to manage and scale applications seamlessly across various cloud environments. Kubernetes will continue to enhance its support for workloads that span different cloud providers, facilitating automated failover, optimized resource allocation, and unified monitoring across distributed systems.

2. Integration with Serverless Computing

Serverless computing, which abstracts infrastructure management entirely, is gaining traction as a way to further simplify cloud application deployment. While Kubernetes traditionally focuses on containerized workloads, there is increasing interest in integrating serverless capabilities with Kubernetes to provide greater flexibility in application architecture. Projects such as **KNative** are already working to bring serverless features to Kubernetes, enabling developers to deploy applications without worrying about the underlying infrastructure or containers. The future of

Kubernetes in the serverless landscape will likely see deeper integration, allowing developers to build event-driven, highly scalable applications without the need for complex orchestration. Kubernetes will evolve to manage serverless workloads alongside traditional containerized applications, enabling hybrid workloads that can scale in real-time based on demand while reducing infrastructure costs.

3. Edge Computing and IoT Integration

Edge computing, which involves processing data closer to where it is generated (at the "edge" of the network), is emerging as a key area for cloud computing. This trend is particularly relevant for Internet of Things (IoT) applications, where low-latency data processing is critical. Kubernetes is well-positioned to extend its capabilities to edge computing environments, as it can manage containerized applications across distributed edge devices. In the future, Kubernetes will expand its reach beyond traditional data centers and public cloud environments into edge nodes, enabling organizations to deploy applications in remote locations, mobile devices, or IoT devices. With edge computing, Kubernetes will enable near-instantaneous data processing and low-latency application performance, which will be crucial for industries like autonomous vehicles, industrial automation, and smart cities. The evolution of Kubernetes' networking and orchestration models will be essential for supporting the demands of edge computing.

4. Artificial Intelligence (AI) and Machine Learning (ML) Integration

As artificial intelligence (AI) and machine learning (ML) applications become more critical in various industries, Kubernetes will play a major role in managing the complex infrastructure required for training and deploying AI models. The scalability and automation provided by Kubernetes make it an ideal platform for managing distributed machine learning workloads, which often require high computational power and the ability to scale resources dynamically.

CONCLUSION

Kubernetes has rapidly emerged as a transformative technology in the cloud computing landscape, revolutionizing the way applications are deployed, managed, and scaled. Its ability to automate the deployment, scaling, and operation of containerized applications has significantly streamlined IT operations, enhanced efficiency, and enabled organizations to fully leverage the benefits of cloud infrastructure. Kubernetes, combined with the flexibility and scalability of cloud computing, offers organizations a robust solution for building, deploying, and managing cloud-native applications across diverse cloud environments, whether public, private, or hybrid. As businesses increasingly adopt microservices architectures and embrace the advantages of containerization, Kubernetes provides the necessary tools to simplify complex operations, improve resource utilization, and enhance application resilience. Its capabilities, such as auto-scaling, self-healing, and rolling updates, ensure that applications remain highly available and performant, even in dynamic and demanding cloud environments. Furthermore, Kubernetes' ability to integrate with a broad ecosystem of tools and services, from service meshes to monitoring solutions, allows organizations to implement a comprehensive and efficient cloud-native strategy. Looking ahead, the future of Kubernetes and cloud computing is promising. The growing adoption of multi-cloud and hybrid cloud strategies, the rise of serverless computing, the integration of AI/ML workloads, and the expansion into edge computing all point to a continued evolution of Kubernetes as a foundational platform for managing distributed applications. As Kubernetes continues to evolve, we can expect even greater automation, intelligence, and security, ensuring that organizations can meet the challenges of modern application development and infrastructure management. In conclusion, Kubernetes, when integrated with cloud computing, provides a powerful and flexible solution for organizations seeking to enhance their operational capabilities, reduce infrastructure costs, and accelerate their digital transformation. By adopting Kubernetes and leveraging its full potential, businesses can achieve greater scalability, efficiency, and agility, positioning themselves for success in an increasingly competitive and dynamic technological landscape. The continued evolution of Kubernetes will undoubtedly play a central role in shaping the future of cloud computing, driving innovation and enabling organizations to stay ahead in the digital era.

REFERENCES

- [1] <https://kubernetes.io/docs/concepts/architecture/>
- [2] <https://www.geeksforgeeks.org/kubernetes-architecture/>
- [3] <https://www.aquasec.com/cloud-native-academy/kubernetes-101/kubernetes-architecture/>
- [4] <https://devtron.ai/blog/kubernetes-architecture-the-ultimate-guide/>
- [5] <https://en.wikipedia.org/wiki/Kubernetes>
- [6] https://www.tutorialspoint.com/kubernetes/kubernetes_architecture.htm