SGOU
Higher Education for All

INTRODUCTION TO CLOUD COMPUTING

Course Code: B24DS02DE
BSc Data Science and Analytics
Discipline Specific Elective Course
Self Learning Material



SREENARAYANAGURU OPEN UNIVERSITY

SREENARAYANAGURU OPEN UNIVERSITY

The State University for Education, Training and Research in Blended Format, Kerala

SREENARAYANAGURU OPEN UNIVERSITY

SGOU

Vision

To increase access of potential learners of all categories to higher education, research and training, and ensure equity through delivery of high quality processes and outcomes fostering inclusive educational empowerment for social advancement.

Mission

To be benchmarked as a model for conservation and dissemination of knowledge and skill on blended and virtual mode in education, training and research for normal, continuing, and adult learners.

Pathway

Access and Quality define Equity.

Introduction to Cloud Computing

Course Code: B24DS02DE

Semester - V

Discipline Specific Elective Course
Undergraduate Programme
BSc Data Science and Analytics
Self Learning Material

(With Model Question Paper Sets)

SGOU

SREENARAYANAGURU
OPEN UNIVERSITY

SREENARAYANAGURU OPEN UNIVERSITY

The State University for Education, Training and Research in Blended Format, Kerala

SREENARAYANAGURU
OPEN UNIVERSITY

INTRODUCTION TO CLOUD COMPUTING

Course Code: B24DS02DE
Semester - V
Discipline Specific Elective Course
BSc Data Science and Analytics

Academic Committee

Dr. T. K. Manoj
Dr. Smitha Dharan
Dr. Satheesh S.
Dr. Vinod Chandra S.S.
Dr. Hari V. S.
Dr. Sharon Susan Jacob
Dr. Ajith Kumar R.
Dr. Smiju I.S.
Dr. Nimitha Aboobaker

Development of the Content

Subi Priya Laxhmi S.B.N.
Dr. Kanitha D.K.
Greeshma P.P.
Shamin S.
Aswathy V.S.

Review

Dr. Sarith Divakar

Linguistics

Dr. Sarith Divakar

Scrutiny

Subi Priya Laxhmi S.B.N.
Dr. Kanitha D.K.
Greeshma P.P.
Shamin S.
Aswathy V.S.

Cover Design

Jobin J.

Co-ordination

Director, MDDC :
Dr. I.G. Shibi
Asst. Director, MDDC :
Dr. Sajeevkumar G.
Coordinator, Development:
Dr. Anfal M.
Coordinator, Distribution:
Dr. Sanitha K.K.

SGOU

Scan this QR Code for reading the SLM on a digital device.

Edition
November 2026

Copyright
© Sreenarayanaguru Open University

ISBN 978-81-180900-4-5

9 788118 090045

All rights reserved. No part of this work may be reproduced in any form, by mimeograph or any other means, without permission in writing from Sreenarayanaguru Open University. Printed and published on behalf of Sreenarayanaguru Open University by Registrar, SGOU, Kollam.

www.sgou.ac.in

Visit and Subscribe our Social Media Platforms

MESSAGE FROM VICE CHANCELLOR

Dear learner,

I extend my heartfelt greetings and profound enthusiasm as I warmly welcome you to Sreenarayanaguru Open University. Established in September 2020 as a state-led endeavour to promote higher education through open and distance learning modes, our institution was shaped by the guiding principle that access and quality are the cornerstones of equity. We have firmly resolved to uphold the highest standards of education, setting the benchmark and charting the course.

The courses offered by the Sreenarayanaguru Open University aim to strike a quality balance, ensuring students are equipped for both personal growth and professional excellence. The University embraces the widely acclaimed "blended format," a practical framework that harmoniously integrates Self-Learning Materials, Classroom Counseling, and Virtual modes, fostering a dynamic and enriching experience for both learners and instructors.

The University is committed to providing an engaging and dynamic educational environment that encourages active learning. The Study and Learning Material (SLM) is specifically designed to offer you a comprehensive and integrated learning experience, fostering a strong interest in exploring advancements in information technology (IT). The curriculum has been carefully structured to ensure a logical progression of topics, allowing you to develop a clear understanding of the evolution of the discipline. It is thoughtfully curated to equip you with the knowledge and skills to navigate current trends in IT, while fostering critical thinking and analytical capabilities.The Self-Learning Material has been meticulously crafted, incorporating relevant examples to facilitate better comprehension.

Rest assured, the university's student support services will be at your disposal throughout your academic journey, readily available to address any concerns or grievances you may encounter. We encourage you to reach out to us freely regarding any matter about your academic programme. It is our sincere wish that you achieve the utmost success.



Regards,
Dr. Jagathy Raj V. P.

01-11-2026

SGOU

Contents

Block 1 Introduction to Cloud Computing 1

Unit 1 Foundations of Cloud Computing 2

Unit 2 Evolution and Milestones 23

Unit 3 Types of Cloud Deployment Models 34

Unit 4 Components of Cloud Computing 54

Block 2 Understanding Cloud based Services 73

Unit 1 SPI Service Models 74

Unit 2 Cloud Service Models 87

Unit 3 Managed High performance Computing and Serverless Cloud services 101

Unit 4 Popular Cloud Computing Service Models 116

Block 3 Managing Cloud Environment 140

Unit 1 Cloud Administration Portals and User Management 141

Unit 2 Identity and Access management 161

Unit 3 Resource Management 179

Unit 4 Ensuring Performance 201

Block 4 Cloud Migration and Associated Technologies 216

Unit 1 Introduction to Cloud Migration 217

Unit 2 Cloud Migration Planning and Assessment 233

Unit 3 Cloud Migration Tools and Techniques 251

Unit 4 Planning and Managing Cloud Migration 266

Block 5 Cloud Automation Tools 285

Unit 1 Introduction to Cloud Automation 286

Unit 2 Infrastructure Automation Tools 304

Unit 3 Configuration Management Tools 320

Unit 4 CI/CD and Orchestration Tools 337

Model Question Paper sets 351

SGOU



Introduction to Cloud Computing

SGOU

Unit 1
Foundations of Cloud Computing

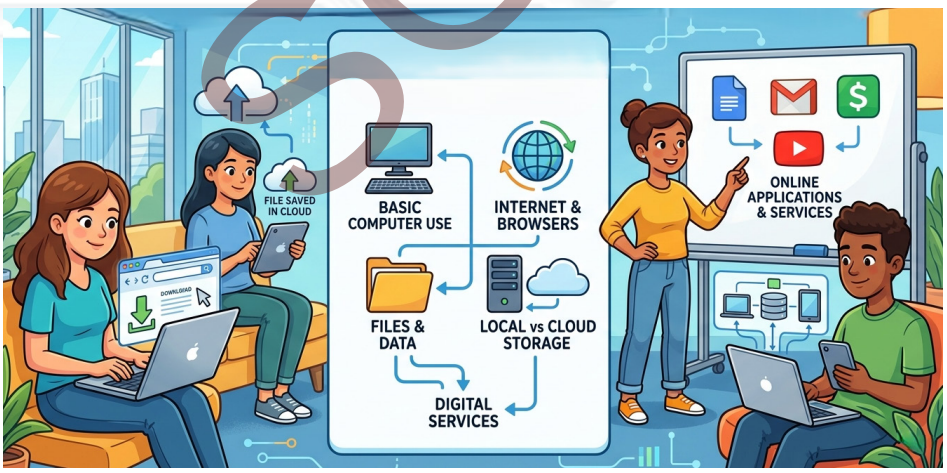
Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the basic concept and purpose of cloud computing
- identify the key characteristics of cloud computing
- describe the advantages and disadvantages of cloud computing
- recognize common applications of cloud computing

Prerequisite

A basic understanding of computers, operating systems, and the internet is required. Familiarity with common activities such as using web browsers, accessing websites, downloading files, and connecting devices to the internet is helpful. For example, using Google Search, Gmail, or online learning platforms provides an understanding of how internet-based services work.



A basic knowledge of data storage and file management is also useful. Concepts such as files, folders, local storage, servers, and data sharing should be familiar. For example, saving a document on a computer differs from storing it in Google Drive or OneDrive, where the data can be accessed from multiple devices through the internet.

General awareness of online applications and digital services provides a useful foundation. Familiarity with services such as Google Docs, Microsoft 365, YouTube, online

SGOU

banking, and online shopping platforms helps connect cloud computing concepts with real-world applications such as remote access, scalable services, centralized storage, and on-demand resources.

Keywords

On-Demand Resources, Remote Access, Scalability, Internet-Based Services, Resource Pooling, Vendor Lock-In

Discussion

1.1.1 Introduction to Cloud Computing

Suppose a learner creates an assignment and saves it in Google Drive. The learner does not need to keep the assignment only on the computer or carry a pen drive. The file is stored on remote servers managed by Google and can be accessed from a laptop, smartphone, or tablet through an Internet connection.

If the learner edits the assignment from another device, the changes can be synchronized across devices. The learner can also share the file with teachers or classmates without physically transferring it. This is a simple real-world example of cloud computing, where computing resources and services are accessed over the Internet rather than being provided entirely through local hardware.

Definition of Cloud Computing

Cloud computing is a technology for delivering computing resources and services over a network, typically the Internet, on demand. These resources include:

- Servers and computing power
- Data storage
- Databases
- Networking
- Software and applications
- Data processing resources

Instead of purchasing and maintaining all the required physical infrastructure, users can obtain computing resources from cloud service providers according to their requirements. Resources can be provisioned, scaled, and released as needed.

The National Institute of Standards and Technology (NIST) defines cloud computing as a model that enables convenient, on-demand network access to a shared pool of configurable computing resources, such as networks, servers, storage, applications, and services, that can be rapidly provisioned and released with minimal management effort or interaction with the service provider.

SGOU

According to NIST, cloud computing has five essential characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. These characteristics explain how cloud computing provides flexible, accessible, and efficient computing resources to users(fig 1.1.1).

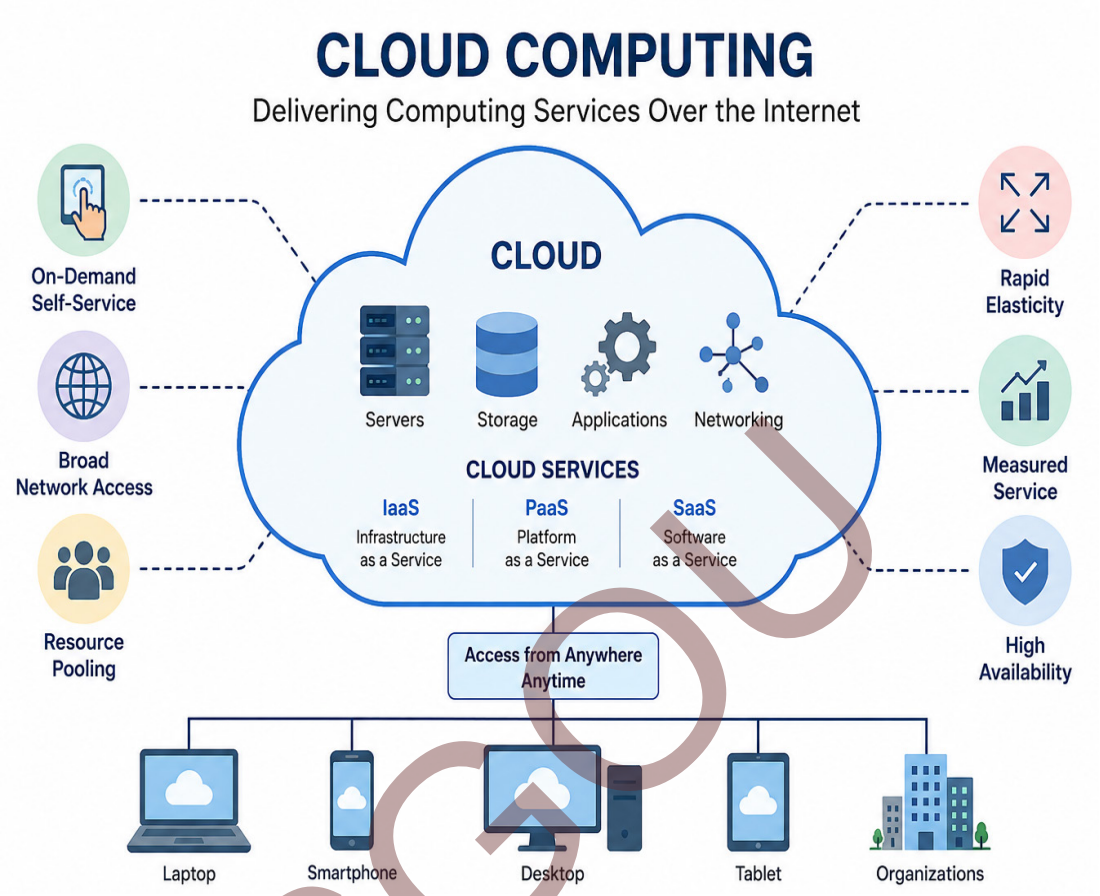


Fig 1.1.1: Concept of Cloud Computing

Cloud computing enables users to store data, run applications, process information, host websites, and access computing resources remotely. This provides greater flexibility and can reduce the need for organizations to maintain extensive physical infrastructure.

Examples: Google Drive and Microsoft OneDrive provide cloud storage, while Amazon Web Services (AWS), Microsoft Azure, and Google Cloud provide a wide range of cloud computing services.

1.1.1.1 Basic Concept and Working

The fundamental concept of cloud computing is the on-demand delivery of shared computing resources as services. The physical infrastructure is operated and maintained by the cloud service provider, while users access the required resources through a network.

A typical cloud environment consists of data centres, servers, storage systems, networking equipment, virtualization technologies, and cloud management software(fig 1.1.2)

SGOU

How Cloud Computing works?

Step 1: User Request

The user accesses a cloud service using a device such as a computer, smartphone, or tablet and sends a request for a particular resource or service.

Step 2: Internet Connection

The request is transmitted through the Internet to the cloud service provider.

Step 3: Request Processing

The cloud platform receives the request and determines the appropriate computing, storage, database, or application resources required to fulfil it.

Step 4: Resource Allocation

The required resources are allocated from the provider's cloud infrastructure. Depending on demand, resources can be scaled up or scaled down.

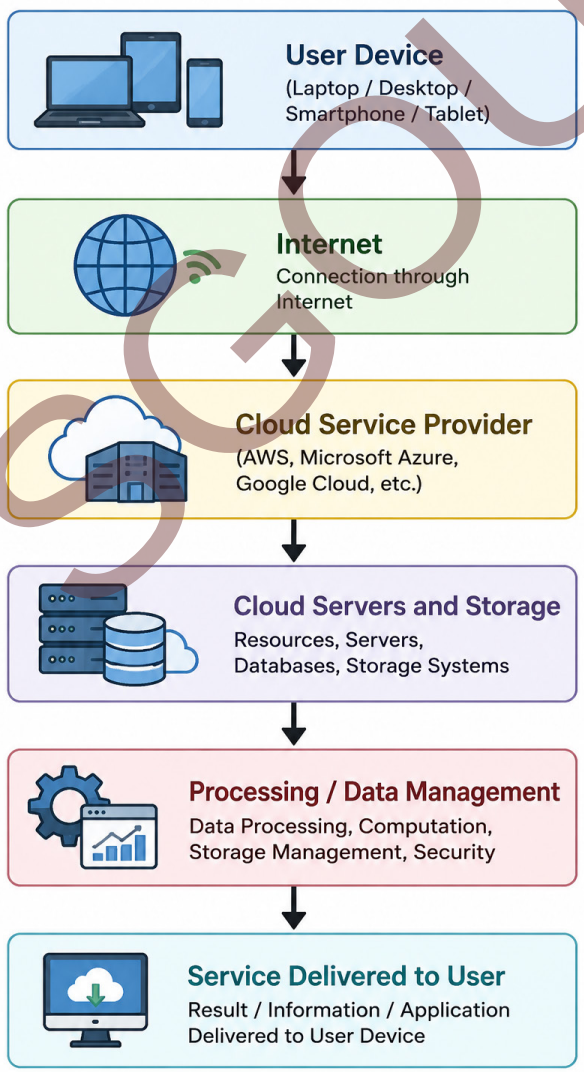


Fig 1.1.2: Working Flow of Cloud Computing

Step 5: Service Execution

The allocated resources perform the requested operation, such as:

- Storing or retrieving a file
- Running an application
- Processing data
- Managing a database
- Hosting a website

Step 6: Service Delivery

The result or requested service is delivered to the user's device through the Internet.

Step 7: Resource Release

When the resources are no longer required, they can be released or reallocated to other users, depending on the cloud service and usage model.

Thus, cloud computing enables users to access computing resources anytime and from anywhere without directly managing the underlying physical infrastructure.

1.1.2 Characteristics of Cloud Computing

Cloud computing has several distinctive characteristics that differentiate it from traditional computing environments. These characteristics describe how cloud resources are provided, accessed, shared, scaled, and monitored. They enable users and organizations to obtain computing services efficiently according to their requirements. The major characteristics include:

A. **On-Demand Self-Service:** On-demand self-service means that users can obtain computing resources automatically whenever they need them, without requiring direct assistance from the cloud service provider. Users can provision resources such as:

- Computing power
- Storage
- Virtual machines
- Databases
- Applications

For example, a company can create a virtual server through a cloud management console within a few minutes without contacting the provider's technical staff. i.e., Resources can be requested, configured, and accessed by users whenever required.

SGOU

B. Broad Network Access: Broad network access means that cloud services are accessible through networks using standard Internet-enabled devices and technologies. Users can access cloud resources from different types of devices, such as:

- Desktop computers
- Laptops
- Smartphones
- Tablets
- Thin clients

For example, a user can access a file stored in Google Drive using a smartphone at home and later access the same file using a computer in a classroom. i.e.,Cloud services are accessible from different locations and devices through a network.

C. Resource Pooling: Resource pooling means that the cloud provider combines its physical and virtual computing resources into a shared pool and dynamically allocates them to multiple users according to their requirements. Resources may include:

- Servers
- Storage
- Memory
- Processing power
- Network bandwidth

The physical location of the resources may not be known to the user. Virtualization allows resources to be efficiently shared among multiple users while maintaining logical separation.

For Example, a cloud provider may use a large pool of servers to provide computing resources to thousands of customers. i.e., Shared infrastructure is dynamically allocated to different users according to demand.

D. Rapid Elasticity: It refers to the ability of cloud resources to be quickly increased or decreased according to workload requirements. When demand increases, additional resources can be provisioned. When demand decreases, unnecessary resources can be released.

For Example, an e-commerce website may require additional computing resources during a major online sale because of increased visitor traffic. After the sale, these resources can be reduced. i..e, Cloud resources can expand or contract rapidly according to demand.

E. Measured Service: This means that cloud resource usage is automatically

SGOU

monitored, measured, and controlled. Cloud providers can measure resources such as:

- CPU usage
- Storage capacity
- Network bandwidth
- Number of users
- Processing time

This supports usage monitoring, resource management, and pay-as-you-use pricing in many cloud services.

For Example, if an organization uses a specific amount of cloud storage and computing resources, the provider can record that usage for billing and monitoring purposes. i.e., Cloud usage is measured and monitored, providing transparency about resource consumption.

F. Scalability and Flexibility: Scalability is the ability of a cloud system to handle changes in workload by adjusting its available resources. Flexibility refers to the ability to choose and modify resources according to changing requirements. Cloud resources can be adjusted based on:

- Number of users
- Data volume
- Application workload
- Business requirements
- Seasonal demand

For Example, an educational platform can increase its server capacity when thousands of learners access online examinations and reduce the capacity when the examination period ends. i.e., Cloud computing allows organizations to adapt resources to changing requirements without major changes to physical infrastructure.

G. High Availability: This refers to the ability of cloud services to remain accessible and operational for most of the required time. Cloud providers use techniques such as:

- Redundant servers
- Data replication
- Multiple data centres
- Backup systems

SGOU

- Fault-tolerant infrastructure
- Automatic failover

These mechanisms reduce the effect of hardware failures or other infrastructure problems.

For Example, if one server becomes unavailable, a cloud platform may redirect workloads to another available server. i.e., Cloud infrastructure is designed to provide continuous and reliable access to services, although no system can guarantee zero downtime.

These key characteristics of cloud computing enable a practical and efficient solution for individuals and organizations with changing computing requirements.

1.1.3 Advantages of Cloud Computing

Cloud computing provides several benefits by allowing individuals and organizations to access computing resources without maintaining extensive physical infrastructure. It improves cost efficiency, scalability, accessibility, collaboration, resource utilization, reliability, and speed of service deployment.

1. Cost Reduction

Cloud computing can reduce the cost of purchasing and maintaining physical IT infrastructure. Organizations can use cloud resources according to their requirements instead of investing heavily in servers, storage devices, networking equipment, and data-centre facilities.

- Reduces initial hardware investment.
- Minimizes infrastructure maintenance costs.
- Reduces expenditure on physical data centres.
- Many services follow a pay-as-you-use pricing model.

Example: A small business can rent cloud computing resources instead of purchasing and maintaining its own servers.

2. Scalability

Scalability allows cloud resources to be increased or decreased according to workload requirements.

- Additional computing resources can be added when demand increases.
- Unused resources can be reduced when demand decreases.
- Supports changing business and application requirements.
- Avoids the need to purchase new hardware for every increase in demand.

Example: An online shopping website can increase its computing capacity during a

SGOU

major sale and reduce it after the sale.

3. Flexibility and Accessibility

Cloud services can be accessed from different locations and devices through an Internet connection. Users can also select and modify cloud resources according to their requirements.

- Supports remote access.
- Enables access from computers, smartphones, and tablets.
- Supports remote and hybrid working.
- Resources can be modified as requirements change.

Example: A learner can access study materials stored in cloud storage from home, college, or another location.

4. Resource Optimization

Cloud computing improves the utilization of computing resources by dynamically allocating resources according to demand.

- Reduces underutilization of hardware.
- Supports sharing of infrastructure among multiple users.
- Resources can be allocated when required.
- Improves overall infrastructure efficiency.

Example: A cloud provider can allocate available processing capacity to applications that currently require more computing power.

5. Improved Collaboration

Cloud-based applications allow multiple users to access, share, and work on the same files or applications.

- Supports real-time or near-real-time collaboration.
- Simplifies file sharing.
- Reduces the need for physical file transfers.
- Enables teams in different locations to work together.

Example: Multiple learners can collaborate on a document using Google Docs and share the updated version with their teacher.

6. Backup and Disaster Recovery

Cloud computing can support data backup and disaster recovery by storing copies of data and applications in geographically distributed infrastructure.

SGOU

- Protects against hardware failure.
- Helps recover data after accidental deletion or system failure.
- Supports business continuity.
- Reduces dependence on a single physical location.

Example: An organization can maintain cloud-based backups of important business data and restore them when the primary system becomes unavailable.

7. Automatic Updates

Cloud service providers generally manage software updates, security patches, and infrastructure maintenance for the services they offer.

- Reduces the need for manual updates.
- Helps maintain current software versions.
- Reduces maintenance workload for users.
- Supports timely security and performance improvements.

Example: A cloud-based application can receive updates from the service provider without requiring every user to manually install the underlying infrastructure updates.

8. Faster Deployment

Cloud computing enables applications and computing resources to be provisioned and deployed quickly compared with setting up physical infrastructure.

- Reduces infrastructure setup time.
- Enables rapid provisioning of virtual machines and services.
- Supports faster application development and testing.
- Helps organizations respond quickly to business requirements.

Example: A development team can create a cloud-based test server within minutes instead of waiting for a physical server to be purchased, installed, and configured.

Cloud computing offers significant advantages by reducing infrastructure costs while improving scalability, accessibility, collaboration, resource utilization, reliability, and deployment speed. These benefits make cloud services suitable for a wide range of personal, educational, business, and organizational applications.

1.1.4 Challenges of Cloud Computing

Although cloud computing provides flexibility, scalability, and cost benefits, it also introduces several technical, security, operational, and legal challenges. Organizations must carefully manage these challenges to ensure that cloud services remain secure, reliable, compliant, and suitable for their requirements.

SGOU

1. Security and Privacy Concerns

Cloud environments store and process data on infrastructure managed by third-party providers, creating concerns about unauthorized access, data breaches, identity theft, and data privacy.

- Sensitive information may be exposed if security controls are inadequate.
- Unauthorized users may attempt to access cloud resources.
- Misconfigured cloud services can create security vulnerabilities.
- Organizations must implement proper authentication, access control, and encryption.

Example: A company may face a data breach if confidential customer information is stored in a cloud storage service with incorrectly configured access permissions.

2. Internet Dependency

Cloud services generally require a reliable network or Internet connection for access.

- Loss of connectivity can prevent users from accessing cloud applications and data.
- Slow Internet connections can affect application performance.
- Remote locations may have limited network availability.

Example: If an Internet connection fails during an online examination conducted through a cloud-based platform, learners may temporarily lose access to the examination system.

3. Downtime and Availability Issues

Cloud providers may experience service interruptions, hardware failures, software problems, or network outages.

- Cloud services are not guaranteed to be available at all times.
- Service interruptions can affect business operations.
- Organizations may depend heavily on the provider's availability and recovery mechanisms.

Example: If a cloud-hosted business application becomes temporarily unavailable, employees may be unable to access essential services.

4. Vendor Lock-In

Vendor lock-in occurs when an organization becomes highly dependent on a particular cloud provider and finds it difficult or expensive to move its applications and data to another provider.

SGOU

- Cloud providers may use different technologies and interfaces.
- Moving large volumes of data can be time-consuming and costly.
- Applications may depend on provider-specific services.

Example: An application designed specifically around one provider's proprietary database services may require significant modification before it can operate on another cloud platform.

5. Limited Control

In cloud computing, the cloud provider manages much of the underlying hardware, infrastructure, networking, and platform environment.

- Users may have limited control over physical infrastructure.
- Certain configuration options may not be available.
- Infrastructure maintenance is largely handled by the provider.

Example: An organization using a public cloud cannot directly access or physically manage the servers on which its application runs.

6. Data Migration Challenges

Data migration involves transferring data and applications from one system or environment to another. Moving large amounts of data to or from the cloud can be complex.

- Large data volumes may require significant transfer time.
- Data formats may differ between systems.
- Migration can cause temporary service disruption.
- Data integrity must be maintained during transfer.

Example: Moving a large organization's database from an on-premises server to a cloud database may require careful planning, testing, and validation.

7. Compliance and Legal Issues

Organizations using cloud services must comply with data protection laws, industry regulations, contractual requirements, and organizational policies.

- Data may be stored in different geographical locations.
- Different countries may have different data protection requirements.
- Organizations may need to maintain audit records and meet regulatory requirements.
- Responsibility for compliance may be shared between the provider and the customer.

SGOU

Example: A healthcare organization must ensure that cloud-based storage and processing of patient information comply with applicable healthcare and data-protection regulations.

8. Performance and Latency Issues

Latency refers to the delay between sending a request and receiving a response. Cloud applications may experience performance problems because of network conditions, geographical distance, or high workloads.

- Network congestion can increase response time.
- Applications requiring real-time processing may be sensitive to latency.
- Heavy workloads can affect performance if resources are insufficient or poorly configured.

Example: A cloud-based video conferencing application may experience delays or interruptions when network latency is high.

Cloud computing challenges mainly involve security, connectivity, reliability, control, migration, compliance, and performance. Proper planning, security measures, service-level agreements, backup strategies, and careful selection of cloud providers can help organizations reduce these risks.

1.1.5 Applications of Cloud Computing

Cloud computing is widely used across education, healthcare, finance, business, entertainment, government, and technology. Organizations use cloud services to store data, host applications, process large volumes of information, support collaboration, and provide services to users without maintaining all computing infrastructure locally(fig 1.1.3).

1. Cloud Storage and Backup

Cloud computing provides online storage for files, documents, photographs, videos, databases, and other digital information.

- Allows users to access data from different devices.
- Provides centralized storage.
- Supports automatic data synchronization.
- Enables backup and recovery of important data.
- Reduces dependence on physical storage devices.

Example: Google Drive and Microsoft OneDrive allow users to store and access files through the Internet.

SGOU

2. Web and Application Hosting

Cloud platforms provide infrastructure for hosting websites and web-based applications.

- Websites can be deployed on cloud servers.
- Computing resources can be increased when traffic grows.
- Cloud infrastructure supports application testing and deployment.
- Organizations can avoid maintaining their own physical web servers.

Example: An online shopping website can host its web application on a cloud platform and scale its resources during periods of high traffic.

3. Software as a Service (SaaS)

Software as a Service (SaaS) provides software applications through the Internet. Users generally access the application through a web browser or dedicated client without installing and maintaining the complete software infrastructure locally.

- Software is centrally managed by the provider.
- Updates are generally handled by the provider.
- Users can access applications from multiple devices.
- Subscription or usage-based pricing is commonly used.

Examples: Google Workspace, Microsoft 365, and Salesforce.

4. Online Education and E-Learning

Cloud computing supports online classes, virtual learning environments, digital libraries, examinations, and collaborative learning.

- Provides access to learning materials from different locations.
- Supports online classes and virtual classrooms.
- Enables cloud-based assignment submission.
- Facilitates collaboration between learners and teachers.
- Supports storage and delivery of educational content.

Example: An educational institution can use cloud-based learning platforms to provide course materials and conduct online assessments.

5. Healthcare Applications

Healthcare organizations use cloud computing for electronic health records, medical data storage, telemedicine, healthcare applications, and data analysis.

SGOU

- Provides centralized access to healthcare information.
- Supports secure data storage and backup.
- Enables remote healthcare services.
- Supports analysis of medical and patient data.
- Facilitates collaboration among healthcare professionals.

Example: A healthcare organization can store patient records in a cloud-based system and provide authorized healthcare professionals with access to the required information.

6. Banking and Financial Services

Cloud computing is increasingly used in banking, digital payments, financial analysis, customer services, and fraud detection.

- Supports online and mobile banking applications.
- Provides scalable infrastructure for financial services.
- Enables analysis of large financial datasets.
- Supports backup and disaster recovery.
- Helps detect suspicious transactions using data-processing systems.

Example: A digital banking platform can use cloud infrastructure to handle large numbers of customer transactions and requests.

7. E-Commerce

E-commerce companies use cloud computing to support online stores, product catalogues, payment systems, customer databases, and order processing.

- Handles large numbers of users.
- Supports rapid scaling during sales and promotional events.
- Stores product and customer information.
- Supports recommendation and personalization systems.
- Enables continuous availability of online stores.

Example: An online shopping platform can increase its computing resources during a festival sale to handle increased customer traffic.

8. Business and Enterprise Applications

Organizations use cloud computing for various business and enterprise functions such as customer relationship management, accounting, human resources, communication, and project management.

SGOU

- Supports remote access to business applications.
- Facilitates collaboration among employees.
- Reduces infrastructure maintenance requirements.
- Centralizes business data and applications.
- Supports organizations with distributed offices.

Example: A company can use a cloud-based customer relationship management system to allow employees in different branches to access and update customer information.

9. Data Analytics and Big Data

Cloud computing provides scalable computing and storage resources for processing large and complex datasets.

- Supports large-scale data storage.
- Provides computing resources for data processing.
- Enables data visualization and reporting.
- Supports real-time and batch data analysis.
- Allows organizations to process datasets without maintaining large local infrastructure.

Example: A retail company can analyze customer transactions and purchasing patterns using cloud-based data analytics services.

10. Artificial Intelligence and Machine Learning

Cloud platforms provide computing resources and specialized services for developing, training, deploying, and managing AI and machine learning applications.

- Provides high-performance computing resources.
- Supports large datasets for model training.
- Enables machine learning model deployment.
- Provides access to AI development tools and services.
- Allows resources to be scaled according to computational requirements.

Example: A company can use cloud-based machine learning services to develop a recommendation system for its online shopping platform.

11. Media and Entertainment

Cloud computing is widely used for video streaming, music streaming, digital content storage, gaming, and content distribution.

SGOU

- Stores large volumes of multimedia content.
- Supports streaming services.
- Enables content delivery to users in different locations.
- Provides scalable infrastructure for large audiences.
- Supports video processing and content management.

Example: Streaming platforms use cloud infrastructure to store and deliver movies, television programmes, and other digital content to users.

12. Government Services

Government organizations use cloud computing to provide digital public services, online portals, data storage, and administrative applications.

- Supports online delivery of government services.
- Provides centralized data management.
- Enables citizens to access services remotely.
- Supports scalability during high-demand periods.
- Can improve efficiency and reduce infrastructure requirements.

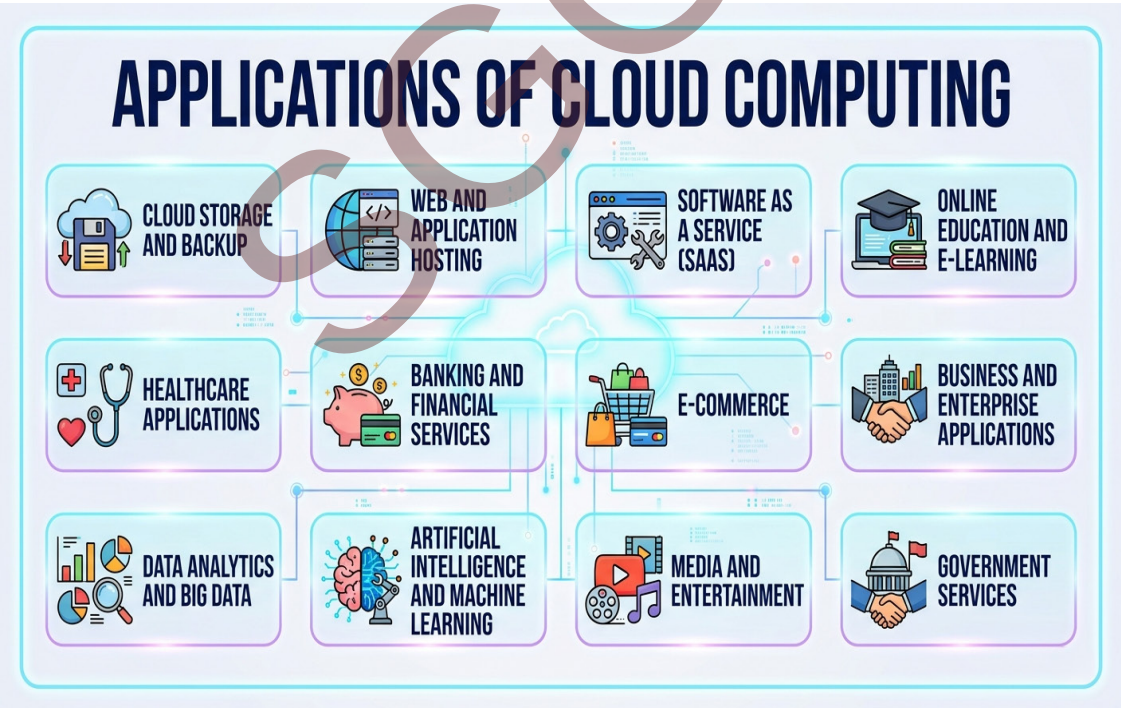


Fig 1.1.3: Applications of Cloud Computing

Example: A government department can use cloud infrastructure to operate an online portal through which citizens submit applications and access public services.

Cloud computing has become an essential part of modern digital services, providing on-demand, scalable, flexible, and accessible computing resources for applications across education, healthcare, finance, business, AI, entertainment, and government.

Cloud computing has transformed the way computing resources and services are delivered by providing flexible, scalable, accessible, and cost-effective solutions. Despite challenges such as security, privacy, and dependency on Internet connectivity, its wide range of applications makes it an essential technology for modern digital services.

Recap

- Cloud computing provides computing resources over the internet.
- Cloud services can be accessed from different locations using internet-connected devices.
- Computing resources include servers, storage, databases, software, and networking.
- On-demand access allows resources to be obtained whenever required.
- Scalability enables resources to increase or decrease according to demand.
- Cloud computing reduces the need for maintaining extensive physical infrastructure.
- Resource pooling allows multiple users to share computing resources efficiently.
- Broad network access supports access through computers, tablets, and smartphones.
- Measured services allow resource usage to be monitored and controlled.
- Cloud computing can reduce hardware and infrastructure costs.
- Remote access supports flexible working and learning environments.
- Cloud services can provide automatic updates and maintenance.
- High availability helps users access services with minimal interruption.
- Dependence on internet connectivity can affect access to cloud services.
- Security and privacy are important considerations in cloud computing.
- Service interruptions can temporarily affect access to cloud-based resources.
- Data stored with cloud providers requires appropriate security measures.

SGOU

- Cloud computing supports applications such as online storage and collaboration.
- Education, healthcare, banking, entertainment, and e-commerce use cloud-based services.
- Cloud computing provides flexible, scalable, and accessible solutions for modern digital applications.

Objective Type Questions

1. What technology provides computing resources over the internet?
2. What feature allows resources to be obtained whenever required?
3. What feature allows cloud resources to increase or decrease based on demand?
4. What refers to sharing computing resources among multiple users?
5. What feature enables access to cloud services through different devices?
6. What mechanism monitors and controls resource usage?
7. What type of infrastructure is reduced through cloud computing?
8. What service is used to store files online?
9. What factor is essential for accessing cloud services remotely?
10. What aspect protects cloud data from unauthorized access?
11. What refers to the ability of a service to remain accessible?
12. What type of cost can cloud computing reduce by minimizing hardware requirements?
13. What application enables users to work together on documents online?
14. What industry uses cloud computing for digital patient services and data management?
15. What industry uses cloud computing for online transactions and financial services?
16. What industry uses cloud computing for online shopping platforms?
17. What industry uses cloud computing for streaming and digital content?
18. What provides software applications through the internet?

SGOU

19. What provides storage capacity through the internet?

20. What technology enables flexible and remote access to computing resources?

Answers

1. Cloud Computing
2. On-Demand
3. Scalability
4. Resource Pooling
5. Accessibility
6. Metering
7. Infrastructure
8. Cloud Storage
9. Internet
10. Security
11. Availability
12. Hardware Cost
13. Collaboration
14. Healthcare
15. Banking
16. E-Commerce
17. Entertainment
18. SaaS
19. Cloud Storage
20. Cloud Computing

SGOU

Assignments

1. Explain the concept and main characteristics of cloud computing with suitable examples.
2. Discuss the major advantages and disadvantages of cloud computing.
3. Describe the different applications of cloud computing in everyday life and various industries.
4. Compare traditional computing with cloud computing based on accessibility, cost, scalability, and resource usage.
5. Explain how cloud computing supports modern services such as online education, healthcare, banking, e-commerce, and entertainment.

Reference

1. Elayidom, M. S., et al. Cloud Computing & Big Data: From the Basics to Practical Use Cases. 1st ed., Cengage.
2. Erl, T., Barcelo Monroy, E., & others. (2023). *Cloud computing: Concepts, technology, security, and architecture* (2nd ed.). Pearson.
3. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2011). *Cloud computing: Principles and paradigms*. Wiley.
4. Sosinsky, B. (2011). *Cloud computing Bible*. Wiley Publishing.
5. Erl, T., Mahmood, Z., & Puttini, R. (2019). *Cloud computing: Concepts, technology & architecture*. Pearson.
6. Chandrasekaran, K. (2023). *Essentials of cloud computing*. CRC Press.

Suggested Reading

1. Google Cloud – Cloud Computing Learning Resources https://cloud.google.com/learn
2. Shimonski, R. (2025). *Foundations of cloud computing*. Elsevier.
3. Wufka, M., & Canonico, M. (2026). *Introduction to cloud computing*. Springer.
4. Yang, J. (2025). *Cloud computing and microservices*. Springer.
5. Oracle – Cloud Infrastructure Learning https://education.oracle.com/learning-explorer

SGOU

Unit 2
Evolution and Milestones

Learning Outcomes

Upon completion of the unit, learner will be able to:

- define cloud computing and recall its historical development from early computing systems
- identify the major stages involved in the transition from mainframe systems to cloud computing
- describe the concept of grid computing and explain its contribution to cloud evolution
- outline the idea of utility computing and state its role in delivering computing as a service
- explain how Web 2.0 technologies influenced the growth and popularity of cloud computing

Prerequisite

Most of you are already familiar with using online services such as Google Drive, YouTube, Netflix, or social media platforms like Instagram and Facebook. You use these services every day to store photos, watch videos, share information, and communicate instantly all through the internet. You might have noticed that these applications do not require installation on your personal computer, nor do you need to worry about managing servers or storage space. These examples show how computing today has become more flexible, more connected, and available anywhere at any time.

In the early days, however, computing was very different. Computers were large, expensive machines used only by big organizations. Over time, technology advanced, and computing systems evolved from centralized mainframes to distributed and grid computing, and later to virtualized and cloud-based environments. This gradual transformation improved the way people share, process, and store information, making computing faster, more reliable, and more accessible to everyone.

By connecting your everyday experiences with these developments, you can better understand how cloud computing came into existence. In this unit, you will explore how computing evolved through different stages starting from mainframes, progressing through grid and utility computing, and influenced by the rise of Web 2.0 to form the powerful cloud systems that support today's digital world.

SGOU

Keywords

Mainframe, Grid Computing, Utility Computing, Web 2.0, Cloud Evolution

Discussion

1.2.1 Evolution of Cloud Computing

The idea of cloud computing started in the 1950s, when people began thinking about how computers could share their power and resources with many users at the same time. The word "cloud" became popular much later, but early systems like distributed computing and time-sharing helped to create the base for today's cloud technology. Over the years, computing developed step by step from mainframe computers to client-server systems, then to grid computing and utility computing, and finally to the modern cloud systems we use today.

Today, cloud computing is an advanced technology that lets users store, manage, and process data on the Internet instead of using their own computers or local servers. It provides many useful services such as data storage, software applications, computing power, servers, development tools, and networking support. These services are available anytime and anywhere, as long as there is an internet connection.

Big companies like Amazon (AWS), Google (Google Cloud Platform), and Microsoft (Azure) are the main providers of cloud services. They offer platforms that are flexible, powerful, and easy to scale, helping both individuals and organizations. Cloud computing has changed the way businesses work by reducing costs, making data easier to access, and improving productivity and flexibility.

1.2.2 Stages in the Evolution of Cloud Computing

The concept of cloud computing did not develop instantaneously; rather, it is the outcome of several decades of continuous technological progress and innovation in the fields of computing and networking. Over time, various computing paradigms have evolved, each introducing new principles, architectures, and techniques that have collectively influenced the formation of modern cloud computing. The evolution began in the era of mainframe computing and gradually progressed through distributed systems, cluster computing, grid computing, virtualization, and service-oriented architectures, ultimately leading to the development of the comprehensive cloud computing environment on which we rely on today. Each stage in this evolutionary process played a significant role in enhancing the efficiency, scalability, and accessibility of computing resources. The following sections provide a detailed explanation of these stages and their contributions to the emergence of cloud computing(fig 1.2.1).

SGOU

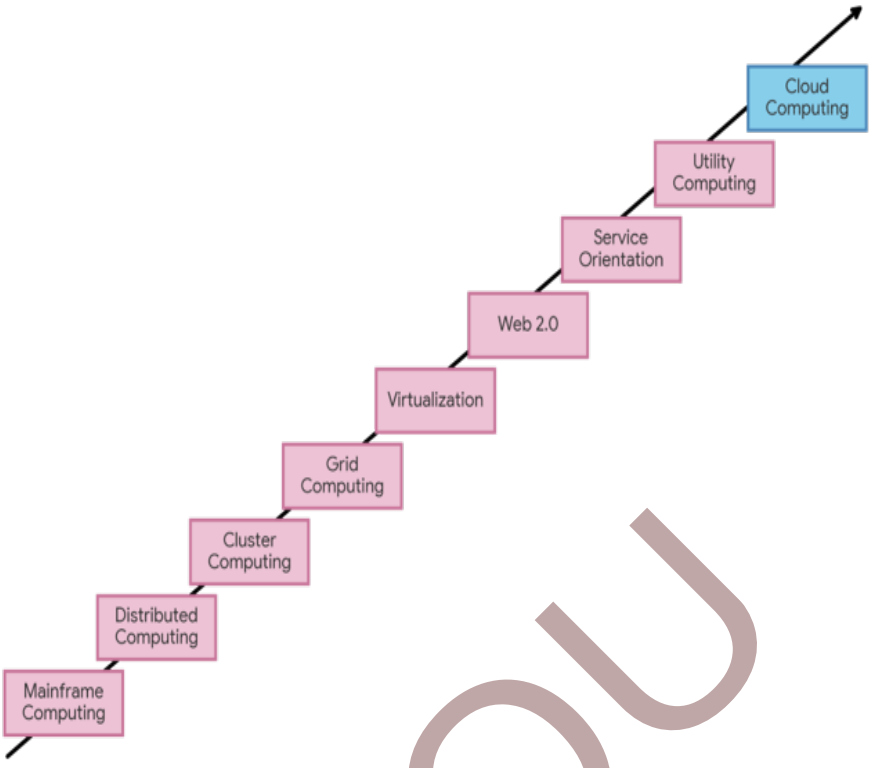


Fig. 1.2.1 Evolution of Cloud Computing

1.2.2.1 Mainframe Computing

The earliest stage in the evolution of cloud computing began with mainframe computers during the 1950s and 1960s. Mainframes were large, powerful, and extremely costly machines primarily used by government institutions, research organizations, and large businesses for processing vast amounts of data. These systems were designed to handle multiple tasks at once and deliver reliable performance with minimal downtime.

Because of their high cost, it was not practical for every user or department to own a separate machine. Instead, multiple users connected to a single mainframe through terminals simple input and output devices without independent processing power. While each terminal appeared to operate as an individual computer to its user, all the computation actually took place in the central mainframe. This technique, known as time-sharing, allowed several users to share the same physical computing resources efficiently.

Mainframes also introduced the concept of centralized control and data management, where storage, processing, and maintenance were all handled at a central location. This reduced redundancy and improved system reliability, as administrators could manage resources more effectively. The principle of providing computing as a shared service originating from the mainframe era became one of the most influential ideas leading to modern cloud computing.

SGOU

1.2.2.2 Distributed Computing

The next stage in the evolution of cloud computing is distributed computing, which became popular during the 1970s and 1980s. In this model, a single large computing task is divided into smaller parts, and these parts are executed simultaneously by multiple computers connected through a network. Each computer, known as a node, performs its assigned portion of the work and communicates with other nodes to share results. Together, they function as a unified system, even though they may be located in different places.

Distributed computing was developed to solve problems that were too large or complex for a single computer to handle efficiently. It allowed organizations to combine the power of several machines, increasing the overall processing speed and performance. This approach also improved reliability, since if one node failed, others could continue working without stopping the entire process.

An important example of distributed computing is the client-server model, where a server provides resources or services, and clients (such as personal computers) request them. This model is still widely used today in web applications and databases.

The key contribution of distributed computing to the evolution of cloud computing is the idea of sharing and coordinating resources across multiple systems. It introduced principles like communication between nodes, workload distribution, and fault tolerance, all of which are fundamental to how cloud systems operate today. Distributed computing shifted the focus from centralized mainframes to collaborative, interconnected systems, paving the way for the flexibility and scalability that define modern cloud computing.

1.2.2.3 Cluster Computing

The next stage in the evolution of cloud computing is known as cluster computing, which gained significance during the late 1980s and 1990s. In this model, a group of interconnected computers, referred to as a cluster, work together to perform tasks as if they were a single powerful system. Each computer within the cluster is called a node, and all nodes are connected through high-speed local area networks. These nodes share the same resources and collectively solve complex computational problems by dividing the workload among themselves.

Cluster computing emerged as an efficient alternative to costly mainframes and supercomputers. Instead of relying on one large machine, organizations could use multiple smaller and less expensive systems to achieve similar performance levels. This approach improved both speed and reliability because the workload was processed in parallel by multiple nodes. Moreover, if one node failed, the remaining nodes continued to function, ensuring fault tolerance and uninterrupted operation.

Cluster systems became widely used in research, weather forecasting, scientific simulations, and financial modeling applications that required intensive data processing. The key idea behind cluster computing was parallelism and coordination, where each computer worked on a different portion of a task simultaneously.

This stage played a crucial role in shaping cloud computing because it demonstrated

SGOU

how linking several computers could provide scalable, flexible, and high-performance computing power. The principles of resource sharing, load balancing, and scalability developed in cluster computing directly influenced the design of modern cloud infrastructures, where multiple servers collaborate to deliver seamless computing services to users.

1.2.2.4 Grid Computing

Following cluster computing, the next major stage in the evolution of cloud computing was grid computing, which became prominent in the late 1990s and early 2000s. Grid computing extended the concept of clusters by connecting computers and resources from different locations, sometimes even across countries or continents, to work together on a common task. Instead of being limited to a local network like in cluster computing, grid computing used wide-area networks, including the Internet, to link diverse systems and organizations.

The main goal of grid computing was to utilize idle or underused computing power from various systems to solve large and complex problems. It allowed institutions, research centers, and organizations to share their computing resources, storage, and data in a coordinated way. For example, a university in one country could contribute its unused processing capacity to help scientists in another country analyze large datasets or perform simulations.

This model was inspired by the idea of the electric power grid, where users can access electricity from a shared infrastructure whenever they need it. Similarly, in grid computing, users could access shared computational power as a utility. This introduced the concept of resource pooling and on-demand access, which are now key characteristics of cloud computing.

Grid computing played an important role in projects such as climate modeling, medical research, and large-scale scientific experiments like those conducted at CERN. It proved that geographically distributed systems could collaborate efficiently, providing enormous processing power through coordinated resource sharing.

The influence of grid computing on cloud computing is significant, it demonstrated that computing could be delivered as a service, accessible remotely through the Internet. The principles of distributed collaboration, resource virtualization, and service-based access developed during this stage became essential building blocks for the cloud computing models that followed.

1.2.2.5 Virtualization

The next major stage in the evolution of cloud computing is virtualization, which transformed the way computing resources are managed and utilized. Virtualization technology allows a single physical machine to be divided into multiple virtual machines (VMs), each functioning as an independent computer with its own operating system and applications. This means one physical server can act as several virtual servers, improving resource utilization and flexibility.

Before virtualization, computing resources were often underused because each

SGOU

application or operating system required a dedicated machine. With virtualization, organizations could run multiple applications on the same hardware without interference, thereby reducing costs and increasing efficiency. This innovation became the core foundation of modern cloud infrastructure, enabling service providers to host and manage thousands of virtual servers for users around the world.

Virtualization also made it easier to allocate, scale, and manage resources dynamically. For instance, if one virtual machine needs more memory or processing power, these resources can be reassigned without interrupting other operations. It also improved system reliability if one virtual machine fails, others continue running normally.

Technologies such as VMware, Microsoft Hyper-V, and KVM (Kernel-based Virtual Machine) popularized this concept, making virtualization practical and accessible. This stage introduced the idea of hardware abstraction, where users no longer needed to worry about the physical details of the underlying system.

Virtualization directly influenced cloud computing by allowing data centers to operate efficiently and deliver computing as a flexible, on-demand service. It laid the groundwork for Infrastructure as a Service (IaaS), where virtual machines, storage, and networking resources are provided to users over the internet. In short, virtualization made the cloud scalable, cost-effective, and reliable key qualities that define today's cloud environments.

1.2.2.6 Web 2.0

The next important stage in the evolution of cloud computing is Web 2.0, which emerged in the early 2000s. This phase marked a major shift in how people used and interacted with the internet. Unlike the early web (known as Web 1.0), where users could only read static pages and access limited content, Web 2.0 introduced dynamic, interactive, and user-centered web applications.

Web 2.0 made the internet a platform for collaboration and sharing. Users were no longer just consumers of information they became contributors through social media, blogs, wikis, and online communities. Popular platforms like Facebook, YouTube, and Wikipedia encouraged users to create and share content, while tools like Google Docs allowed people to collaborate online in real time. This transformation required powerful servers and scalable systems capable of handling millions of simultaneous users and large volumes of data.

The development of technologies such as AJAX (Asynchronous JavaScript and XML), HTML5, and web application frameworks made websites more responsive and interactive. These advancements encouraged the movement of data storage and processing from personal computers to remote servers introducing the early idea of accessing applications and data "from anywhere."

Web 2.0 played a vital role in shaping cloud computing because it demonstrated the need for internet-based services that could grow according to user demand. It also helped popularize the concept of Software as a Service (SaaS), where applications run on the web instead of being installed locally on a user's device.

SGOU

In summary, Web 2.0 transformed the internet into a powerful, service-oriented platform that encouraged user participation, data sharing, and collaboration. These principles directly influenced the design and purpose of cloud computing, making it a natural next step in the evolution of computing technologies.

1.2.2.7 Service-Oriented Architecture (SOA)

The next significant stage in the evolution of cloud computing is Service-Oriented Architecture (SOA). This model emerged in the early 2000s as a method for designing software systems in which different components, called services, communicate and cooperate over a network. Each service performs a specific function such as processing a payment, retrieving data, or verifying a user and can be reused by multiple applications.

In SOA, services are loosely coupled, meaning they are independent of each other and can be combined in various ways to build complex systems. This flexibility made software development faster, more modular, and easier to maintain. For example, an organization could develop a single authentication service and use it across multiple applications instead of creating separate ones for each program.

The introduction of SOA also encouraged standardized communication protocols such as XML, SOAP (Simple Object Access Protocol), and later REST (Representational State Transfer), which allowed services built on different platforms to interact seamlessly. This interoperability was a major step forward, as it helped break down the barriers between different systems and technologies.

SOA played a critical role in the growth of cloud computing because it introduced the concept of delivering computing functions as services. The core idea that computing resources like software, platforms, and infrastructure could be accessed and used over the internet came directly from the principles of SOA.

In essence, SOA became the architectural foundation for cloud service models such as SaaS (Software as a Service), PaaS (Platform as a Service), and IaaS (Infrastructure as a Service). It helped organizations build systems that were more flexible, scalable, and reusable, which perfectly aligned with the goals of cloud computing. Thus, SOA served as the bridge between traditional software systems and the modern service-based cloud environment.

1.2.2.8 Utility Computing

The next milestone in the evolution of cloud computing is Utility Computing, which introduced a revolutionary way of delivering computing resources. Utility computing is based on the idea of providing computing services such as storage, processing power, and applications just like public utilities such as electricity, water, or telephone services. In this model, users can access the resources they need on demand and pay only for what they use, rather than purchasing and maintaining expensive hardware or software.

Utility computing emerged in the early 2000s and was driven by the need for cost-effective and efficient computing solutions. Before this model, organizations had to invest heavily in their own servers and infrastructure, which often remained underused during low-demand periods. Utility computing solved this issue by allowing users to

SGOU

scale resources up or down depending on their requirements, ensuring better resource utilization and cost savings.

This approach was supported by advances in virtualization, distributed computing, and network technologies, which made it possible to allocate resources dynamically and securely over the internet. Early examples of utility computing include services provided by companies like Amazon (with Amazon Web Services), Google, and IBM, which offered computing power and storage on a metered basis.

Utility computing also introduced important business and operational concepts that shaped modern cloud computing, such as on-demand resource provisioning, measured service, and scalability. It transformed computing from a product that organizations owned into a service that they could rent as needed.

In summary, utility computing brought the concept of computing-as-a-service into reality. It allowed individuals and businesses to access powerful computing resources without worrying about infrastructure management. This idea directly evolved into today's cloud computing, where users enjoy flexible, affordable, and easily scalable services delivered through the internet.

1.2.2.9 Cloud Computing

The final and most advanced stage in this evolutionary journey is Cloud Computing. Cloud computing represents the combination and refinement of all the earlier computing models distributed systems, grid computing, virtualization, service orientation, and utility computing into a single, unified platform for delivering computing services over the internet.

In cloud computing, resources such as servers, storage, databases, applications, and networking components are made available to users through online platforms. These resources are hosted in large data centers managed by cloud service providers such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud. Users can access them remotely from anywhere and at any time using any device with an internet connection.

One of the defining features of cloud computing is its on-demand self-service nature. Users can quickly obtain computing resources whenever needed, without requiring direct human interaction with the service provider. It also offers elasticity and scalability, allowing resources to expand or shrink automatically based on workload demands. This ensures high efficiency and cost-effectiveness, as users only pay for what they use.

Cloud computing services are typically categorized into three main models:

- **Infrastructure as a Service (IaaS) :** provides virtualized computing resources like servers and storage.
- **Platform as a Service (PaaS) :** offers platforms and tools for developing and deploying applications.
- **Software as a Service (SaaS) :** delivers ready-to-use software applications over the internet.

SGOU

You will learn more about the concepts, characteristics, and service models of cloud computing in detail in the upcoming units.

Recap

- Cloud computing originated from the idea of sharing computing resources in the 1950s.
- Mainframe systems allowed multiple users to access a single powerful computer.
- Distributed and client-server models improved communication between systems.
- Cluster computing enabled faster processing by connecting multiple computers.
- Grid computing allowed global sharing of computing resources across organizations.
- Virtualization made it possible to run many virtual machines on a single physical system.
- Web 2.0 introduced interactive and user-driven web applications.
- Service-Oriented Architecture (SOA) supported modular and reusable software services.
- Utility computing introduced the pay-as-you-use model for computing services.
- Cloud computing combined all these ideas to provide scalable, flexible, and on-demand services.
- Major cloud providers include Amazon AWS, Microsoft Azure, and Google Cloud.
- Cloud services are categorized into IaaS, PaaS, and SaaS models.
- Cloud computing ensures cost efficiency, flexibility, and accessibility from anywhere.
- The evolution of computing laid the foundation for modern cloud technologies.

SGOU

Objective Type Questions

1. What type of computer was mainly used in the early days of computing?
2. Which computing model allows resource sharing among multiple systems?
3. Which computing model focuses on large-scale resource sharing across networks?
4. What technology enables the creation of virtual machines on a single physical computer?
5. Which version of the web introduced user-generated and interactive content?
6. Which computing model is based on a pay-as-you-use concept?
7. What is the modern computing model that integrates all earlier models?
8. What characteristic of cluster computing ensures the system continues working if one node fails?
9. Which computing model served as the direct foundation for Infrastructure as a Service (IaaS)?

Answers to Objective Type Questions

1. Mainframe
2. Distributed
3. Grid
4. Virtualization
5. Web 2.0
6. Utility
7. Cloud
8. Fault-tolerance
9. Virtualization

SGOU

Assignments

1. Explain how cloud computing evolved from mainframe systems.
2. Describe the role of distributed computing in the development of cloud computing.
3. What is virtualization, and why is it considered the foundation of cloud computing?
4. How did Web 2.0 contribute to the growth of cloud-based applications?
5. Discuss the concept of utility computing and its importance in the emergence of modern cloud services.

Reference

1. Buyya, R., Vecchiola, C., & Selvi, S. T. (2013). *Mastering cloud computing: Foundations and applications programming*. McGraw Hill Education (India) Pvt. Ltd.
2. Sosinsky, B. (2011). *Cloud computing bible*. Wiley India Pvt. Ltd.
3. Velte, T., Velte, A., & Elsenpeter, R. (2010). *Cloud computing: A practical approach*. McGraw Hill Education.

Suggested Reading

1. Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud computing: Concepts, technology & architecture*. Pearson Education.
2. Srinivasan, A. (2014). *Cloud computing: A practical approach for learning and implementation*. Pearson Education India.

SGOU

Unit 3
Types of Cloud Deployment Models

Learning Outcomes

Upon completion of the unit, learner will be able to:

- define the concept of cloud deployment models
- describe the features of Public, Private, Hybrid, and Community Clouds
- explain the advantages and disadvantages of different cloud deployment models
- explain the factors considered when selecting a suitable cloud deployment model

Prerequisite

Think about how different organizations have different requirements when using cloud computing. A student or small business may need simple and affordable cloud services, while a bank, hospital, or government organization may require strict security, privacy, and control over its data. Some organizations may need to handle sudden increases in workload, while others may need to keep certain applications within their own infrastructure.

You may already be familiar with cloud computing and the basic resources it provides, such as servers, storage, databases, networking, and applications. You may also have a basic understanding of virtualization and how cloud resources can be accessed through the Internet. These concepts provide a foundation for understanding why cloud resources can be organized and managed in different ways.

This topic takes these familiar concepts further by exploring Public, Private, Hybrid, and Community Cloud deployment models. You will see how each model differs in terms of ownership, access, security, control, cost, scalability, and management, and how these differences help organizations choose a suitable deployment model for their specific needs.

Keywords

Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud, Cloud Security

SGOU

Discussion

1.3.1 Introduction to Cloud Deployment Models

Cloud computing provides on-demand access to computing resources such as servers, storage, databases, networking, software, and applications through the Internet. These resources can be organized and managed in different ways according to the requirements of an organization. The method used to organize, manage, and provide access to cloud resources is known as a cloud deployment model.

A cloud deployment model defines how cloud infrastructure and services are hosted, owned, managed, and made available to users. It determines who can access the cloud resources, who manages the infrastructure, how data is stored, and the level of control an organization has over its cloud environment.

The four primary types of cloud deployment models are shown in below figure. 1.3.1.

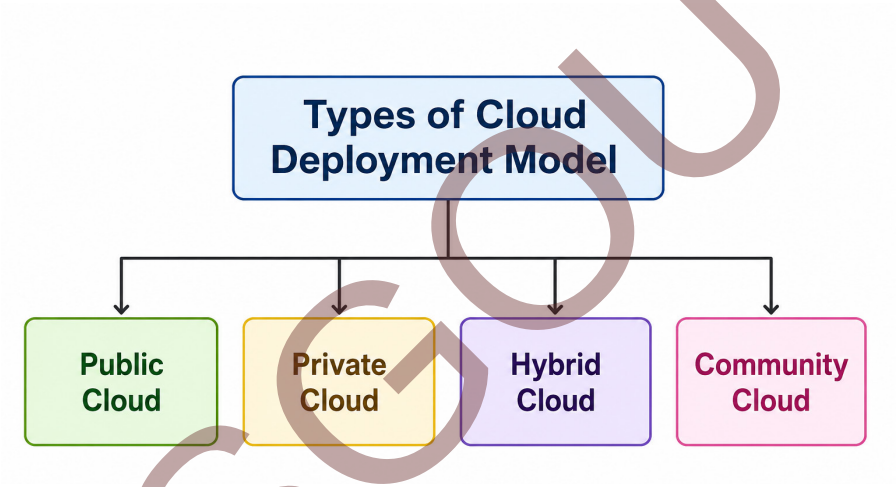


Fig. 1.3.1 Types of Cloud Deployment Model

1.3.2 Public Cloud

A Public Cloud is a cloud deployment model in which computing resources and services are provided by a third-party cloud service provider over the Internet. The cloud infrastructure is owned and managed by the provider, while the resources are made available to multiple users or organizations.

In a public cloud, users can access services such as computing power, storage, databases, networking, and software without having to purchase and maintain the underlying physical infrastructure. Resources are generally allocated according to the user's requirements, and the user typically pays based on the resources or services used.

Public cloud providers manage the physical servers, data centers, networking equipment, storage systems, maintenance, and other underlying infrastructure. Users are responsible for managing their applications, data, and configurations according to the services they use.

SGOU

Suppose a small company wants to host its website. Instead of purchasing physical servers and setting up its own data center, the company can rent computing and storage resources from a public cloud provider. The company can use the required resources through the Internet and increase or decrease them according to the website traffic.

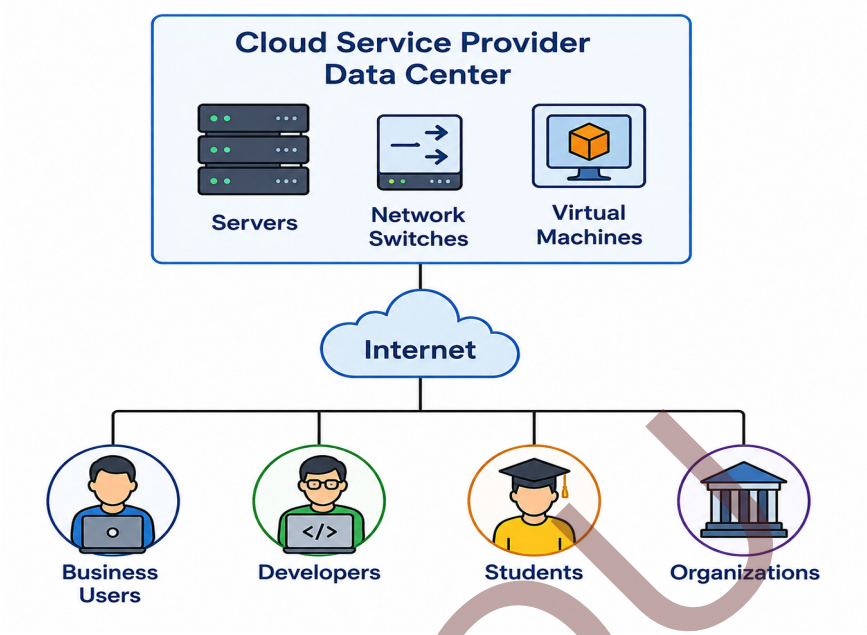


Fig. 1.3.2 Public Cloud Architecture

The figure 1.3.2 shows how a Public Cloud provides computing resources and services to multiple users through the Internet. The Cloud Service Provider Data Center contains resources such as servers, network switches, and virtual machines, which are owned and managed by the cloud service provider.

The Internet acts as the communication medium between the provider's data center and the users. Different users, such as business users, developers, students, and organizations, can access the required cloud services based on their needs.

Users do not need to purchase or maintain the physical infrastructure themselves. For example, a developer can use virtual machines for application development, while a business can use cloud servers to host its website.

The figure 1.3.3 shows three well-known public cloud providers: Amazon Web Services (AWS), Microsoft Azure, and Google Cloud. These providers offer various cloud services such as computing, storage, databases, networking, and application hosting through the Internet.

SGOU

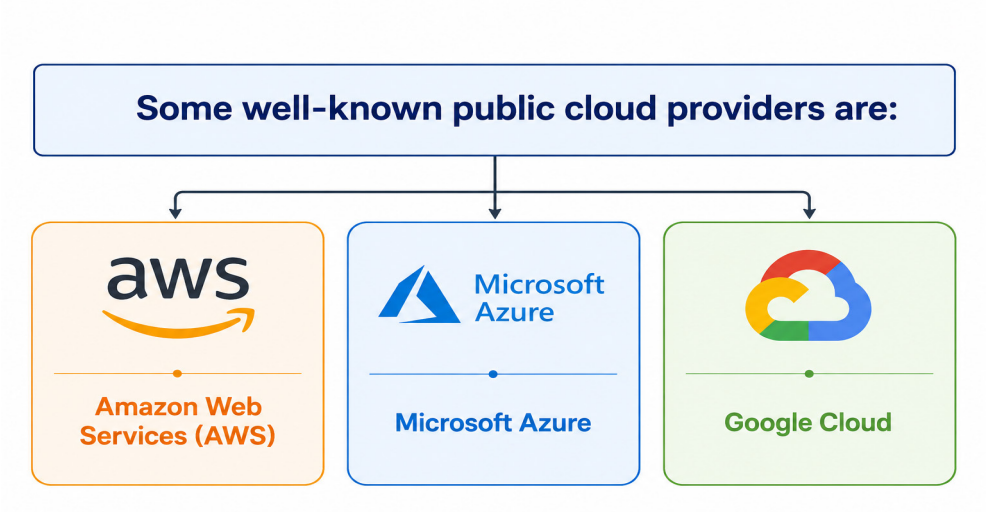


Fig. 1.3.3 Examples of Public Cloud providers

1.3.2.1 Features of Public Cloud

1. Multi-tenancy

Public clouds use a multi-tenant architecture, in which computing resources such as CPU, storage, and network infrastructure are shared by multiple customers. Although the underlying infrastructure is shared, each customer's data and resources are logically isolated using technologies such as virtualization and access controls. This efficient sharing of infrastructure improves resource utilization and helps reduce the cost of cloud services.

Example: Multiple organizations can use virtual machines on the same physical server while their applications and data remain logically separated.

2. Elasticity

Public clouds support automatic scaling of resources according to workload requirements. When application traffic increases, additional resources such as virtual machines or containers can be provisioned. When the workload decreases, the additional resources can be released. This helps maintain application performance without permanently maintaining excess infrastructure.

Example: An online shopping application can automatically obtain additional computing resources during a festival sale and release them when traffic returns to normal.

3. Pay-as-you-use

Public cloud services generally follow a pay-as-you-use pricing model, where users pay for the resources they consume, such as computing time, storage capacity, or data transfer. There is no need to make a large initial investment in physical servers and networking equipment.

Example: A company using 100 GB of cloud storage pays for the storage it uses rather than purchasing and maintaining its own storage hardware.

SGOU

4. Managed Services

The cloud provider manages much of the underlying infrastructure, including hardware maintenance, software updates, security patches, and infrastructure-level backup services, depending on the service being used. This reduces the need for organizations to maintain physical infrastructure themselves.

Example: A company using a cloud-based database service does not need to manage the physical database servers.

5. Global Availability

Public cloud providers operate data centers in different geographical regions, allowing users to deploy applications closer to their customers. This can help reduce latency and improve availability. Multiple regions can also support backup, redundancy, and disaster recovery strategies.

Example: A company serving customers in India, Europe, and the United States can deploy its application in suitable cloud regions to provide better performance to users in different locations.

6. On-Demand Self-Service

Users can provision cloud resources such as virtual machines, storage, and databases whenever required, usually through a web portal or API, without directly interacting with the cloud provider's staff.

Example: A developer can create a virtual machine within a few minutes through the cloud provider's management console.

1.3.2.2 Advantages of Public Cloud

Public Cloud offers several advantages that make it suitable for individuals, businesses, and organizations.

1. Cost-Effective

Organizations can avoid major capital expenditure (CapEx) associated with purchasing servers, storage devices, networking equipment, and other infrastructure. The cloud provider manages infrastructure maintenance, upgrades, and data-center operations. Users generally pay for the services and resources they consume, making public cloud suitable for organizations with limited initial investment.

2. High Scalability

Public cloud platforms allow users to increase or decrease computing resources quickly according to workload requirements. This flexibility helps organizations efficiently manage changing workloads without maintaining excess infrastructure.

3. Anywhere Accessibility

Since public cloud services are generally available through the Internet, users can

SGOU

access applications and data from different locations using computers, laptops, tablets, or mobile devices, subject to appropriate access permissions. This supports remote working and collaboration among geographically distributed users.

Reliability and Redundancy

Public cloud providers typically operate multiple data centers and redundant infrastructure to improve service availability. If a failure occurs in one component or location, properly designed applications can use alternative resources or locations to maintain service availability.

Rapid Deployment

Public cloud resources such as virtual machines, databases, storage, and applications can be provisioned quickly using web-based management consoles or APIs. This reduces the time required to set up infrastructure and helps organizations develop and launch applications faster.

Reduced Infrastructure Maintenance

The cloud provider is responsible for maintaining the underlying physical infrastructure, including hardware maintenance, infrastructure upgrades, and data-center operations. This reduces the organization's maintenance workload and allows it to focus on its core activities.

1.3.2.3 Disadvantages of Public Cloud

1. Security Concerns

Although public cloud providers implement various security measures, the underlying infrastructure is shared among multiple customers. Organizations may therefore have concerns about data privacy and unauthorized access, particularly when handling highly sensitive information.

2. Limited Customization

Public cloud users are generally limited to the hardware, software, and configuration options provided by the cloud provider. This may make it difficult to meet highly specialized requirements, such as custom hardware configurations or specific security mechanisms.

3. Performance Variability

Since public cloud resources operate on shared infrastructure, application performance may vary depending on workload, network conditions, and resource availability. This can sometimes result in network latency or reduced application performance.

4. Compliance and Regulatory Issues

Organizations operating in regulated industries may need to comply with specific legal, security, and data-protection requirements. Requirements related to data storage location, privacy, and data processing may affect the use of public cloud services.

SGOU

Therefore, organizations must ensure that the selected cloud services comply with applicable regulations and organizational policies.

5. Dependence on Internet Connectivity

Access to public cloud services generally depends on a reliable Internet connection. Poor or interrupted connectivity can affect the availability and performance of cloud-based applications and services.

1.3.3 Private Cloud

A Private Cloud is a cloud deployment model in which cloud infrastructure and services are dedicated to a single organization. Unlike a public cloud, the resources are not shared with unrelated organizations. The infrastructure may be owned and managed by the organization itself or by a third-party service provider.

A private cloud provides greater control, customization, security, and privacy over cloud resources. It is particularly suitable for organizations that handle sensitive data or have specific security, compliance, and performance requirements.

For example, a bank can use a private cloud to manage customer information, financial transactions, and internal banking applications. Since the infrastructure is dedicated to the bank, it can implement its own security policies, access controls, and infrastructure configurations.

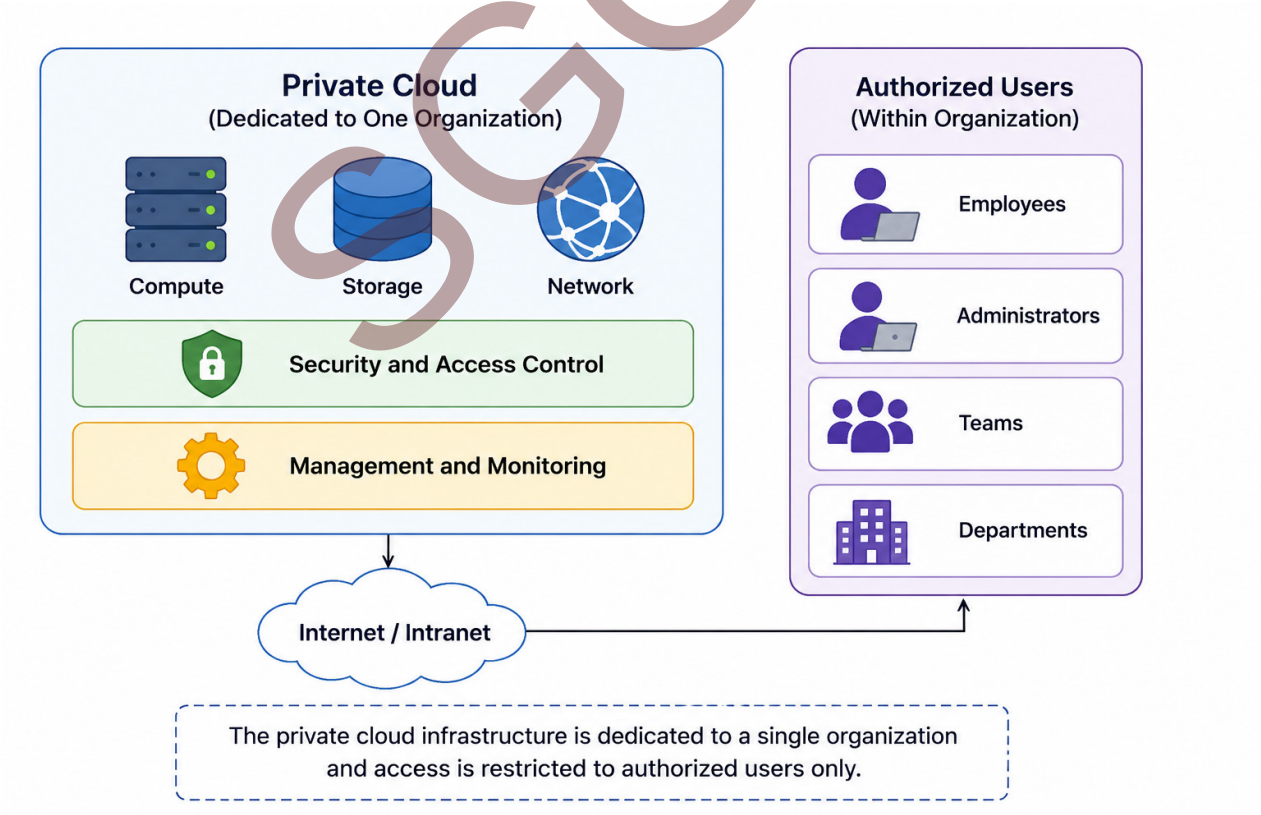


Fig. 1.3.4 Private Cloud Architecture

SGOU

The figure 1.3.4 illustrates the basic architecture of a Private Cloud, where cloud resources are dedicated to a single organization. It includes essential resources such as compute, storage, and network, along with security and access control and management and monitoring components.

The private cloud is connected through the Internet/Intranet to authorized users such as employees, administrators, teams, and departments. Access is restricted to authorized users within the organization, providing greater control, security, and privacy over cloud resources.

Examples of Private Cloud Platforms

Several platforms and solutions can be used to build or implement private cloud environments. Some well-known examples are:

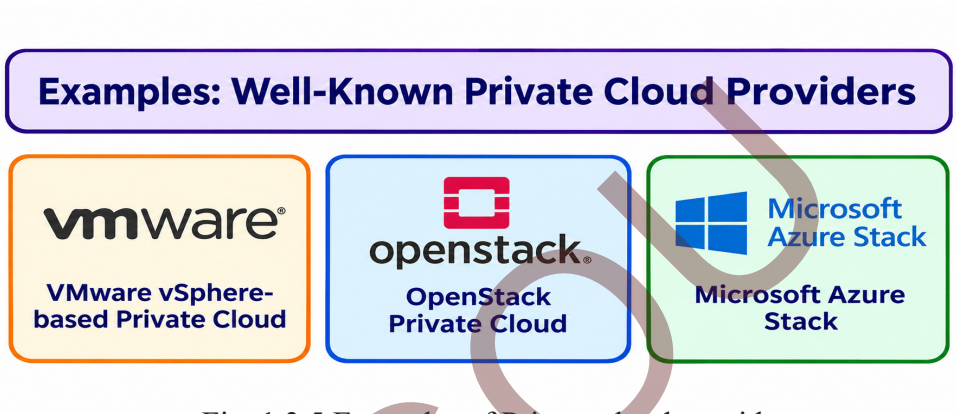


Fig. 1.3.5 Examples of Private cloud providers

1.3.3.1 Features of Private Cloud

1. Exclusive Access

A private cloud is dedicated to a single organization. External organizations do not share the cloud infrastructure, which provides greater control and reduces risks associated with multi-tenancy.

2. High Security and Control

The organization has greater control over data, networks, configurations, identity management, and security mechanisms. This allows strict implementation of organizational security policies and access controls.

3. Customization

Private clouds support customized infrastructure and configurations according to the organization's requirements. Organizations can select suitable hardware, operating systems, networking configurations, and security mechanisms.

4. On-Premise or Hosted

A private cloud can be deployed on-premises within an organization's own data center or hosted and managed by a third-party cloud provider. This provides flexibility in choosing how the infrastructure is deployed and managed.

SGOU

5. High Performance

Since the infrastructure is dedicated to a single organization, resources are not shared with other tenants. This can provide more predictable and consistent performance for applications and workloads.

Advantages of Private Cloud

1. **Enhanced Security**

Organizations can store sensitive data within their own controlled environment, reducing exposure to external threats and providing greater control over data protection.

2. **Greater Compliance**

Private cloud enables organizations to meet legal, regulatory, and industry-specific requirements related to data security, privacy, and storage.

3. **Customizable Environment**

Organizations have greater control over the cloud environment and can customize network configurations, storage policies, security settings, and resource allocation according to their requirements.

4. **Predictable Performance**

Since the infrastructure is dedicated to a single organization, resources are not shared with other users. This provides consistent performance, reliability, and lower latency for applications and workloads.

5. **Greater Control**

Organizations have greater control over data, infrastructure, security policies, access permissions, and resource management. This allows them to configure the cloud environment according to their specific requirements.

6. **Improved Data Management**

Private cloud provides greater control over data storage, access, processing, and backup. Organizations can establish their own policies for managing and protecting their data.

Disadvantages of Private Cloud

1. **High Cost**

Organizations need to invest in hardware, data-center facilities, networking equipment, cooling systems, IT staff, and maintenance, resulting in higher initial and operational costs.

2. **Limited Scalability**

Scaling the infrastructure may require purchasing and installing additional hardware, which can be time-consuming and expensive compared with public cloud environments.

SGOU

3. Maintenance Responsibility

The organization is generally responsible for system updates, security patches, hardware maintenance, troubleshooting, and repairs, unless the private cloud is managed by a third-party provider.

4. Requires Skilled IT Personnel

Managing a private cloud requires skilled technical staff to handle infrastructure, networking, security, virtualization, monitoring, and troubleshooting.

5. Infrastructure Management Complexity

Designing, deploying, and managing a private cloud can be complex and time-consuming, particularly when the organization has a large or highly customized infrastructure.

1.3.4 Hybrid Cloud

A Hybrid Cloud is a cloud deployment model that combines two or more cloud environments, typically a Public Cloud and a Private Cloud, connected through secure networking technologies. It allows organizations to keep sensitive applications and data in the private cloud while using the public cloud for less sensitive or additional workloads.

Hybrid Cloud provides a balance between the security and control of a private cloud and the flexibility and scalability of a public cloud. It is useful for organizations that experience changing workloads, require backup and disaster recovery, or need to distribute applications across different cloud environments.

The model also allows data and applications to move or share resources between the private and public cloud when required, based on organizational needs and policies.

Hybrid Cloud architecture

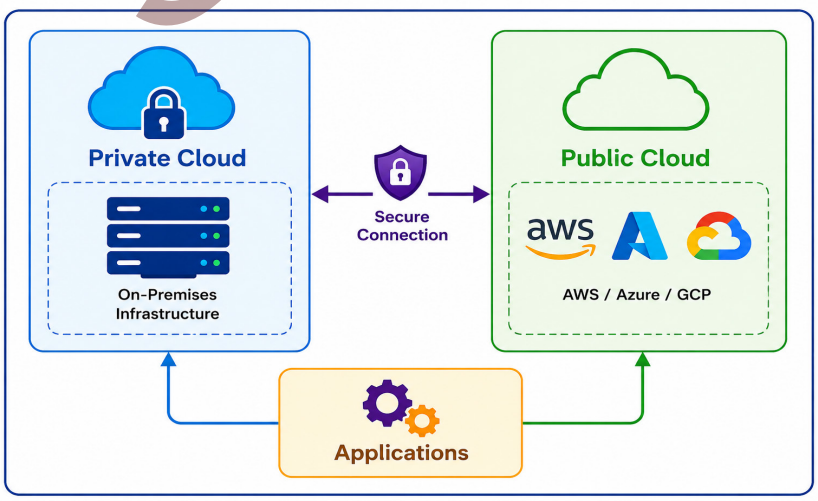


Fig. 1.3.6 Hybrid Cloud Architecture

SGOU

For example, an online shopping company can keep customer and payment information in its private cloud while using the public cloud to handle additional website traffic during major sales. This allows the organization to maintain security for sensitive data while gaining extra computing resources when needed.

The figure 1.3.6 illustrates a Hybrid Cloud architecture, where a Private Cloud and a Public Cloud are connected through a secure connection. The private cloud contains dedicated, on-premises infrastructure, while the public cloud provides scalable resources through providers such as AWS, Azure, and GCP. Applications can use resources from either environment based on organizational requirements. This architecture provides flexibility, security, scalability, and efficient resource utilization.

Features of Hybrid Cloud

1. Combination of Public and Private Cloud

Hybrid cloud combines private and public cloud environments, allowing organizations to use the advantages of both models according to their requirements.

2. Secure Connectivity

The private and public cloud environments are connected through secure networking technologies, such as VPNs, dedicated connections, or other secure communication mechanisms.

3. Flexible Workload Management

Organizations can place workloads in the appropriate cloud environment based on factors such as security, performance, cost, and resource requirements.

4. Scalability

Hybrid cloud allows organizations to use the scalable resources of the public cloud when additional computing or storage capacity is required.

5. Data and Application Portability

Data and applications can be transferred or accessed across the connected cloud environments when required, subject to security policies and technical compatibility.

6. Enhanced Security and Control

Sensitive data and critical applications can remain in the private cloud, while less sensitive workloads can use the public cloud. This provides greater control over important resources.

7. Resource Optimization

Organizations can distribute workloads between private and public clouds to achieve better utilization of resources and cost efficiency.

SGOU

Advantages of Hybrid Cloud

1. **Cost optimization:** Critical data and applications can be stored in the private cloud, while high-volume or less sensitive workloads can be handled using public cloud resources when required. This can help reduce infrastructure and operational costs.
2. **High scalability:** Public cloud resources can be used during periods of peak demand, allowing organizations to handle increased workloads without permanently expanding their private cloud infrastructure.
3. **Strong security:** Sensitive data and critical applications can be maintained in the private cloud, providing greater control over data security, privacy, and compliance requirements.
4. **Improved business continuity:** Hybrid cloud environments support effective backup and disaster recovery strategies, helping organizations maintain or restore services during system failures or unexpected events.
5. **Greater Flexibility:** Organizations can choose the most appropriate cloud environment for different applications and workloads based on security, performance, cost, and resource requirements.

Disadvantages of Hybrid Cloud

1. **Complex Management:** Managing both private and public cloud environments can be complex, as organizations need to coordinate different infrastructures, platforms, security policies, and management tools.
2. **Higher Implementation Cost:** Setting up and integrating a private cloud with public cloud services may require significant investment in hardware, networking, security, software, and skilled personnel.
3. **Integration Complexity:** Connecting different cloud environments and ensuring that applications, databases, and services work together properly can be technically challenging.
4. **Requires Skilled Personnel:** Hybrid cloud environments require professionals with knowledge of cloud infrastructure, networking, security, virtualization, and cloud management to operate and maintain them effectively.
5. **Data Management Challenges:** Managing data across multiple cloud environments can be difficult. Organizations need to ensure consistent data access, synchronization, backup, and security across both environments.

1.3.5 Community Cloud

A Community Cloud is a cloud computing model shared by several organizations that have common goals, security requirements, privacy needs, or regulatory requirements. Instead of each organization building and maintaining its own private cloud, the organizations collaborate and share a common cloud infrastructure.

SGOU

A Community Cloud may be managed by one of the participating organizations or by a third-party cloud service provider. Access to the cloud is generally restricted to authorized organizations within the community.

Community Clouds are commonly used by government departments, universities, healthcare institutions, and financial organizations that operate under a common framework or have similar requirements.

The Community Cloud model provides a balance between cost, security, and collaboration. Organizations can share infrastructure and operational costs while maintaining common security policies, access controls, and regulatory requirements.

Architecture of Community Cloud

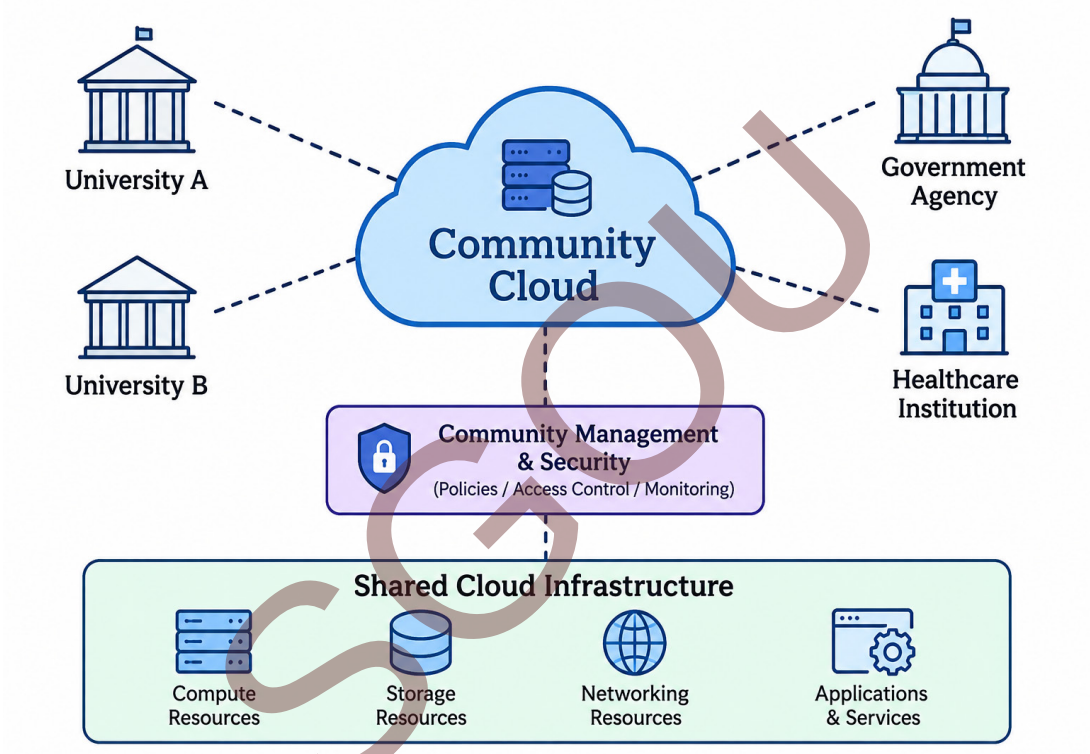


Fig. 1.3.7 Community Cloud Architecture

The Community Cloud Architecture shows how multiple organizations with common requirements can share a common cloud environment. In the diagram, University A, University B, a Government Agency, and a Healthcare Institution are connected to the Community Cloud through secure connections.

The Community Cloud provides shared computing and storage resources to all participating organizations. A Community Management and Security layer manages common policies, access control, and monitoring to ensure that only authorized users can access the resources.

The Shared Cloud Infrastructure provides compute resources, storage resources, networking resources, and applications and services. These resources are shared among the community members according to their requirements and access permissions.

Features of Community Cloud

1. **Shared Infrastructure:** A community cloud provides a common cloud infrastructure that is shared by multiple organizations belonging to the same community.
2. **Common Purpose:** A community cloud is designed for organizations that have similar business objectives, security requirements, or operational needs.
3. **Enhanced Security:** A community cloud provides security measures and access controls according to the specific security requirements of the participating organizations.
4. **Cost Sharing:** The cost of infrastructure, maintenance, and management is shared among the participating organizations, reducing the individual cost for each organization.
5. **Controlled Access:** Access to the community cloud is restricted to authorized organizations and users, ensuring that only permitted members can use the resources.
6. **Regulatory Compliance:** A community cloud helps organizations meet common legal, regulatory, and industry-specific requirements related to data security and privacy.

Advantages of Community Cloud

1. **Reduced Operational Cost:** Organizations can reduce their overall operational expenses by sharing common infrastructure, services, and management costs.
2. **Better Resource Utilization:** Shared resources can be used more efficiently, reducing the wastage of computing, storage, and networking capacity.
3. **Improved Collaboration:** Organizations can work together more effectively by using a common platform for sharing information, applications, and services.
4. **Industry-Specific Benefits:** Organizations belonging to the same industry can use cloud services designed specifically for their common business needs and practices.
5. **Simplified Data Exchange:** Data and information can be exchanged more easily between participating organizations, improving communication and workflow.

Disadvantages of Community Cloud

1. **Complex Management:** Managing a cloud environment shared by multiple organizations can be difficult because each organization may have different requirements and policies.
2. **Dependency on Other Organizations:** The performance and management

SGOU

of the cloud may depend on the cooperation and decisions of other participating organizations.

3. **Limited Flexibility:** Organizations may not always be able to make individual changes because the cloud environment is designed to meet the common requirements of the community.

4. **Potential Performance Issues:** Heavy resource usage by one or more organizations may affect the performance of shared services if resources are not properly managed.

5. **Difficult Decision-Making:** Decisions regarding policies, upgrades, costs, and resource usage may take more time because multiple organizations are involved.

1.3.6 Selection Criteria for Cloud Deployment Models

Choosing the appropriate cloud deployment model is an important decision for an organization because it affects cost, security, performance, compliance, control, and future scalability. Since the requirements of organizations differ across industries, the selection should be based on a systematic evaluation of business objectives, workload requirements, security needs, budget, technical capabilities, and existing infrastructure.

1.3.6.1 Nature of the Business and Workload Requirements

The type of business and workload requirements are important factors when selecting a cloud deployment model. Organizations with static and predictable workloads, such as informational websites, can use public or private clouds effectively. Organizations with dynamic workloads, such as e-commerce applications that experience seasonal traffic increases, can benefit from the flexibility of public or hybrid clouds.

For mission-critical applications, such as banking and healthcare systems, where reliability, security, and confidentiality are essential, private or community clouds may be more suitable. For example, a hospital managing electronic health records may prefer a private cloud because it requires consistent performance and strict protection of sensitive patient data.

1.3.6.2 Security and Compliance Requirements

Security and compliance are major factors in selecting a cloud deployment model. Organizations that handle highly sensitive information, such as financial, healthcare, or defense-related data, require strong security controls, access restrictions, and data isolation. Private or community clouds are generally suitable for such requirements.

Workloads containing moderately sensitive data, such as internal business applications, can be hosted using hybrid clouds, while low-sensitivity workloads, such as public websites and development or testing environments, can use public clouds.

Organizations must also consider regulations such as GDPR, HIPAA, and PCI-DSS, which may require strict data protection, access control, auditing, and data governance.

SGOU

When regulations require data to remain in a specific location or under controlled access, private or hybrid cloud models may be more appropriate.

1.3.6.3 Budget and Cost Considerations

Cost is an important factor when selecting a cloud deployment model. Public clouds generally follow a pay-as-you-go pricing model, which reduces the need for large initial investments in hardware and infrastructure.

Private clouds require higher initial investment in hardware, software, infrastructure, maintenance, and skilled IT staff. Hybrid clouds provide a balance by combining private infrastructure with public cloud resources.

Organizations should consider the Total Cost of Ownership (TCO), which includes infrastructure, maintenance, staffing, security, upgrades, and operational expenses. Organizations that prioritize low cost and minimal infrastructure management may prefer public clouds, while organizations requiring greater control may choose private clouds.

1.3.6.4 Control and Customization Requirements

Different cloud deployment models provide different levels of control and customization. Private clouds provide greater control over infrastructure, data, network configuration, and security policies, making them suitable for organizations with strict governance requirements.

Hybrid clouds provide a combination of control and flexibility by allowing sensitive workloads to remain in private infrastructure while less-critical workloads can run on public cloud platforms.

Public clouds provide less direct control because the infrastructure is managed by the cloud service provider. Organizations requiring dedicated hardware, customized network configurations, or specialized applications may therefore prefer private or hybrid cloud models.

1.3.6.5 Performance and Reliability Requirements

Performance and reliability depend on factors such as network latency, resource availability, data center location, and infrastructure capacity. Public cloud providers generally offer high availability, distributed infrastructure, load balancing, and disaster recovery facilities across multiple locations.

Private cloud performance depends largely on the organization's infrastructure, resource capacity, and IT management capabilities. For example, a global e-commerce company that requires fast access for customers in different regions may benefit from a public or hybrid cloud with geographically distributed infrastructure.

1.3.6.6 Scalability Requirements

Scalability is an important consideration when selecting a cloud deployment model. Organizations experiencing sudden or unpredictable increases in demand can benefit from the elastic scaling capabilities of public clouds.

SGOU

Private clouds are generally more suitable for organizations with stable and predictable growth, where resources can be planned in advance. Hybrid clouds provide greater flexibility through cloud bursting, where additional workloads can temporarily move to public cloud resources during periods of high demand.

For example, during a festive sales period, an e-commerce organization can use public cloud resources to handle increased customer traffic while keeping its core systems in a private cloud.

1.3.6.7 Data Location and Residency Requirements

Data location and residency requirements are influenced by data protection laws and data sovereignty regulations. Some organizations may be required to store sensitive information within a particular country or geographical region.

Government organizations and businesses handling highly sensitive data may prefer private clouds when strict data residency is required. Organizations operating internationally can use public cloud services that provide data centers in different geographical regions.

Community clouds can be suitable when organizations within the same industry, such as banks, healthcare institutions, or research organizations, need to follow common data residency, security, and regulatory requirements.

1.3.6.8 Integration with Existing Infrastructure

Many organizations already use on-premise infrastructure, legacy applications, databases, and security systems. The ability to integrate these existing systems with cloud services is therefore an important selection criterion.

A hybrid cloud is particularly suitable because it allows organizations to connect their existing on-premise infrastructure with public cloud services. Secure technologies such as VPNs and dedicated network connections can be used to support communication between the different environments.

1.3.6.9 Vendor Lock-In Concerns

Vendor lock-in occurs when an organization becomes highly dependent on a particular cloud provider's services, tools, technologies, or architecture. This can make it difficult and costly to move applications and data to another provider.

Organizations that require greater flexibility and portability may choose hybrid or multi-cloud strategies to reduce dependence on a single provider. Hybrid cloud environments can allow workloads to be moved between private and public environments when required.

1.3.6.10 IT Skill Level and Management Capabilities

The availability of skilled IT professionals is another important factor in selecting a cloud deployment model. Organizations with limited IT staff or technical expertise may prefer public cloud services because the cloud provider manages much of the

SGOU

infrastructure, maintenance, updates, and security.

Organizations with experienced IT teams can manage private or hybrid cloud environments more effectively. Before selecting a deployment model, organizations should evaluate their expertise in cloud administration, cybersecurity, infrastructure management, and system integration.

1.3.6.11 Disaster Recovery and Business Continuity Needs

Disaster recovery and business continuity requirements should also be considered when selecting a cloud deployment model. Organizations should identify objectives such as the Recovery Time Objective (RTO) and Recovery Point Objective (RPO) to minimize service interruptions and data loss.

Public and hybrid clouds can provide access to geographically distributed infrastructure, backup facilities, and disaster recovery services, which can help organizations recover quickly from system failures or disasters.

In a private cloud, the organization is generally responsible for designing, implementing, and maintaining its own backup and disaster recovery infrastructure. This can increase the cost, complexity, and management requirements of the deployment.

Recap

- Cloud deployment models define how cloud infrastructure and services are hosted, owned, managed, and accessed.
- Public Cloud provides shared cloud resources through a third-party cloud service provider.
- Key Public Cloud features include multi-tenancy, elasticity, pay-as-you-use, managed services, global availability, and on-demand self-service.
- Public Cloud offers cost efficiency, scalability, accessibility, reliability, rapid deployment, and reduced infrastructure maintenance.
- Private Cloud provides dedicated cloud infrastructure for a single organization with greater security, control, privacy, and customization.
- Private Cloud features include exclusive access, high security, customization, flexible deployment, and predictable performance.
- Hybrid Cloud combines Private Cloud and Public Cloud to provide security, flexibility, and scalability.
- Hybrid Cloud supports flexible workload management, secure connectivity, resource optimization, and data/application portability.
- Community Cloud is shared by organizations with common goals, security requirements, privacy needs, or regulatory requirements.

SGOU

- Community Cloud supports cost sharing, collaboration, controlled access, better resource utilization, and regulatory compliance.
- Each deployment model has its own advantages and disadvantages based on cost, security, control, scalability, and management requirements.
- Selection of a suitable deployment model depends on workload, security, compliance, budget, control, performance, scalability, data location, integration, IT skills, and disaster recovery needs.

Objective Type Questions

1. Which cloud deployment model provides resources to multiple users through a third-party provider?
2. Which cloud deployment model is dedicated to a single organization?
3. Which cloud deployment model combines public and private cloud environments?
4. Which cloud deployment model is shared by organizations with common requirements?
5. What feature of public cloud allows multiple customers to share the underlying infrastructure?
6. Which public cloud feature allows users to pay only for the resources they consume?
7. Which public cloud feature allows users to provision resources whenever required without directly contacting the provider?
8. Which factor refers to the ability of a cloud environment to handle increasing workloads?
9. Which cloud model allows additional public cloud resources to be used during periods of high demand?

SGOU

Answers to Objective Type Questions

1. Public Cloud
2. Private Cloud
3. Hybrid Cloud
4. Community Cloud

5. Multi-tenancy
6. Pay-as-you-use
7. On-Demand Self-Service
8. Scalability
9. Hybrid Cloud

Assignments

1. Explain the four major cloud deployment models with suitable examples.
2. Explain the features, advantages, and disadvantages of Public Cloud.
3. Describe the features, advantages, and disadvantages of Private Cloud.
4. Explain the features and advantages of Hybrid Cloud. Also mention its disadvantages.
5. Explain the important factors to be considered when selecting a suitable cloud deployment model.

Reference

1. https://www.geeksforgeeks.org/cloud-computing/cloud-computing/
2. https://nptel.ac.in/courses/106105167

Suggested Reading

1. Hiran, K. K., Doshi, R., Fagbola, T., & Mahrishi, M. (2019). *Cloud Computing: Master the Concepts, Architecture and Applications with Real-World Examples and Case Studies* (1st ed.). BPB Publications.
2. Jamsa, K. (2013). *Cloud Computing* (1st ed.). Jones & Bartlett Student Edition.

SGOU

Unit 4
Components of Cloud Computing

Learning Outcomes

Upon completion of the unit, learner will be able to:

- identify the major components of cloud computing
- describe the functions of front-end, network, cloud platform, virtualization, servers, storage, and databases
- explain the basic infrastructure of cloud computing and its features
- describe how cloud components work together to process user requests
- explain the advantages of basic cloud infrastructure

Prerequisite

Think about the many online services you use every day. When you store a document in Google Drive, send an email, access an online learning platform, watch a video, or use a web-based application, you are using services that depend on computing resources located somewhere beyond your own device. Although the process appears simple from the user's side, several components work together in the background to receive your request, process it, store or retrieve data, and send the required result back to you.

You may already be familiar with computers, web browsers, mobile applications, the Internet, and basic networking. You may also know that servers are used to provide services and that data can be stored on computers or storage devices. These basic ideas provide a foundation for understanding cloud computing.

In this unit, you will go behind these familiar online activities to understand what happens inside a cloud environment. You will explore the major components of cloud computing, such as users, front-end systems, networks, cloud platforms, virtualization, servers, storage, databases, and cloud services. You will also learn about basic cloud infrastructure, its important features, how its components work together to fulfil a user's request, and the advantages it provides. This will help you understand how the cloud services you use every day are actually delivered.

SGOU



Keywords

Virtualization, Cloud Services, Cloud Platform, Network Infrastructure, Cloud Storage, IaaS, PaaS, SaaS

Discussion

1.4.1 Introduction to Components of Cloud Computing

Cloud computing is a combination of several hardware, software, networking, and virtualization components that work together to provide computing resources and services over the internet. These components enable users to access computing power, storage, applications, databases, and networking resources whenever required.

The major components of cloud computing can be broadly divided into front-end components, back-end components, cloud-based service delivery components, and network infrastructure. In addition to these components, cloud computing relies on basic infrastructure such as data centers, physical servers, storage systems, virtualization technologies, hypervisors, and virtual machines.

The front-end components allow users to interact with cloud services, while the back-end components provide the computing resources required to process user requests. Network infrastructure connects users with cloud resources, and virtualization allows physical resources to be efficiently divided and managed as virtual resources.

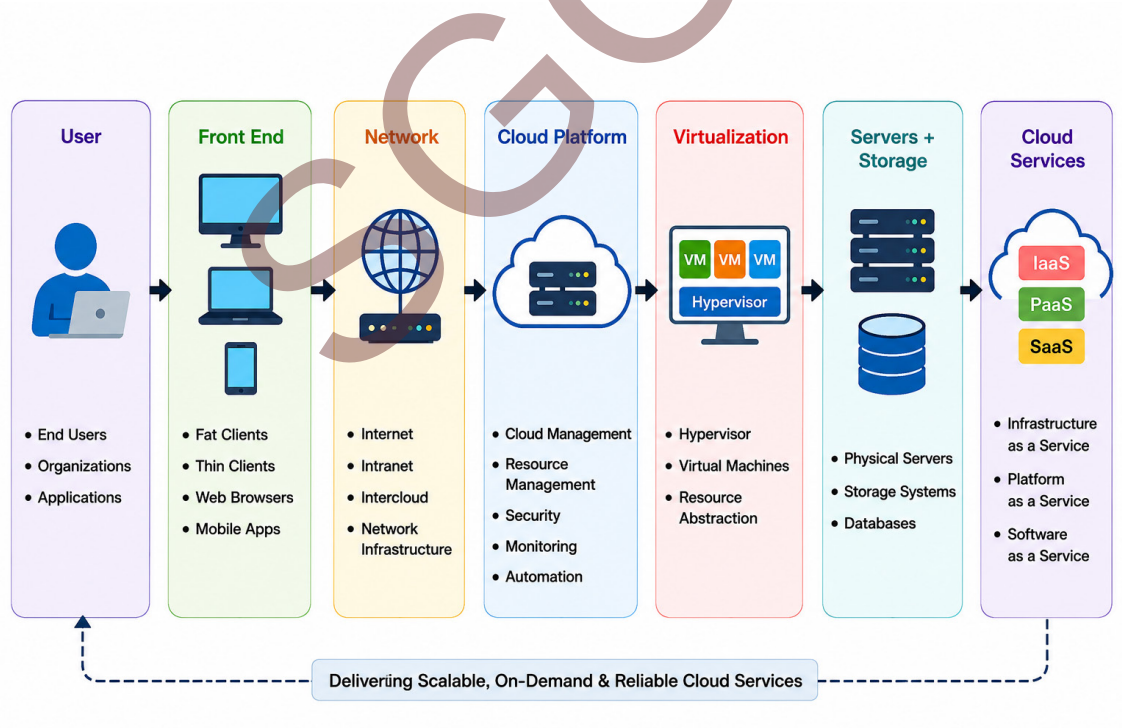


Fig. 1.4.1 Components of cloud computing

1. User

The user is the individual, organization, or application that accesses and uses cloud computing resources and services. A user may request different types of services, such as storing files, running an application, processing data, or accessing an online software application.

The user does not necessarily need to know where the physical servers or storage systems are located. Cloud computing hides much of the underlying infrastructure and allows users to access resources through a simple interface.

Types of Cloud Users

1. **Individual Users** – People who use cloud services for personal activities such as storing photos, documents, and other files.
2. **Business Users** – Organizations that use cloud computing for applications, databases, data storage, communication, and business operations.
3. **Developers** – Software developers who use cloud platforms to develop, test, and deploy applications.
4. **Applications** – Software applications can also act as cloud service consumers by requesting computing resources or data from other cloud services.

2. Front-End

The front end is the user-facing part of a cloud computing environment. It includes the devices, applications, interfaces, and software through which users access cloud services.

The front end acts as the connection point between the user and the cloud environment. When a user performs an action, such as opening a cloud application or uploading a file, the front-end application sends the request to the cloud through the network.

Major Front-End Components

A. Fat Client (Thick Client)

A fat client, also called a thick client, is a device or application that performs a significant amount of processing locally.

It generally has:

- High processing capability
- Considerable memory
- Large local storage
- Locally installed applications

A fat client can perform many operations without continuously depending on cloud servers.

SGOU

Example:

A desktop computer running locally installed video-editing software is a typical example of a fat-client environment.

B.Thin Client

A thin client is a lightweight device or application that depends heavily on remote servers for processing and storage.

It generally has:

- Limited processing capability
- Limited local storage
- Less powerful hardware
- Greater dependence on network connectivity

Example:

A user accessing a cloud-based application through a web browser can use a thin-client approach because most application processing takes place on the cloud server.

C. Web Browsers

Web browsers provide an interface for accessing cloud-based applications and services.

Examples:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

Users can access services such as online office applications, learning platforms, cloud storage, and web-based business applications through browsers.

D. Mobile Applications

Mobile applications allow users to access cloud services through smartphones and tablets. These applications communicate with cloud servers through the internet or mobile networks.

Example:

A mobile application used to access cloud-based banking or file-storage services communicates with the corresponding cloud backend.

3. Network

A network is an essential component of cloud computing that provides communication

SGOU

between users, applications, cloud platforms, servers, and storage systems. It allows requests, data, and responses to travel between users and cloud resources.

Since most cloud resources are located in remote data centers, reliable network connectivity is necessary for accessing cloud services.

Functions of Network in Cloud Computing

- **Connectivity:** Connects users with cloud services.
- **Data Transfer:** Transfers files and other data between users and cloud resources.
- **Resource Access:** Enables access to remote computing, storage, and applications.
- **Communication:** Allows servers, applications, and databases to communicate.
- **Security:** Supports secure communication through firewalls, encryption, and access controls.
- **Remote Access:** Allows users to access cloud services from different locations.
- **Load Distribution:** Helps distribute requests among multiple servers.

Major Network Components

A. Internet

The Internet is a global public network that allows users to access cloud services from different locations.

It enables:

- Remote access to cloud applications
- Data transfer
- Communication between users and cloud servers
- Access to public cloud platforms

Example:

A user accessing Google Drive from a home computer uses the Internet to communicate with the cloud service.

B. Intranet

An intranet is a private network used within an organization. It provides controlled access to internal applications, databases, files, and private cloud resources.

SGOU

Features:

- Provides secure internal connectivity.
- Restricts access to authorized users.
- Supports internal applications and databases.
- Allows employees to access private cloud resources.

Example

Employees accessing an organization's private cloud-based ERP system through the organization's internal network.

C. Intercloud

The intercloud refers to interconnected cloud environments that allow different cloud platforms or providers to communicate and work together.

It supports:

- Workload migration
- Resource sharing
- Multi-cloud operations
- Backup and disaster recovery
- Improved service availability

Example:

An organization may host its main application on one cloud provider and maintain its backup data on another cloud provider.

D. Network Infrastructure

Network infrastructure includes the physical and virtual components that provide connectivity and communication within cloud environments.

Major components include:

1. **Routers:** Connect different networks and determine the path for data transmission.
2. **Switches:** Connect servers and other devices within a network and forward data to the appropriate destination.
3. **Firewalls:** Monitor and control network traffic to protect cloud resources from unauthorized access.
4. **Load Balancers:** Distribute incoming requests across multiple servers to improve performance and availability.

SGOU

5. **Network Interfaces:** Provide connectivity between servers, virtual machines, storage systems, and networks.
6. **Virtual Networks:** Provide logically isolated networks for cloud resources and allow organizations to control communication between them.
7. **VPN:** Provides a secure connection between users or organizational networks and cloud resources over a public network.

4. Cloud Platform

The cloud platform is the core software environment that manages and coordinates cloud resources and services. It acts as an intermediate layer between the users and the underlying cloud infrastructure, ensuring that computing, storage, networking, and application resources are available when required.

Major Functions of a Cloud Platform

1. **Resource Management** – Manages computing, storage, and networking resources and allocates them according to application requirements.
2. **Resource Provisioning** – Creates and configures cloud resources such as virtual machines, storage, and networks when users request them.
3. **Security Management** – Protects cloud resources using authentication, authorization, access controls, encryption, and security policies.
4. **Monitoring** – Monitors resource usage, system performance, availability, and application health.
5. **Automation** – Automatically performs tasks such as resource allocation, scaling, backup, and system configuration.
6. **User Management** – Manages users, accounts, roles, permissions, and access to cloud services.
7. **Service Management** – Coordinates the delivery and operation of cloud services such as IaaS, PaaS, and SaaS.

Example

When a user requests a virtual machine, the cloud platform receives the request, checks the available resources, creates or assigns the required virtual machine, allocates CPU and memory, and provides access to the user.

5. Virtualization

Virtualization is a technology that creates virtual versions of physical computing resources such as servers, storage, and networks. It allows a single physical server to run multiple virtual machines (VMs), each functioning as an independent computing environment.

Virtualization is a fundamental technology in cloud computing because it improves resource utilization, flexibility, scalability, and cost efficiency.

SGOU

Main Functions of Virtualization

1. **Resource Sharing** – Allows multiple virtual machines to share the resources of a single physical server.
2. **Resource Utilization** – Improves the use of CPU, memory, storage, and other physical resources.
3. **Isolation** – Keeps virtual machines independent so that problems in one VM do not directly affect other VMs.
4. **Resource Allocation** – Allows computing resources to be allocated according to the requirements of different workloads.
5. **Rapid Provisioning** – New virtual machines can be created and configured quickly.
6. **Scalability** – Resources can be adjusted according to changing workload requirements.

6. Servers

A server is a computer system that provides computing resources and services to other computers, applications, or users over a network. In cloud computing, servers form the physical computing foundation on which applications, virtual machines, and cloud services operate.

Cloud providers use powerful servers in data centers to process user requests, run applications, manage databases, and provide computing resources on demand.

How Virtualization Works

In a traditional computing environment, a physical server may be dedicated to a single application, which can result in unused resources. Virtualization divides the physical resources of a server into multiple virtual machines (VMs). Each virtual machine operates as an independent computing environment with its own operating system and applications. A hypervisor manages the physical server and distributes resources such as CPU, memory, storage, and network resources among the virtual machines according to their requirements. This allows multiple applications or workloads to run efficiently on the same physical server.

The figure 1.4.2 illustrates how a hypervisor enables multiple virtual machines to share the resources of a single physical computer. Each virtual machine runs its own operating system and applications, while the hypervisor manages access to the underlying hardware resources.

SGOU

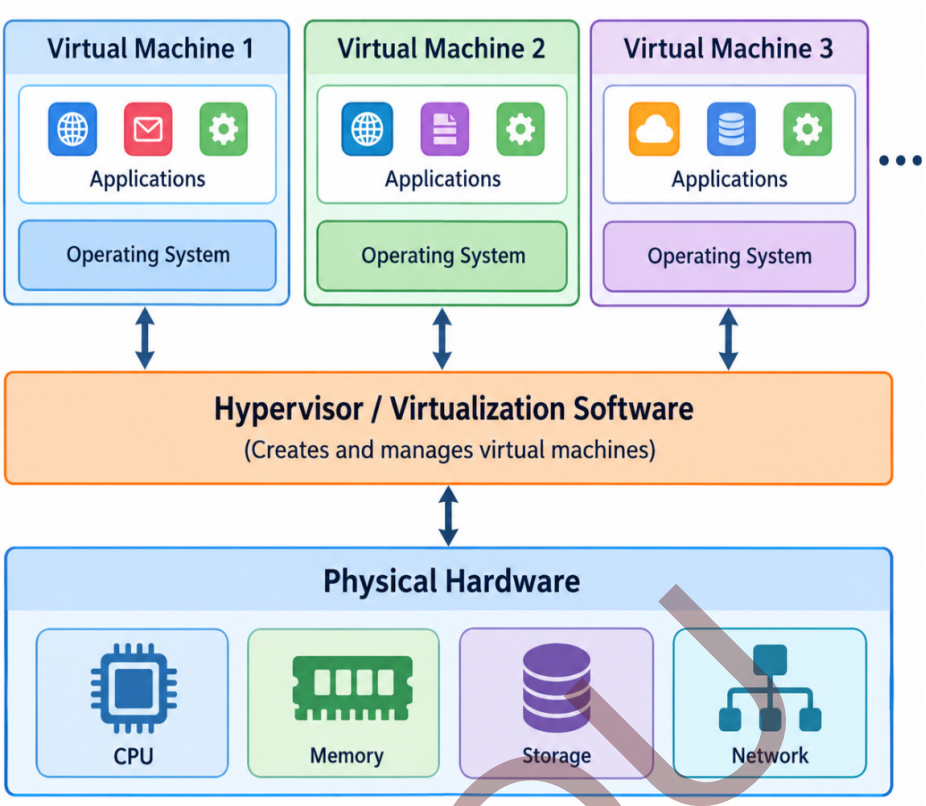


Fig. 1.4.2 How virtualization works

Functions of Servers in Cloud Computing

1. **Processing Data** – Servers provide CPU and memory resources required to process user requests and application workloads.
2. **Running Applications** – Cloud applications and services are executed on servers.
3. **Hosting Virtual Machines** – Physical servers provide the hardware resources required to run multiple virtual machines through virtualization.
4. **Managing Data** – Servers work with storage systems and databases to store, retrieve, and process data.
5. **Handling User Requests** – Servers receive requests from users and applications and return the required responses.
6. **Supporting Scalability** – Additional servers can be added when more computing resources are required.

Types of Servers

1. **Web Server** – Receives web requests and delivers web pages or web content to users.

SGOU

2. **Application Server** – Runs application programs and handles application-related requests.
3. **Database Server** – Stores, manages, and processes data in databases.
4. **File Server** – Provides centralized storage and access to files.
5. **Virtualization Server** – Provides physical resources for creating and running virtual machines.

Example

When a user accesses an online shopping application, the application server processes the user's request, the database server retrieves product and customer information, and the web server delivers the required information to the user's device.

7. Storage

Cloud storage is a service that provides space for storing and managing data on cloud infrastructure. It allows users and applications to store files, documents, images, videos, databases, backups, and other digital information without depending entirely on local storage devices.

Cloud storage is designed to provide scalability, availability, reliability, and flexible access to data.

Functions of Cloud Storage

1. **Data Storage** – Stores files, application data, databases, backups, and other digital information.
2. **Scalability** – Storage capacity can be increased or decreased according to the requirements of users and applications.
3. **Data Access** – Authorized users and applications can access stored data through a network from different locations.
4. **Data Backup** – Important data can be copied and stored for recovery in case of accidental deletion or system failure.
5. **Data Replication** – Multiple copies of data can be maintained to improve availability and reliability.
6. **Data Management** – Provides mechanisms for organizing, managing, and controlling access to stored data.

Types of Cloud Storage

a. **Object Storage** – Stores data as objects and is suitable for large amounts of unstructured data such as images, videos, documents, and backups.

b. **Block Storage** – Provides storage volumes that can be attached to virtual machines and is commonly used for operating systems and databases.

SGOU

c. **File Storage** – Provides shared file-based storage that can be accessed by multiple users or applications.

8. Databases

A database is an organized collection of data that allows information to be stored, managed, updated, and retrieved efficiently. In cloud computing, databases are hosted and managed within cloud infrastructure, allowing applications to access data whenever required.

Cloud databases are widely used for storing customer information, transaction records, application data, student records, product details, and other structured or unstructured information.

Functions of Cloud Databases

1. **Data Storage** – Stores large amounts of application and organizational data.
2. **Data Retrieval** – Allows applications and authorized users to quickly retrieve required information.
3. **Data Management** – Supports adding, modifying, deleting, and organizing data.
4. **Data Security** – Provides access controls and security mechanisms to protect stored information.
5. **Backup and Recovery** – Supports backup of important data and recovery in case of data loss or system failure.
6. **Scalability** – Database resources can be increased according to the growth of data and application requirements.
7. **High Availability** – Replication and other techniques can help keep database services available even when individual components fail.

9. Cloud Services

Cloud services are computing resources, platforms, and software applications provided to users through the cloud. They allow users and organizations to access required resources over a network without purchasing and maintaining the complete physical infrastructure themselves.

Cloud services are mainly classified into Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

1. IaaS – Infrastructure as a Service

Infrastructure as a Service (IaaS) provides virtualized computing resources through the cloud. It offers basic infrastructure components such as virtual machines, storage, and networking, allowing organizations to use and manage computing resources without purchasing and maintaining physical hardware. The cloud provider manages

SGOU

the underlying physical infrastructure, while users can configure and manage the operating systems, applications, and other resources according to their requirements.

Examples of IaaS: Amazon EC2 (AWS), Azure Virtual Machines, and Google Compute Engine.

2. PaaS – Platform as a Service

Platform as a Service (PaaS) provides a cloud-based platform for developing, testing, and deploying applications. It includes tools and services such as runtime environments, databases, and middleware. Developers can use these facilities to build and run applications without having to manage the underlying hardware, operating systems, and other infrastructure components.

Examples of PaaS include: Google App Engine, Microsoft Azure App Service, and AWS Elastic Beanstalk.

3. SaaS – Software as a Service

Software as a Service (SaaS) provides ready-to-use software applications over the internet. Users can access these applications through web browsers or mobile applications without having to install or maintain the software on their devices. The cloud provider manages the software, including its maintenance, updates, and availability.

Examples of SaaS include: Google Workspace, Microsoft 365, and Salesforce.

1.4.2 Basic Cloud Infrastructure

Basic Cloud Infrastructure refers to the fundamental technologies and resources required to provide cloud computing services. It consists of physical hardware, virtualized resources, networking components, storage systems, and management tools that work together to deliver cloud services.

This infrastructure enables users to access computing power, servers, storage, applications, and network resources whenever required, without having to purchase, operate, or maintain the underlying physical hardware themselves.

1.4.2.1 Features of Basic Cloud Infrastructure

Basic cloud infrastructure provides several important features that help cloud services deliver computing resources in a flexible, reliable, and efficient manner. The major features are:

- **Virtualization:** Allows multiple virtual machines to run on a single physical server, making better use of available hardware resources.
- **Scalability:** Allows computing resources such as processing power, memory, and storage to be increased or decreased according to the requirements.
- **Resource Pooling:** Enables computing, storage, and networking resources to be shared among multiple users while keeping their resources logically separated.

SGOU

- **High Availability:** Uses backup and redundant resources to keep cloud services available and reduce service interruptions.
- **Elasticity:** Automatically adds or removes resources when the workload changes, helping the system handle changing demands efficiently.
- **Measured Service:** Monitors resource usage and allows users to be charged based on the resources they consume.
- **Global Access:** Allows users to access cloud resources and services from different locations through an internet connection.

1.4.2.2 How Components Work Together

Basic cloud infrastructure works by connecting users with computing, storage, and networking resources available in the cloud. When a user requests a cloud service, the request passes through the network and is processed by the cloud infrastructure. The required resources are then allocated, the requested task is performed, and the result is returned to the user.

The following figure 1.4.3 illustrates how the different components of basic cloud infrastructure work together to process a user's request and deliver the required service.

Step 1: User Request

The process begins when a user, organization, or application sends a request for a cloud service.

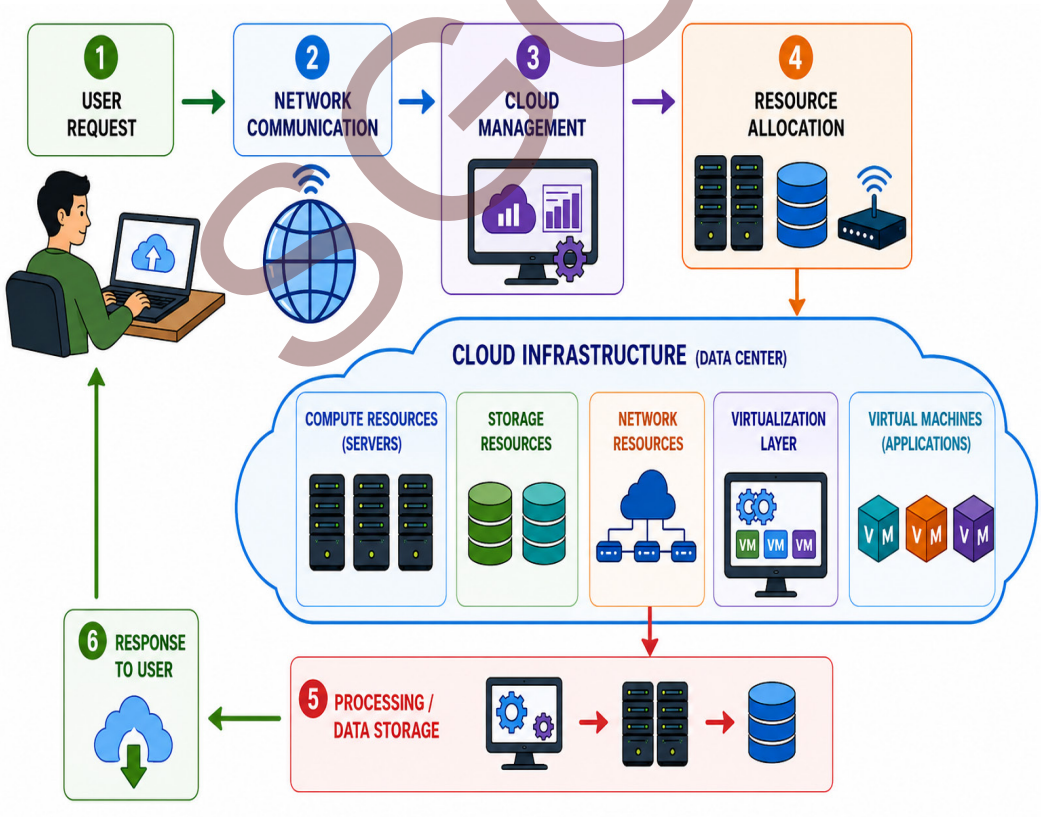


Fig. 1.4.3 Working of Basic Cloud Infrastructure

SGOU

The user may request services such as:

- Storing a file
- Running an application
- Processing data
- Accessing a database
- Using an online software application

The request is made through a web browser, mobile application, or other client application.

Step 2: Network Communication

The user's request is transmitted through a network, usually the Internet, to the cloud environment. The network provides communication between the user's device and the cloud infrastructure. It carries the user's request to the cloud and later carries the response back to the user.

Example:

When a user uploads a file to cloud storage, the file and the upload request are transmitted through the Internet to the cloud.

Step 3: Cloud Management

The request reaches the cloud management system, which manages and coordinates the available cloud resources. It receives the user's request, authenticates the user, checks the access permissions, identifies the resources required to fulfil the request, and verifies their availability. Based on these requirements, the cloud management system determines the appropriate resources needed to process the user's request.

Step 4: Resource Allocation

After understanding the user's request, the cloud management system allocates the required resources from the available resource pool. These resources include compute resources such as CPU and memory, storage resources for storing data, network resources for communication, and virtual machines for running applications.

Virtualization allows physical resources to be divided into multiple virtual resources and allocated to different users or applications according to their requirements. This helps in efficient resource utilization, reduces resource wastage, and allows multiple users to share the same physical infrastructure.

Step 5: Processing and Storage

After the required resources are allocated, the cloud infrastructure starts processing the user's request. The allocated servers, virtual machines, storage systems, databases, and network resources work together to complete the requested task.

SGOU

A server or virtual machine provides the computing power needed to run applications and perform calculations. For example, when a user opens a cloud-based application, the server processes the user's request and runs the required application. If the request involves storing or retrieving a file, the cloud storage system performs the required operation. Similarly, a cloud database stores, retrieves, and manages the information required by an application.

The network resources help different cloud components communicate with each other. For example, the server may need to communicate with a database to retrieve information and with a storage system to access a file.

During this step, the cloud infrastructure may perform several operations such as processing data, running applications, storing files, retrieving information, and managing database records. Once the requested operation is completed, the required result or data is prepared to be sent back to the user.

Step 6: Response to User

After the requested operation is completed, the result is sent back to the user through the network. The cloud system prepares the required information and sends it from the cloud infrastructure to the user's device.

The response may include requested data, a processed result, a downloaded file, an application output, or a confirmation that the requested operation has been completed. The user can then view or use the result through a web browser, mobile application, or other client application.

Example: When a student uploads an assignment to cloud storage, after the file is successfully stored, the cloud service sends a confirmation message to the student indicating that the upload was completed successfully.

1.4.2.3 Advantages of Basic Cloud Infrastructure

Basic cloud infrastructure provides several advantages by allowing computing, storage, networking, and other resources to be managed efficiently. The major advantages are:

1. Efficient Resource Utilization

Cloud infrastructure uses virtualization and resource pooling to make better use of physical resources. Multiple users and applications can share the available computing, storage, and network resources without requiring separate physical systems.

2. Scalability

Cloud infrastructure allows resources such as CPU, memory, storage, and network capacity to be increased or decreased according to requirements. This helps the system handle changes in workload efficiently.

3. Cost Efficiency

Organizations can use cloud resources without purchasing and maintaining large

SGOU

amounts of physical hardware. Resources can be provided according to requirements, which helps reduce infrastructure and maintenance costs.

4. Flexibility

Cloud infrastructure provides flexible access to computing resources and services. Users can obtain the required resources whenever needed and from different locations through a network.

5. High Availability

Cloud infrastructure uses backup systems, redundancy, and resource replication to keep services available. If one component fails, another available component can help continue the service.

6. Easy Management

Cloud infrastructure provides management tools for monitoring, configuring, allocating, and controlling resources. This makes it easier for cloud providers and organizations to manage their infrastructure.

7. Better Performance

Cloud infrastructure can distribute workloads across multiple servers and resources. This helps improve processing performance, response time, and overall system efficiency.

8. Easy Backup and Recovery

Cloud infrastructure supports data backup and recovery by maintaining copies of important data. This helps protect information against accidental deletion, hardware failure, or other system problems.

SGOU

Recap

- Cloud computing components work together to provide computing resources and services over the Internet.
- Front-end provides the interface through which users access cloud services.
- The network connects users, applications, servers, storage, and other cloud resources.
- Cloud platform manages and coordinates cloud resources and services.
- Virtualization divides physical resources into virtual resources for efficient sharing and allocation.
- Servers, storage, and databases provide essential computing and data-handling capabilities.
- IaaS, PaaS, and SaaS represent different levels of cloud services.

- Basic cloud infrastructure combines physical hardware, virtualization, networking, storage, and management tools.
- Important infrastructure features include scalability, resource pooling, elasticity, high availability, and global access.
- Cloud infrastructure processes a request through user request, network communication, cloud management, resource allocation, processing and storage, and response to the user.
- Basic cloud infrastructure supports efficient resource utilization, flexibility, cost efficiency, better performance, availability, and backup and recovery.

Objective Type Questions

1. What is the user-facing part of a cloud computing environment?
2. What type of client performs most processing locally?
3. What type of client depends heavily on remote servers for processing and storage?
4. What network allows users to access cloud services from different locations?
5. What is the private network used within an organization called?
6. What technology creates virtual versions of physical computing resources?
7. What component provides computing resources for cloud applications and services?
8. What type of storage stores data as objects?
9. What type of storage provides volumes that can be attached to virtual machines?
10. Which cloud service model provides virtual machines, storage, and networking?
11. Which cloud service model provides an environment for developing and deploying applications?
12. Which cloud service model provides ready-to-use software applications?

SGOU

Answers to Objective Type Questions

1. Front-end
2. Fat Client
3. Thin Client
4. Internet
5. Intranet
6. Virtualization
7. Server
8. Object Storage
9. Block Storage
10. Infrastructure as a Service (IaaS)
11. Platform as a Service (PaaS)
12. Software as a Service (SaaS)

Assignments

1. Explain the major components of cloud computing and describe their roles in providing cloud services.
2. Explain the basic cloud infrastructure and discuss its important features.
3. Describe the working of basic cloud infrastructure from a user request to the final response.
4. Explain virtualization in cloud computing and discuss its functions and importance.
5. Explain the major advantages of basic cloud infrastructure.

SGOU

Reference

1. https://www.geeksforgeeks.org/software-engineering/cloud-computing-infrastructure/
2. https://nptel.ac.in/courses/106105167

Suggested Reading

1. Hiran, K. K., Doshi, R., Fagbola, T., & Mahrishi, M. (2019). *Cloud Computing: Master the Concepts, Architecture and Applications with Real-World Examples and Case Studies* (1st ed.). BPB Publications.
2. Jamsa, K. (2013). *Cloud Computing* (1st ed.). Jones & Bartlett Student Edition.

SGOU



Understanding Cloud based Services

SGOU

Unit 1
SPI Service Models

Learning Outcomes

Upon completion of the unit, learner will be able to:

- identify the three cloud service models: IaaS, PaaS, and SaaS
- explain the key features and working of IaaS, PaaS, and SaaS
- describe the advantages and limitations of different cloud service models
- differentiate the responsibilities of the cloud provider and the user in each service model

Prerequisite

Think about the digital world around you. Where do your photos and documents go when you save them online? How can you use email, video conferencing, or office applications without installing complex software on your computer? And how do developers build and deploy applications without purchasing expensive servers and other infrastructure?

The answers to these interesting questions lie in the world of cloud service models. Cloud computing has changed the way we access and use technology by making computing resources, development platforms, and software available as services over the Internet. Instead of owning and maintaining everything, users can simply access the services they need.

Before beginning this unit, you only need a basic understanding of computers, the Internet, software applications, servers, and data storage. With this foundation, you are ready to explore the three major cloud service models: IaaS, PaaS, and SaaS.

As you move through this unit, you will discover who manages the infrastructure, who develops the applications, and how ready-to-use software reaches users. You will also understand how these three service models differ in terms of control, responsibility, and services provided.

Keywords

IaaS, PaaS, SaaS, Cloud Service Provider, Infrastructure, Platform, Software, Virtualization, Scalability



SGOU

Discussion

2.1.1 Introduction to Cloud Service Models

Cloud computing delivers computing resources and services over the Internet, enabling users and organizations to access IT resources without owning and maintaining the underlying physical infrastructure. Instead of investing in physical servers, storage devices, networking equipment, and software, users can obtain these resources as cloud-based services whenever they are required.

Cloud service models define the types of services offered by a cloud service provider and the level of responsibility shared between the provider and the user. Based on the level of control and management provided, cloud computing is primarily classified into three fundamental service models:

1. Infrastructure as a Service (IaaS)
2. Platform as a Service (PaaS)
3. Software as a Service (SaaS)

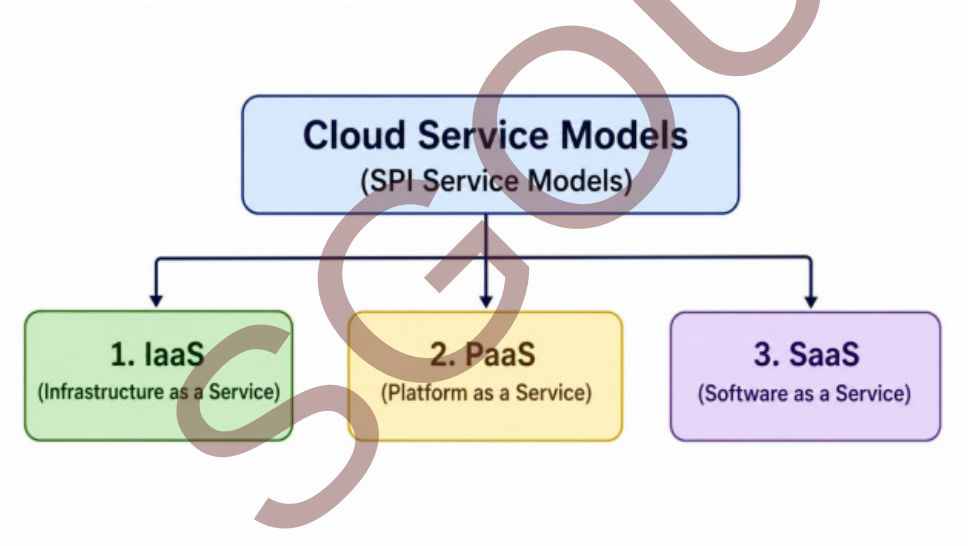


Fig. 2.1.1 Cloud Service Models

These three models are collectively known as the SPI Service Models, where SPI stands for SaaS, PaaS, and IaaS. They represent different levels of cloud service abstraction, ranging from infrastructure resources in IaaS to complete software applications in SaaS.

2.1.2 Infrastructure as a Service (IaaS)

Infrastructure as a Service (IaaS) is a cloud computing model that provides essential IT infrastructure, such as servers, storage, networking, and virtual machines, over the Internet. Instead of purchasing and maintaining physical hardware, users can obtain these resources from a cloud service provider and use them according to their requirements.

SGOU

In the IaaS model, the cloud provider manages the underlying physical infrastructure, while users have control over the virtual computing environment. Users can install operating systems, deploy applications, and manage their data without owning the physical servers and other hardware.

The resources provided through IaaS can be used to support websites, applications, databases, and other computing tasks. Since the infrastructure is delivered virtually, users can access and configure the required resources without setting up a physical data centre.

IaaS is particularly useful for organizations that require flexible computing infrastructure without the cost and responsibility of maintaining their own data centres. It is also suitable for data-intensive applications that require significant storage and computing resources.

Examples: Amazon EC2, Microsoft Azure Virtual Machines, Google Compute Engine.

2.1.2.1 Key Features of IaaS

Infrastructure as a Service provides several features that make computing infrastructure more flexible and convenient. The major features of IaaS are discussed below.

1. On-Demand Resource Provisioning

IaaS enables users to obtain computing resources whenever they are required. Virtual servers, storage, and networking resources can be provisioned without purchasing and installing physical hardware.

2. Scalability

The resources can be increased or decreased according to changing workload requirements. For example, additional computing resources can be allocated when a data processing task requires more processing power.

3. Virtualization

IaaS uses virtualization technology to provide virtual versions of computing resources. This enables multiple virtual machines and resources to operate on the same physical infrastructure.

4. Flexible Resource Configuration

Users can select and configure computing resources according to their requirements. They can choose suitable processing capacity, storage, operating systems, and other infrastructure settings.

5. Pay-as-You-Use Model

Users generally pay only for the computing resources they use. This helps organizations avoid a large initial investment in physical infrastructure.

SGOU

6. Remote Accessibility

IaaS resources can be accessed and managed through the Internet from different locations, enabling users to work with cloud infrastructure remotely.

2.1.2.2 Components of IaaS

Infrastructure as a Service consists of several essential components that work together to provide computing infrastructure to users over the Internet. The major components of IaaS are discussed below.

1. Virtual Servers

Virtual servers provide the computing environment required to run applications and services. Users can create and configure virtual machines by selecting the required processing power, memory, and operating system.

2. Storage

Cloud storage provides space for storing data, application files, databases, and backup copies. Users can select and expand storage capacity according to their requirements.

3. Networking

Networking components provide connectivity between users, virtual servers, storage systems, and other cloud resources. They enable the communication and transfer of data within the cloud environment.

4. Virtualization

Virtualization is a key technology used in IaaS to create virtual versions of computing resources. It enables physical servers to host multiple virtual machines, allowing infrastructure resources to be used efficiently.

5. Data Centres

The physical infrastructure required to deliver IaaS services is located in data centres maintained by the cloud service provider. These data centres contain physical servers, storage devices, networking equipment, and other supporting infrastructure.

2.1.2.3 Working of IaaS

In the IaaS model, the cloud service provider owns and maintains the physical infrastructure, including servers, storage systems, networking equipment, and data centres. Through virtualization technology, these physical resources are divided into virtual resources and made available to users over the Internet.

When a user requires computing resources, the user can select the required virtual server, storage capacity, and networking resources. These resources can then be configured based on the requirements of the application or service.

After obtaining the virtual infrastructure, the user can install the required operating system and applications, configure the software environment, and manage the data. The

SGOU

cloud provider is responsible for maintaining the underlying physical infrastructure, while the user manages the resources and software environment provided to them.

The basic working of IaaS can be represented as follows:

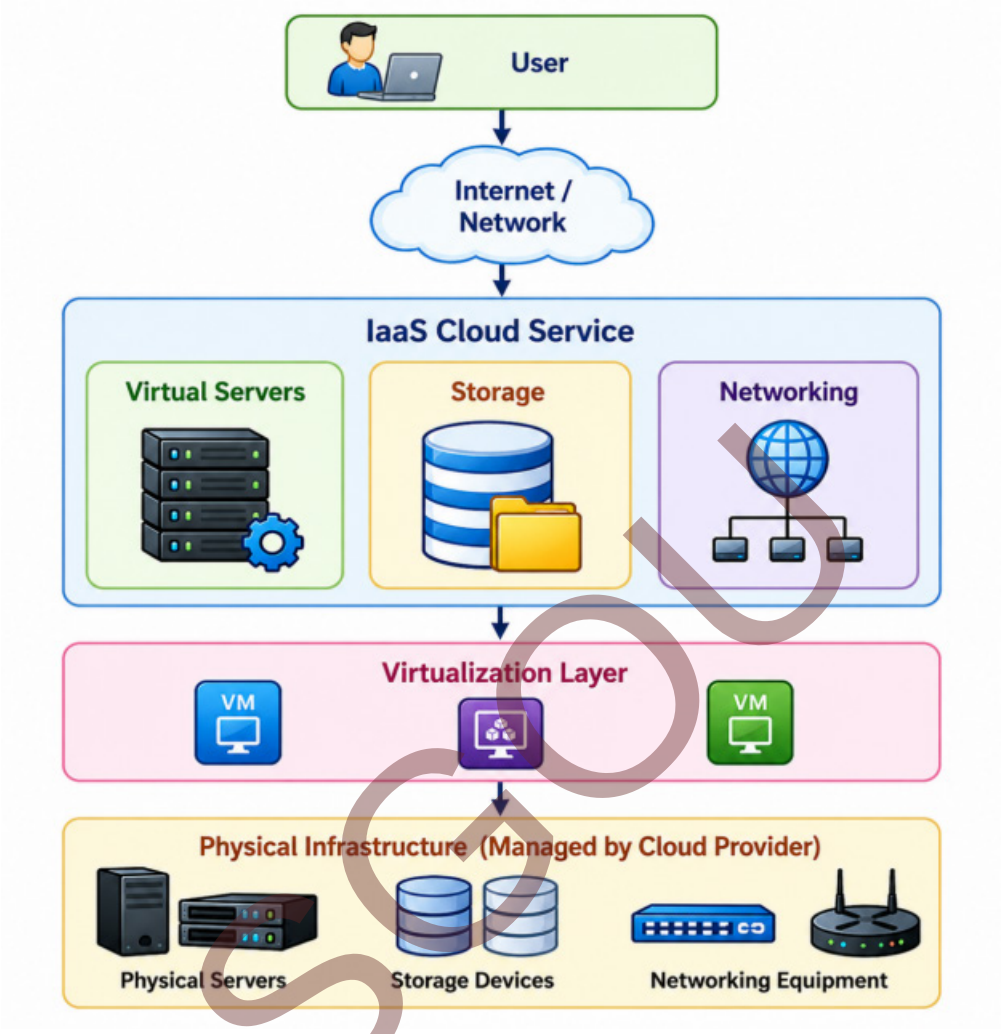


Fig. 2.1.2 Basic working of Infrastructure as a Service

2.1.2.4 Advantages of IaaS

1. **Reduced Initial Cost:** IaaS eliminates the need to purchase expensive physical servers, storage devices, and networking equipment, thereby reducing the initial investment required for computing infrastructure.
2. **Scalability:** Computing resources can be increased or decreased according to workload requirements. Users can access additional resources when needed without purchasing new hardware.
3. **Flexibility:** Users can select and configure virtual servers, storage, and networking resources according to their specific requirements. They can also choose suitable operating systems and install the required applications.
4. **Reduced Hardware Maintenance:** The cloud provider manages and

maintains the physical servers, storage systems, networking equipment, and other underlying infrastructure, reducing the maintenance responsibilities of the user.

2.1.2.5 Limitations of IaaS

Despite its advantages, Infrastructure as a Service also has certain limitations.

1. Requires Technical Knowledge

Users are responsible for managing operating systems, applications, and configurations. Therefore, a certain level of technical knowledge is required to effectively use IaaS resources.

2. Security Responsibilities

Although the cloud provider manages the physical infrastructure, users are responsible for securing their operating systems, applications, and data. Improper configuration may lead to security risks.

3. Dependence on Internet Connectivity

Access to IaaS resources depends on a reliable network or Internet connection. Poor connectivity may affect access to cloud-based infrastructure and services.

4. Cost Management

Although IaaS follows a pay-as-you-use model, costs may increase if resources are not properly monitored and managed. Unused or unnecessary resources can result in additional expenses.

2.1.3 Platform as a Service (PaaS)

Platform as a Service (PaaS) is a cloud service model that provides a ready-to-use platform for developing, testing, deploying, and managing applications. It provides developers with the necessary programming languages, development tools, databases, middleware, operating systems, and runtime environments without requiring them to manage the underlying hardware and infrastructure.

In PaaS, the cloud service provider manages the servers, storage, networking, virtualization, operating system, and runtime environment, while the user mainly focuses on developing, deploying, and managing applications and application data. This allows developers to concentrate on application development rather than infrastructure management.

PaaS acts as a middle layer between Infrastructure as a Service (IaaS) and Software as a Service (SaaS). It provides more development support than IaaS while giving users more control over applications than SaaS.

Examples: Google App Engine, Microsoft Azure App Service, and Heroku.

SGOU

2.1.3.1 Key Features of PaaS

1. **Ready-to-Use Development Environment:** PaaS provides a pre-configured environment with programming languages, frameworks, libraries, runtime environments, and development tools required for application development.
2. **Application Development and Deployment:** Developers can develop, test, and deploy applications on the cloud without setting up and managing physical servers.
3. **Scalability:** PaaS allows applications to increase or decrease computing resources according to workload and user demand.
4. **Integrated Development Tools:** PaaS provides tools for coding, debugging, testing, version control, and application deployment, supporting the application development process.
5. **Database and Middleware Support:** PaaS provides access to databases, storage, and middleware services, helping applications store data and communicate with other services.
6. **Automatic Maintenance and Updates:** The cloud provider manages operating system updates, security patches, platform upgrades, and maintenance, reducing the workload of developers.

2.1.3.2 Working of PaaS

PaaS provides a cloud-based platform where developers can develop, test, deploy, and manage applications without directly managing the underlying hardware and infrastructure.

Step 1: Access the PaaS Platform

The developer accesses the PaaS platform through the Internet using a web browser, development environment, or other supported tools.

Step 2: Develop the Application

The developer uses the programming languages, frameworks, libraries, and development tools provided by the PaaS platform to create the application.

Step 3: Use Platform Services

The application can use services such as databases, storage, authentication, and messaging provided by or integrated with the PaaS platform.

Step 4: Test the Application

The developer tests and debugs the application using the tools available on the PaaS platform.

Step 5: Deploy the Application

After testing, the application is deployed to the PaaS environment, where the platform provides the required runtime environment and computing resources.

SGOU

Step 6: Manage and Scale the Application

The PaaS platform helps monitor and scale the application according to demand. The cloud provider manages infrastructure maintenance, updates, and security patches.

2.1.3.3 Advantages of PaaS

1. Faster Application Development

PaaS provides ready-to-use development tools, frameworks, and runtime environments, helping developers build and deploy applications more quickly.

2. Reduced Infrastructure Management

The cloud service provider manages the servers, storage, networking, operating systems, and platform maintenance, allowing developers to focus on application development.

3. Lower Development Cost

PaaS reduces the need to purchase, install, and maintain expensive hardware and software infrastructure, thereby reducing development and operational costs.

4. Easy Scalability

Applications can be easily scaled up or down according to changing workload and user demand.

5. Easy Collaboration

PaaS provides a common cloud-based development environment that enables developers and teams in different locations to work and collaborate efficiently.

6. Automatic Updates and Maintenance

The cloud provider handles software updates, security patches, platform upgrades, and maintenance, reducing the administrative workload of developers.

2.1.3.4 Limitations of PaaS

1. **Vendor Lock-in:** Applications developed using a specific PaaS platform may become dependent on that provider's technologies and services. This can make it difficult to move the application to another cloud platform.
2. **Limited Control:** Users have limited control over the underlying infrastructure, operating system, and platform configuration, as these are managed by the cloud service provider.
3. **Security and Privacy Concerns:** Since applications and data are hosted on the cloud provider's infrastructure, users may have concerns regarding data security, privacy, and compliance.
4. **Limited Customization:** Some PaaS platforms may restrict the ability to customize the platform environment according to specific application

SGOU

requirements.

5. **Dependence on the Service Provider:** The availability and performance of applications depend on the reliability of the PaaS provider. Service outages or technical issues may affect the application.
6. **Migration Challenges:** Moving applications and data from one PaaS provider to another can be complex and time-consuming, particularly when provider-specific tools and services are used.

2.1.4 SaaS (Software as a Service)

Software as a Service (SaaS) is a cloud service model that provides ready-to-use software applications over the Internet. Users can access and use these applications through a web browser or mobile application without installing, managing, or maintaining the software on their own computers or servers.

In SaaS, the cloud service provider manages the entire infrastructure and application environment, including the servers, storage, networking, virtualization, operating systems, application software, updates, and maintenance. Users mainly access and use the software according to their requirements.

SaaS allows users to use software on a subscription or pay-as-you-use basis, reducing the need to purchase, install, and maintain software and related infrastructure. This makes software easily accessible from different locations and devices through an Internet connection.

Examples of SaaS Providers: Salesforce, Google Workspace, Microsoft 365, Dropbox, and Zoho.

2.1.4.1 Key Features of SaaS

1. **Ready-to-Use Applications:** SaaS provides fully functional software applications that users can access and use directly without installing or configuring the software.
2. **Internet-Based Access:** Users can access SaaS applications through the Internet using a web browser or mobile application from different locations and devices.
3. **No Installation or Maintenance:** The cloud service provider handles software installation, updates, maintenance, and technical support, reducing the workload of users.
4. **Subscription-Based Model:** SaaS applications are commonly offered through a subscription or pay-as-you-use model, allowing users to pay according to their requirements.
5. **Automatic Updates:** Software updates, new features, and security patches are automatically managed by the service provider, ensuring that users have access to the latest version of the application.
6. **Scalability and Flexibility:** SaaS applications can easily scale according

SGOU

to the number of users and changing requirements, providing flexibility for individuals and organizations.

2.1.4.2 Working of SaaS

SaaS delivers ready-to-use software applications to users through the Internet. The cloud service provider hosts and manages the application and its underlying infrastructure, while users simply access and use the software according to their requirements.

First, the user accesses the SaaS application through the Internet using a web browser or mobile application. The user then signs in using the required login credentials to securely access the application and its services.

Once access is provided, the user can perform the required tasks using the ready-to-use features and functions of the software. The user's requests are processed by the application, and the related data is stored and managed in the cloud infrastructure.

The SaaS provider manages the complete software environment, including the application software, servers, storage, networking, security, updates, and maintenance. Users do not need to install, maintain, or update the software on their own systems.

Since the application is delivered through the Internet, users can access the software and their data from different devices and locations, provided they have a suitable Internet connection.

2.1.4.3 Advantages of SaaS

1. Easy Access

SaaS applications can be accessed through the Internet using a web browser or mobile application, allowing users to use the software from different locations and devices.

2. No Installation Required

Users can access and use the software without installing it on their own computers or managing the required hardware and software infrastructure.

3. Reduced Cost

SaaS commonly follows a subscription or pay-as-you-use model, reducing the need for large initial investments in software licenses, hardware, and maintenance.

4. Automatic Updates and Maintenance

The SaaS provider manages software updates, security patches, maintenance, and technical improvements, ensuring that users can access the latest version of the application.

5. Scalability

SaaS services can easily scale according to the number of users and changing requirements, providing flexibility to individuals and organizations.

SGOU

6. Easy Collaboration

Many SaaS applications allow multiple users to access, share, and work on the same data or documents, supporting effective collaboration.

2.1.4.4 Limitations of SaaS

1. Dependence on Internet Connectivity

SaaS applications generally require a stable Internet connection. Poor or unavailable connectivity can affect access to the software and its services.

2. Limited Control and Customization

Users have limited control over the software, infrastructure, and configuration. Some SaaS applications may also provide limited options for customization.

3. Security and Privacy Concerns

Since application data is stored and processed on the cloud provider's infrastructure, users may have concerns regarding data security, privacy, and compliance.

4. Dependence on the Service Provider

The availability and performance of the application depend on the reliability of the SaaS provider. Service outages or technical problems may temporarily affect access.

Recap

- Cloud service models define how computing resources, platforms, and software are delivered to users through the Internet.
- The three major cloud service models are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), collectively known as the SPI model.
- IaaS provides virtualized computing resources such as servers, storage, networking, and virtual machines.
- In IaaS, the cloud provider manages the physical infrastructure, while users have greater responsibility for managing their operating systems, applications, and data.
- PaaS provides a ready-to-use platform with development tools, programming environments, databases, and middleware for building and deploying applications.
- In PaaS, the provider manages the infrastructure and platform, allowing developers to focus mainly on application development and deployment.
- SaaS provides fully functional, ready-to-use software applications that users can access through the Internet.
- In SaaS, the provider manages the entire software environment and underlying infrastructure, while users mainly access and use the application.

SGOU

- The three service models differ mainly in terms of the services provided, level of user control, and responsibility shared between the provider and the user.
- IaaS provides the highest level of user control, whereas SaaS requires the least management from the user.
- PaaS acts as an intermediate model, offering a balance between infrastructure control and ready-to-use software services.

Objective Type Questions

1. What does IaaS stand for?
2. What does PaaS stand for?
3. What does SaaS stand for?
4. Which cloud service model provides virtualized computing resources such as servers and storage?
5. Which cloud service model provides a platform for developing and deploying applications?
6. Which cloud service model provides ready-to-use software applications?
7. What is the collective name for IaaS, PaaS, and SaaS?
8. Who manages the underlying physical infrastructure in IaaS?

SGOU

Answers to Objective Type Questions

1. Infrastructure as a Service
2. Platform as a Service
3. Software as a Service
4. IaaS
5. PaaS
6. SaaS
7. SPI Model
8. Cloud Service Provider

Assignments

1. Explain the three major cloud service models—IaaS, PaaS, and SaaS—with suitable examples.
2. Describe the key features and working of Infrastructure as a Service (IaaS).
3. Explain how Platform as a Service (PaaS) supports application development and deployment. Discuss its advantages and limitations.
4. Describe the working of Software as a Service (SaaS) and explain its major advantages and limitations.
5. Compare IaaS, PaaS, and SaaS based on the services provided, level of user control, and responsibilities of the cloud service provider and user.

Reference

1. https://nptel.ac.in/courses/106105167
2. https://www.geeksforgeeks.org/cloud-computing/types-of-cloud/

Suggested Reading

1. Hiran, K. K., Doshi, R., Fagbola, T., & Mahrishi, M. (2019). Cloud Computing: Master the Concepts, Architecture and Applications with Real-World Examples and Case Studies (1st ed.). BPB Publications.
2. Jamsa, K. (2013). Cloud Computing (1st ed.). Jones & Bartlett Student Edition.

SGOU

Unit 2
Cloud Service Models

Learning Outcomes

Upon completion of the unit, learner will be able to:

- define emerging cloud service models such as STaaS, DBaaS, MaaS, FaaS, BPaaS, SecaaS, and IdMaaS
- describe the purpose and key features of different cloud service models
- identify suitable cloud service models for different organizational requirements
- compare various cloud service models based on their purpose and applications

Prerequisites

Have you ever wondered how your photos are stored safely online, how a company manages thousands of customer records, how an application responds instantly when an event occurs, or how organizations protect their cloud resources and control user access? These everyday activities are powered by different cloud services working behind the scenes.

A basic understanding of computer networks, the Internet, data storage, databases, software applications, and programming concepts will help learners connect these real-world activities with cloud computing. Familiarity with fundamental cloud concepts such as IaaS, PaaS, SaaS, and cloud deployment models will provide a strong foundation for exploring specialized cloud services.

With this foundation, learners can discover how Storage as a Service (STaaS) provides flexible data storage, how Database as a Service (DBaaS) manages databases, how Function as a Service (FaaS) executes code without server management, and how services such as BPaaS, SecaaS, and IdMaaS support business processes, security, and identity management.

Exploring these concepts will help learners understand what happens behind the scenes when modern digital services are accessed, how organizations select appropriate cloud services for different requirements, and how these services contribute to scalability, efficiency, security, and cost-effective technology management. This

SGOU

foundation will also prepare learners to explore the growing role of cloud computing in software development, business operations, cybersecurity, data management, and other technology-driven fields.

Keywords

Storage as a Service (STaaS), Database as a Service (DBaaS), Management as a Service (MaaS), Function as a Service (FaaS), Business Process as a Service (BPaaS), Security as a Service (SecaaS), Identity as a Service (IdMaaS)

Discussion

2.2.1 Introduction to Cloud Service Models

In today's digital era, cloud computing has become an important part of everyday life. Activities such as watching movies on Netflix, storing study materials on Google Drive, and accessing online applications demonstrate the widespread use of cloud services. In the past, organisations had to purchase expensive servers, install and maintain software, and manage complex IT systems. Today, resources such as computing power, databases, applications, and storage can be accessed through the Internet whenever required.

The early cloud service models such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), laid the foundation for cloud computing. As technology advanced and user requirements became more specialised, several additional cloud service models emerged to provide greater flexibility and specialised support.

These models include Storage as a Service (STaaS), Database as a Service (DBaaS), Management as a Service (MaaS), Function as a Service (FaaS), Business Process as a Service (BPaaS), Security as a Service (SECaaS), and Identity Management as a Service (IdMaaS).

For example, STaaS enables users to store and access data, DBaaS supports cloud-based database management, and FaaS allows specific functions to be executed automatically without managing servers. Similarly, BPaaS supports business operations, while SECaaS and IdMaaS provide cloud-based security and identity management services.

Together, these specialised service models demonstrate how cloud computing has evolved to meet a wide range of computing and business requirements. The following sections explain each of these cloud service models, including their concepts, advantages, disadvantages, and applications.

2.2.2 Storage as a Service (STaaS)

Storage as a Service (STaaS) is a cloud computing service model that provides users and organisations with data storage facilities through the Internet. Instead of purchasing and maintaining physical storage devices, users can obtain storage space from a cloud service provider as and when required.

SGOU

The cloud service provider manages the underlying storage infrastructure, including the hardware and software required to store, manage, and access data. Users can store files in the cloud and access them from different locations and devices through an Internet connection.

STaaS helps organisations reduce the cost and effort involved in purchasing, maintaining, and upgrading storage infrastructure. It also provides flexibility, as storage capacity can be increased or decreased according to changing requirements. Many STaaS providers also offer additional features such as data backup, recovery, sharing, and encryption.

Common examples of cloud-based storage services include Google Drive, Dropbox, and iCloud.

2.2.2.1 Advantages of STaaS

Storage as a Service offers several benefits to individuals and organisations by providing flexible and easily accessible cloud-based storage facilities.

1. **Cost-effective:** Users do not need to invest in expensive storage hardware or its maintenance. They can pay according to their storage requirements.
2. **Scalability:** Storage capacity can be increased or decreased easily based on changing data requirements.
3. **Accessibility:** Data can be accessed from different locations and devices through an Internet connection.
4. **Data Backup and Recovery:** Cloud storage services can provide backup and recovery facilities to help protect data from loss.
5. **Reduced Maintenance:** The cloud service provider manages storage hardware, software updates, and other maintenance activities.
6. **Collaboration:** Multiple users can share and access files, making collaboration easier.

2.2.2.2 Disadvantages of STaaS

Although Storage as a Service offers several benefits, it also has certain limitations.

1. Data Security Concerns: Storing sensitive data on third-party servers may create privacy and security concerns.
2. Internet Dependency: Access to cloud-stored data depends on the availability of a reliable Internet connection.
3. Possible Downtime: Service outages may temporarily prevent users from accessing their stored data.
4. Additional Costs: Data transfer, retrieval, or other additional services may sometimes result in extra charges.

SGOU

2.2.3 Database as a Service (DBaaS)

Database as a Service (DBaaS) is a cloud-based service model that enables users and organisations to access and manage databases without installing, configuring, or maintaining the underlying database infrastructure.

The cloud service provider manages tasks such as database setup, maintenance, updates, backup, recovery, security, and scaling. This allows users to focus mainly on storing, accessing, and managing data for their applications.

DBaaS provides database resources on demand and enables organisations to use scalable database services without investing heavily in physical servers and database infrastructure. For example, a developer can use a cloud-based database to store user information for a mobile application or manage customer and product data for an e-commerce application.

Common examples of DBaaS include Amazon RDS, Microsoft Azure SQL Database, Google Cloud SQL, and Oracle Autonomous Database.

2.2.3.1 Advantages of DBaaS

1. **Reduced Administrative Effort:** The service provider handles tasks such as database setup, maintenance, updates, and infrastructure management.
2. **Cost Efficiency:** Organisations can use database resources according to their requirements without investing heavily in their own infrastructure.
3. **Scalability:** Database capacity can be increased or decreased according to changes in workload and data requirements.
4. **High Availability and Reliability:** Cloud database services help provide reliable access to data and support continuous availability.
5. **Automatic Backup and Recovery:** Regular backups and recovery facilities help protect data against loss.

2.2.3.2 Disadvantages of DBaaS

1. **Data Security and Privacy Concerns:** Storing sensitive data with a third-party provider may create security and privacy concerns.
2. **Limited Control:** Users may have limited control over certain database configurations and the underlying infrastructure.
3. **Internet Dependency:** Accessing and managing cloud databases generally requires a reliable Internet connection.
4. **Performance Variability:** Database performance may be affected by factors such as network latency and shared cloud resources.

SGOU

2.2.4 Management as a Service (MaaS)

Management as a Service (MaaS) is a cloud computing service model that provides IT management functions through the Internet. It enables organisations to use cloud-based tools and services to monitor, manage, and maintain their IT infrastructure, applications, and networks.

The main purpose of MaaS is to simplify IT management and reduce the operational workload of organisations. It can support activities such as performance monitoring, resource management, troubleshooting, configuration management, and reporting through a centralized cloud-based platform.

For example, an organisation can use MaaS tools to monitor the performance of its cloud servers, track resource usage, detect system issues, and manage IT operations remotely. Common examples of MaaS-related services include ServiceNow, IBM Cloud management services, and cloud-based IT monitoring and management platforms.

2.2.4.1 Advantages of MaaS

Management as a Service offers several benefits by simplifying the monitoring and management of IT resources.

1. **Reduced IT Workload:** MaaS reduces the effort required to manage IT infrastructure, applications, and networks.
2. **Cost Efficiency:** Organisations can reduce the cost of purchasing and maintaining their own management tools and infrastructure.
3. **Continuous Monitoring:** Cloud-based management services can support continuous monitoring of system performance and resource usage.
4. **Scalability:** MaaS can support the management of growing IT infrastructure as organisational requirements increase.
5. **Centralised Management:** IT resources and services can be monitored and managed through a centralised platform.
6. **Expert Support:** Organisations can benefit from specialised management tools and professional expertise provided through the service.

2.2.4.2 Disadvantages of MaaS

Although Management as a Service offers several benefits, it also has certain limitations.

1. **Data Security Risks:** Management and monitoring data stored in the cloud may be exposed to security and privacy risks.
2. **Dependency on the Service Provider:** Organisations may become dependent on the service provider for important IT management functions.
3. **Possible Service Downtime:** Provider-side outages may temporarily affect monitoring and management activities.

SGOU

4. **Integration Complexity:** Integrating cloud-based management services with existing or legacy systems can sometimes be challenging.

5. **Additional Costs:** Advanced management features, additional monitoring, or specialised services may involve extra charges.

2.2.5 Function as a Service (FaaS)

Function as a Service (FaaS) is a cloud computing service model that allows developers to execute individual functions or small pieces of code in response to specific events without managing servers or underlying infrastructure. It is an important part of serverless computing, where the cloud provider manages server provisioning, scaling, resource allocation, and maintenance.

In FaaS, developers upload their code as separate functions, and the cloud platform automatically executes the required function whenever a specific event occurs. These events may include a web request, file upload, database update, or message received from an application. For example, when a user uploads an image to a website, an FaaS function can automatically resize the image or generate a thumbnail.

Popular FaaS services include AWS Lambda, Azure Functions, Google Cloud Functions, IBM Cloud Code Engine, and Oracle Cloud Infrastructure Functions.

2.2.5.1 Advantages of FaaS

1. **Cost-efficient:** Users pay primarily for the resources and execution time consumed by their functions.

2. **Automatic scalability:** FaaS automatically adjusts computing resources according to workload and demand.

3. **Faster deployment:** Developers can deploy functions without setting up or managing servers.

4. **Improved productivity:** FaaS reduces backend and infrastructure management, allowing developers to focus on application development.

5. **High availability and reliability:** The cloud provider manages the underlying infrastructure and provides mechanisms to maintain service availability.

6. **Reduced infrastructure management:** The cloud provider manages servers, operating systems, runtime environments, and other infrastructure components, reducing administrative effort.

7. **Event-driven execution:** Functions are automatically executed when specific events occur, making FaaS suitable for applications that respond to user actions or system events.

2.2.5.2 Disadvantages of FaaS

1. **Cold start latency:** A delay may occur when a function is invoked after remaining inactive for some time.

SGOU

2. **Limited execution time:** FaaS functions are generally designed for short-running tasks and cannot run indefinitely.

3. **Vendor lock-in:** Functions may depend on provider-specific services, making migration between cloud providers difficult.

4. **Debugging challenges:** Debugging and tracing problems can be difficult when applications consist of multiple distributed functions.

5. **Not suitable for long-running applications:** FaaS is mainly designed for short, event-driven tasks and may not be suitable for applications requiring continuous or lengthy processing.

6. **Resource limitations:** Functions may have restrictions on memory, CPU, storage, execution time, and other resources imposed by the cloud provider.

7. **Monitoring and testing complexity:** Testing, monitoring, and managing interactions among multiple functions can become difficult as the number of functions increases.

2.2.6 Business Process as a Service (BPaaS)

Business Process as a Service (BPaaS) is a cloud-based service model in which business processes and business process outsourcing (BPO) services are delivered through the cloud. It enables organizations to automate, manage, and execute business processes without owning or maintaining the underlying IT infrastructure.

BPaaS can be used for various business activities such as payroll processing, human resource management, finance, accounting, marketing, customer service, and supply chain management. The service provider manages the required infrastructure, applications, and technologies, allowing organizations to focus on their core business activities.

For example, an organization can use a cloud-based payroll service to automatically calculate employee salaries, deductions, and generate payslips without installing and maintaining its own payroll software and servers.

BPaaS can integrate with other cloud service models such as Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) to provide end-to-end business solutions. These services can be configured according to organizational requirements and can scale as business needs change.

2.2.6.1 Advantages of BPaaS

1. **Cost-effective:** Reduces operational, infrastructure, software, and maintenance costs.

2. **Scalable and flexible:** Services can be easily adjusted according to workload and changing business requirements.

SGOU

3. **Focus on core business:** Allows organizations to outsource non-core business activities and concentrate on their primary objectives.
4. **Automation and efficiency:** Automates routine business processes and improves operational efficiency.
5. **Faster deployment:** Cloud-based business processes can be implemented quickly with minimal IT intervention.
6. **Global accessibility:** Business processes and services can be accessed from different locations through the Internet.
7. **Reduced IT management:** The service provider manages the underlying infrastructure, software, updates, and maintenance.
8. **Standardized processes:** Provides established business workflows and practices that can improve consistency and process quality.

2.2.6.2 Disadvantages of BPaaS

1. **Data security concerns:** Sensitive business information is stored and processed on third-party cloud infrastructure, creating potential security risks.
2. **Vendor lock-in:** Dependence on a particular service provider can make it difficult and costly to switch to another provider.
3. **Customization limitations:** BPaaS solutions may provide standardized processes that may not fully support highly specific business requirements.
4. **Internet dependency:** Reliable Internet connectivity is required to access and use cloud-based business processes.
5. **Compliance challenges:** Organizations must ensure that the BPaaS provider follows relevant legal, regulatory, and industry requirements.
6. **Limited control:** Organizations have less direct control over the underlying infrastructure and how the service is managed.
7. **Integration challenges:** Connecting BPaaS with existing organizational applications and systems may require additional configuration or integration efforts.

2.2.7 Security as a Service (SecaaS)

Security as a Service (SecaaS) is a cloud-based service model in which a third-party provider delivers cybersecurity services over the Internet. It allows organizations to outsource various security functions such as authentication, antivirus protection, intrusion detection, encryption, firewall management, and data loss prevention to specialized cloud providers.

In this model, security tools, applications, and infrastructure are hosted and managed by the service provider. Organizations can access these security services through the

SGOU

Internet and typically pay through a subscription or usage-based pricing model, similar to Software as a Service (SaaS).

For example, an organization can use a cloud-based security service to detect suspicious login attempts, block malicious traffic, and protect sensitive data without installing and maintaining all the required security systems on its own premises.

Popular Security as a Service (SecaaS) providers include McAfee Cloud Security, Symantec Cloud Security (Broadcom), Trend Micro Cloud One, Palo Alto Networks Prisma Access, Cisco Umbrella, Zscaler Cloud Security, Okta, and CrowdStrike Falcon.

2.2.7.1 Advantages of SecaaS

1. **Cost-effective:** Reduces the cost of purchasing security hardware, software, and maintaining in-house security infrastructure.
2. **Always up to date:** Security software, threat intelligence, and security definitions can be updated automatically by the service provider.
3. **Scalable and flexible:** Security services can be easily scaled according to the organization's changing requirements.
4. **Expert management:** Security operations are managed by cybersecurity professionals with specialized knowledge and expertise.
5. **Quick deployment:** Cloud-based security services can be implemented faster than many traditional on-premises security solutions.
6. **Continuous security monitoring:** Cloud security providers can continuously monitor systems and networks to identify and respond to potential security threats.

2.2.7.2 Disadvantages of SecaaS

1. **Data privacy concerns:** Sensitive organizational data may be processed or stored by third-party security providers, creating privacy concerns.
2. **Internet dependency:** Reliable Internet connectivity is generally required to access and manage cloud-based security services.
3. **Vendor lock-in:** Moving security services and configurations from one provider to another can be complex and costly.
4. **Limited control:** Organizations may have less direct control over security tools, configurations, and infrastructure compared with an internally managed solution.
5. **Compliance challenges:** Organizations must ensure that the service provider meets applicable security, legal, and industry-specific compliance requirements.
6. **Dependence on service provider:** Any service interruption, technical failure, or security issue at the provider's end may affect the organization's security operations.

SGOU

2.2.8 Identity as a Service (IdMaaS)

Identity as a Service (IDaaS), also known as Identity Management as a Service (IdMaaS), is a cloud-based Identity and Access Management (IAM) service that provides authentication, authorization, and identity management functionalities over the Internet. It enables organizations to securely manage user identities and control access to applications, data, and systems without maintaining identity management infrastructure on their premises.

IdMaaS providers manage various identity-related functions such as user provisioning, Single Sign-On (SSO), Multi-Factor Authentication (MFA), password management, and directory integration through cloud-based platforms.

For example, an organization can use an IdMaaS solution to allow employees to log in once and securely access multiple cloud applications such as email, file storage, and business applications using Single Sign-On (SSO).

2.2.8.1 Advantages of IdMaaS

1. **Centralized Identity Management:** Simplifies user authentication and access control across multiple applications and systems.
2. **Improved Security:** Provides security features such as Multi-Factor Authentication (MFA), Single Sign-On (SSO), and access controls.
3. **Cost-effective:** Reduces the need to purchase, operate, and maintain on-premises identity management infrastructure.
4. **Scalability:** Easily supports the addition or removal of users as the organization grows.
5. **Easy Integration:** Can integrate with various cloud-based and on-premises applications and systems.
6. **Reduced Administrative Effort:** Automates identity-related tasks such as user provisioning, access management, and account deactivation.

2.2.8.2 Disadvantages of IdMaaS

1. **Dependence on Internet Connectivity:** Access and authentication may be affected when a reliable Internet connection is unavailable.
2. **Vendor Lock-in:** Switching to another identity service provider can be difficult when the existing solution is deeply integrated with organizational systems.
3. **Data Privacy Concerns:** Storing identity and user information in the cloud may create privacy and data protection concerns.
4. **Limited Customization:** Some IdMaaS solutions may not fully support highly specific organizational identity management requirements.

SGOU

5. **Potential Downtime:** Service interruptions at the provider's end can prevent users from accessing applications and systems.

6. **Dependence on Provider Security:** The security of user identities and authentication information depends partly on the security measures implemented by the service provider.

2.2.9 Comparison of Cloud Service Models

Table 2.2.1 Comparison of Cloud Service Models

Cloud Model	Main Purpose	Key Service Type	Users / Target Audience	Examples/ Providers
Storage as a Service (STaaS)	To store, manage, and access data remotely through the cloud	Cloud-based storage	Businesses and individuals needing scalable data storage	Amazon S3, Azure Blob Storage, Google Cloud Storage
Database as a Service (DBaaS)	To provide database hosting, management, and maintenance through the cloud	Managed databases	Developers and Organizations	Amazon RDS, Azure SQL Database, Oracle Cloud
Management as a Service (MaaS)	To monitor, manage, and maintain IT infrastructure and services	Cloud-based IT management	IT teams and enterprises	IBM Cloud, SolarWinds, ServiceNow
Function as a Service (FaaS)	To execute code in response to events without managing servers	Serverless computing	Developers and application creators	AWS Lambda, Azure Functions, Google Cloud Functions
Business Process as a Service (BPaaS)	To deliver business processes and operations through the cloud	Business process automation	Enterprises and large organizations	SAP, Oracle, IBM
Security as a Service (SecaaS)	To provide cybersecurity services and protection through the cloud	Cloud-based security	Businesses and organizations of all sizes	McAfee, Zscaler, Cisco

SGOU

Identity as a Service (IdMaaS)	To manage user identities, authentication, and access control	Identity and access management	Enterprises and organizations using cloud applications	Okta, Microsoft Entra ID, Ping Identity

Recap

- Cloud computing provides computing resources and services through the Internet without requiring users to manage the underlying infrastructure.
- Emerging cloud service models provide specialized services to meet specific business and technological requirements.
- Storage as a Service (STaaS) provides scalable cloud-based storage for storing, managing, and accessing data.
- Database as a Service (DBaaS) provides managed databases without requiring organizations to maintain database infrastructure.
- Management as a Service (MaaS) helps organizations monitor and manage IT infrastructure and services through the cloud.
- Function as a Service (FaaS) allows developers to execute individual functions in response to events without managing servers.
- Business Process as a Service (BPaaS) delivers business processes such as payroll, HR, finance, and customer service through the cloud.
- Security as a Service (SecaaS) provides cloud-based cybersecurity services such as authentication, encryption, threat detection, and data protection.
- Identity as a Service (IdMaaS) provides cloud-based identity and access management, including authentication, authorization, SSO, and MFA.
- Each cloud service model has its own purpose, benefits, limitations, and target users.
- Cloud service models help organizations improve scalability, flexibility, efficiency, security, accessibility, and cost management.
- Selecting an appropriate cloud service model depends on the organization's requirements, workload, security needs, budget, and level of control.

SGOU

Objective Type Questions

1. What does STaaS stand for?
2. What is the main purpose of STaaS?
3. What does DBaaS stand for?
4. Which cloud service model provides managed databases?
5. What does MaaS stand for in cloud computing?
6. Which service model allows developers to execute code without managing servers?
7. What does FaaS stand for?
8. Name one popular FaaS provider.
9. Which cloud service model delivers business processes through the cloud?
10. What does BPaaS stand for?
11. Which cloud service model provides cybersecurity services through the cloud?
12. What does IdMaaS stand for?

Answers to Objective Type Questions

1. Storage as a Service
2. Cloud-based data storage
3. Database as a Service
4. DBaaS
5. Management as a Service
6. FaaS
7. Function as a Service
8. AWS Lambda
9. BPaaS
10. Business Process as a Service
11. SecaaS
12. Identity Management as a Service

SGOU

Assignments

1. Explain in detail the concept of Storage as a Service (STaaS).
2. Compare and contrast Database as a Service (DBaaS) and Function as a Service (FaaS).
3. Explain how Management as a Service (MaaS) helps organizations monitor and manage IT infrastructure. Provide suitable real-world examples.
4. Discuss the role of Security as a Service (SecaaS) and Identity as a Service (IdMaaS) in enhancing cloud security.
5. Explain Business Process as a Service (BPaaS) and discuss how it helps organizations automate and optimize business operations using cloud-based solutions.

Reference

1. https://nptel.ac.in/courses/106105167
2. https://www.geeksforgeeks.org/cloud-computing/storage-as-a-service/
3. https://www.geeksforgeeks.org/software-engineering/overview-of-database-as-a-service/

Suggested Reading

1. Hiran, K. K., Doshi, R., Fagbola, T., & Mahrishi, M. (2019). *Cloud Computing: Master the Concepts, Architecture and Applications with Real-World Examples and Case Studies* (1st ed.). BPB Publications.
2. Jamsa, K. (2013). *Cloud Computing* (1st ed.). Jones & Bartlett Student Edition.

SGOU

Unit 3

Managed High performance Computing and Serverless Cloud Services

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the concepts and key features of AWS Parallel Computing Service (AWS PCS)
- describe the architecture and working principles of AWS PCS and its major components
- familiarize the concepts and key features of Serverless Computing
- analyze the advantages, limitations, and common use cases of Serverless Computing

Prerequisite

The increasing demand for processing large volumes of data and delivering applications quickly has made scalable and flexible cloud computing technologies essential. AWS Parallel Computing Service (AWS PCS) is important for understanding how multiple computing resources can work together to handle computationally intensive tasks efficiently. For example, weather forecasting systems may need to process large amounts of environmental data using multiple computing resources within a short period. Serverless Computing is important for understanding how applications can automatically execute tasks in response to events without requiring continuous server management. For example, in an online shopping system, placing an order can automatically trigger functions to update inventory, process payment information, and send confirmation messages. These technologies demonstrate how cloud platforms can allocate computing resources according to workload requirements. Studying AWS PCS helps learners understand the efficient execution of large-scale and parallel workloads in the cloud. Studying Serverless Computing helps learners understand event-driven application execution and automatic resource scaling. Together, these technologies illustrate how modern cloud platforms reduce infrastructure management while improving scalability and operational efficiency. Therefore, learning AWS PCS and Serverless Computing is essential for understanding how cloud environments support diverse real-world computing and application requirements.

SGOU

Keywords

Parallel Computing, Slurm, Compute Nodes, Automatic Scaling, Event-Driven Execution, Resource Provisioning

Discussion

Cloud computing has progressed significantly beyond basic services such as virtual machines, storage, and networking. Modern cloud platforms now offer specialized services designed to support computationally intensive workloads and simplify application development. Managed High-Performance Computing (HPC) and Serverless Cloud Services are two important approaches that enable organizations to perform complex tasks efficiently while minimizing the need to maintain physical infrastructure.

Managed High-Performance Computing (HPC) refers to cloud-based services that provide powerful and coordinated computing resources for applications that require substantial processing capabilities. HPC environments commonly combine multiple processors, high-speed interconnects, large memory capacities, and parallel computing techniques to process complex workloads within a shorter period. In a managed HPC environment, the cloud service provider takes responsibility for several infrastructure-related tasks, including resource provisioning, cluster configuration, software environment setup, monitoring, maintenance, and system management. As a result, researchers, engineers, data scientists, and organizations can concentrate on their computational workloads without having to manage the underlying hardware and infrastructure. Managed HPC is widely applicable to areas such as scientific simulations, weather prediction, artificial intelligence, engineering and design simulations, genomic research, and large-scale data analysis.

Serverless Cloud Services offer a different model in which developers can execute applications or individual functions without having to provision or manage servers directly. The cloud provider automatically allocates the required computing resources, executes the application when an event or request occurs, scales the resources according to workload requirements, and manages the underlying infrastructure. In most serverless models, users are charged according to factors such as execution time, number of requests, or resources consumed. Serverless computing is particularly suitable for event-driven applications, application programming interfaces (APIs), data processing tasks, workflow automation, and applications with unpredictable or fluctuating workloads.

2.3.1 AWS PCS (AWS Parallel Computing Service)

AWS Parallel Computing Service (AWS PCS) is a managed High-Performance Computing (HPC) service offered by Amazon Web Services (AWS). It enables organizations, researchers, engineers, and data scientists to create and operate HPC clusters in the AWS Cloud without having to manually manage the entire underlying infrastructure.

High-Performance Computing involves combining multiple computing resources

SGOU

to solve computationally intensive problems within a shorter period. In traditional HPC environments, organizations need to purchase and configure servers, establish networks, install job schedulers, manage storage, monitor hardware, and handle hardware failures. AWS PCS reduces much of this administrative burden by providing a managed environment for running HPC workloads.

AWS PCS is based on Slurm, an open-source workload manager widely used in HPC environments. Slurm handles job submission, scheduling, and execution across the computing resources within the cluster. This allows users to continue working with familiar HPC concepts while benefiting from the scalability and flexibility of the AWS Cloud.

Many scientific, engineering, and data-intensive workloads require considerably more computing power than a single computer can provide. Applications such as weather and climate modelling, molecular modelling, computational fluid dynamics (CFD), seismic analysis, and large-scale data processing may require multiple processors or computing nodes to operate together.

In a traditional High-Performance Computing (HPC) environment, an organization must establish and maintain various components, including physical servers, high-speed networking, shared storage, a workload scheduler, and computing nodes. It also requires system administration, monitoring, software configuration, hardware maintenance, and capacity planning.

Traditional HPC environment:

Physical Servers → High-Speed Network → Shared Storage → Job Scheduler → Compute Nodes → HPC Jobs

Establishing and maintaining such an environment can require considerable time, technical expertise, infrastructure, and financial investment. Organizations must also estimate their computing requirements in advance, which may lead to underutilized resources when demand is low.

AWS Parallel Computing Service (AWS PCS) provides a managed, cloud-based approach for running Slurm-based HPC workloads. It simplifies the deployment and operation of HPC environments while allowing users to configure supporting AWS resources such as networking, storage, IAM, and computing resources.

AWS PCS environment:

AWS PCS Cluster → Slurm Controller → Queues → Compute Node Groups → EC2 Compute Instances → HPC Jobs

AWS PCS can also support dynamic computing capacity for pending jobs. Suitable EC2 compute instances can be launched when additional capacity is needed and scaled down as the workload decreases.

Therefore, AWS PCS is suitable for organizations requiring scalable, cloud-based HPC computing while reducing the operational effort associated with deploying, maintaining, and scaling the infrastructure required by traditional HPC environments.

SGOU

2.3.1.1 AWS PCS Architecture and Core Components

The AWS Parallel Computing Service (AWS PCS) Architecture shown in Fig. 2.3.1 illustrates the flow of an HPC job from user submission to execution. It explains how users submit jobs, how Slurm schedules them, how AWS PCS provisions the required Amazon EC2 compute resources, and how compute nodes access shared storage to read input data and write output results.

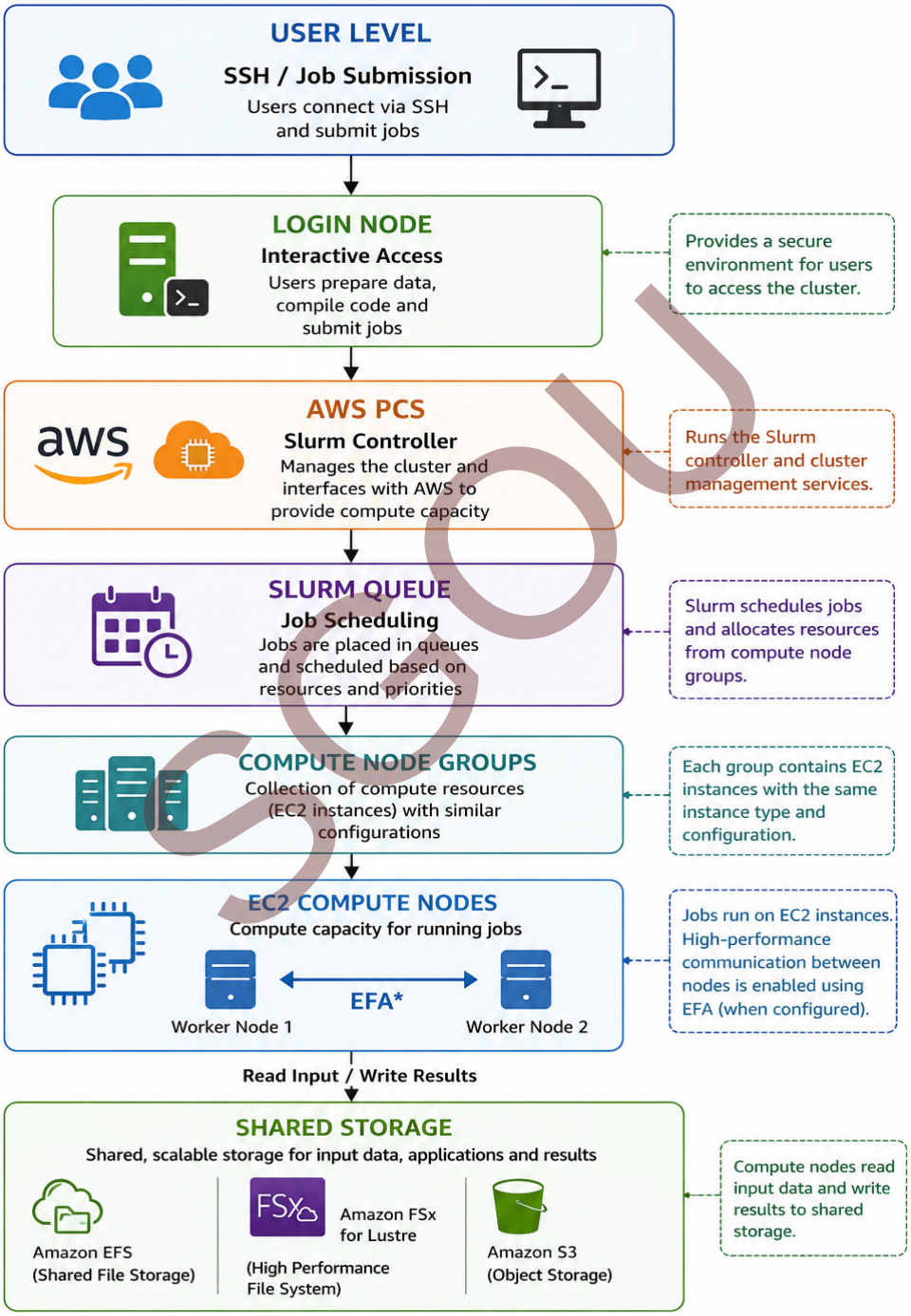


Fig 2.3.1 AWS HPC Slurm Cluster Architecture

1. User Level

At the top of the architecture, users access the HPC environment through SSH (Secure Shell). They connect to the Login Node, where they can prepare data, compile applications, configure the required environment, and submit HPC jobs. Users do not directly manage individual worker nodes. Instead, jobs are submitted to the Slurm-based HPC environment.

User → SSH → Login Node → Job Submission

2. Login Node

The Login Node acts as the interactive entry point to the HPC cluster. Users use the login node to:

- Prepare input data and create job scripts.
- Compile applications and configure environmental dependencies.
- Submit jobs using standard Slurm commands, such as sbatch.
- Monitor job status and access the results after execution.

The login node functions primarily as an administrative workspace, while computationally intensive processing is performed by the worker compute nodes.

3. AWS PCS – Slurm Controller

After a user submits a job, it enters the AWS PCS-managed Slurm environment. AWS PCS provides a managed HPC environment based on Slurm, where the Slurm controller manages workload scheduling and coordinates job execution.

It determines:

- The resources required by the job.
- The queue through which the job should be processed.
- The compute nodes that can execute the job.
- The appropriate time for the job to run.

Thus, AWS PCS provides the managed layer for the Slurm-based HPC cluster.

4. Slurm Queue

The Slurm Queue stores submitted jobs until the necessary computing resources become available. Slurm considers factors such as:

- Requested CPU resources.
- Memory requirements.
- Number of required nodes.

SGOU

- Job priority.
- Available computing capacity.

When suitable resources become available, Slurm schedules the job and assigns the required compute resources.

Job Submission → Queue → Resource Allocation → Job Execution

5. Compute Node Groups

Compute Node Groups represent groups of computing resources configured for the HPC cluster. Each node group can be configured with specific EC2 instance types and settings based on the requirements of particular workloads.

For example, an HPC environment may include:

- CPU-oriented compute nodes.
- GPU-oriented compute nodes.
- Memory-optimized compute nodes.

The queue and compute node-group configurations work together to determine where different jobs are executed.

6. EC2 Compute Nodes

The actual HPC workloads run on **Amazon EC2 compute instances**, which function as the worker nodes responsible for performing the calculations required by submitted jobs. Computationally intensive applications can use multiple compute nodes to execute tasks in parallel.

For example:

Worker Node 1 → Part of the computation
Worker Node 2 → Another part of the computation

The results are subsequently combined as required by the application.

7. EFA – Elastic Fabric Adapter

The architecture shows **EFA (Elastic Fabric Adapter)** connecting the worker nodes. EFA provides high-performance networking for suitable HPC workloads that require rapid communication between compute instances.

For example:

Worker Node 1 ⇄ EFA ⇄ Worker Node 2

This capability is particularly useful for distributed HPC applications in which compute nodes need to communicate frequently during execution. The EFA connection is optional and is used when it is configured and supported by the selected instance type and environment.

SGOU

8. **Read Input / Write Results**

During execution, compute nodes need to access the input data and store the results generated by the application. Therefore, the compute nodes perform two primary storage operations:

Read Input Data → Process Data → Write Results

This enables computational nodes to work with data stored in shared or accessible cloud storage.

9. **Shared Storage**

The bottom section of the architecture represents storage services that can be used for HPC workloads. Fig. 2.3.1 includes the following storage services:

Amazon EFS:

Provides shared file storage that can be accessed by multiple compute resources.

Amazon FSx for Lustre:

Provides a high-performance file system suitable for demanding HPC workloads and large-scale data processing.

Amazon S3:

Provides object storage for datasets, application data, and results.

The appropriate storage service is selected according to the application's performance, access, and data-management requirements.

2.3.1.2 Advantages of AWS Parallel Cluster Service (AWS PCS)

1. **Reduced Infrastructure Management:** Organizations can minimize the physical infrastructure they need to purchase, configure, and maintain.
2. **Scalability:** Computing resources can be scaled up or down based on workload requirements.
3. **Faster HPC Deployment:** Cloud-based infrastructure can significantly reduce the time needed to set up an HPC environment compared with building a physical cluster.
4. **Flexible Resource Selection:** Organizations can choose suitable computing resources according to the specific requirements of their applications.
5. **Familiar Slurm Environment:** Users can continue using Slurm-based HPC workflows without having to adopt a completely different job-management system.
6. **Cloud Integration:** HPC workloads can be integrated with other AWS cloud services for storage, networking, data processing, and application development.

SGOU

7. **Suitable for Variable Workloads:** Organizations with workloads that fluctuate significantly over time can benefit from flexible, cloud-based resource provisioning.

2.3.1.3 Limitations and Considerations

Although AWS PCS offers several advantages, organizations should consider the following factors before adopting it.

- **Cost:** Running a large number of compute instances for extended periods can lead to substantial cloud expenses. Therefore, effective resource planning and cost monitoring are essential.
- **Application Compatibility:** Not all applications can automatically take advantage of parallel computing. Applications may need to be specifically designed or configured to utilize multiple processors or compute nodes efficiently.
- **Data Transfer:** Transferring very large datasets to or from the cloud may require considerable time and can also result in additional data transfer costs.
- **Technical Expertise:** Although AWS PCS reduces the effort involved in infrastructure management, users still require knowledge of HPC concepts, Linux environments, Slurm, and the AWS platform.
- **Performance Requirements:** Certain applications may require specialized hardware or very low-latency communication. Consequently, suitable computing and networking resources must be selected carefully.

2.3.2 Serverless Computing

Serverless Computing is a modern cloud-based execution model that allows developers to develop and deploy applications without directly managing server infrastructure. In traditional computing environments, developers or system administrators are responsible for managing servers, operating systems, storage, scaling, security, and software updates. In contrast, serverless computing transfers these infrastructure management responsibilities to the cloud service provider, which handles them automatically.

In this model, the developer primarily writes a small unit of code known as a function and uploads it to the cloud platform. The cloud provider manages the execution of the function, allocates the required computing resources, and automatically scales the resources up or down according to the number of incoming requests. This makes Serverless Computing particularly suitable for applications that require rapid deployment and flexible resource scaling.

Common examples of serverless platforms include AWS Lambda, Google Cloud Functions, and Microsoft Azure Functions.

2.3.2.1 Working Principle of Serverless Computing

Serverless computing operates primarily on an **event-driven model**. The developer uploads a function, which is a small piece of code written in a programming language

SGOU

such as Python, JavaScript, or Java, to the cloud platform.

The function remains inactive until a specific event triggers its execution. Such an event may be an HTTP request, file upload, database update, message received from another application, or a scheduled timer. When the event occurs, the cloud provider automatically executes the function within a lightweight execution environment, such as a container.

After completing its task, the function returns the result to the user or stores it in a database. The computing resources allocated for the function are then released. The user is charged only for the actual time taken to execute the function, which is generally measured in milliseconds. Therefore, there is no charge for the period during which the function remains idle.

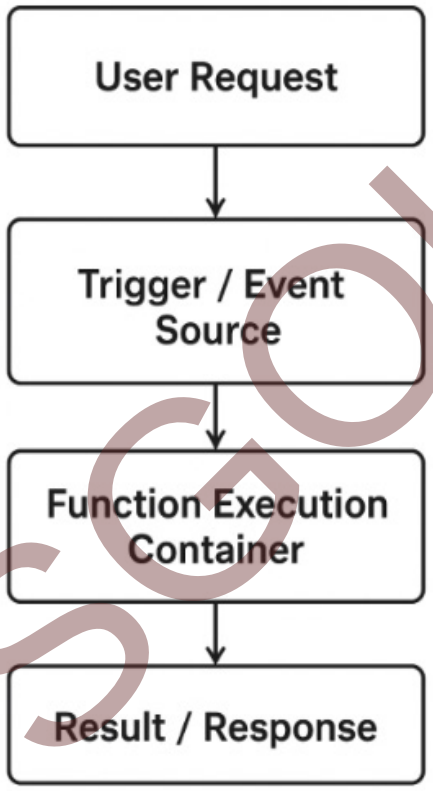


Fig 2.3.2 Serverless Architecture

The figure illustrates how a user request or system event triggers a function, which runs within a container, and then returns the result to the user.

2.3.2.2 Key Features of Serverless Computing

One of the primary features of Serverless Computing is Event-Driven Execution, where functions are executed only when a particular event occurs. This feature makes serverless computing suitable for real-time applications such as file processing, online transactions, and messaging.

Another important feature is Automatic Scaling. When the number of user requests increases, the cloud provider automatically launches additional instances of the function

to handle the increased workload. When the demand decreases, the number of instances is automatically reduced without requiring manual intervention.

Serverless functions are generally Stateless, which means that they do not retain information between successive executions. Any information that needs to be preserved must therefore be stored using external services such as databases or cloud storage.

No Server Management is another major advantage of serverless computing. Developers are not required to configure servers, install software, perform updates, or manage system uptime, as these responsibilities are handled by the cloud provider.

Finally, serverless computing uses a Metered Billing model, in which users are charged according to the number of function executions and the duration of each execution, often measured in milliseconds. This usage-based approach can make serverless computing highly cost-efficient.

2.3.2.3 Advantages of Serverless Computing

Serverless Computing offers several advantages by eliminating the need for organizations to manage physical or virtual servers. It helps reduce operational costs because organizations do not have to spend on server hardware, power, cooling, and server administration.

It provides high scalability, as the cloud platform automatically increases or decreases computing resources according to changes in demand. This makes serverless computing suitable for applications that experience unpredictable or fluctuating traffic.

Serverless computing also supports faster application development, allowing developers to concentrate primarily on application logic without spending significant time on infrastructure configuration and management.

Another important advantage is fine-grained billing, where users pay only for the computing resources consumed during application execution. This is particularly beneficial for applications that operate intermittently rather than continuously.

2.3.2.4 Disadvantages of Serverless Computing

Serverless Computing can experience cold start delays. When a function has not been invoked for a certain period, the system may need additional time to initialize its execution environment. As a result, the first request may have a slightly higher response time.

Another limitation is the restricted execution time. Many cloud providers set a maximum runtime for serverless functions, often limited to a few minutes. Therefore, serverless computing may not be suitable for processes that require extended execution periods.

Vendor lock-in is another concern. When an application is developed using the specific tools and services of a particular cloud provider, migrating it to another platform can become difficult.

SGOU

Serverless computing also makes debugging and monitoring more challenging because developers do not have direct access to the underlying servers and have limited control over the execution environment.

For example, consider an e-commerce website where a serverless function is automatically triggered when a customer places an order. The function can update inventory, process payment confirmation, send an order confirmation email, and update the delivery system in real time, without requiring manual server management.

Similarly, when a user uploads an image, a serverless function can be automatically triggered to resize the image, apply a filter, and store the processed image in cloud storage.

2.3.2.5 Common Use Cases of Serverless Computing

Serverless Computing is widely adopted for various real-world applications. It is commonly used in chatbots and microservices, where individual functions are designed to perform specific and independent tasks.

It is also suitable for real-time data processing, such as processing continuous data streams generated by sensors, social media platforms, and other connected systems.

Serverless computing is particularly useful for Internet of Things (IoT) applications, where large numbers of devices periodically transmit data to cloud-based systems for processing and analysis.

Another important application is notification and alert systems, in which serverless functions are automatically triggered to send messages, emails, or alerts when predefined events or conditions occur.

Recap

- Cloud computing has evolved beyond basic virtual machines, storage, and networking.
- Managed High-Performance Computing (HPC) provides powerful resources for computationally intensive workloads.
- HPC commonly uses multiple processors, large memory, high-speed networks, and parallel computing techniques.
- Managed HPC reduces the infrastructure management responsibilities of organizations.
- AWS Parallel Computing Service (AWS PCS) is a managed cloud-based HPC service offered by Amazon Web Services.
- AWS PCS enables users to create and operate HPC clusters without manually managing the entire underlying infrastructure.

SGOU

- AWS PCS uses Slurm for job submission, scheduling, and execution.
- The login node provides users with an entry point for preparing data, compiling applications, and submitting jobs.
- Slurm queues hold submitted jobs until the required computing resources become available.
- Compute node groups provide different types of computing resources according to workload requirements.
- Amazon EC2 compute instances perform the actual computational tasks in an AWS PCS environment.
- EFA can provide high-performance communication between compute nodes for suitable HPC workloads.
- AWS PCS can use services such as Amazon EFS, Amazon FSx for Lustre, and Amazon S3 for data storage.
- Serverless Computing allows developers to execute applications or functions without directly managing servers.
- Serverless computing uses event-driven execution, automatic scaling, stateless functions, and metered billing, but may face limitations such as cold starts, execution-time restrictions, and vendor lock-in.

Objective Type Questions

1. Which cloud-based approach is designed for computationally intensive workloads using multiple processors and parallel computing techniques?
2. Which workload manager is used by AWS Parallel Computing Service (AWS PCS)?
3. Which AWS service provides managed HPC clusters based on Slurm?
4. Which AWS service provides virtual compute instances for HPC workloads?
5. Which networking technology can provide high-performance communication between HPC compute nodes?
6. Which AWS storage service provides object storage for datasets and results?
7. Which AWS storage service provides a high-performance file system suitable for HPC workloads?
8. Which AWS storage service provides shared file storage accessible by multiple compute resources?

SGOU

9. Which serverless model executes functions in response to specific events?
10. What is the name given to a small unit of code executed in a serverless environment?
11. Which serverless AWS service is commonly used to execute functions without managing servers?
12. What type of scaling automatically increases or decreases resources according to demand?
13. What type of billing charges users based on function execution and duration?
14. What issue occurs when a serverless function requires additional time to initialize its execution environment?

Answers to Objective Type Questions

1. HPC
2. Slurm
3. AWS PCS
4. EC2
5. EFA
6. S3
7. Lustre
8. EFS
9. Event-driven
10. Function
11. Lambda
12. Autoscaling
13. Metered
14. Cold-start

SGOU

Assignments

1. Explain the concept of Managed High-Performance Computing (HPC) and discuss its major characteristics and applications.
2. Describe AWS Parallel Computing Service (AWS PCS) and explain how it simplifies the deployment and management of HPC workloads.
3. Explain the architecture and core components of AWS PCS, including the login node, Slurm controller, queues, compute node groups, EC2 compute nodes, EFA, and shared storage.
4. Compare a traditional HPC environment with an AWS PCS environment, highlighting the differences in infrastructure management, scalability, resource provisioning, and workload execution.
5. Discuss the advantages of AWS PCS and explain how it helps organizations manage computationally intensive workloads efficiently.
6. Explain the limitations and considerations of AWS PCS, including cost, application compatibility, data transfer, technical expertise, and performance requirements.
7. Describe Serverless Computing, its working principle, and the role of event-driven execution, automatic scaling, and cloud-managed infrastructure.
8. Explain the advantages, disadvantages, and common use cases of Serverless Computing with suitable examples.

Reference

1. Amazon Web Services. (n.d.). AWS Parallel Computing Service documentation. Amazon Web Services.
2. Amazon Web Services. (n.d.). AWS Lambda documentation. Amazon Web Services.
3. Pethuru Raj, R., & Raman, A. (2018). Cloud computing: An overview. CRC Press.
4. Erl, T., Puttini, R., & Mahmood, Z. (2013). Cloud computing: Concepts, technology & architecture. Pearson.

SGOU

Suggested Reading

1. Sterling, T., Anderson, M., & Brodowicz, M. (2017). High performance computing: Modern systems and practices. Morgan Kaufmann.
2. Amazon Web Services. (n.d.). AWS PCS user guide. Amazon Web Services.
3. Amazon Web Services. (n.d.). AWS Lambda – Serverless compute. Amazon Web Services.
4. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2011). Cloud computing: Principles and paradigms. John Wiley & Sons.

SGOU

Unit 4
Popular Cloud Computing Service Models

Learning Outcomes

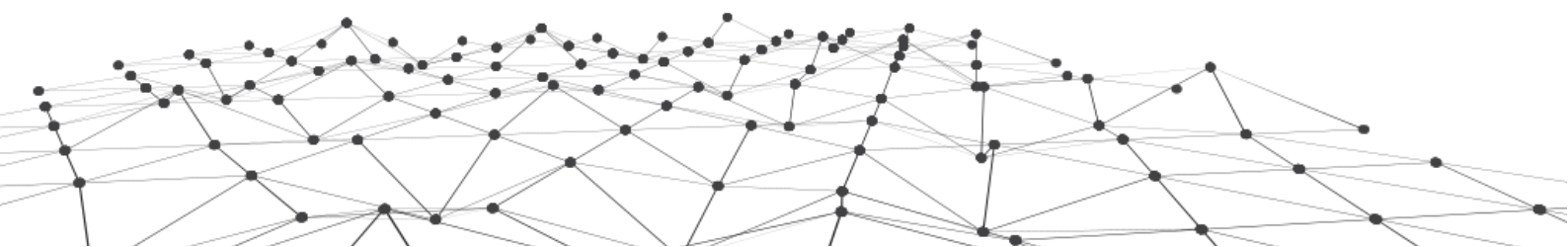
Upon completion of the unit, learner will be able to:

- identify the major features of AWS, Microsoft Azure, and Google Cloud
- explain the basic architecture and core services of AWS, Azure, and Google Cloud
- describe the advantages, and limitations of AWS, Azure, and Google Cloud
- compare AWS, Azure, and Google Cloud based on different aspects

Prerequisite

Learners should have a general awareness of how organizations use digital technologies to store data, run applications, and deliver online services. They should be familiar with common computing resources such as servers, databases, storage systems, and networks. Knowledge of how organizations manage increasing workloads and changing user demands will help learners understand the importance of cloud platforms. Studying AWS, Microsoft Azure, and Google Cloud Platform (GCP) is important because these platforms provide scalable resources without requiring organizations to maintain all physical infrastructure themselves. For example, an online university may experience a sudden increase in users during admission or examination periods. A cloud platform can provide additional computing and storage resources during such peak periods and reduce them when demand decreases. AWS, Azure, and GCP also provide services for storing learner records, hosting applications, managing databases, analyzing data, and supporting AI-based applications. Comparing these platforms helps organizations select suitable services based on factors such as workload requirements, scalability, cost, security, and availability. Understanding their architectures and services also helps learners recognize how real-world applications are deployed and managed in cloud environments. Therefore, studying these three major cloud platforms provides practical knowledge for understanding how modern organizations build, operate, and scale digital services.

SGOU



Keywords

Cloud Infrastructure, Availability Zones, Shared Responsibility Model, Object Storage, Cost Management.

Discussion

Cloud computing service models provide users with on-demand access to computing resources such as servers, storage, databases, networking, and software over the Internet. Several leading cloud providers offer a wide range of services to support application development, data management, analytics, and enterprise computing. Among the most popular cloud platforms are Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).

2.4.1 Introduction to Amazon Web Services (AWS)

Amazon Web Services (AWS) is a comprehensive cloud computing platform that provides a wide range of technology services through the Internet. Introduced in 2006, AWS has significantly changed the way individuals, businesses, and government organizations access and utilize computing resources.

AWS enables users to obtain essential resources such as computing power, storage, and databases on demand, without having to purchase and maintain their own physical data centres. Its cloud-based approach provides greater flexibility and allows organizations to select and use the resources they require according to their needs.

Table 2.4.1 Before and After AWS: Transformation in Cloud Computing

Aspect	Before AWS	After AWS
1.Infrastructure Cost	High upfront infrastructure costs	No upfront costs; pay-as-you-go pricing
2.Scalability	Limited scalability; resources had to be purchased in advance	Instant scalability based on demand
3.Deployment	Long time required to deploy infrastructure and applications	Faster deployment with on-demand resources
4.Global Reach	Establishing infrastructure across different regions was difficult and expensive	Global reach with a few clicks through AWS Regions
5.Disaster Recovery & Availability	Disaster recovery was complex and required significant infrastructure	Built-in disaster recovery options and high availability

SGOU

2.4.1.1 Architecture of AWS

AWS operates through a global infrastructure consisting of Regions and Availability Zones (AZs). A Region represents a specific geographical area, and each Region contains multiple Availability Zones. These Availability Zones are independent data centers that are interconnected through high-speed networks, providing reliable communication and data transfer.

This architecture supports fault tolerance, high availability, and disaster recovery. If one Availability Zone experiences a failure, applications and workloads can be designed to operate through another Availability Zone, thereby reducing service interruptions and minimizing downtime.

AWS services can be accessed and managed through the AWS Management Console, a web-based interface, as well as the AWS Command Line Interface (CLI) and Software Development Kits (SDKs). Users can create and manage virtual machines through Amazon EC2, store objects using Amazon S3, manage databases with Amazon RDS or Amazon DynamoDB, configure cloud networks using Amazon VPC, and deliver content efficiently through Amazon CloudFront.

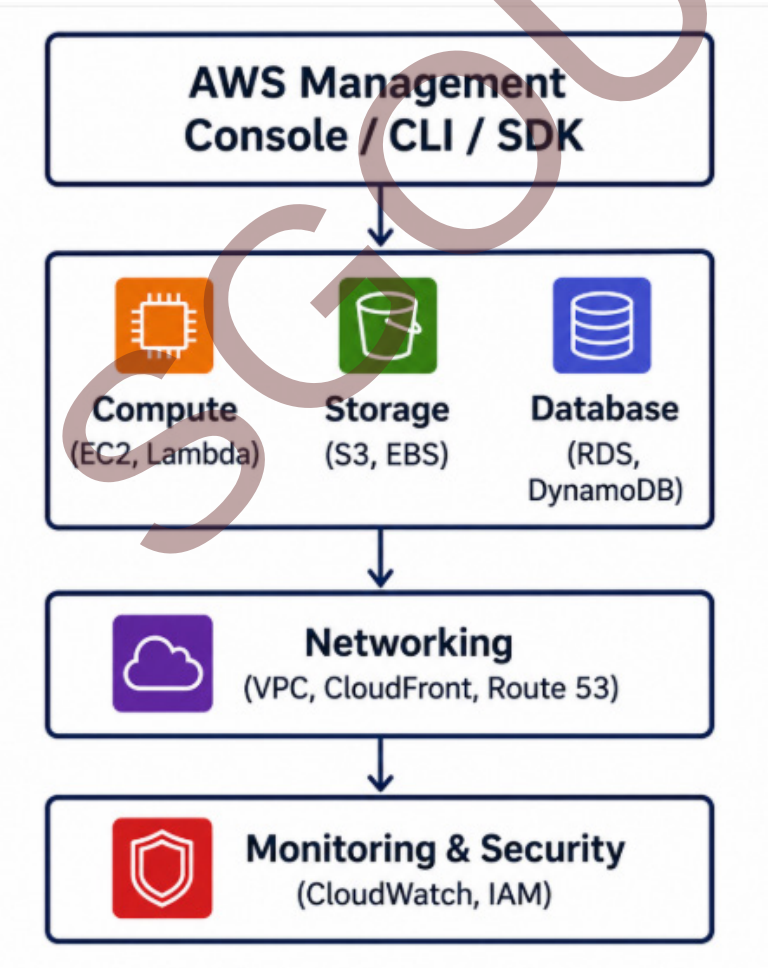


Fig. 2.4.1 Simplified Architecture of AWS Cloud Services

SGOU

2.4.1.2 Key features of AWS include:

- **On-demand resources:** Computing, storage, databases, and other services can be accessed whenever required.
- **Reduced infrastructure costs:** Organizations can avoid the expense of purchasing and maintaining physical data centres and hardware.
- **Pay-as-you-go pricing:** Users generally pay for the resources and services they actually consume.
- **Flexible scalability:** Computing resources can be increased or decreased according to changing workloads and business requirements.

2.4.1.3 Shared Responsibility Model

The AWS Shared Responsibility Model explains how security responsibilities are divided between AWS and its customers. AWS is responsible for protecting the underlying infrastructure that provides cloud services, while customers are responsible for securing the data, applications, operating systems, and resources that they deploy or configure within the AWS environment. The exact division of responsibility can vary depending on the AWS service being used.

AWS Responsibility – Security of the Cloud

AWS is responsible for the security of the cloud, which means protecting the infrastructure used to operate AWS services. This includes the physical data centers, servers, storage devices, networking equipment, virtualization infrastructure, and core AWS services. AWS also implements physical security measures and follows appropriate security and compliance practices to protect its facilities and infrastructure.

For example, when a customer runs an application on an Amazon EC2 instance, AWS is responsible for securing the physical server, data center, networking infrastructure, and underlying virtualization environment on which the EC2 instance operates. The customer does not have to physically protect or maintain these data center facilities.

Customer Responsibility – Security in the Cloud

Customers are responsible for security in the cloud, which involves protecting the resources, applications, and data that they use on AWS. This may include encrypting sensitive information, configuring network security controls, managing users and permissions through AWS Identity and Access Management (IAM), applying operating-system patches where applicable, and implementing appropriate application-level security mechanisms.

For example, consider a university that deploys an online admission system on Amazon EC2. AWS is responsible for protecting the physical infrastructure and underlying cloud platform. However, the university is responsible for configuring appropriate firewall or security-group rules, creating suitable IAM permissions, protecting applicant information through encryption, applying security updates to the operating system, and securing the admission application against unauthorized access.

SGOU

Similarly, if the university stores student records in Amazon S3, AWS is responsible for securing the underlying storage infrastructure, while the university must configure appropriate S3 access permissions, encryption, and data protection policies. If confidential student information is accidentally made publicly accessible because of an incorrect access configuration, that is generally a customer-side security responsibility.

Thus, the Shared Responsibility Model can be summarized as "security of the cloud" being primarily AWS's responsibility and "security in the cloud" being the customer's responsibility. Understanding this distinction helps organizations identify the security controls they must implement themselves while relying on AWS to protect the underlying cloud infrastructure.

2.4.1.4 Core AWS Services

AWS provides a wide range of cloud services that support computing, storage, databases, networking, application development, and many other requirements. Among these, several fundamental services are particularly important for understanding the AWS cloud environment. These services allow organizations to run applications, store data, manage databases, and establish secure and efficient network connections without having to maintain all the underlying physical infrastructure.

1. Compute Services

Amazon EC2 (Elastic Compute Cloud) provides resizable virtual computing resources in the AWS Cloud. An EC2 instance can be considered a virtual server whose operating system and computing capacity can be selected according to application requirements. Users can choose operating systems such as Linux or Windows and configure resources such as CPU, memory, storage, and networking. For example, an organization developing an online admission application can run its web application on an EC2 instance and increase the computing capacity when the number of users increases.

AWS Lambda is a serverless computing service that allows developers to execute code without directly managing servers. The developer uploads the required code, and Lambda executes it when a specified event or trigger occurs. For example, when a user uploads an image to an Amazon S3 bucket, a Lambda function can automatically be triggered to resize or process the image. Since the infrastructure is managed by AWS, developers can concentrate primarily on the application logic.

AWS Elastic Beanstalk is a Platform as a Service (PaaS) that simplifies the deployment and management of web applications. Developers can upload their application code, while AWS manages several underlying deployment tasks, including capacity provisioning, load balancing, scaling, and application health monitoring. For example, a developer can upload a Python or Java web application to Elastic Beanstalk without manually configuring every server required to run the application.

2. Storage Services

Amazon S3 (Simple Storage Service) is an object storage service designed to store and retrieve large amounts of data from anywhere through the web. It can be used to

SGOU

store files such as images, videos, documents, application backups, and other digital objects. For example, an educational institution can use Amazon S3 to store recorded lectures, learner documents, examination materials, and backup files. S3 is designed to provide highly durable and scalable storage for different types of data.

Amazon EBS (Elastic Block Store) provides persistent block-level storage that can be attached to Amazon EC2 instances. It functions similarly to a virtual hard disk associated with a virtual server and is suitable for applications that require persistent storage. For example, an EC2 instance running a database application can use an EBS volume to store database files and other application data that must remain available even when the instance is stopped and restarted.

Amazon Glacier, now provided through Amazon S3 Glacier storage classes, is designed for low-cost, long-term data archiving. It is suitable for information that needs to be retained for a long period but does not require frequent access. For example, an organization may use archival storage to retain old financial records, historical documents, system backups, or completed examination records for future reference.

3. Database Services

Amazon RDS (Relational Database Service) is a managed service for running relational databases in the AWS environment. It supports database engines such as MySQL, PostgreSQL, Oracle, Microsoft SQL Server, and Amazon Aurora. AWS manages many routine database administration tasks, allowing organizations to focus on their applications and data. For example, an online university admission system can use Amazon RDS to store learner information, application details, programme information, and transaction records.

Amazon DynamoDB is a managed NoSQL database service designed to provide fast and predictable performance with automatic scalability. It is suitable for applications that need to handle large amounts of data and a high number of requests without requiring traditional relational database structures. For example, an online shopping application can use DynamoDB to manage product information, shopping-cart data, or user activity that needs to be accessed quickly at large scale.

4. Networking Services

Amazon VPC (Virtual Private Cloud) allows users to create a logically isolated virtual network within the AWS Cloud. Organizations can define network components such as IP address ranges, subnets, routing rules, and security controls according to their requirements. For example, a university can create a VPC for its online admission system and place application servers and databases in appropriately configured network segments while controlling access between them.

Amazon CloudFront is a Content Delivery Network (CDN) that helps deliver static and dynamic web content to users with reduced latency. It uses a distributed network of locations to cache and deliver frequently requested content closer to end users. For example, an educational website containing video lectures, images, and downloadable

SGOU

resources can use CloudFront to deliver these resources more efficiently to learners located in different geographical regions.

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. It helps translate human-readable domain names into the IP addresses or other endpoints associated with applications and services. For example, when a user enters a university website address such as www.example.edu, Route 53 can help direct the request to the appropriate application endpoint hosted in AWS.

2.4.1.5 AWS Pricing Models and Cost Management

1. Pay-as-You-Go Pricing

AWS follows a usage-based pricing approach, where customers are charged according to the resources and services they actually consume. The cost may depend on factors such as computing resources, including EC2 instances, storage services such as Amazon S3, data transfer, and the number of requests or service executions, such as AWS Lambda invocations. This flexible pricing model is particularly suitable for organizations whose workloads vary over time.

2. On-Demand Instances

On-Demand Instances allow users to pay for computing resources based on the actual duration of usage, without requiring long-term commitments or upfront payments. Depending on the instance type, billing may be calculated per hour or per second. This option is useful for short-term, irregular, or unpredictable workloads, as well as for users who are new to AWS and want to experiment with its services. It is also appropriate for applications that require continuous availability and cannot easily be interrupted. On-Demand Instances provide considerable flexibility and are commonly used for development, testing, and prototyping.

3. Reserved Pricing

AWS Reserved pricing is designed for workloads that have predictable and consistent resource requirements. Users can make a long-term commitment, typically for one or three years, to obtain computing capacity at a lower cost than standard On-Demand pricing. Reserved pricing is available for services such as EC2, RDS, and Redshift. Depending on the service and commitment, users can achieve significant cost savings. AWS also provides different payment choices, including full upfront payment, partial upfront payment, or no upfront payment.

4. Spot Instances

Spot Instances enable users to utilize unused EC2 computing capacity at significantly reduced prices. The pricing can vary according to the availability of resources and market demand. Since Spot Instances may be interrupted when AWS requires the capacity, they are most appropriate for workloads that can tolerate interruptions. Common applications include batch processing, large-scale data analysis, testing, and other flexible computing tasks. This pricing option can provide substantial savings compared with On-Demand Instances.

SGOU

5. AWS Free Tier

AWS provides Free Tier benefits that allow users to explore selected AWS services without or with limited charges, subject to the applicable service limits and current AWS terms. These benefits can include limited usage of services such as EC2, Amazon S3, and AWS Lambda. The Free Tier is particularly useful for students, developers, startups, and organizations that want to learn about AWS, develop applications, or test cloud-based solutions before committing to paid usage.

6. AWS Pricing Calculator

The AWS Pricing Calculator is a tool that helps users estimate the expected cost of their AWS infrastructure. Users can select the required services, specify configurations and expected usage levels, and obtain an estimated cost. This allows organizations to plan their cloud budgets more effectively and compare different infrastructure configurations before deployment.

7. Cost Management and Optimization

AWS provides several tools to monitor cloud usage and control expenditure. Services such as AWS Cost Explorer help users analyze spending patterns and identify areas of high resource consumption, while AWS Budgets allows organizations to establish customized spending thresholds and receive notifications when costs or usage approach predefined limits. These tools support continuous cost monitoring and help organizations optimize their cloud resources while avoiding unnecessary expenditure.

2.4.2 Introduction to Microsoft Azure

Microsoft Azure, introduced in 2010, is a widely adopted cloud computing platform that enables users to build, deploy, and manage applications efficiently. It provides a broad range of services, including computing, storage, networking, analytics, databases, cognitive services, and Internet of Things (IoT). With its "Pay-As-You-Go" pricing model, Azure offers scalable virtual resources without requiring significant upfront investment in physical infrastructure.

- Offers virtual machines, analytics, monitoring tools, and high-speed data processing capabilities.
- Reduces the need for extensive physical infrastructure and large initial investments.
- Uses a cost-effective pricing model in which users pay only for the resources they consume.

2.4.2.1 Microsoft Azure: Working

Microsoft Azure is a cloud computing platform that provides a wide range of services for developing, deploying, and managing applications and IT resources. It supports public cloud services and can also be integrated with private infrastructure to enable hybrid cloud environments. Azure makes extensive use of virtualization technology, where a hypervisor abstracts the underlying physical hardware and allows multiple virtual machines (VMs) to operate independently on the same physical server.

SGOU

- **Virtualization:** Hypervisors enable multiple virtual machines to share the resources of a single physical server while operating as separate computing environments.
- **Operating System Support:** Azure virtual machines can run different operating systems, including Microsoft Windows and Linux, according to application requirements.
- **Scalable Infrastructure:** Azure operates large-scale, geographically distributed data centers containing interconnected servers and networking resources, enabling applications to achieve scalability, reliability, availability, and high performance.

2.4.2.2 Microsoft Azure Architecture

Microsoft Azure is a globally distributed cloud computing platform that provides the infrastructure and services required to develop, deploy, and manage applications at scale.

The architecture of Microsoft Azure is designed to deliver scalable, secure, reliable, and high-performance cloud services. Azure organizes its infrastructure into regions, availability zones, and resource groups, enabling organizations to efficiently deploy, manage, and protect their cloud resources.

An Azure Region is a geographical location that contains one or more data centers situated within a specific area. These data centers are interconnected through high-speed, low-latency networks. Regions allow organizations to select suitable locations for deploying applications and storing data based on factors such as performance, regulatory requirements, and the geographical location of users. For example, Azure has regions such as Central India, South India, West US, and North Europe.

Within supported regions, Azure provides Availability Zones, which are physically separate data center locations with independent power, cooling, and networking systems. They are designed to protect applications and data from infrastructure failures. If one availability zone becomes unavailable because of a power failure, natural disaster, or technical problem, resources in another zone can continue operating. This improves high availability and fault tolerance.

Another key component of Azure architecture is the Resource Group. A resource group is a logical container used to organize related Azure resources, such as virtual machines, databases, storage accounts, and networking components. Resources associated with a particular application or project can be placed within the same resource group, making it easier to manage, monitor, secure, update, and remove them collectively.

Azure provides several tools for users to access and manage its services. The Azure Portal is a web-based graphical interface that provides a dashboard for managing cloud resources. Administrators and experienced users can use Azure PowerShell and the Azure CLI (Command-Line Interface) to execute commands and automate management tasks. Developers can also use Azure SDKs for programming languages such as Python, Java, and .NET to develop and manage cloud-based applications.

SGOU

Azure Compute Services provide the processing resources required to execute applications and workloads. Azure Virtual Machines (VMs) allow users to run Windows and Linux operating systems in the cloud. Azure App Service is used for hosting web applications, mobile applications, and APIs. For event-driven workloads, Azure Functions provides a serverless environment in which code can be executed automatically in response to events without requiring direct server management.

Azure provides different storage services to meet various data storage requirements. Azure Blob Storage is suitable for storing unstructured data such as images, videos, and documents. Azure Files provides shared file storage that can be accessed by multiple systems. Azure Disk Storage provides persistent storage for virtual machines and functions similarly to a physical hard disk. These storage services offer scalability, security, and accessibility for cloud-based applications.

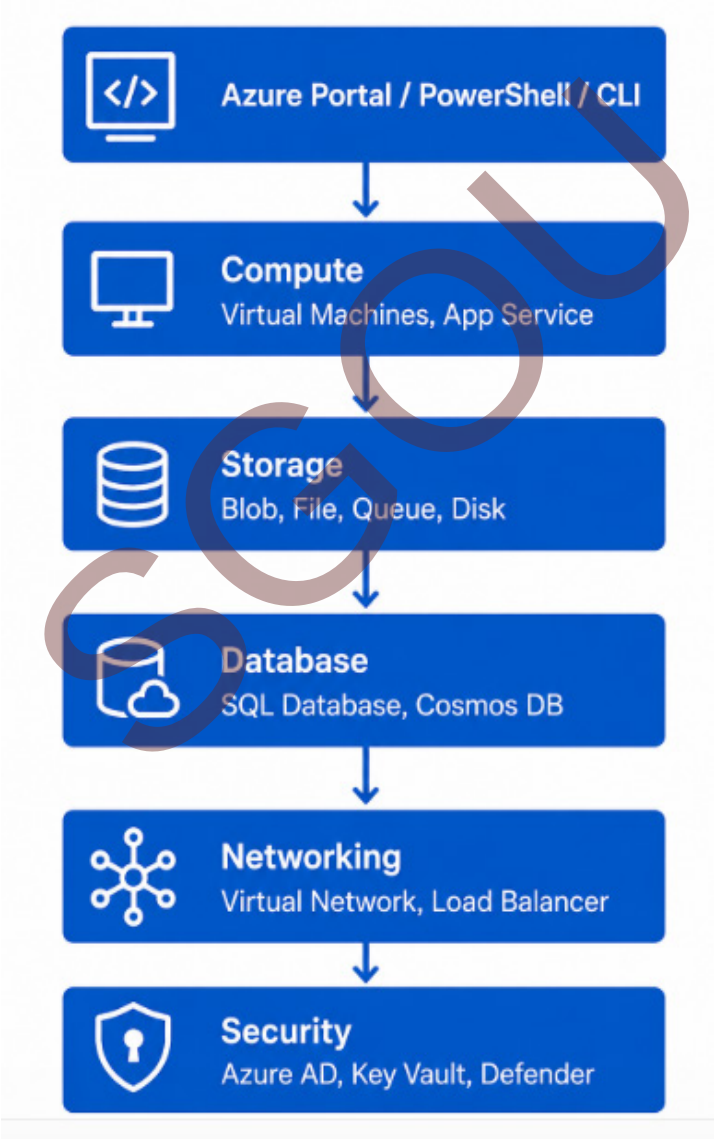


Fig. 2.4.2 Simplified Architecture of Microsoft Azure Cloud

For database management, Azure provides services such as Azure SQL Database, a fully managed relational database service based on SQL Server technologies, and Azure Cosmos DB, a globally distributed NoSQL database designed for applications requiring high performance and large-scale data processing.

Azure Networking provides secure communication between cloud resources. Azure Virtual Network (VNet) enables resources to communicate with one another within a secure network environment. Azure Load Balancer distributes incoming network traffic across multiple servers or resources, helping improve application availability and performance.

Azure also provides various security and identity management services. Microsoft Entra ID (formerly Azure Active Directory) manages user identities and access to Azure resources. Azure Key Vault securely stores and manages sensitive information such as passwords, certificates, encryption keys, and other application secrets. Together, these services help organizations protect their applications, data, identities, and cloud infrastructure.

2.4.2.3 Key features of Azure

One of the key features of Microsoft Azure is its Hybrid Cloud Support, which enables organizations to connect their on-premises infrastructure with Azure cloud services. Organizations do not need to completely replace their existing systems; instead, they can operate some resources within their local data centers while using other resources in the cloud. This hybrid approach provides greater flexibility, improved control, and potential cost savings.

Azure also provides advanced Artificial Intelligence (AI) and Machine Learning (ML) services that support the development of intelligent applications. These services can be used to create chatbots, image recognition systems, predictive analytics applications, and automated decision-making solutions. Azure Machine Learning and Azure AI services simplify the integration of AI capabilities into applications, even for developers with limited data science expertise.

Another significant feature of Azure is its strong developer support. Azure integrates with development platforms such as Visual Studio and GitHub, enabling developers to build, test, and deploy applications efficiently. Azure DevOps provides tools for source control, software testing, and continuous integration and continuous deployment (CI/CD), supporting streamlined software development and delivery.

Azure is also designed to provide high availability and disaster recovery. Cloud services can be backed up and restored when data loss or system failures occur. Azure provides service availability commitments that help organizations maintain continuous access to critical applications, making it suitable for business and public-sector workloads.

Security is another major strength of Azure. The platform supports various international security, privacy, and compliance standards, including GDPR, ISO, HIPAA, and SOC requirements. Data can be protected through encryption, access controls, identity management, and monitoring mechanisms. Azure also provides

SGOU

security services that help organizations identify potential threats and respond to security incidents effectively.

2.4.2.4 Scaling and Management of Azure

Microsoft Azure provides several features and services for efficiently managing cloud resources and adjusting computing capacity according to workload requirements. These capabilities help organizations maintain application performance while optimizing resource utilization and operational costs.

- **Auto-Scaling:** Azure supports automatic scaling, allowing computing resources to be increased or decreased dynamically according to workload demand. This helps applications maintain performance during periods of high demand while reducing resource consumption and costs when demand decreases.
- **Resource Groups:** Azure resources can be logically organized into resource groups, making it easier to manage related resources as a single unit. Resource groups simplify resource organization, monitoring, access control, deployment, and lifecycle management.
- **Azure Resource Manager (ARM):** Azure Resource Manager provides a unified management layer for deploying, configuring, monitoring, and controlling Azure resources. It supports templates and declarative deployment, enabling resources to be deployed consistently and managed efficiently. ARM also facilitates role-based access control, resource organization, and centralized management of cloud resources.

2.4.2.5 Advantages of Microsoft Azure

Microsoft Azure is particularly beneficial for organizations that already use Microsoft products and technologies. Since many organizations are familiar with Windows and other Microsoft tools, migrating to and adopting Azure can be easier and faster. Azure also supports both Windows and Linux operating systems, providing greater flexibility in deploying different types of applications and workloads.

Azure provides seamless hybrid cloud integration, allowing organizations to combine on-premises infrastructure with cloud-based resources. This is especially useful for organizations that cannot move all their systems to the cloud because of security, regulatory, or legal requirements. Azure also supports automatic scalability, enabling computing resources to be increased or reduced according to changes in workload and demand.

Another important advantage is strong security and compliance. Microsoft invests significantly in cloud security, and Azure provides features such as built-in firewalls, data encryption, identity and access management, and threat detection. These capabilities make Azure suitable for handling sensitive data and critical workloads, including those used by financial institutions and government organizations.

Azure also provides a wide range of cloud services through a single platform. Organizations can access services for website hosting, artificial intelligence, databases,

SGOU

virtual networking, storage, and many other requirements. These services can be managed through a centralized interface, making cloud administration more convenient, integrated, and efficient.

2.4.2.6 Disadvantages of Microsoft Azure

Although Microsoft Azure offers numerous benefits, it also has certain limitations. Users who are unfamiliar with Microsoft technologies may find the Azure platform difficult to understand initially. Since Azure provides a wide range of services and features, beginners may find it challenging to identify and use the appropriate services.

Another limitation is its complex pricing structure. Azure charges customers based on resource usage, including computing time, storage consumption, and data transfer. This usage-based pricing can make it difficult for beginners to accurately predict the total cost of their cloud resources. In addition, certain Azure services may require separate Microsoft licenses, which can increase the overall expenditure.

Azure performance can also vary depending on the geographical region selected. When the nearest Azure data center is located far from the users, network latency may increase and result in slower response times. Furthermore, migrating existing applications and systems to Azure may require appropriate technical expertise, detailed planning, and configuration to ensure a smooth transition.

2.4.2.7 Cost Management Tools in Azure

Managing cloud expenditure is an important aspect of cloud computing because organizations often use multiple services, resources, and workloads simultaneously. Microsoft Azure provides several tools that help organizations estimate, monitor, analyze, control, and optimize cloud costs. These tools enable administrators and organizations to understand where money is being spent and identify opportunities to reduce unnecessary expenditure.

The major Azure cost management tools include Azure Cost Management and Billing, Azure Pricing Calculator, and Azure Advisor.

1. **Azure Cost Management and Billing**

Azure Cost Management and Billing is a set of capabilities that helps organizations monitor and manage their Azure expenditure. It provides information about how cloud resources are being used and how much they are costing over a selected period.

Major Features

- **Cost Analysis:** Users can examine their Azure spending and identify which services, subscriptions, resource groups, or resources are contributing to the overall cost.
- **Budget Management:** Organizations can create budgets for subscriptions or other relevant scopes and track their spending against those budgets.
- **Cost Alerts:** Notifications can be configured to alert users when spending reaches specified thresholds or when certain cost-related conditions occur.

SGOU

- **Cost Breakdown:** Expenditure can be analyzed according to factors such as services, resource groups, subscriptions, and usage.
- **Cost Optimization:** Historical spending information can be used to identify resources that are underutilized or unnecessarily consuming resources.
- **Reporting:** Cost information can be reviewed through reports and dashboards to support financial planning and cloud governance.

Example

Suppose an organization has deployed several virtual machines, databases, and storage resources. At the end of a month, its Azure bill is higher than expected. Using Cost Analysis, the administrator can determine which services are responsible for the increase. If an unused virtual machine is identified as a major source of unnecessary expenditure, it can be stopped, resized, or removed.

Thus, Azure Cost Management and Billing helps organizations understand their actual spending and maintain control over cloud expenditure.

2. Azure Pricing Calculator

The Azure Pricing Calculator is a planning and estimation tool that helps users determine the approximate cost of Azure services before deploying them.

Users can select the services they intend to use and specify configuration and usage parameters. The calculator then provides an estimated cost based on the selected options.

Major Features

- **Service Selection:** Users can select services such as virtual machines, storage, databases, networking, and other Azure resources.
- **Configuration-Based Estimation:** Users can specify factors such as region, instance type, operating system, storage capacity, and expected usage.
- **Usage Estimation:** The expected amount of resource consumption can be entered to obtain a more realistic cost estimate.
- **Pricing Comparison:** Different configurations can be evaluated to determine which option may provide better cost efficiency.
- **Planning Support:** Organizations can estimate their expected cloud expenditure before migrating applications or deploying new workloads.

Example

Consider an organization planning to host a web application on Azure. Before deployment, the administrator can use the Azure Pricing Calculator to estimate the cost of:

1. Virtual machines or application hosting,

2. Storage,

SGOU

3. Database services,

4. Network traffic, and

5. Other required Azure services.

The administrator can then compare different configurations and select an appropriate solution based on the organization's performance requirements and budget.

Therefore, the Azure Pricing Calculator is primarily useful for planning and estimating future Azure costs rather than monitoring actual expenditure.

3. Azure Advisor

Azure Advisor is a recommendation service that analyzes Azure resources and usage patterns and provides personalized recommendations for improving the efficiency of cloud resources.

It evaluates resources against established best practices and can identify opportunities to improve cost efficiency, performance, reliability, security, and operational effectiveness.

Major Areas of Recommendations

- **Cost Optimization:** Identifies opportunities to reduce unnecessary expenditure, such as underutilized resources or resources that could potentially be resized.
- **Performance:** Provides recommendations for improving the performance of applications and Azure resources.
- **Reliability:** Identifies potential improvements that can increase application availability and resilience.
- **Security:** Highlights recommendations that can help strengthen the security of Azure resources.
- **Operational Excellence:** Provides suggestions for improving the management and operational efficiency of cloud resources.

Example

Suppose an organization has a virtual machine that consistently uses only a small portion of its allocated computing capacity. Azure Advisor may identify the resource as a potential candidate for rightsizing. The organization can evaluate the recommendation and, if appropriate, move to a smaller and less expensive configuration.

Similarly, Advisor may identify other resources that are not being used efficiently and provide recommendations that can help reduce costs.

SGOU

Table 2.4.2 Comparison of Azure Cost Management Tools

Tool	Primary Purpose	When It Is Used	Example
Azure Cost Management and Billing	Monitor and analyze actual Azure expenditure	During and after resource usage	Identify which resource group is responsible for increased spending
Azure Pricing Calculator	Estimate the expected cost of Azure services	Before deployment or during planning	Estimate the monthly cost of hosting an application
Azure Advisor	Provide recommendations for optimizing resources	During ongoing Azure operations	Recommend resizing an underutilized virtual machine

As an example, these tools can be used as follows: An organization planning a new application can first use the Azure Pricing Calculator to estimate the expected monthly cost. After deploying the application, Azure Cost Management and Billing can be used to monitor actual expenditure and compare it with the planned budget. Azure Advisor can then provide recommendations for improving resource utilization and reducing unnecessary costs.

Therefore, these tools help organizations follow a continuous plan–monitor–optimize approach to cloud cost management. They support better financial planning, efficient resource utilization, and improved control over Azure expenditure.

2.4.3 Google Cloud Platform (GCP)

Google Cloud Platform (GCP) is a comprehensive cloud computing platform provided by Google. It offers a wide range of infrastructure, platform, and managed services that enable individuals and organizations to develop, deploy, operate, and manage applications and digital services using Google's global cloud infrastructure.

- **Compute and Storage:** GCP provides a variety of computing services for running applications and workloads, along with scalable storage solutions for managing structured and unstructured data. It also offers virtual machine and container-based computing options.
- **Data Analytics and Artificial Intelligence:** GCP provides powerful data processing and analytics services such as BigQuery, along with AI and machine learning capabilities through platforms such as Vertex AI. These services enable organizations to process large datasets, discover insights, and develop and deploy machine learning models.
- **Scalability and Security:** GCP uses Google's globally distributed infrastructure to provide scalable computing and networking resources. It also incorporates security mechanisms such as identity and access management, encryption, monitoring, and threat protection to help safeguard applications and data.

GCP enables organizations to select and combine cloud services according to their

SGOU

computing, storage, networking, data analytics, artificial intelligence, and application development requirements.

2.4.3.1 Architecture of Google Cloud Platform

The architecture of Google Cloud Platform (GCP) consists of several logical layers that work together to provide cloud services in an efficient and organized manner. The architecture separates major functions such as user interaction, computing, storage, databases, networking, monitoring, and security, enabling reliable and secure operation of cloud resources.

At the top level, users and administrators access GCP through different management interfaces, including:

- **Web-based Google Cloud Console**
- **Command-Line Interfaces (CLI)**
- **Application Programming Interfaces (APIs)**

This access layer allows users to create and manage resources, configure cloud services, monitor system performance, and control various aspects of their cloud environment.

The Compute Layer is responsible for running applications and processing workloads. It includes:

- **Compute Engine**, which provides virtual machines that function similarly to traditional physical servers.
- **Google Kubernetes Engine (GKE)**, which manages containerized applications and supports automatic scaling according to workload requirements.

The Storage Layer provides facilities for storing different types of data. It includes:

- **Cloud Storage**, which is used to store large volumes of unstructured data such as images, videos, documents, and backups.
- **Persistent Disk**, which provides storage volumes that can be attached to virtual machine instances.

The Database Layer provides services for managing structured and unstructured data. Important services include:

- **BigQuery**, a cloud-based data warehouse designed for large-scale data analytics.
- **Cloud SQL**, which supports relational database systems such as MySQL, PostgreSQL, and SQL Server.
- **Firestore**, a NoSQL database primarily used for web and mobile applications requiring real-time data synchronization.

SGOU

The Networking Layer enables communication between cloud resources and manages network traffic. It includes:

- **Virtual Private Cloud (VPC)**, which allows users to create and manage private networks within the cloud environment.
- **Load Balancing**, which distributes incoming traffic across multiple servers or resources to prevent overloading.
- **Content Delivery Network (CDN)**, which improves content delivery by caching data at locations closer to users around the world.

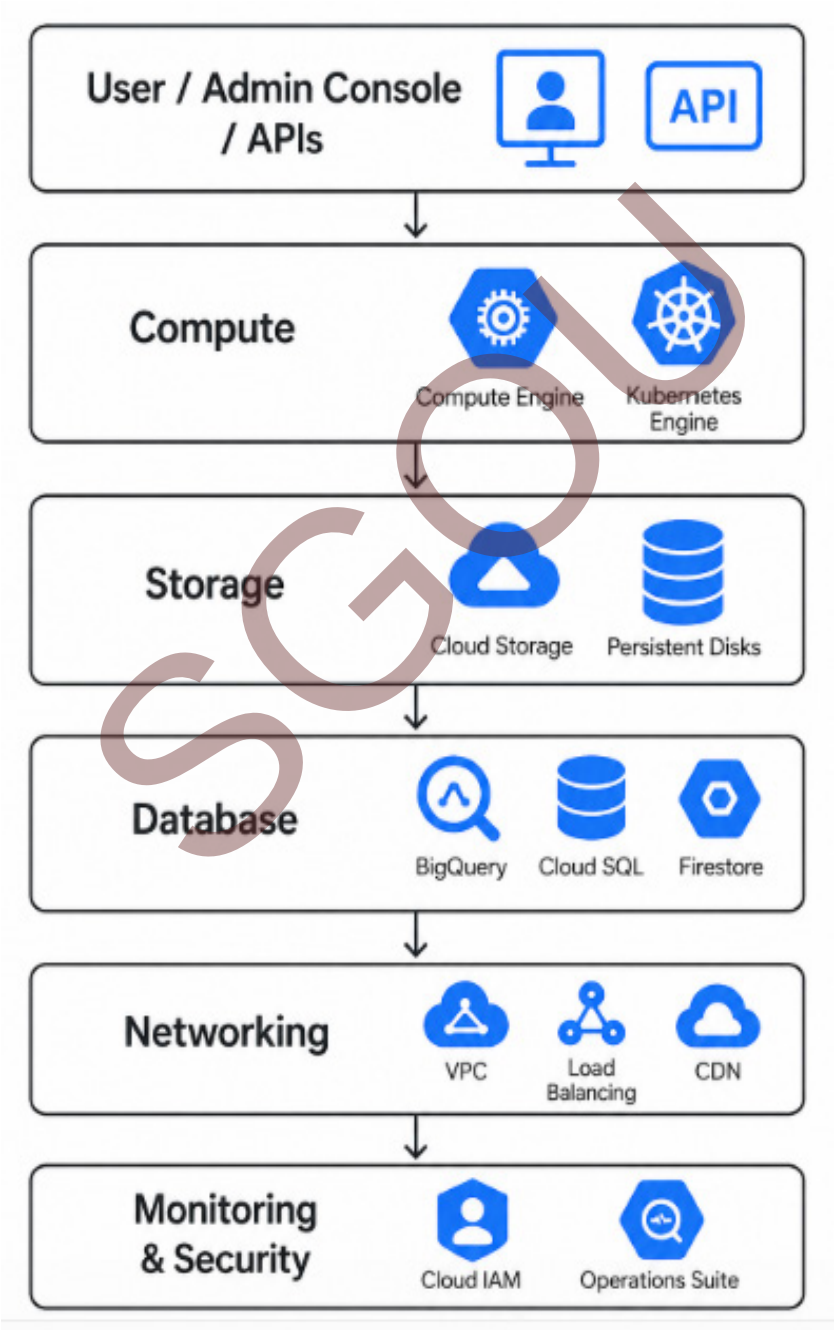


Fig. 2.4.3 Google Cloud Platform (GCP) architecture

SGOU

Finally, the Monitoring and Security Layer helps protect cloud resources and ensures that they operate effectively. It includes:

- **Cloud IAM (Identity and Access Management)**, which manages user identities and controls access to specific cloud resources.
- **Google Cloud Operations Suite**, which provides monitoring, logging, performance analysis, and alerting capabilities for maintaining system health.

Together, these interconnected layers provide a secure, reliable, scalable, and efficient cloud architecture capable of supporting a wide range of modern computing, storage, networking, and data-processing requirements.

2.4.3.2 Key Features of Google Cloud Platform

1. AI and Machine Learning Services

Google Cloud Platform provides advanced Artificial Intelligence (AI) and Machine Learning (ML) tools such as Vertex AI, AutoML, and TensorFlow. These services enable developers to build, train, and deploy machine learning models efficiently. They can be applied in areas such as image recognition, speech processing, recommendation systems, and predictive analytics.

2. Powerful Data Analytics

Google BigQuery is a powerful cloud-based data analytics and data warehousing service capable of processing very large datasets rapidly without requiring users to manage the underlying servers. It is particularly useful for organizations handling large volumes of data, including those in finance, healthcare, and e-commerce.

3. Strong Open-Source Support

GCP provides extensive support for open-source technologies such as Kubernetes, TensorFlow, and Apache Spark. This allows developers to use widely adopted tools and frameworks that they are already familiar with, while reducing dependence on proprietary technologies.

4. High Sustainability Standards

Google emphasizes energy efficiency and environmental sustainability in the operation of its data centers. The company uses renewable energy sources for many of its operations and has implemented measures to improve energy efficiency, making sustainability an important aspect of GCP's cloud infrastructure.

5. Transparent and Flexible Pricing

GCP follows a usage-based pricing model that allows customers to pay for the resources they consume. It also provides pricing benefits such as sustained-use discounts, which can automatically reduce costs for workloads that operate for extended periods, without requiring long-term commitments.

SGOU

2.4.3.3 Advantages of Google Cloud Platform

- **Advanced AI and Machine Learning:** GCP offers powerful AI and machine learning services that are widely used for developing and deploying intelligent applications.
- **High Performance and Low Latency:** GCP operates on Google's advanced global network infrastructure, providing fast data transfer, reliable performance, and low network latency.
- **Strong Open-Source Support:** Its extensive support for open-source technologies provides developers with greater flexibility and allows them to work with widely used tools and frameworks.
- **Cost-Effectiveness:** GCP can be a cost-effective option for startups, educational institutions, and academic projects, particularly for workloads with limited or variable resource requirements.
- **User-Friendly Interface:** GCP provides intuitive management tools and interfaces that simplify cloud resource management, making the platform easier for beginners to understand and use.

2.4.3.4 Disadvantages of Google Cloud Platform

Although Google Cloud Platform (GCP) offers a wide range of cloud services, it has certain limitations.

- **Smaller Market Share:** GCP has a smaller market presence compared with major cloud providers such as AWS and Microsoft Azure.
- **Limited Enterprise Adoption:** Some large organizations have comparatively lower adoption of GCP because AWS and Azure have had a longer presence in the enterprise cloud market.
- **Regional Service Availability:** Certain GCP services may have limited availability in particular geographical regions, which can restrict deployment options for some users.
- **Fewer Third-Party Integrations:** Compared with the extensive ecosystem of AWS, GCP may offer fewer third-party tools, integrations, and supporting solutions.

Table 2.4.3 Comparative Analysis: AWS vs. Microsoft Azure vs. Google Cloud

Feature	Amazon Web Services (AWS)	Microsoft Azure	Google Cloud
Initial Cloud Platform Launch	2006	2010	2011
Parent Company	Amazon	Microsoft	Google
Service/Product Portfolio	240+ services	200+ products and services	150+ products

SGOU

Compute Service	Amazon EC2	Azure Virtual Machines	Compute Engine
Storage Service	Amazon S3, Amazon EBS	Azure Blob Storage	Cloud Storage
Database Service	Amazon RDS, Amazon DynamoDB	Azure SQL Database, Azure Cosmos DB	BigQuery, Cloud SQL
AI and ML	Amazon SageMaker	Azure AI / Azure AI Foundry	Vertex AI
Global Reach	39 geographic Regions	70+ Regions	43 Regions
Hybrid Cloud Support	Strong	Strong	Strong
Open-Source Focus	High	High	High
Best Suited For	Broad enterprise workloads, scalability and diverse cloud services	Microsoft-centric environments, enterprise workloads and hybrid cloud	Data analytics, AI/ ML and cloud-native workloads

Note: The information presented in the table is based on data collected as of August 2026.

Recap

- Cloud computing provides on-demand access to computing resources over the Internet.
- AWS, Microsoft Azure, and Google Cloud are major cloud computing platforms.
- AWS was introduced in 2006 and provides a broad range of cloud services.
- AWS reduces the need for organizations to purchase and maintain physical data centers.
- AWS supports scalability, flexible deployment, global reach, and high availability.
- AWS infrastructure is organized into Regions and Availability Zones (AZs).
- The AWS Shared Responsibility Model divides security responsibilities between AWS and its customers.
- Amazon EC2 provides resizable virtual computing resources for running applications.
- Amazon S3 provides scalable object storage for files, documents, backups, and other data.

SGOU

- Amazon RDS provides managed relational database services, while DynamoDB supports NoSQL workloads.
- Amazon VPC, CloudFront, and Route 53 provide important networking and content-delivery capabilities.
- AWS offers different pricing approaches, including Pay-as-You-Go, On-Demand, Reserved, and Spot pricing.
- Microsoft Azure provides cloud services for computing, storage, networking, databases, analytics, AI, and IoT.
- Azure uses Regions, Availability Zones, Resource Groups, and Azure Resource Manager to organize and manage cloud resources.
- Google Cloud provides strong capabilities in data analytics, artificial intelligence, machine learning, scalability, and open-source technologies.

Objective Type Questions

1. In which year was Amazon Web Services (AWS) introduced?
2. Which AWS service provides resizable virtual computing resources?
3. Which AWS service is used for object storage?
4. Which AWS service provides serverless code execution?
5. Which AWS service provides managed relational database capabilities?
6. Which AWS service is a managed NoSQL database?
7. What does VPC stand for in AWS?
8. Which AWS service functions as a Content Delivery Network (CDN)?
9. What is the AWS security model that divides security responsibilities between AWS and its customers called?
10. Which Azure service provides virtual machines?
11. Which Azure service is used for serverless event-driven code execution?
12. What is the logical container used to organize related Azure resources called?
13. Which Azure tool is used to estimate the expected cost of Azure services before deployment?
14. Which Google Cloud service is designed for large-scale data analytics and data warehousing?
15. Which Google Cloud service provides AI and machine learning capabilities for building and deploying models?

SGOU

Answers to Objective Type Questions

1. 2006
2. Amazon EC2
3. Amazon S3
4. AWS Lambda
5. Amazon RDS
6. Amazon DynamoDB
7. Virtual Private Cloud
8. Amazon CloudFront
9. Shared Responsibility Model
10. Azure Virtual Machines
11. Azure Functions
12. Resource Group
13. Azure Pricing Calculator
14. BigQuery
15. Vertex AI

Assignments

1. Explain the concept of cloud computing service models and discuss the major cloud platforms, AWS, Microsoft Azure, and Google Cloud.
2. Describe the architecture of Amazon Web Services (AWS), including Regions, Availability Zones, and major methods of accessing and managing AWS services.
3. Explain the AWS Shared Responsibility Model and distinguish between AWS responsibilities and customer responsibilities with suitable examples.
4. Discuss the core AWS services for computing, storage, databases, and networking, with suitable examples of their applications.
5. Explain the major AWS pricing models and cost-management tools, including Pay-as-You-Go, On-Demand, Reserved pricing, Spot Instances, Free Tier, and AWS Pricing Calculator.

SGOU

6. Describe the architecture of Microsoft Azure and explain its major components, including Regions, Availability Zones, Resource Groups, compute, storage, database, networking, and security services.
7. Explain the key features, advantages, and disadvantages of Microsoft Azure, with suitable examples.
8. Describe the architecture, key features, advantages, and disadvantages of Google Cloud, and compare it with AWS and Microsoft Azure.

Reference

1. Author/Entity Name. (2022). Introduction to Cloud Service Models: IaaS, PaaS, and SaaS. General Cloud Computing Textbook / Educational Resource.
2. Amazon Web Services (AWS). (2024). AWS Global Infrastructure and Key Services. AWS Documentation/Whitepaper.
3. Microsoft Azure. (2023). Microsoft Azure Architecture and Hybrid Cloud Capabilities. Microsoft Azure Documentation/Whitepaper.
4. Google Cloud Platform (GCP). (2023). GCP's Strength in AI, Data Analytics, and Open-Source Technologies. Google Cloud Documentation / Blog.

Suggested Reading

1. Chou, E. (2023). Cloud Computing: Concepts, Technology & Architecture. Pearson Education. (For a deeper understanding of the underlying technology and architectural patterns).
2. Microsoft. (2024). Azure Well-Architected Framework: Reliability and Operational Excellence. Microsoft Documentation. (To explore how Azure implements fault tolerance and high availability in detail).
3. Gundersen, G. (2022). The Kubernetes Book. O'Reilly Media. (To understand container orchestration and its relevance to Google Cloud Platform's strategy).
4. Amazon Web Services. (2024). AWS Security Best Practices: Identity and Access Management (IAM). AWS Documentation. (To gain expertise in the security features highlighted as key to AWS).

SGOU



Managing Cloud Environment

SGOU

Unit 1
Cloud Administration Portals and User Management

Learning Outcomes

Upon completion of the unit, learner will be able to:

- identify the features of cloud administration portals
- explain the process of cloud account creation and management
- differentiate user roles and permissions in cloud environments
- manage cloud subscriptions using administration portals
- explain the importance of user management in cloud administration

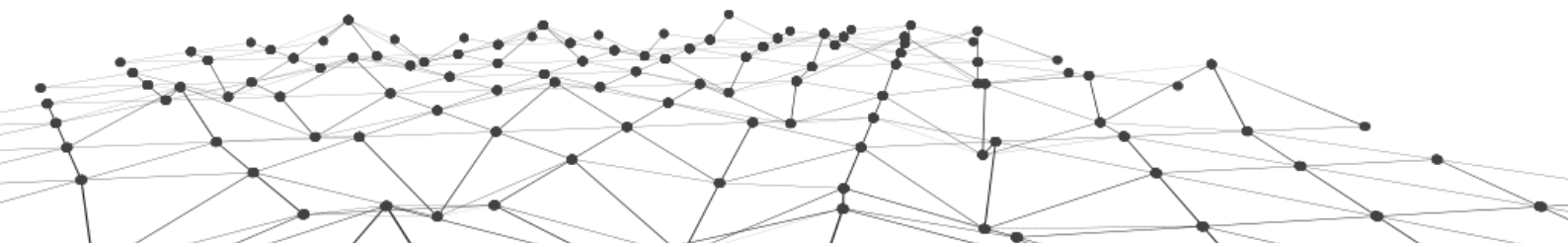
Prerequisite

Every day, millions of people send emails, attend online classes, store photographs, make digital payments, and collaborate with colleagues using cloud based applications. These services are expected to work continuously, providing secure and uninterrupted access to users around the world. Most users simply sign in and start working without thinking about the complex activities taking place behind the scenes.

For every new employee joining an organization, a user account must be created. When a student enrolls in a university, access to learning platforms and digital resources must be provided. If an employee leaves the organization, the account should be removed to prevent unauthorized access. At the same time, cloud resources must be monitored, security policies must be enforced, and operational costs must be controlled. As organizations continue to grow, performing these activities manually becomes increasingly difficult and may lead to security risks, service interruptions, and unnecessary expenses.

This is why cloud administration has become an essential part of modern cloud computing. It provides the methods and tools required to manage users, resources, security, subscriptions, and system performance in an organized and efficient manner. Understanding cloud administration enables learners to appreciate how cloud environments are managed to ensure reliability, security, and cost effectiveness.

SGOU



In this section, you will explore the fundamentals of cloud administration, understand its objectives and components, examine cloud administration portals, and learn how user accounts, roles, permissions, and subscriptions are managed in a cloud environment.

Keywords

Identity Management, Access Management, Authentication, Authorization, User Roles, Permissions, Account Verification, Subscription Management

Discussion

3.1.1 Overview of Cloud Administration

Almost every digital service used today relies on cloud computing. Online banking, email services, video conferencing, learning management systems, e-commerce websites, and social media platforms all use cloud infrastructure to deliver services over the Internet.

Although these services appear simple to users, several administrative tasks take place in the background. New users must be registered, existing users must be granted appropriate access, resources must be monitored, and cloud services must remain secure and available. These responsibilities are performed through cloud administration.

This unit introduces the basic concepts of cloud administration and explains how cloud administration portals help organizations manage users, resources, and subscriptions efficiently.

Suppose a college decides to use cloud services for managing admissions, online classes, examinations, and digital certificates. Thousands of students and staff members will access these services every day.

Now consider a few situations.

- A newly admitted student needs access to the Learning Management System (LMS).
- A faculty member joins the institution and requires access to course materials.
- An employee leaves the institution, and the account must be removed.
- The institution wants to monitor the monthly cloud expenses.

Who performs these activities?

These activities are part of cloud administration.

Cloud administration is the process of managing cloud resources, user accounts, security settings, subscriptions, and services using cloud administration tools. It ensures that the right users have the right level of access to the required cloud resources while maintaining security and efficient resource utilization.

SGOU

In simple terms, cloud administration is similar to managing a large digital workplace where users, applications, and resources are continuously added, updated, monitored, and protected.

Example 1

A software company has employees working in different departments.

Table 3.1.1 Departments and Cloud Requirements

Department	Cloud Requirement
Human Resources	Employee records
Finance	Billing and accounting applications
Development	Virtual machines and databases
Marketing	File sharing and collaboration tools
Management	Reports and dashboards

Each department uses different cloud resources. Cloud administration ensures that every employee can access only the resources required for their work.

As organizations grow, the number of cloud users and resources also increases. Managing these resources manually becomes difficult and may lead to security risks, operational delays, and increased costs.

Cloud administration provides a structured approach to managing cloud environments.

Some important reasons for cloud administration are:

Secure Access

Only authorized users should access cloud resources. Authentication and authorization mechanisms help prevent unauthorized access.

Efficient Resource Management

Cloud resources such as virtual machines, storage, databases, and applications should be allocated according to organizational requirements.

Cost Control

Most cloud providers follow a pay-as-you-use pricing model. Continuous monitoring helps organizations avoid paying for unused resources.

Centralized Management

Cloud administration portals allow organizations to manage users, services, and subscriptions from a single interface.

Improved Reliability

Regular monitoring helps identify system issues before they affect users.

SGOU

3.1.1.1 Objectives of Cloud Administration

Now that you have understood why cloud administration is important, the next question is: What exactly does cloud administration try to achieve?

Whenever an organization uses cloud services, it expects the system to be secure, organized, cost-effective, and easy to manage. Cloud administration works toward these goals through a set of well-defined objectives.

The first objective is managing users effectively. Every user should have a separate account, and the account should remain active only for the required period. Creating accounts for new users, updating user information, resetting passwords, and removing inactive accounts are all part of this objective.

The second objective is protecting cloud resources. Cloud environments often store sensitive information such as student records, financial data, customer details, or business documents. Proper administration ensures that only authorized users can access these resources.

Another important objective is efficient resource management. Cloud platforms provide storage, virtual machines, databases, and networking services. These resources should be allocated according to the actual needs of the organization so that they are neither underutilized nor wasted.

Cloud administration also focuses on monitoring system performance. Continuous monitoring helps identify problems such as excessive resource usage, service failures, or unusual login activities before they affect users.

Finally, cloud administration aims at controlling cloud costs. Since cloud services are usually billed based on usage, monitoring subscriptions and resource consumption helps organizations avoid unnecessary expenses.

These objectives work together to create a cloud environment that is secure, reliable, and economically efficient.

3.1.1.2 Components of Cloud Administration

To achieve the objectives discussed above, cloud administration is divided into several functional components. Each component handles a specific responsibility, and together they ensure the smooth operation of a cloud environment.

The most visible component is user management. This component deals with creating user accounts, updating user details, resetting passwords, and removing accounts that are no longer required. It forms the foundation of cloud administration because every cloud activity begins with a user identity.

Closely related to user management is identity management. Before a user can access any cloud resource, the system must verify who the user is. Identity management handles authentication through methods such as usernames, passwords, multi-factor authentication, or security tokens.

SGOU

Once the identity is verified, the next component, access management comes into action. It determines what the user is allowed to do. For example, a student may only view course materials, while a faculty member can upload assignments and modify course content.

Cloud administration also includes resource management. This component is responsible for creating, configuring, monitoring, and deleting cloud resources such as storage accounts, databases, virtual machines, and applications. Efficient resource management helps organizations maintain good performance while avoiding unnecessary costs.

Another important component is subscription management. Organizations often subscribe to different cloud plans and services. Subscription management helps track billing information, monitor resource usage, and ensure that cloud services remain within the allocated budget.

To keep the cloud environment healthy, monitoring is continuously performed. Monitoring provides information about system performance, resource utilization, availability, and operational health. It helps administrators detect issues early and take corrective action.

The final component is security management. Security policies, encryption, auditing, and access controls are implemented to protect cloud resources from unauthorized access and cyber threats.

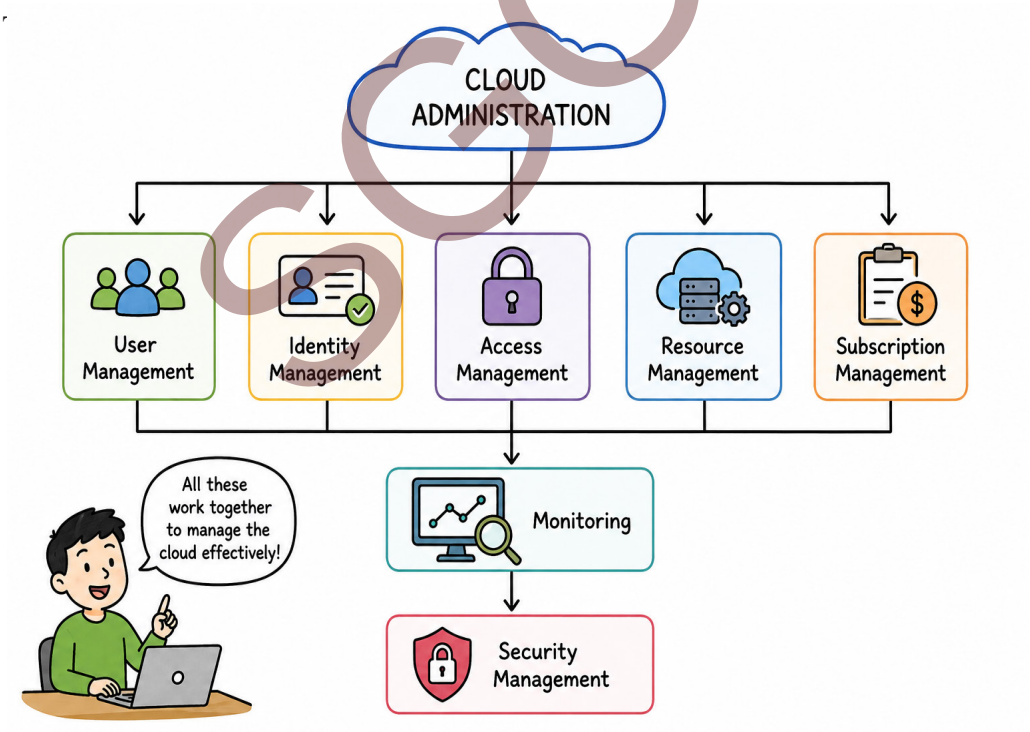


Figure 3.1.1 Major components of cloud administration.

SGOU

Notice how these components are interconnected. User management provides the identities, identity management verifies them, access management grants permissions, resource management allocates cloud resources, subscription management controls costs, monitoring tracks system health, and security management protects the entire environment. This integrated approach makes cloud administration both efficient and secure.

3.1.1.3 Cloud Administration Portals

A university decides to migrate its digital services to the cloud. The Learning Management System (LMS), student information system, institutional email, digital library, and research repositories will all be hosted on a cloud platform. Since these services will be used by thousands of students, faculty members, researchers, and administrative staff, the university requires an efficient mechanism to manage its cloud environment.

The IT team must create user accounts, monitor cloud resources, assign permissions, review system health, and track cloud expenditure. Performing these tasks separately for each service would be both time-consuming and inefficient. To simplify cloud management, cloud service providers offer a centralized management interface known as a cloud administration portal.

A cloud administration portal is a web-based interface that enables administrators to manage cloud resources and services from a single location. It serves as the primary point of interaction between the user and the cloud platform. Through the portal, administrators can create resources, manage users, monitor services, configure security settings, and review billing information.

Although each cloud provider designs its portal differently, the basic purpose remains the same to simplify cloud administration through a centralized and user-friendly interface.

3.1.1.4 Exploring a Cloud Administration Portal

After signing in successfully, the administrator is presented with the **dashboard**, which acts as the starting point for managing the cloud environment. Similar to the home page of a website, the dashboard provides quick access to important information and frequently used services.

Instead of displaying every available cloud service, the dashboard focuses on information that requires immediate attention. It may present active resources, recent activities, system notifications, usage statistics, billing summaries, or alerts related to cloud services.

For the university, the dashboard may indicate that new user accounts have been created, storage utilization has increased, or a scheduled backup has completed successfully. Such information helps administrators understand the current state of the cloud environment without navigating through multiple pages.

SGOU

Navigating through the Portal

A cloud platform may offer hundreds of services. Locating the required service would be difficult if every option appeared on a single page. To make navigation easier, cloud administration portals organize services into logical categories.

These categories generally include Compute, Storage, Networking, Databases, Identity Management, Security, Monitoring, and Billing.

Suppose the university plans to create additional storage for its digital library. Rather than searching through numerous pages, the administrator simply sWelects the Storage category from the navigation menu. Similarly, selecting Identity Management opens the tools required for creating users and assigning permissions.

Although the names of categories may vary slightly across cloud providers, the overall navigation approach remains similar.

Searching for Cloud Services

As cloud platforms continue to evolve, new services are introduced regularly. Remembering the location of every service is neither practical nor necessary. To simplify navigation, cloud administration portals provide a search facility.

The search bar allows administrators to locate a service by entering its name or a related keyword. For example, entering *Virtual Machine*, *Storage*, or *Database* immediately displays the corresponding service.

This feature becomes particularly useful in large cloud environments where numerous services are available. Instead of spending time browsing through menus, administrators can access the required service within seconds.

3.1.1.5 Common features of Cloud Administration Portals

Although AWS, Microsoft Azure, and Google Cloud Platform have different user interfaces, they provide a similar set of administrative features. Once an administrator becomes familiar with one portal, learning another becomes much easier because many of the concepts remain the same.

The following sections describe the most used features available in cloud administration portals.

Dashboard

The dashboard provides an overview of the cloud environment. It displays important information such as resource status, notifications, recent activities, and billing summaries. Since the dashboard presents frequently accessed information in one place, administrators can quickly assess the health of the cloud environment.

SGOU

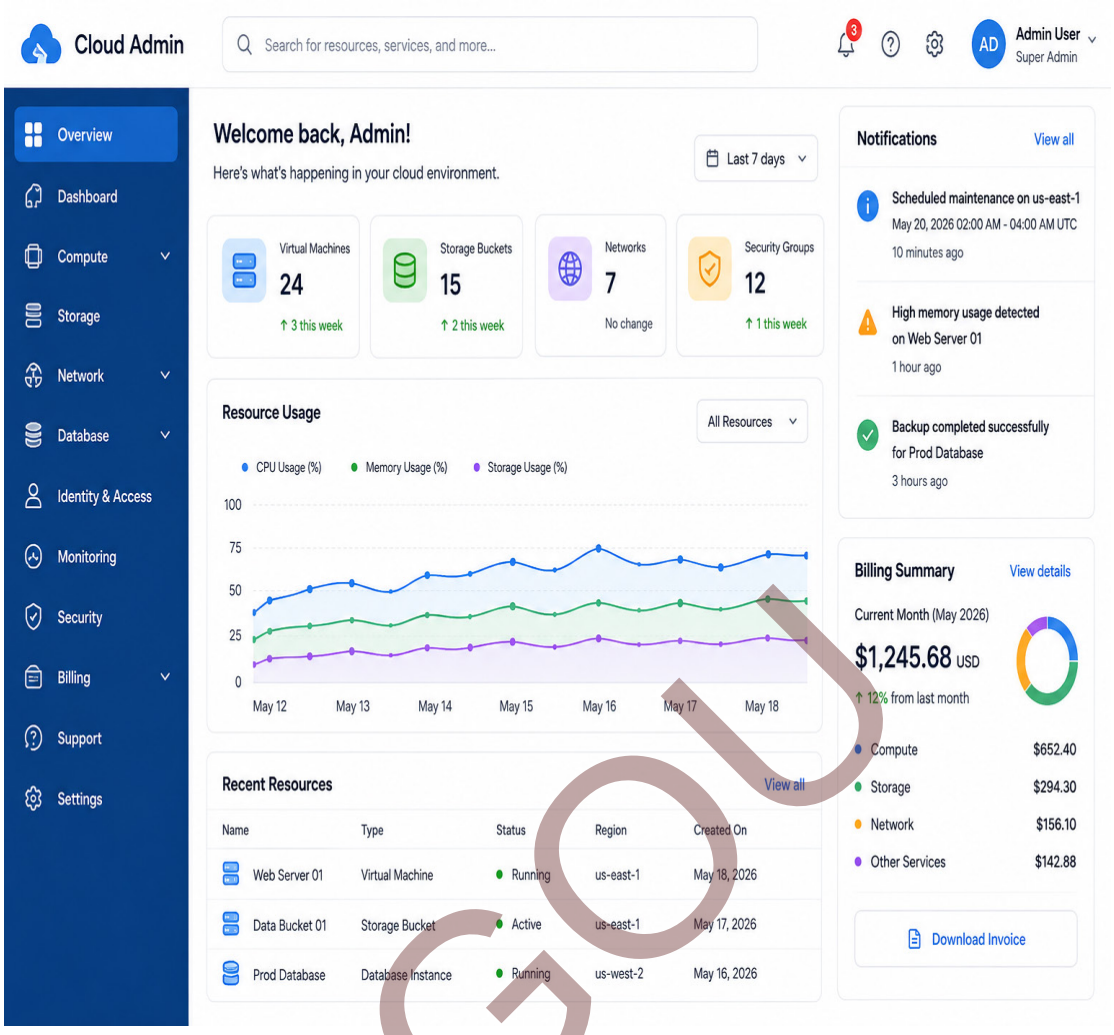


Fig.3.1.2 Cloud Administration Dashboard

Resource Management

One of the primary purposes of a cloud administration portal is to create and manage cloud resources. Depending on organizational requirements, administrators can provision virtual machines, storage services, databases, networking components, and other cloud resources. Existing resources can also be modified or removed whenever necessary.

Monitoring

Cloud resources operate continuously and require regular observation. Monitoring tools collect information about system performance, CPU utilization, memory usage, storage capacity, and network activity. If an abnormal condition occurs, alerts are generated so that corrective action can be taken promptly.

Billing and Subscription

Cloud providers generally charge based on resource usage. The administration portal therefore includes billing tools that display current expenditure, estimated costs,

SGOU

invoices, and subscription details. These tools help organizations monitor spending and make informed decisions about resource utilization.

Identity and Access Management

Managing users is one of the most important administrative responsibilities. Cloud administration portals include dedicated services for creating user accounts, assigning roles, and controlling access to cloud resources. Since user management is a major topic in this unit, it will be discussed in detail in the upcoming sections.

3.1.2 Account Creation

A cloud administration portal provides access to various management functions. However, before any cloud resource can be accessed or managed, users must have a valid account. Whether it is a student accessing an online learning platform, a faculty member uploading course materials, or an administrator managing cloud resources, each person requires a unique identity within the cloud environment.

Account creation is therefore one of the first and most important tasks performed through a cloud administration portal. It enables organizations to identify users, provide secure access, and manage cloud resources in a systematic manner.

Consider a university that has recently adopted a cloud platform for its academic and administrative services. At the beginning of every academic year, new students join various programmes, faculty members are appointed, and administrative staff take up different responsibilities. Before these users can access institutional email, cloud storage, the Learning Management System (LMS), or other cloud-based applications, individual cloud accounts must be created for each of them.

Creating separate accounts for every user offers several advantages. It helps the institution identify who is accessing cloud resources, assign appropriate permissions, maintain security, and keep records of user activities. Sharing a common account among multiple users makes it difficult to monitor activities and increases the risk of unauthorized access.

3.1.2.1 What is a Cloud Account?

A cloud account is a digital identity that allows a user to access services provided by a cloud platform. Similar to an email account or an online banking account, a cloud account uniquely identifies a user and stores information required for authentication and authorization.

A cloud account generally contains the following information:

- User name
- Email address
- Password

SGOU

- Contact information
- Assigned role
- Account status

Once the account is created, the user can sign in to the cloud administration portal or other cloud services using the assigned credentials.

3.1.2.2 Steps in Creating a Cloud Account

After collecting the necessary user information, the next step is to create the cloud account. Although the appearance of cloud administration portals varies from one cloud provider to another, the basic account creation process is quite similar. Understanding these steps helps administrators create user accounts efficiently while ensuring that only authorized users gain access to cloud resources.

The following sequence describes the typical account creation process followed in most cloud platforms.

Step 1: Sign in to the Cloud Administration Portal

The process begins by signing in to the cloud administration portal using an administrator account. Only users with appropriate administrative privileges can create new user accounts.

After successful authentication, the administrator is presented with the portal dashboard, from where different management functions can be accessed.

Step 2: Open the User Management Section

The administrator then navigates to the **User Management** or **Identity Management** section of the portal. This section contains options for creating new users, updating user information, resetting passwords, and managing existing accounts.

The exact name of this section may differ depending on the cloud platform, but its purpose remains the same.

Step 3: Create a New User Account

The administrator selects the option to add a new user and enters the required information, such as the user's name, email address, username, and other organizational details.

Some cloud platforms may also request additional information, such as the department, employee ID, or student registration number.

Step 4: Assign Login Credentials

The new user requires credentials to access the cloud environment. During account creation, the administrator assigns a username and either creates a temporary password or allows the system to generate one automatically.

SGOU

For security reasons, many organizations require users to change the temporary password during their first login.

Step 5: Save the Account

After verifying the entered information, the administrator saves the account. The cloud platform creates a unique digital identity for the user.

The account is now registered in the organization's cloud environment but may require additional configuration before it can be used.

Step 6: Notify the User

The final step is to inform the user that the account has been created. Login credentials or an activation link are usually sent to the registered email address. The user can then activate the account and sign in to the cloud services.

3.1.2.3 Account Verification

Creating a cloud account is only the first step. Before the account can be used, the cloud platform must verify that the information provided during registration is valid. This process is known as account verification.

Account verification helps confirm the identity of the user and ensures that only authorized individuals can access cloud resources. It also reduces the possibility of fake or duplicate accounts being created within the cloud environment.

The verification process varies from one cloud platform to another. However, most cloud service providers follow similar methods to validate newly created accounts.

Email Verification

One of the most common verification methods is email verification. After a new account is created, the cloud platform sends a verification message to the email address provided during registration.

The user opens the email and selects the verification link. Once the link is verified, the cloud platform confirms that the email address belongs to the registered user and activates the account.

Email verification also enables the cloud platform to send future notifications such as password reset links, security alerts, and account updates.

Mobile Number Verification

Some cloud platforms request a mobile number during account creation. A One-Time Password (OTP) is sent to the registered mobile number.

The user enters the OTP in the verification page. If the entered OTP matches the generated code, the mobile number is verified successfully.

Mobile number verification provides an additional level of confidence that the account belongs to the intended user.

SGOU

Identity Verification

In some organizations, particularly those handling sensitive information, verifying an email address alone may not be sufficient.

For example, before creating cloud accounts for faculty members and administrative staff, a university may verify employee records maintained by the Human Resources department. Similarly, student accounts may be created only after admission has been confirmed.

This additional verification ensures that cloud accounts are issued only to eligible users.

Administrator Approval

Certain cloud environments require administrator approval before a newly created account becomes active.

After the user submits the registration details, the request is reviewed by an administrator. The administrator verifies the information, assigns the appropriate user role, and activates the account.

This approach is commonly used in educational institutions and business organizations where cloud resources contain confidential information.

3.1.3 User Roles and Permissions

After a cloud account has been created and verified, the user is ready to access cloud services. However, not every user should have the same level of access. A student should not be able to modify examination records, and a faculty member should not have permission to manage the university's cloud subscription. Similarly, administrative staff should not be allowed to alter cloud security settings unless such responsibilities are part of their job.

This is where user roles and permissions become important. They help organizations control who can access cloud resources and what actions each user is allowed to perform.

A user role is a collection of responsibilities assigned to a user within a cloud environment. Instead of assigning permissions individually to every user, users performing similar responsibilities are grouped under the same role.

For example, a university may have students, faculty members, departmental administrators, and cloud administrators. Although thousands of students use the cloud platform, they generally require the same type of access. Therefore, all students can be assigned the Student role.

Similarly, faculty members can be assigned the Faculty role, while cloud administrators receive the Administrator role.

Assigning roles simplifies user management because permissions are associated with the role rather than with individual users.

SGOU

For example, A university has 3,000 students enrolled in different programmes. Instead of configuring permissions separately for each student, the IT team assigns all students to a common **Student** role.

Whenever a new student joins the university, the administrator simply assigns the Student role to the newly created account. The required permissions are applied automatically.

3.1.3.1 Understanding Permissions

While a role defines who a user is, a permission defines what the user is allowed to do.

Permissions specify the actions that users can perform on cloud resources. These actions may include viewing information, creating resources, modifying existing resources, or deleting resources.

Some common permissions are:

- View resources
- Create resources
- Edit resources
- Delete resources
- Manage users
- Configure security settings

Permissions ensure that users perform only the tasks necessary for their responsibilities.

A faculty member logs in to the university's cloud portal.

The faculty member can:

1. upload lecture notes
2. create assignments
3. evaluate student submissions

However, the faculty member cannot:

1. create new user accounts
2. modify subscription plans
3. change security policies

These restrictions are determined by the permissions assigned to the Faculty role.

SGOU

3.1.3.2 Relationship between Roles and Permissions

At this point, an important question arises. If every user is assigned a role, why are permissions needed? Isn't assigning a role enough?

To answer this question, let us return to the university introduced earlier.

Suppose a new faculty member joins the Department of Computer Science. After the cloud account is created and verified, the administrator assigns the Faculty role to the new user. Does this role alone allow the faculty member to use cloud resources?

The answer is yes, but only because the Faculty role already contains a set of predefined permissions. These permissions specify the activities that faculty members are allowed to perform, such as uploading course materials, creating assignments, and viewing student submissions.

Now consider another user who is assigned the Student role. Although both users can sign in to the same cloud platform, they experience different levels of access. A student can view learning materials and submit assignments but cannot modify course content or create new user accounts. The difference lies not in the cloud account itself but in the permissions associated with the assigned role.

This illustrates an important principle of cloud administration. A role identifies the responsibilities of a user, while permissions define the actions that the user is allowed to perform. When a role is assigned, the corresponding permissions are automatically made available to that user.

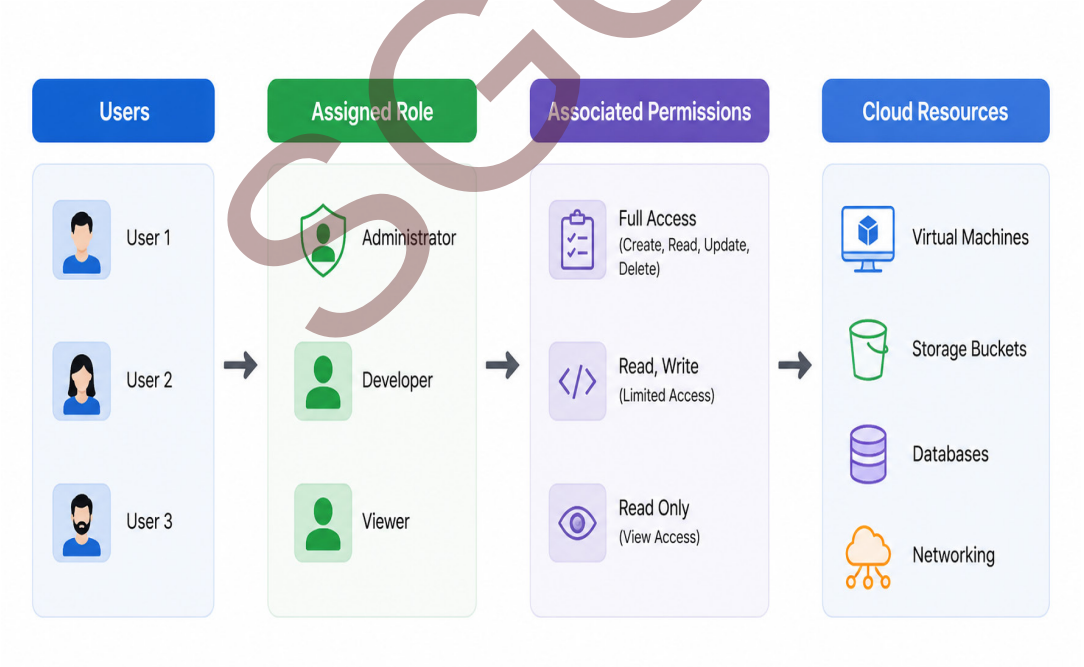


Fig. 3.1.3 Role-Based Access Control in Cloud Computing

SGOU

This approach greatly simplifies administration. Instead of configuring permissions separately for every individual, the administrator assigns an appropriate role. Any user receiving that role automatically inherits the same set of permissions. If the responsibilities of the role change, the administrator updates the permissions once, and the changes apply to all users assigned to that role.

3.1.4 Subscription and Management

Until now, the university has created cloud accounts for its users and assigned appropriate roles and permissions. Students can access learning materials, faculty members can manage academic content, and administrators can perform management tasks through the cloud administration portal.

As the number of users and cloud services increases, another important responsibility emerges. The university must ensure that sufficient cloud resources are available while keeping operational costs under control. This is achieved through subscription management.

A cloud subscription is an agreement between the cloud service provider and the customer that defines how cloud services are accessed, managed, and billed. It specifies the cloud services available to the organization, the usage limits, and the pricing model.

Subscription management involves monitoring resource usage, managing subscriptions, controlling expenditure, and ensuring that cloud resources continue to meet the organization's requirements.

3.1.4.1 Why is a Cloud Subscription required?

Cloud service providers offer a wide range of services such as virtual machines, databases, storage, networking, and artificial intelligence tools. These services are generally not provided without a subscription. Before using them, an organization must select a subscription that matches its requirements.

The type of subscription depends on factors such as the number of users, storage requirements, computing resources, and expected workload.

For a university, the subscription should be capable of supporting students, faculty members, digital learning resources, online assessments, and administrative applications. As these requirements change over time, the subscription may also need to be reviewed and updated.

Example

Initially, the university uses cloud services only for its Learning Management System. After a few months, it decides to host the digital library and institutional email on the same cloud platform. Since these additional services require more storage and computing resources, the university upgrades its cloud subscription to accommodate the increased workload.

SGOU

3.1.4.2 Components of a Cloud Subscription

A cloud subscription includes several elements that determine how cloud services are used and managed.

One important component is the subscription plan. The plan specifies the cloud services available to the organization and the limits associated with those services.

Another component is resource allocation. The subscription determines the amount of computing power, storage, and networking resources that can be used.

The subscription also defines the billing model. Depending on the cloud provider, organizations may pay according to resource usage, choose a fixed monthly plan, or adopt other pricing options.

Many cloud platforms also provide usage reports. These reports help organizations monitor resource consumption and identify opportunities to optimize costs.

3.1.4.3 Managing Cloud Subscriptions

Managing a subscription is an ongoing activity rather than a one-time task. As organizational requirements change, administrators regularly review subscription details to ensure that cloud resources remain sufficient and cost-effective.

One important responsibility is monitoring resource usage. By examining usage reports, administrators can identify whether computing resources, storage, or network services are being fully utilized.

Another responsibility is reviewing subscription costs. Since many cloud platforms follow a pay-as-you-use model, monitoring expenditure helps organizations avoid unnecessary expenses.

Administrators also review subscription details whenever new cloud services are introduced or when existing services are no longer required. This ensures that the subscription continues to match the organization's needs.

Example

At the end of the academic year, the university observes that cloud storage usage has increased significantly because of digital learning materials and recorded online classes. After reviewing the usage reports, the administrator upgrades the storage capacity to ensure uninterrupted service for the next academic year.

3.1.1.4 Good Practices in Subscription Management

Effective subscription management helps organizations obtain the required cloud services while avoiding unnecessary costs.

Some recommended practices include:

- reviewing resource usage regularly
- selecting subscription plans that match organizational requirements

SGOU

- removing unused cloud resources
- monitoring billing reports periodically
- upgrading or modifying subscriptions only when necessary

Following these practices helps organizations use cloud resources efficiently while maintaining financial control.

Recap

Overview of Cloud Administration

- Cloud administration manages users, resources, and services.
- It ensures security, reliability, and cost efficiency.
- Centralized management simplifies cloud operations.

Components of Cloud Administration

- Identity management verifies users.
- Access management controls permissions.
- Resource management manages cloud resources.
- Monitoring tracks system health.
- Security management protects cloud resources.

Cloud Administration Portal

- Provides centralized management of cloud services.
- Dashboard gives an overview of cloud resources.
- Search and navigation enable quick access to services.
- Billing and monitoring support efficient administration.

Account Creation and Verification

- Every user requires a unique cloud account.
- Account verification confirms user identity.
- Email, OTP, and administrator approval improve security.

User Roles and Permissions

- Roles define user responsibilities.
- Permissions define allowed actions.

SGOU

- Users inherit permissions through assigned roles.
- Role-based access simplifies administration.

Subscription Management

- A subscription defines cloud services and billing.
- Monitor resource usage and costs regularly.
- Upgrade subscriptions only when required.
- Remove unused resources to reduce expenses.

Objective Type Questions

1. Which interface provides centralized cloud management?
2. What verifies the identity of a cloud user?
3. What determines the actions a user can perform?
4. What defines the responsibilities of a user?
5. Which component monitors cloud resource health?
6. Which component protects cloud resources from cyber threats?
7. Which portal feature displays resource status and notifications?
8. What allows users to quickly locate cloud services?
9. What uniquely identifies a cloud user?
10. Which verification method uses an OTP?
11. Which verification method uses a verification link?
12. Who creates new cloud user accounts?
13. What confirms the validity of a newly created account?
14. Which agreement defines cloud services and billing?
15. Which model charges users based on resource usage?
16. Which reports help optimize cloud costs?
17. Which component manages cloud expenditure?
18. Which component allocates cloud resources?
19. Which process grants users access to cloud resources?
20. Which practice helps reduce unnecessary cloud costs?

SGOU

Answers to Objective Type Questions

1. Cloud Administration Portal
2. Identity Management
3. Permission
4. Role
5. Monitoring
6. Security Management
7. Dashboard
8. Search
9. Cloud Account
10. Mobile Number Verification
11. Email Verification
12. Administrator
13. Account Verification
14. Cloud Subscription
15. Pay as you use
16. Usage Reports
17. Subscription Management
18. Resource Management
19. Authentication
20. Removing Unused Resources

SGOU

Assignments

1. Explain the objectives and major components of cloud administration. How do these components work together to ensure efficient cloud management?
2. Describe the process of creating and verifying a cloud account. Why is account verification important in a cloud environment?
3. Explain the relationship between user roles and permissions in cloud computing. How does role based access control simplify cloud administration?
4. A university has admitted 2,500 new students and recruited 100 faculty members. Analyze how cloud administration, user roles, permissions, and

account verification can be used to provide secure and efficient access to cloud resources.

5. A company notices a significant increase in its monthly cloud bill. Analyze the possible reasons for the increased cost and recommend subscription management practices that can help optimize cloud resource utilization and reduce expenses.

Reference

1. Erl, T., Mahmood, Z., & Puttini, R. (2013). *Cloud computing: Concepts, technology & architecture*. Pearson.
2. Buyya, R., Vecchiola, C., & Selvi, S. T. (2013). *Mastering cloud computing: Foundations and applications programming*. Morgan Kaufmann.
3. Velte, A. T., Velte, T. J., & Elsenpeter, R. (2019). *Cloud computing: A practical approach* (2nd ed.). McGraw Hill.
4. Rhoton, J. (2013). *Cloud computing explained: Implementation handbook for enterprises*. Recursive Press.
5. Sosinsky, B. (2011). *Cloud computing bible*. Wiley.

Suggested Reading

1. Microsoft. (2024). *Microsoft Azure documentation*. https://learn.microsoft.com/azure
2. Amazon Web Services. (2024). *AWS documentation*. https://docs.aws.amazon.com
3. Google Cloud. (2024). *Google Cloud documentation*. https://cloud.google.com/docs
4. NIST. (2011). *The NIST definition of cloud computing (Special Publication 800-145)*. National Institute of Standards and Technology. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf
5. Red Hat. (2024). *What is cloud computing?* https://www.redhat.com/en/topics/cloud-computing

SGOU

Unit 2
Identity and Access management

Learning Outcomes

Upon completion of the unit, learner will be able to:

- define identity and access management, and the AAA model
- explain the functions of authentication, authorization, and accounting in the AAA model
- describe the concept and working of Single Sign On (SSO)
- differentiate between authentication, authorization, and accounting
- analyze the significance of the AAA model and SSO in securing cloud environments

Prerequisite

As people increasingly rely on cloud services for education, business, banking, healthcare, and communication, protecting digital identities and controlling access to resources have become critical challenges. Organizations must ensure that only legitimate users can access cloud resources while maintaining a balance between security and user convenience. At the same time, they need mechanisms to monitor user activities and simplify access to multiple cloud applications without compromising security. Understanding how cloud platforms achieve these objectives is essential for anyone involved in cloud computing.

This topic introduces the concepts of Identity and Access Management (IAM), the AAA model, and Single Sign-On (SSO). It explains how cloud platforms verify user identities, determine access permissions, maintain activity records, and enable users to access multiple applications through a single authentication. These concepts form the foundation of secure access control in modern cloud environments and are widely implemented by leading cloud service providers. By learning this topic, you will gain the knowledge required to understand and manage secure user access in cloud-based systems. Let us begin by exploring the fundamentals of Identity and Access Management.

SGOU



Keywords

Identity Management, Authentication, Authorization, Accounting, AAA Model, Single Sign-On (SSO), Identity Provider (IdP), Multi-Factor Authentication (MFA)

Discussion

3.2.1 Fundamentals of Identity and Access Management

In the previous section, you learned how cloud administration portals help administrators manage cloud resources, create user accounts, assign roles and permissions, and manage subscriptions. These activities ensure that cloud resources are organized and available to authorized users. However, creating user accounts and assigning roles alone are not sufficient to secure a cloud environment.

Consider a university cloud portal where thousands of students, faculty members, and administrative staff access digital services every day. Students log in to view learning materials and submit assignments. Faculty members upload course content, evaluate assignments, and record grades. Administrative staff manage admissions, maintain student records, and generate institutional reports.

Although all these users access the same cloud platform, should every user be allowed to access every resource available in the system?

The answer is clearly no. A student should not be able to modify examination records or change institutional policies. Similarly, a faculty member should not have permission to create administrator accounts or manage cloud subscriptions. Every user should access only the resources required to perform assigned responsibilities.

This raises two important questions.

1. How does the cloud platform recognize who is requesting access?
2. How does it determine whether the user has permission to access a particular resource?

The answers to these questions lie in Identity and Access Management (IAM).

Identity and Access Management is a framework of policies, processes, and technologies that identifies users and controls their access to digital resources. It ensures that the right user can access the right resource at the right time while preventing unauthorized access. IAM is one of the most important security mechanisms in cloud computing because cloud environments are shared by many users with different responsibilities.

Without Identity and Access Management, any authenticated user might gain access to confidential information or perform actions beyond assigned responsibilities. Such situations could result in data breaches, service disruptions, financial losses, and violations of organizational security policies.

SGOU

To understand the importance of IAM, consider a public library. Every member receives a unique membership card before borrowing books. The card identifies the member and records borrowing activities. Visitors who are not registered members cannot borrow books. Library staff, however, have additional privileges such as issuing membership cards, updating book records, and managing library operations.

A cloud environment follows a similar approach. Every user is assigned a unique digital identity that enables the cloud platform to recognize the user and determine the resources that the user is permitted to access.

The term identity refers to the unique digital representation of a user within a cloud environment. It distinguishes one user from another by storing information such as a username, email address, employee ID, or student registration number.

Table 3.2.1 Examples of digital identities used in a university cloud platform.

User	Digital Identity	Typical Responsibility
Student	Student ID and institutional email	Access learning materials and submit assignments
Faculty Member	Employee ID and institutional email	Upload course materials and evaluate assignments
Administrator	Administrator account	Manage users, resources, and cloud services

Although all these users access the same cloud platform, each user has a unique identity. This enables the cloud system to recognize users individually and associate them with appropriate responsibilities.

Having a digital identity alone is not sufficient. The cloud platform must also determine whether the identified user is permitted to access a requested resource. This process is known as access management.

Access management controls who can access a resource and specifies the operations that can be performed after access is granted. It ensures that users receive only the permissions required to perform their assigned responsibilities. This security principle is known as the principle of least privilege, which limits users to the minimum level of access necessary to complete their tasks.

As organizations continue to adopt cloud computing, the number of users, applications, and cloud resources continues to grow rapidly. Managing identities and controlling access manually becomes increasingly difficult and may lead to security vulnerabilities. Identity and Access Management provides a systematic approach to managing digital identities, controlling access, and protecting cloud resources.

In the following sections, you will explore the fundamental functions of Identity and Access Management, understand how users are authenticated and authorized, and learn

SGOU

how cloud platforms ensure secure access to digital resources.

3.2.1.1 Identity Management

Identity is one of the most valuable assets in a cloud environment. Every activity performed in the cloud, such as accessing files, creating virtual machines, or sharing documents, begins with identifying the user. Before granting access to any cloud resource, the system must first determine who is requesting the service. This process is known as identity management.

Identity management is the process of creating, maintaining, and managing digital identities for users, devices, and applications within an organization. It ensures that every authorized entity has a unique identity, allowing the cloud platform to distinguish one user from another.

Consider a college that uses a cloud-based Learning Management System (LMS). Thousands of students, faculty members, and administrative staff access the system every day. Each user is assigned a unique username or institutional email address. Even if two students have the same name, their registration numbers and login credentials are different. This uniqueness enables the cloud platform to identify users accurately.

A digital identity is a collection of information that uniquely represents a user in a cloud environment. It contains various attributes that help identify the user and determine appropriate access rights.

Common attributes of a digital identity include:

- Username
- User ID or Employee ID
- Email address
- Department or Organization
- Designation or Role
- Contact information
- Authentication credentials

These attributes help cloud platforms recognize users and associate them with specific organizational responsibilities.

Example: Identity Management in a University

Consider a university that uses a cloud platform for academic management.

A newly admitted student receives a registration number, institutional email address, and login account. The student can access learning materials, assignments, and examination results.

A faculty member receives a different identity with additional privileges for uploading

SGOU

lecture notes, creating assignments, and evaluating students.

An administrator receives a highly privileged identity that allows user management, resource allocation, and security configuration.

Although all three users access the same cloud platform, their digital identities are unique and are managed separately. This enables the cloud platform to recognize each user correctly and provide services according to organizational policies.

3.2.2 AAA Model

Identity management forms the foundation of cloud security by creating and maintaining unique digital identities for users. However, recognizing a user is only the first step. Before granting access to cloud resources, the platform must verify that the claimed identity is genuine, determine the resources the user is permitted to access, and maintain a record of the user's activities. These three essential security functions are collectively provided by the AAA model, which stands for Authentication, Authorization, and Accounting. The AAA model is widely used in cloud environments to ensure secure access, enforce access control, and support auditing and compliance. The first component of the AAA model, authentication, verifies the identity of the user.

3.2.2.1 Authentication

Identity management enables a cloud platform to recognize users by maintaining their digital identities. However, simply having a digital identity is not sufficient to access cloud resources. A user may claim to be someone else by entering another person's username or email address. Therefore, before granting access, the cloud platform must verify whether the user is genuinely the owner of the claimed identity. This verification process is known as authentication.

Authentication is the process of verifying the identity of a user, device, or application before granting access to cloud resources. It answers the question:

"Are you really who you claim to be?"

Authentication is the first line of defense in cloud security. Only after successful authentication can a user proceed to access cloud services. If authentication fails, access is denied, protecting sensitive information from unauthorized users.

After a user attempts to access a cloud service, the authentication process begins. The user first enters a username or email address and then provides authentication credentials such as a password, fingerprint, or verification code. The cloud platform verifies these credentials by comparing them with the information stored in its authentication database. If the credentials are valid, the user's identity is confirmed and access is granted. If the credentials do not match, authentication fails and access is denied.

Figure 3.2.1 illustrates the authentication process.

SGOU

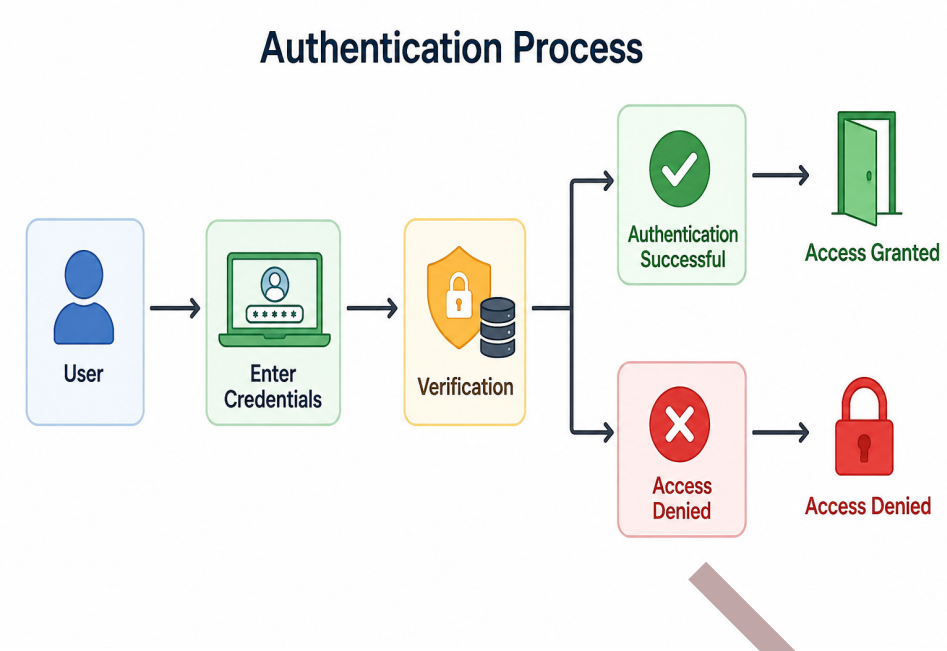


Fig.3.2.1 Authentication Process

Authentication Factors

Authentication is based on one or more factors that verify a user's identity. These factors are classified into different categories.

- Knowledge Factor

A knowledge factor is something the user knows. Examples include: Password, PIN (Personal Identification Number), Security question and answer

Passwords are the most commonly used authentication method. However, weak or reused passwords can increase security risks.

- Possession Factor

A possession factor is something the user possesses. Examples are Mobile phone, Smart card, Security token, One Time Password (OTP)

When logging into an online banking application, a verification code is often sent to the registered mobile number. Access is granted only after entering the correct OTP.

- Inherence Factor

An inherence factor is something the user is.

Examples include:

- Fingerprint
- Face recognition
- Iris scan

SGOU

- Voice recognition

Biometric authentication provides stronger security because these characteristics are unique to each individual.

Types of Authentication

Cloud platforms use different authentication methods depending on the required level of security.

1. **Single Factor Authentication (SFA)**

Single Factor Authentication uses only one authentication factor, usually a password.

For example, a student logs into the university portal using only a username and password.

Although simple to use, Single Factor Authentication provides limited security because passwords can be guessed, stolen, or compromised.

2. **Multi Factor Authentication (MFA)**

Multi Factor Authentication verifies a user using two or more authentication factors.

For example, after entering a password, a user receives an OTP on a registered mobile phone. Access is granted only after both the password and OTP are verified.

Many cloud providers, including Microsoft Azure, Amazon Web Services (AWS), and Google Cloud, recommend Multi Factor Authentication for protecting cloud accounts.

Authentication successfully verifies the identity of a user. However, authentication alone does not determine what the authenticated user is allowed to do within the cloud environment. Once the identity has been verified, the cloud platform must decide which resources the user can access and which operations can be performed. This process is known as authorization, discussed in the next section.

3.2.1.2 Authorization

Authentication confirms the identity of a user, but it does not determine what the user is allowed to do after signing in. Once the identity has been verified, the cloud platform must decide which resources the authenticated user can access and what operations the user can perform. This decision-making process is known as authorization.

Authorization is the process of granting or denying access to cloud resources based on the permissions assigned to a user. It determines what an authenticated user is allowed to access and the actions that can be performed on those resources.

In simple terms, authentication answers the question, "Who are you?" whereas authorization answers the question, "What are you allowed to do?"

Consider a university cloud portal used by students, faculty members, and administrators. After successfully logging in, all users are authenticated. However, each user is allowed to perform only specific tasks according to assigned responsibilities.

SGOU

For example:

- A student can access learning materials, submit assignments, and view examination results.
- A faculty member can upload lecture notes, create assignments, and evaluate students.
- An administrator can create user accounts, assign roles, manage subscriptions, and configure security settings.

Although all three users are authenticated, they have different levels of authorization.

In cloud environments, permissions are generally assigned to roles instead of individual users. This approach is known as **Role Based Access Control (RBAC).**

Suppose the university has 3,000 students. Instead of assigning identical permissions separately to every student, administrators create a single role called *Student*. All permissions required by students are assigned to this role. Whenever a new student joins the university, assigning the Student role automatically grants the required permissions.

Similarly, separate roles such as Faculty Member and Administrator can be created with different permission levels. Table 3.2.2 shows examples of authorization based on user roles.

Table 3.2.2. Examples of authorization based on user roles.

User Role	Authorized Activities
Student	View course materials, submit assignments, view grades
Faculty Member	Upload lecture notes, create assignments, evaluate students
Administrator	Manage users, assign roles, configure cloud services

This approach simplifies administration because permissions are managed at the role level rather than individually for every user.

3.2.1.3 Accounting

Authentication verifies the identity of a user, and authorization determines the resources that the authenticated user can access. However, cloud platforms also need to maintain a record of user activities after access has been granted. These records help organizations monitor system usage, investigate security incidents, and comply with regulatory requirements. This function is known as accounting, the third component of the AAA model.

Accounting is the process of recording, monitoring, and maintaining information about the activities performed by users, devices, or applications in a cloud environment. It keeps track of who accessed the system, when the access occurred, which resources

SGOU

were used, and what actions were performed.

Unlike authentication and authorization, which focus on granting secure access, accounting focuses on maintaining a history of user activities. This information is valuable for auditing, troubleshooting, performance analysis, and security investigations.

What information does accounting record?

Accounting records various details related to user activities in the cloud. Some commonly recorded information includes:

- User identity
- Login and logout time
- Duration of the user session
- Resources accessed
- Operations performed
- IP address or device information
- Success or failure of login attempts

These records are usually stored in log files or audit logs maintained by the cloud platform.

Consider a university cloud portal used by students and faculty members.

A student logs into the portal at 9:00 a.m., downloads lecture notes, submits an assignment, and logs out at 9:45 a.m. Throughout this session, the cloud platform records the student's login time, accessed resources, assignment submission, logout time, and the device used for accessing the portal.

Similarly, when a faculty member uploads lecture notes or records examination grades, these activities are also stored in audit logs. If any dispute or security issue arises later, administrators can review these records to determine who performed the activity and when it occurred.

Having understood each component individually, it is useful to see how they work together during a typical user login session. The next section explains the working of the AAA model and illustrates how authentication, authorization, and accounting cooperate to provide secure access to cloud resources.

3.2.1.4 Working of the AAA Model

The three components of the AAA model do not function independently. Instead, they work together in a sequence to provide secure and controlled access to cloud resources. Whenever a user attempts to access a cloud service, the cloud platform first verifies the user's identity, then checks the user's access rights, and finally records the activities performed during the session. This coordinated process ensures that only authorized users can access cloud resources while maintaining a complete record of their activities.

SGOU

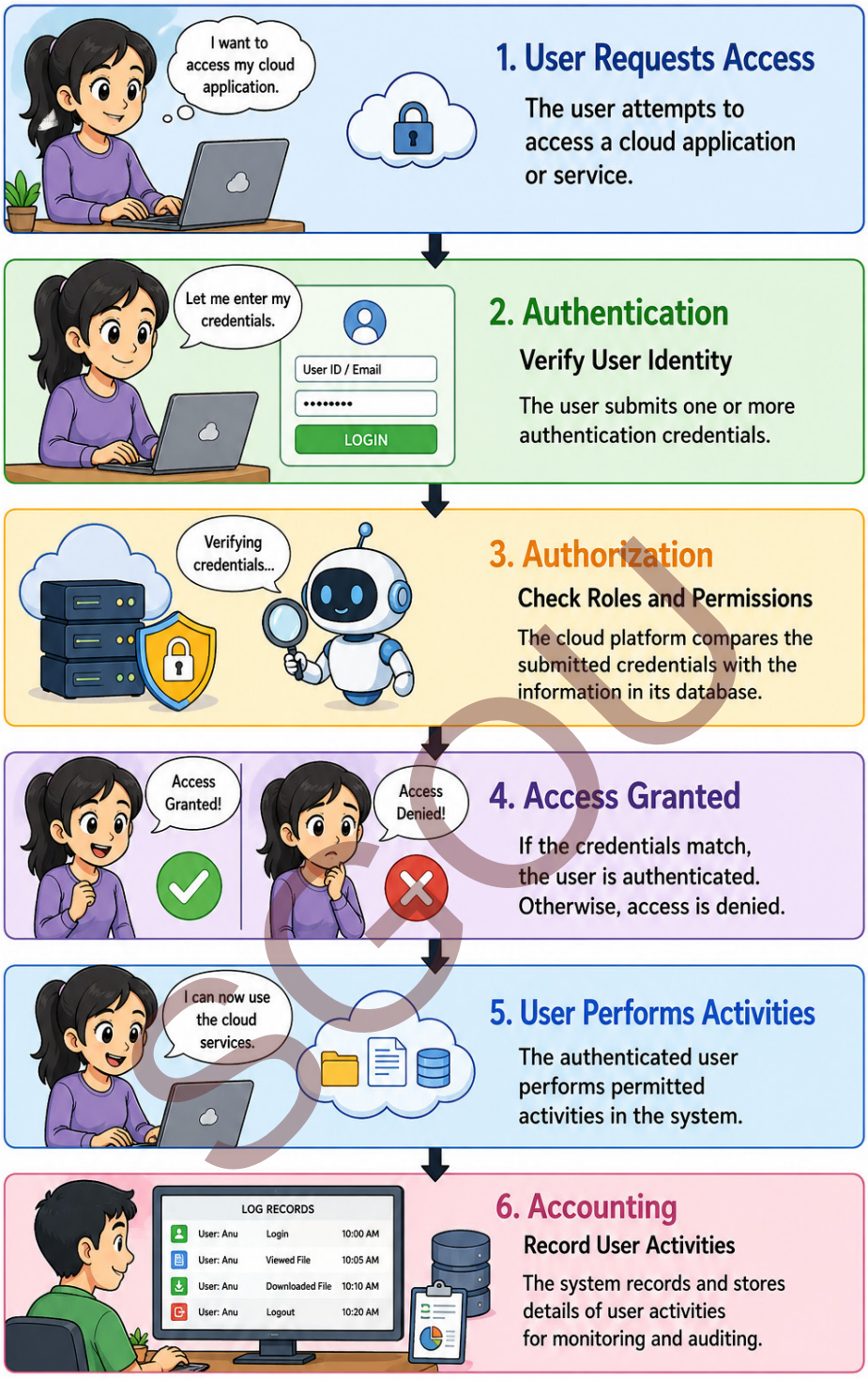


Fig. 3.2.3 Working of AAA model

The working of the AAA model can be understood through the following steps.

Step 1: User Requests Access

The process begins when a user attempts to access a cloud application or service by entering login credentials such as a username and password.

For example, a student signs in to the university's Learning Management System (LMS) to access course materials.

Step 2: Authentication Verifies the User

The cloud platform verifies the user's identity by checking the submitted credentials against the stored authentication information.

If the credentials are correct, the user is authenticated. Otherwise, access is denied.

Step 3: Authorization Checks Permissions

After successful authentication, the cloud platform determines what the authenticated user is allowed to access. It examines the user's assigned role and associated permissions before granting access to cloud resources.

For example, a student can view lecture notes and submit assignments but cannot modify examination records. Similarly, a faculty member can upload lecture materials but cannot manage cloud subscriptions.

Step 4: User Accesses Cloud Resources

Once authorization is completed successfully, the user is allowed to perform only the activities permitted by the assigned role.

The cloud platform continuously enforces access restrictions throughout the user session.

Step 5: Accounting Records User Activities

As the user interacts with cloud resources, the cloud platform records important information such as login time, accessed resources, performed operations, and logout time.

These records are stored in audit logs and can later be used for monitoring, troubleshooting, and security investigations.

The AAA model forms the foundation of secure access control in modern cloud computing. However, users often need to access multiple cloud applications during their daily work. Repeatedly entering login credentials for every application can be inconvenient and time-consuming. To overcome this limitation, cloud platforms provide a mechanism called Single Sign-On (SSO), which enables users to access multiple applications after signing in only once. The next section explains the concept and working of Single Sign-On (SSO).

3.2.3 Single Sign-On (SSO)

So far, you have learned how the AAA model secures access to cloud resources by verifying user identities, controlling access permissions, and maintaining activity records. Although this model provides strong security, users often access multiple cloud applications during their daily work. Entering separate login credentials for every

SGOU

application can be inconvenient, time-consuming, and may even encourage users to adopt insecure practices such as reusing passwords.

Consider a university that uses different cloud applications for online learning, library services, examination management, email, and student information systems. If students and faculty members have to sign in separately to each application, they must remember multiple usernames and passwords. This not only reduces productivity but also increases the burden of managing multiple accounts.

Is there a way to access all these applications by signing in only once?

The answer is Single Sign-On (SSO).

Single Sign-On (SSO) is an authentication mechanism that allows users to access multiple applications or services by signing in only once. After successful authentication, users can move between authorized applications without entering their credentials again during the same session.

In other words, a single login provides access to multiple related applications, making the authentication process faster and more convenient.

3.2.3.1 How Single Sign-On Works

Single Sign-On relies on a central authentication service known as an Identity Provider (IdP). Instead of each application verifying user credentials independently, all applications trust the Identity Provider to authenticate users.

The working of Single Sign-On generally follows these steps.

1. **User Requests Access**

The user attempts to access a cloud application.

2. **Redirection to Identity Provider**

The application redirects the user to the Identity Provider for authentication.

3. **User Authentication**

The user enters the required credentials. The Identity Provider verifies the user's identity.

4. **Authentication Token Issued**

After successful authentication, the Identity Provider generates a secure authentication token.

5. **Access to Multiple Applications**

The user presents the authentication token while accessing other connected cloud applications. Since the token has already been verified, the user is not required to log in again.

SGOU

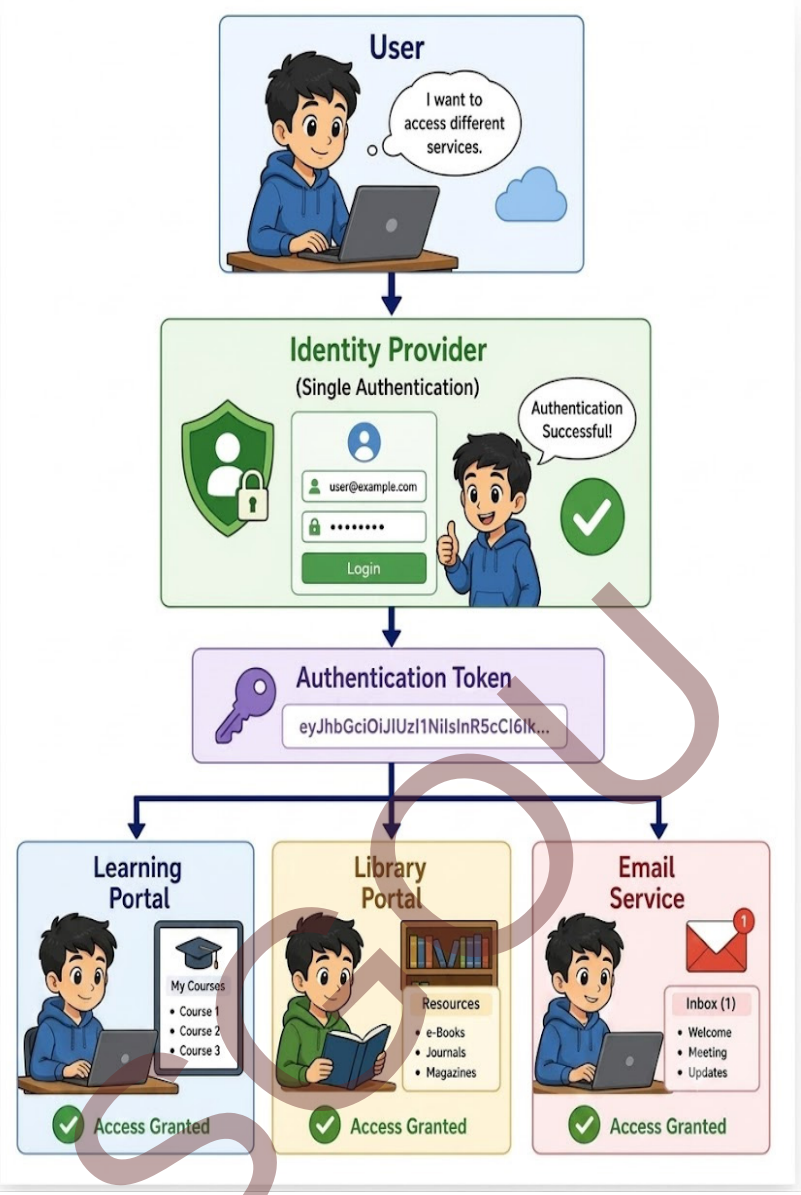


Fig.3.2.4 Working of Single Sign-On

Advantages of Single Sign-On

Single Sign-On offers several benefits for both users and organizations.

- Eliminates repeated login procedures.
- Improves user convenience and productivity.
- Reduces password fatigue.
- Minimizes password reset requests.
- Simplifies user account management.

- Supports centralized authentication.
- Enhances user experience across multiple cloud services.

Limitat

Although Single Sign-On offers many advantages, it also has certain limitations.

- If the SSO account is compromised, multiple connected applications may become accessible to an attacker.
- Failure of the Identity Provider may temporarily affect access to connected applications.
- Initial implementation and integration may require careful planning.
- Strong authentication methods, such as Multi-Factor Authentication (MFA), are often required to improve security.

The AAA model and Single Sign-On complement each other in cloud security.

The AAA model ensures secure identity verification, access control, and activity monitoring, while Single Sign-On improves the authentication experience by allowing users to authenticate once and access multiple cloud applications.

For example, when a user signs in through Single Sign-On, the Authentication component of the AAA model verifies the user's identity. Authorization determines the applications and resources the user is allowed to access, and Accounting records the user's activities across the cloud environment. Thus, Single Sign-On operates in conjunction with the AAA model to provide both security and convenience.

Identity and Access Management, the AAA model, and Single Sign-On together form the foundation of secure access control in cloud computing. They help organizations protect digital resources, simplify user management, and provide seamless access to cloud services while maintaining security and accountability.

SGOU

Recap

- Identity and Access Management (IAM) manages digital identities and controls access to cloud resources.
- Every user is assigned a unique digital identity.
- Identity management creates and maintains user identities.
- The AAA model consists of Authentication, Authorization, and Accounting.
- Authentication verifies the identity of a user.
- Authorization determines the resources a user can access.
- Accounting records user activities for monitoring and auditing.
- Role Based Access Control (RBAC) assigns permissions based on user roles.
- The principle of least privilege grants only the minimum required permissions.
- Single Factor Authentication uses one authentication factor.
- Multi Factor Authentication (MFA) uses two or more authentication factors.
- Single Sign-On (SSO) enables access to multiple applications with one login.
- An Identity Provider (IdP) authenticates users in an SSO environment.
- Authentication tokens enable seamless access to multiple applications.
- SSO improves user convenience and reduces password fatigue.
- Audit logs help monitor user activities and support security investigations.
- The AAA model provides secure identity verification, access control, and activity monitoring.
- IAM, AAA, and SSO together strengthen cloud security and simplify user access.

SGOU

Objective Type Questions

1. What does IAM stand for?
2. Which framework consists of Authentication, Authorization, and Accounting?
3. Which process verifies a user's identity?
4. Which process determines user access permissions?
5. Which component records user activities?
6. Which access control model assigns permissions based on roles?

7. Which authentication method uses more than one verification factor?
8. Which authentication method uses only one verification factor?
9. Which mechanism allows users to access multiple applications after a single login?
10. Which entity authenticates users in an SSO environment?
11. Which digital credential is issued after successful authentication in SSO?
12. Which file stores records of user activities?
13. Which identifier uniquely represents a user in a cloud environment?
14. Which factor uses fingerprints or facial recognition for authentication?

Answers to Objective Type Questions

1. Identity and Access Management (IAM)
2. AAA Model
3. Authentication
4. Authorization
5. Accounting
6. Role Based Access Control (RBAC)
7. Multi Factor Authentication (MFA)
8. Single Factor Authentication (SFA)
9. Single Sign-On (SSO)
10. Identity Provider (IdP)
11. Authentication Token
12. Audit Log
13. Digital Identity
14. Inherence Factor

SGOU

Assignments

1. Explain the concept of Identity and Access Management (IAM). Discuss its significance in securing cloud environments.
2. Describe the components of the AAA model. Explain how Authentication, Authorization, and Accounting work together to provide secure access to cloud resources.
3. Explain the concept and working of Single Sign-On (SSO). Discuss its advantages and limitations in cloud computing.
4. A university uses separate login credentials for its Learning Management System, Digital Library, Email Service, and Examination Portal. Analyze how implementing Single Sign-On (SSO) can improve user experience while maintaining cloud security.
5. An organization discovers that an employee accessed confidential files beyond their job responsibilities. Analyze how the AAA model and the Principle of Least Privilege could have prevented this security incident.

Reference

1. Stallings, W. (2020). *Network security essentials: Applications and standards* (7th ed.). Pearson.
2. Ferraiolo, D. F., Kuhn, D. R., & Chandramouli, R. (2007). *Role-based access control* (2nd ed.). Artech House.
3. Joshi, J. B. D., Bertino, E., Ghafoor, A., & Latif, U. (Eds.). (2011). *Secure information and network management*. Springer.
4. Erl, T., Mahmood, Z., & Puttini, R. (2013). *Cloud computing:Concepts, technology & architecture*. Pearson.
5. Buyya, R., Vecchiola, C., & Selvi, S. T. (2013). *Mastering cloud computing: Foundations and applications programming*. Morgan Kaufmann.

SGOU

Suggested Reading

1. National Institute of Standards and Technology. (2011). *The NIST definition of cloud computing (Special Publication 800-145)*. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf
2. Microsoft. (2024). *Microsoft Entra documentation*. https://learn.microsoft.com/entra
3. Amazon Web Services. (2024). *AWS Identity and Access Management (IAM) documentation*. https://docs.aws.amazon.com/iam/
4. Google Cloud. (2024). *Identity and Access Management documentation*. https://cloud.google.com/iam/docs
5. OpenID Foundation. (2024). *OpenID Connect*. https://openid.net/connect/

SGOU

Unit 3
Resource Management

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the different types of cloud resources and their roles in cloud environments
- describe the process of resource provisioning and de-provisioning in cloud environments
- compare different resource management models used in cloud computing.
- explain the role of auto scaling, quotas, and resource limits in optimizing cloud resource utilization
- identify the purpose of resource tagging in organizing and managing cloud resources

Prerequisite

A cloud application may start with only a few users, but its resource requirements can change within minutes. A sudden increase in users can demand more computing power, storage, and network capacity. When the demand decreases, keeping all those resources active can waste money. At the same time, allowing resources to grow without control can create another problem: excessive resource consumption.

So, how does a cloud environment decide which resources are needed, when they should be added, when they should be released, and who is allowed to use them? How can resources be organized when hundreds of cloud services are running at the same time? These questions are at the heart of resource management.

Understanding resource management helps in developing the skill of selecting, allocating, monitoring, adjusting, and controlling cloud resources efficiently. It also provides a foundation for understanding how cloud platforms maintain performance while avoiding unnecessary resource usage and cost.

Now, let us explore how cloud resources are managed systematically through resource management models, auto scaling, quotas and limits, and tagging.



SGOU

Keywords

Resource provisioning, Resource de-provisioning, Dynamic resource management, Auto scaling, Quotas, Service limits, Tagging, Resource utilization

Discussion

3.3.1 Introduction to Resource Management

Imagine that a college computer laboratory has 50 computers, but only 20 students are using them at a particular time. If all the computers remain powered on throughout the day, electricity is wasted. On the other hand, if 80 students arrive and only 50 computers are available, some students will have to wait. This situation highlights the importance of managing available resources efficiently. Similar challenges arise in cloud computing, where computing resources must be allocated according to user requirements while avoiding unnecessary costs and resource wastage.

Cloud computing provides a wide range of resources such as virtual machines, storage, databases, and networking services over the Internet. These resources are shared among multiple users and organizations. As the number of users and applications increases, it becomes essential to ensure that resources are available whenever needed without affecting performance or increasing operational costs. This process of planning, allocating, monitoring, and controlling cloud resources is known as resource management.

Resource management is one of the fundamental responsibilities of cloud service providers and cloud administrators. It ensures that the right resources are delivered to the right users at the right time. Efficient resource management improves system performance, enhances user experience, reduces operational expenses, and ensures optimal utilization of cloud infrastructure.

Consider an online shopping application during a festive sale. On normal days, a few servers may be sufficient to handle customer requests. However, during the sale, millions of users may access the website simultaneously. If additional computing resources are not made available, the website may become slow or even unavailable. Once the sale is over, keeping the extra servers running unnecessarily would increase operational costs. Effective resource management addresses these challenges by allocating resources when demand increases and releasing them when demand decreases.

Cloud resource management involves several activities. It begins with identifying the type of resources required for an application. Resources are then provisioned to users or services based on their requirements. As workloads change, resources may be increased, decreased, or removed to maintain efficiency. Cloud platforms also monitor resource usage continuously to ensure that available resources are utilized effectively. Policies, quotas, and limits help prevent excessive consumption of resources by individual users or applications.

SGOU

Resource management is not limited to allocating resources. It also focuses on maintaining high availability, ensuring security, balancing workloads, reducing costs, and supporting business continuity. Proper resource management enables cloud environments to respond dynamically to changing workloads without interrupting services.

Modern cloud platforms such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform provide automated resource management features. These platforms can automatically monitor workloads, provision additional resources when required, enforce usage limits, and remove unused resources. Such automation minimizes manual intervention and allows organizations to focus on application development rather than infrastructure management.

Figure 3.3.1 illustrates the basic resource management process in a cloud environment.

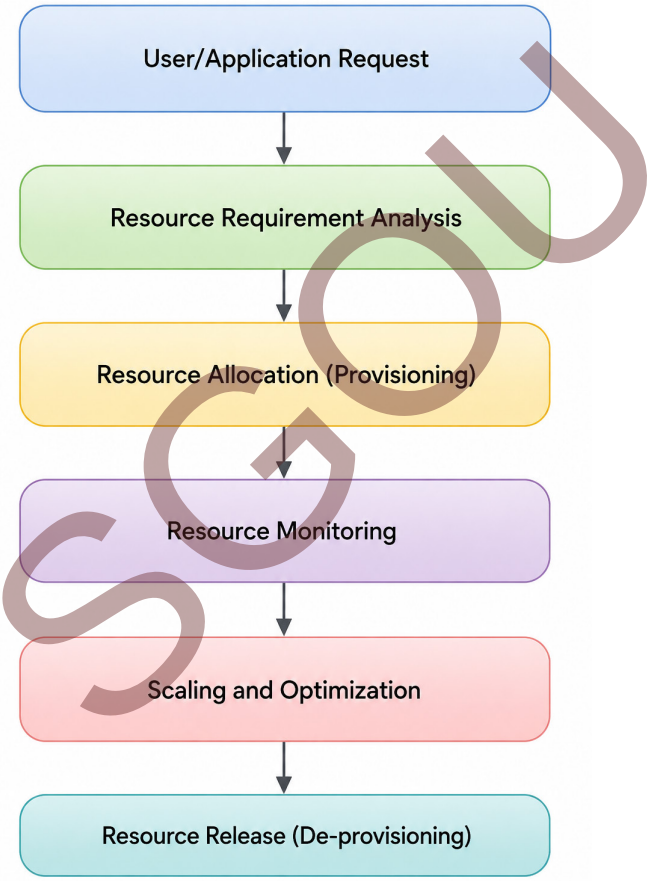


Fig.3.3.1 Basic Resource management process

The effectiveness of resource management directly influences the overall performance of cloud computing systems. Efficient management ensures that computing resources are available whenever required, minimizes resource wastage, controls operational costs, and maintains service quality. Therefore, understanding the principles and techniques of resource management is essential for designing, deploying, and maintaining reliable cloud-based applications.

The following sections discuss the different types of cloud resources, methods of provisioning and de-provisioning resources, resource management models, auto scaling, quotas and limits, and resource tagging. These concepts collectively provide the foundation for effective cloud resource management.

3.3.2 Types of Resources

A cloud environment consists of different types of resources that work together to provide applications and services to users. Each resource performs a specific function. Compute resources provide processing power, storage resources store information, network resources provide communication, database resources organize and manage data, and security resources protect applications and information.

Consider a college admission portal hosted in the cloud. During normal days, only a small number of applicants may access the portal. However, when admissions open, thousands of students may simultaneously register, upload documents, check application status, and make payments. The portal therefore requires adequate computing power, storage capacity, network connectivity, database services, and security mechanisms.

This scenario helps in understanding how different cloud resources work together to support a real-world application.

3.3.2.1 Compute Resources

Compute resources provide the processing power required to execute applications and services. They include virtual machines, virtual CPUs, memory, containers, and serverless computing resources.

A virtual machine (VM) is a software-based computer that runs on a physical server. A cloud user can select the required number of virtual CPUs, memory, operating system, and other configurations according to the application requirements.

For example, the college admission portal may use virtual machines to process student registrations, validate submitted information, generate application numbers, and handle payment-related requests. During the initial days of admission, the number of requests may increase considerably. Additional computing resources may therefore be required to maintain the performance of the portal.

Cloud platforms also provide containers. A container packages an application together with the libraries and dependencies required to run it. Containers can be created and removed quickly, making them useful for applications with changing workloads.

Serverless computing provides another approach to managing compute resources. In this model, users do not directly manage the underlying servers. The cloud platform provides computing capacity when a function needs to execute.

SGOU

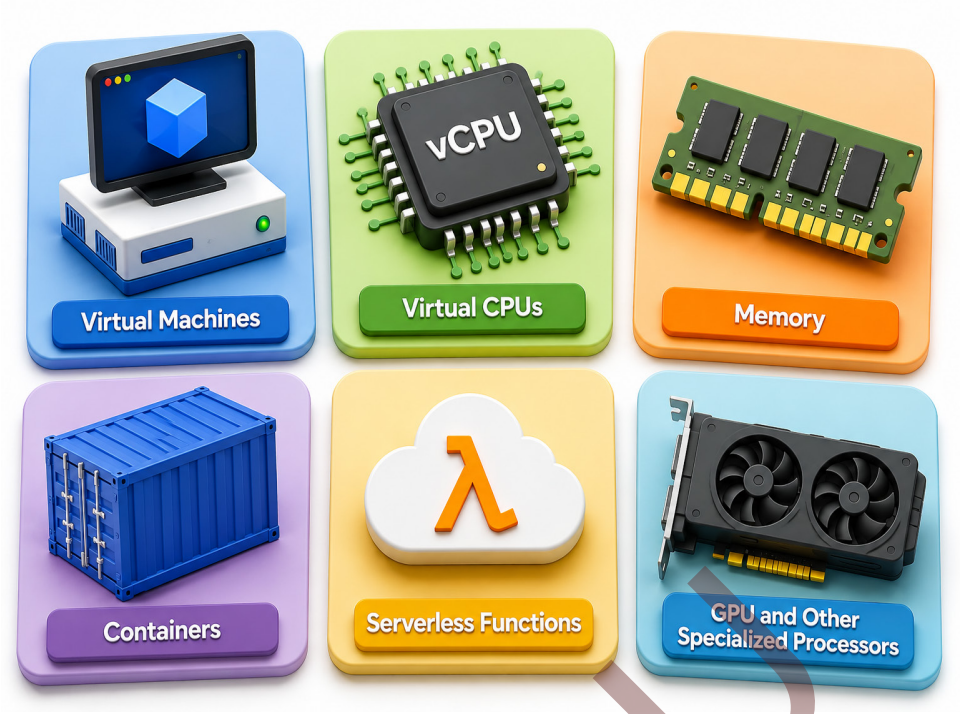


Fig.3.3.2 Common Compute resources

Suppose 500 students normally access an admission portal every day, but 20,000 students access it on the final day of admission. Would the same amount of computing resources be sufficient? The answer illustrates why compute resources must be managed according to workload.

3.3.2.2 Storage Resources

Storage resources are used to store application data, files, documents, images, videos, backups, and other digital information. Cloud storage allows organizations to store large amounts of information without maintaining physical storage infrastructure.

In the college admission portal, applicants may upload photographs, identification documents, certificates, mark sheets, and other supporting documents. These files require appropriate storage resources.

Different storage technologies can be selected according to the nature of the data. Object storage is commonly suitable for documents, images, and other files. Block storage can be used for applications requiring high-performance storage volumes. File storage provides shared access to files from multiple systems.

Storage resources can differ in capacity, performance, availability, durability, and cost. Therefore, selecting the appropriate storage resource is an important part of resource management.

For example, recently uploaded admission documents may require frequent access, whereas old admission records and backups may be accessed less frequently. Different storage options can be selected to balance performance and cost.

SGOU

Activity:

The admission portal stores student photographs, uploaded certificates, application records, and backup files. Which of these types of information would require frequent access? Which information could be moved to lower-cost storage after the admission process is completed?

3.3.2.3 Network Resources

Network resources provide communication between users, applications, servers, databases, and other cloud services. They allow students to access the college admission portal through the Internet.

Network resources include virtual networks, subnets, IP addresses, routers, gateways, load balancers, firewalls, and other networking components.

A virtual network provides an isolated network environment for cloud resources. It can be divided into smaller networks called subnets. IP addresses identify resources within a network, while routers and gateways help direct traffic between different networks.

A load balancer distributes incoming requests among multiple servers. This prevents one server from becoming overloaded and can improve application availability.

For example, during the last few hours before admission closes, thousands of students may try to access the portal at the same time. A load balancer can distribute their requests among several application servers, helping the portal handle the increased traffic.

Network resources help to:

- Connect cloud resources
- Control network traffic
- Improve application availability
- Distribute workloads
- Isolate resources
- Support secure communication

3.3.2.4 Database Resources

Database resources provide mechanisms for storing, organizing, retrieving, and managing application data. Cloud database services allow organizations to use databases without having to manage all the underlying physical infrastructure.

For the college admission portal, the database may contain applicant names, application numbers, contact details, course preferences, payment status, admission status, and other application-related information.

SGOU

Relational databases organize information into tables consisting of rows and columns. They are suitable when relationships between different types of information need to be maintained. For example, applicant information can be related to course information and payment records.

Non-relational databases, commonly known as NoSQL databases, can handle flexible data structures and are useful for applications requiring large-scale or rapidly changing data.

Database resources may include database instances, storage capacity, processing capacity, memory, database connections, backup resources, and replication resources.

During the admission period, database activity may increase significantly because students are continuously submitting applications, checking application status, and updating information. Adequate database resources are therefore required to maintain performance.

Check your understanding:

Why would the admission portal require more database processing capacity on the final day of admission than on an ordinary day?

3.3.2.5 Security Resources

Security resources protect cloud applications, data, users, and infrastructure from unauthorized access and other security threats.

Security is particularly important for a college admission portal because it handles personal information, uploaded documents, login credentials, and payment-related information.

Security resources include Identity and Access Management (IAM), access policies, encryption services, security groups, firewalls, encryption keys, digital certificates, and security monitoring services.

Identity and Access Management controls who can access particular cloud resources and what operations they are allowed to perform. For example, an admission administrator may be permitted to access application records, while a technical support employee may have access only to specific system resources.

Encryption protects sensitive information by converting it into a protected form. It can be used for information stored in cloud storage as well as information transmitted between the student's device and the admission portal.

Firewalls and security groups control network traffic according to predefined rules. They can allow legitimate connections and block unauthorized traffic.

Security monitoring and logging can record activities such as login attempts, changes to cloud resources, and access to sensitive information. These records can help identify unusual or unauthorized activities.

SGOU

Important security resources include:

- Identity and access management
- Access policies
- Firewalls
- Security groups
- Encryption mechanisms
- Encryption keys
- Digital certificates
- Security monitoring and logging

Integration of Cloud Resources

The different types of cloud resources work together to support the complete operation of an application. For instance, When a student uploads a mark sheet to a college admission portal, the different cloud resources work together in the following sequence:

1. Network resources transfer the mark sheet from the student's device to the cloud.
2. Compute resources process the upload request and handle the required application operations.
3. Storage resources store the uploaded mark sheet securely.
4. Database resources record information related to the uploaded document, such as the applicant and document details.
5. Security resources control access and ensure that the document is available only to authorized users.

Thus, cloud resources are interconnected and work together to provide a complete service. Effective resource management requires proper planning, coordination, and utilization of all these resources.

3.3.3 Resource Provisioning and De-provisioning

Cloud applications require different types of resources to perform their operations. However, simply identifying these resources is not sufficient. The resources must be made available when required and released when they are no longer needed. Consider the college admission portal discussed earlier. When admission begins, the number of students accessing the portal increases. Additional computing power, storage, database capacity, and network resources may be required. After the admission process ends, keeping all these additional resources active may result in unnecessary resource consumption and increased cost.

SGOU

The processes of making resources available when needed and releasing them when they are no longer required are known as **resource provisioning and de-provisioning.**

3.3.3.1 Resource Provisioning

Resource provisioning is the process of creating, configuring, and allocating cloud resources according to the requirements of an application, user, or workload.

In a traditional computing environment, obtaining additional resources may require purchasing physical servers, installing operating systems, configuring networks, and setting up storage devices. This process can take considerable time. Cloud computing makes provisioning faster because resources can be created and configured through cloud management interfaces or automated tools.

For example, when applications open for a college admission portal, the number of users may increase significantly. Additional virtual machines can be provisioned to handle the increased number of requests. Additional storage may also be provisioned if applicants need to upload certificates and other documents.

Resource provisioning generally involves the following steps:

1. Identify the requirement – Determine what type and amount of resources are needed.
2. Select suitable resources – Choose appropriate compute, storage, network, database, or other resources.
3. Configure the resources – Specify parameters such as CPU, memory, storage capacity, network settings, and access permissions.
4. Allocate the resources – Make the configured resources available to the application or user.
5. Monitor the resources – Observe resource usage and performance to determine whether the allocated resources are sufficient.

3.3.3.2 Resource De-provisioning

Resource de-provisioning is the process of removing, releasing, or disabling cloud resources when they are no longer required.

Provisioning makes resources available, whereas de-provisioning removes unnecessary resources. Both processes are important for efficient resource management.

Return to the college admission portal. During the final day of admission, additional virtual machines may have been provisioned to handle heavy traffic. Once the admission period ends, the number of users may fall significantly. Keeping all the additional virtual machines running would consume resources unnecessarily.

The unused resources can therefore be de-provisioned.

Resource de-provisioning generally involves the following steps:

SGOU

1. Identify unused resources – Determine which resources are no longer required.
2. Check dependencies – Ensure that removing a resource will not affect other applications or services.
3. Backup required data – Preserve important information before removing resources, if necessary.
4. Release or terminate resources – Stop, delete, or release the resources according to the requirement.
5. Verify the release – Check that the resources have been successfully removed and that the application continues to operate correctly.

A Complete Scenario

Let us connect the concepts with the college admission portal.

1. Before admission begins: Four virtual machines, adequate database capacity, and required storage are provisioned.
2. Admission opens: Thousands of students begin accessing the portal. Additional computing and database resources are provisioned.
3. Peak admission period: More students upload certificates and application documents. Additional storage and processing capacity are made available.
4. Admission activity decreases: The number of users gradually falls, so some additional resources are no longer required.
5. After admission closes: Temporary virtual machines and other unnecessary resources are de-provisioned.
6. Final state: Only the resources required for maintaining application records and related services remain active.

This scenario demonstrates how provisioning and de-provisioning help a cloud application respond to changing requirements.

3.3.4 Overview of Resource Management Models

Cloud applications experience different levels of workload depending on their usage patterns. Some applications have a relatively stable number of users, while others experience significant changes in demand at different times.

Consider a food delivery application. During normal hours, the application may receive a moderate number of orders. During lunch and dinner hours, the number of users and orders may increase considerably. During weekends or special occasions, the demand may be even higher.

Such changes create the need for suitable approaches to manage cloud resources. A resource management model defines how resources are allocated, monitored, adjusted, and released according to the requirements of an application.

SGOU

Three important approaches are static resource management, dynamic resource management, and policy-based resource management.

3.3.4.1 Static Resource Management

Static resource management is an approach in which a fixed amount of cloud resources is allocated to an application in advance. The resource configuration generally remains unchanged unless it is manually modified.

Consider a food delivery application that is configured with four virtual machines to handle its regular workload. These four virtual machines continue to run regardless of whether the application receives 500 orders or 1,000 orders during a particular period.

Static resource management follows a simple sequence. First, the resource requirements of the application are determined based on its expected workload. The required resources are then allocated to the application. Once the application starts running, the allocated resources remain fixed throughout its execution, even if the workload changes. This approach provides predictable resource availability but may lead to underutilization when the application requires fewer resources.



Fig.3.3.3 Static Resource management process

3.3.4.2 Dynamic Resource Management

Dynamic resource management adjusts cloud resources according to changes in workload or demand. Resources can be added when demand increases and reduced when demand decreases.

Unlike static resource management, resources are not permanently fixed. The cloud environment can respond to changing requirements.

For the food delivery application, additional computing resources can be made available during lunch and dinner when the number of users and orders increases. When the rush ends, unnecessary resources can be released.

Example

Suppose the application requires:

- 3 virtual machines during low demand
- 6 virtual machines during normal demand
- 10 virtual machines during peak demand

During the evening rush, additional virtual machines can be added to handle the increased number of orders. After the rush, the additional machines can be released.

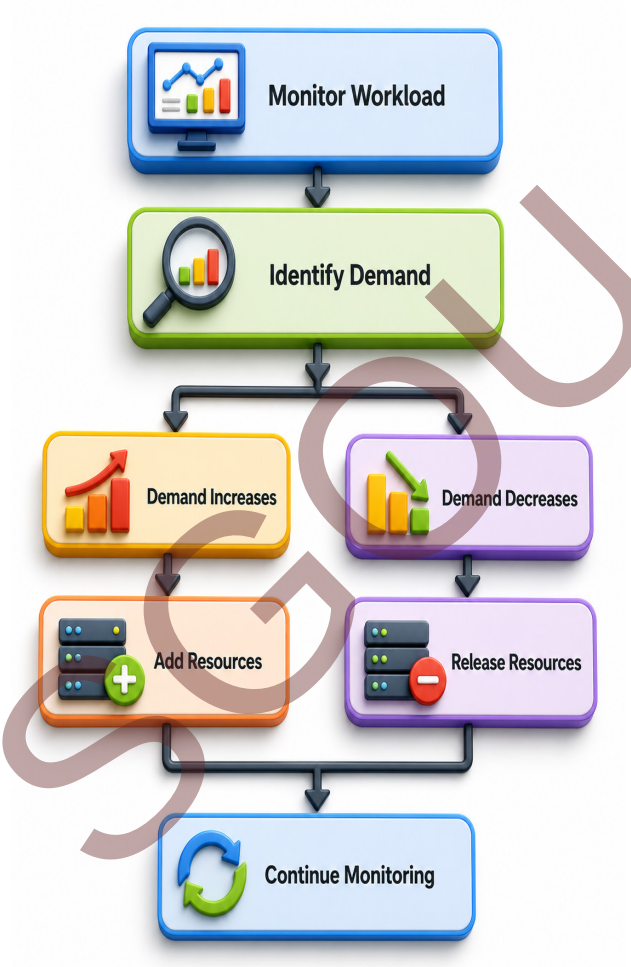


Fig.3.3.3 Dynamic Resource management process

3.3.4.3 Policy-Based Resource Management

Policy-based resource management manages cloud resources according to predefined rules or policies. These policies specify conditions under which resources should be allocated, modified, restricted, or released.

Instead of making every resource management decision manually, specific rules can be defined in advance.

For example:

If the average CPU utilization exceeds 70% for five minutes, add two virtual machines.

Another policy could be:

If CPU utilization remains below 30% for ten minutes, remove one virtual machine.

3.3.5 Auto Scaling

Resource management models provide different ways of controlling cloud resources according to application requirements. In particular, dynamic and policy-based resource management allow resources to change when workload conditions change. But how can these changes take place automatically without requiring continuous manual intervention?

Auto scaling provides an answer to this question. It enables a cloud environment to automatically increase or decrease resources based on the workload, predefined conditions, or application requirements.

Consider a food delivery application. During afternoon hours, the application may receive a moderate number of requests. During dinner time, thousands of users may simultaneously browse restaurants, place orders, and track deliveries. After the dinner rush, the workload may decrease again. Maintaining the same number of resources throughout the day may result in either insufficient resources during peak periods or unnecessary resource usage during low-demand periods.

Auto scaling helps the application respond to these changes by automatically adjusting the resources.

3.3.5.1 Types of Auto Scaling

The main purpose of auto scaling is to ensure that an application has sufficient resources when demand increases and does not retain unnecessary resources when demand decreases.

For example, suppose the food delivery application normally operates with four virtual machines. During the dinner rush, the number of requests increases and the existing resources become heavily utilized. Auto scaling can detect the increased workload and add additional virtual machines. When the demand decreases, the additional machines can be removed.

This provides a balance between performance and resource efficiency.

Auto scaling can be performed mainly in two ways.

Horizontal Scaling

Horizontal scaling changes the number of resource instances.

- **Scale out:** Add more instances when demand increases.

SGOU

- **Scale in:** Remove instances when demand decreases.

Vertical Scaling

Vertical scaling changes the capacity of an existing resource.

- **Scale up:** Increase the capacity of an existing resource.
- **Scale down:** Decrease its capacity when the additional capacity is no longer required.

For example, a virtual machine may initially have 2 CPU cores and 4 GB memory. When the workload increases, its capacity may be increased to 4 CPU cores and 8 GB memory.

3.3.5.2 Benefits of Auto Scaling

Auto scaling provides several benefits in cloud environments.

1. **Improved application performance**

Additional resources can be provided when workload increases, helping applications handle more users and requests.

2. **Efficient resource utilization**

Resources are increased when required and reduced when demand decreases. This reduces unnecessary resource allocation.

3. **Cost optimization**

Unused resources can be automatically released, helping reduce unnecessary cloud expenditure.

4. **Better availability**

Adequate resources can be made available during periods of high demand, reducing the possibility of service disruption caused by resource shortages.

5. **Automatic response to workload changes**

The cloud environment can respond to changing demand without requiring continuous manual intervention.

6. **Improved scalability**

Applications can handle variations in workload more effectively by increasing or decreasing available resources.

7. **Reduced administrative effort**

Automated resource adjustment reduces the need for frequent manual monitoring and configuration.

SGOU

3.3.6 Quotas and Limits

Auto scaling allows cloud resources to increase or decrease according to changing workload conditions. However, resources cannot be increased without any restriction. Cloud environments need mechanisms to control how many resources can be created or consumed by users and applications. Quotas and limits provide this control.

Consider an organization that uses cloud resources for several applications. If every user could create an unlimited number of virtual machines, storage volumes, or database instances, resources could be consumed excessively. This could affect availability, increase costs, and interfere with the resource requirements of other applications. Quotas and limits help prevent such situations.

3.3.6.1 Resource Quotas

A resource quota is a restriction placed on the amount of a particular cloud resource that a user, account, project, or organization can use.

For example, an organization may define a quota allowing a project to use a maximum of 20 virtual machines. If a user attempts to create the 21st virtual machine, the request may be rejected unless the quota is increased.

Quotas can be defined for different types of resources, such as:

- Number of virtual machines
- Number of virtual CPUs
- Amount of storage
- Number of database instances
- Number of IP addresses
- Number of network resources

3.3.6.2 Service Limits

A service limit is a restriction associated with the capacity or usage of a particular cloud service. These limits are generally established by the cloud service provider to maintain the performance, reliability, and stability of its services.

Service limits may apply to:

- Number of resource instances
- Storage capacity
- Network connections
- API requests
- Database connections

SGOU

- Requests within a specific period
- Capacity of individual resources

For example, a research application may send a large number of requests to a cloud-based data processing service. Even if the project has sufficient computing resources, the service may restrict requests when a permitted request rate is exceeded.

Service limits can vary depending on the cloud service, resource type, account, region, or service configuration.

3.3.7 Tagging

Quotas and limits help control how much of a cloud resource can be used. However, organizations may have hundreds or thousands of cloud resources running at the same time. Managing these resources becomes difficult if there is no simple way to identify which resource belongs to which application, project, department, or purpose.

Tagging provides a simple way to organize and identify cloud resources. A tag is a key-value pair attached to a cloud resource to provide additional information about it.

For example, a university may use cloud resources for teaching, research, administration, and student projects. Tags can be used to identify the purpose and ownership of these resources.

A tag may look like:

Project → DataAnalytics

Department → ComputerScience

Environment → Development

Owner → ResearchTeam

The resource itself does not change when a tag is added. Instead, the tag provides information that helps users and administrators identify and manage the resource.

Tagging makes cloud resource management more organized and efficient by providing meaningful information about each resource. It helps identify resources based on their project, owner, department, environment, or purpose, making it easier to search, monitor, and manage them. Tags can also support cost tracking, accountability, and automation by allowing resources to be grouped and managed according to predefined categories. Thus, effective tagging improves visibility and control over cloud resources, particularly when an organization manages a large and complex cloud environment.

Benefits of Tagging

1. Easy Resource Identification

Tags help users quickly identify the purpose, ownership, or project associated with a resource.

SGOU

2. Better Resource Organization

Resources can be grouped according to projects, departments, environments, or other categories.

3. Cost Tracking

Tags can help organizations associate resource usage with particular projects or departments. This supports better analysis of cloud expenditure.

4. Resource Management

Tags make it easier to locate resources that need maintenance, modification, or removal.

5. Automation

Cloud management tools can use tags to perform automated actions. For example, resources tagged as:

Environment → Testing

could be scheduled for shutdown outside working hours.

6. Improved Accountability

Tags can identify the person, team, department, or project responsible for a resource.

Recap

- Cloud resource management ensures efficient use of computing, storage, network, database, and security resources.
- Compute resources provide processing power for applications.
- Storage resources store files, applications, and data.
- Network resources provide communication and connectivity.
- Database resources store and manage application data.
- Security resources protect data and control access to cloud resources.
- Resource provisioning creates, configures, and allocates resources when required.
- Provisioning can be manual, automated, or dynamic.
- Resource de-provisioning releases or removes resources that are no longer required.
- De-provisioning helps reduce resource wastage and unnecessary costs.
- Static resource management maintains a fixed amount of resources.

SGOU

- Static management is suitable for stable and predictable workloads.
- Dynamic resource management adjusts resources according to workload changes.
- Dynamic management adds resources when demand increases and releases them when demand decreases.
- Policy-based resource management uses predefined rules to manage resources.
- Policies can be based on CPU, memory, users, traffic, response time, or cost.
- Auto scaling automatically adjusts resources according to workload or demand.
- Horizontal scaling adds or removes resource instances.
- Scale out means adding instances.
- Scale in means removing instances.
- Vertical scaling changes the capacity of an existing resource.
- Scale up means increasing resource capacity.
- Scale down means decreasing resource capacity.
- Auto scaling improves performance and resource utilization.
- Auto scaling can reduce unnecessary cloud costs.
- Resource quotas restrict the amount of resources a user or project can consume.
- Service limits restrict the capacity or usage of a particular cloud service.
- Quotas and limits help control resource consumption and cloud expenditure.
- Quotas and limits support fair resource allocation and service stability.
- Tagging assigns descriptive information to cloud resources using key-value pairs.
- Tags can identify projects, owners, departments, environments, and purposes.
- Tagging makes resources easier to search, organize, and manage.
- Tagging supports cost tracking, accountability, and automation.
- Effective resource management improves performance, scalability, availability, resource utilization, and cost efficiency.

SGOU

Objective Type Questions

1. What is the process of allocating, monitoring, and controlling cloud resources called?
2. What type of cloud resource provides processing power?
3. What type of cloud resource is used to store data and files?
4. What type of cloud resource provides communication between cloud components?
5. What type of cloud resource is used to store and manage application data?
6. What type of cloud resource protects cloud applications, data, and infrastructure?
7. What is a software-based computer running on a physical server called?
8. What technology packages an application with its libraries and dependencies?
9. What computing model allows users to execute functions without managing the underlying servers?
10. What component distributes incoming network requests among multiple servers?
11. What is the process of creating and allocating cloud resources called?
12. What is the process of releasing or removing cloud resources when they are no longer required called?
13. Which resource management model allocates a fixed amount of resources in advance?
14. Which resource management model adjusts resources according to workload changes?
15. Which resource management model manages resources using predefined rules?
16. What technology automatically increases or decreases cloud resources according to workload?
17. Which type of scaling changes the number of resource instances?
18. Which type of scaling changes the capacity of an existing resource?
19. What is the addition of resource instances during increased demand called?
20. What is the removal of resource instances when demand decreases called?
21. What is the increase in the capacity of an existing resource called?

SGOU

22. What is the decrease in the capacity of an existing resource called?
23. What mechanism restricts the amount of a cloud resource that a user, project, or organization can consume?
24. What restriction is imposed on the usage or capacity of a particular cloud service?
25. What is a key-value pair attached to a cloud resource called?

Answers to Objective Type Questions

1. Resource management
2. Compute resource
3. Storage resource
4. Network resource
5. Database resource
6. Security resource
7. Virtual machine
8. Container
9. Serverless computing
10. Load balancer
11. Provisioning
12. De-provisioning
13. Static resource management
14. Dynamic resource management
15. Policy-based resource management
16. Auto scaling
17. Horizontal scaling
18. Vertical scaling
19. Scale out
20. Scale in
21. Scale up

SGOU

22. Scale down
23. Resource quota
24. Service limit
25. Tag

Assignments

1. Explain the different types of cloud resources and describe the role of compute, storage, network, database, and security resources in a cloud environment.
2. Explain resource provisioning and de-provisioning. Describe the major steps involved in each process with a suitable example.
3. Explain auto scaling in cloud computing. Describe horizontal scaling and vertical scaling, including scale-out, scale-in, scale-up, and scale-down operations.
4. A college admission portal experiences a sudden increase in users during the final hours of admission. Analyze how resource provisioning, auto scaling, and de-provisioning can be used to maintain application performance while avoiding unnecessary resource usage.
5. A food delivery application experiences moderate demand during normal hours, high demand during lunch and dinner, and varying demand on weekends. Analyze which resource management model would be most appropriate for the application. Justify your choice based on workload variation, resource utilization, and cost.

SGOU

Reference

1. Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud computing: Concepts, technology & architecture*. Prentice Hall.
2. Manvi, S. S., & Shyam, G. K. (2021). *Cloud computing: Concepts and technologies*. CRC Press. This book covers cloud architectures, virtualization, workload management, resource provisioning, resource allocation, and resource adaptation.
3. Patni, S., Saxena, D., & Singh, A. K. (2025). *Resource management in cloud computing: Concepts and implementation*. Springer. This is particularly

relevant to your unit because it focuses directly on cloud resource management, including resource optimization and scaling.

4. Shyam, G. K., & Bharti, N. (2025). *Cloud computing: Concepts and technologies* (2nd ed.). CRC Press. The book includes dedicated coverage of capacity planning, auto-scaling, resource management, cloud security, and cloud platforms.

Suggested Reading

1. Amazon Web Services. (n.d.). *AWS Well-Architected Framework: Reliability—Service quotas and constraints*. AWS Well-Architected Framework
2. Qu, C., Calheiros, R. N., & Buyya, R. (2018). Auto-scaling web applications in clouds: A taxonomy and survey. *ACM Computing Surveys, 51*(4), 1–33.
3. Mampage, A., Karunasekera, S., & Buyya, R. (2021). A holistic view on resource management in serverless computing environments: Taxonomy and future directions. *ACM Computing Surveys*.

SGOU

Unit 4
Ensuring Performance

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the role of load balancing in improving cloud application performance
- apply suitable load-balancing techniques to distribute workloads effectively
- analyze load distribution to identify performance issues
- explain the importance of compliance and auditability in cloud environments
- evaluate cloud activities using compliance and audit records

Prerequisite

Cloud applications are expected to serve users efficiently even when the number of requests changes continuously. As workloads increase, some servers may become overloaded while others remain underutilized, affecting response time and overall application performance. At the same time, cloud activities must follow organizational policies, legal requirements, and applicable standards, with reliable records available to verify what happened.

Understanding how workloads are distributed, how performance is analyzed, and how compliance and auditability are maintained is therefore essential for managing cloud applications effectively. These concepts help you understand not only whether a cloud system is performing well, but also whether its activities can be properly controlled, traced, and evaluated.

With this foundation, let us explore Performance, Compliance and Auditability and understand how load balancing, compliance controls, and audit records contribute to effective cloud management.

Keywords

Load balancing, performance monitoring, compliance, auditability, workload distribution, audit trails, compliance controls, cloud resources

SOU



Discussion

3.4.1 Introduction to Performance, Compliance and Auditability

Cloud applications are expected to provide services efficiently even when the number of users and requests changes continuously. A sudden increase in workload can place excessive demand on some computing resources while other available resources remain underutilized. This can lead to increased response time, reduced throughput, and poor application performance. Effective management of cloud resources is therefore essential for maintaining satisfactory service quality.

Load balancing is one of the important techniques used to distribute incoming workloads among multiple available resources. Instead of allowing a single server to handle all requests, a load balancer distributes requests according to a selected technique. This helps in utilizing resources more effectively and can improve the performance of cloud applications.

Performance alone, however, is not sufficient for effective cloud management. Cloud resources and activities must also comply with applicable laws, regulations, standards, organizational policies, and contractual requirements. Compliance ensures that required controls and practices are followed. Auditability provides evidence about cloud activities so that they can be traced, reviewed, and evaluated.

This section examines the role of load balancing in cloud application performance, different techniques for distributing workloads, methods for analyzing load distribution, and the importance of compliance and auditability. It also explains how compliance requirements and audit records can be used to evaluate cloud activities.

3.4.2 Load Balancing in Cloud Applications

Consider a cloud application that is accessed simultaneously by a large number of users. Suppose four application servers are available, but all incoming requests are directed to only one of them. The selected server may become overloaded while the remaining servers have considerable unused capacity.

A better approach is to distribute the incoming requests among the available servers. This process is known as load balancing.

Load balancing is the process of distributing incoming network requests or application workloads across multiple available computing resources. A device or software component called a load balancer receives requests from clients and determines which backend resource should handle each request.

The users generally interact with the application through the load balancer rather than selecting a particular backend server themselves. The load balancer acts as an intermediary between the clients and the available resources.

The objective of load balancing is not necessarily to distribute exactly the same number of requests to every server. The objective is to distribute workloads effectively

SGOU

according to factors such as server capacity, current workload, and the characteristics of the application.

3.4.2.1 Role of Load Balancing in Performance

Application performance can be evaluated using several indicators. Important indicators include response time, throughput, resource utilization, and error rate.

Response time is the time taken by the system to respond to a request. If a server becomes overloaded, response time may increase.

Throughput refers to the amount of work completed within a given period. A system capable of processing more requests successfully in the same period generally has higher throughput.

Resource utilization indicates how much of a resource's available capacity is being used. CPU and memory utilization are common examples.

Error rate indicates the proportion of requests that fail.

Consider the following situation:

Server	CPU Utilization
A	90%
B	40%
C	35%

Server A is experiencing substantially higher CPU utilization than Servers B and C. If new requests continue to be directed to Server A, its response time may increase further.

A suitable load-balancing mechanism can distribute subsequent requests to other available resources, reducing unnecessary workload concentration.

However, load balancing cannot solve every performance problem. An application may experience poor performance because of inefficient program logic, a slow database, insufficient network capacity, or another resource bottleneck. Load balancing addresses the distribution of workloads; it does not automatically eliminate every underlying performance problem.

Think about it:

Three servers receive almost the same number of requests, but one server has 90% CPU utilization while the other two have less than 40%. Does this necessarily mean that the load is balanced?

The answer requires examining not only request counts but also the nature of the requests, server capacity, and other performance indicators.

The next step is to understand how a load balancer decides where to send each request.

SGOU

3.4.3 Load-Balancing Techniques

A load balancer needs a method for selecting an appropriate backend resource for each incoming request. Different load-balancing techniques use different criteria for making this decision.

Some techniques follow a fixed distribution pattern, while others consider the capacity or current workload of the available servers.

3.4.3.1 Round-Robin

Round-robin is a simple load-balancing technique in which requests are distributed sequentially among the available servers.

Suppose three servers, A, B, and C, are available. The requests can be distributed as follows:

Request 1 → A
Request 2 → B
Request 3 → C
Request 4 → A
Request 5 → B
Request 6 → C

The sequence continues repeatedly.

Round-robin works well when the servers have similar capacities and the requests require approximately similar amounts of processing.

For example, three identical servers processing similar web requests can use round-robin effectively.

The main limitation is that ordinary round-robin does not consider the current workload of a server. If Server A is already processing several long-running requests, another request may still be sent to A when its turn arrives.

Therefore, round-robin is most suitable when the resources and workload are relatively uniform.

3.4.3.2 Weighted Round-Robin

Weighted round-robin extends round-robin by assigning different weights to servers.

A weight represents the relative capacity or preference assigned to a server. A server with a higher weight receives a larger share of requests.

SGOU

Consider:

Server	Weight
A	3
B	2
C	1

Server A is assigned the highest weight and therefore receives a larger share of the workload than B and C.

This is useful when servers have different capacities.

For example, suppose Server A has more CPU and memory than Servers B and C. Treating all three servers equally may not use their capacities efficiently. Weighted round-robin allows the more powerful server to handle a larger proportion of requests.

The weights should be selected carefully. A high weight does not guarantee good performance if the workload changes significantly or if the server has another bottleneck.

3.4.3.3 Least-Connections

The least-connections technique directs a new connection to the server that currently has the fewest active connections.

Consider:

Server	Active Connections
A	20
B	8
C	14

A new connection would normally be directed to Server B because it has the fewest active connections.

This approach is useful when connections have different durations. A server handling several long-running connections may continue to be busy even though it has received fewer new requests.

Unlike round-robin, least-connections consider the current condition of the servers.

However, the number of active connections is not always a perfect measure of workload. A small number of computationally intensive connections may consume more resources than many simple connections.

Therefore, the suitability of least-connections depends on whether active connection count provides a useful indication of workload for the particular application.

SGOU

3.4.3.4 Choosing a Suitable Load-Balancing Technique

The selection of a load-balancing technique should be based on the characteristics of the workload and the available resources.

Table 3.4.1 Comparison of Load Balancing Techniques and their Suitability

Technique	Basis of Distribution	Suitable Situation
Round-robin	Sequential order	Similar servers and similar workloads
Weighted round-robin	Relative server capacity	Servers with different capacities
Least-connections	Number of active connections	Workloads with varying connection durations

Consider the following situations:

Situation 1: Three identical servers process similar, short-lived requests.

Round-robin can be a suitable choice.

Situation 2: Three servers have different processing capacities.

Weighted round-robin can provide a more appropriate distribution.

Situation 3: Connections remain active for different lengths of time.

Least-connections may provide a better distribution than simple round-robin.

The correct technique therefore depends on the actual characteristics of the cloud application.

Activity 1: Choose the Technique

A cloud application uses four identical servers. User connections remain active for significantly different periods.

1. Which technique would be more suitable: round-robin or least-connections?
2. Why might simple round-robin result in uneven active workloads?
3. Would the choice change if one server had twice the capacity of the others?

The selection of a technique is only the beginning. The next step is to examine whether the chosen technique is actually distributing workloads effectively.

3.4.4 Applying and Analyzing Load Distribution

After a load-balancing technique is applied, the resulting workload distribution should be monitored and analyzed.

Suppose an application has four servers and the following distribution is observed:

SGOU

Server	Requests
A	250
B	245
C	255
D	250

The request counts are relatively close to one another.

Now consider another situation:

Server	Requests
A	450
B	100
C	120
D	330

The second distribution is clearly uneven.

An uneven distribution may indicate an unsuitable load-balancing technique, differences in server capacity, differences in workload characteristics, or another condition affecting request distribution.

3.4.4.1 Workload Distribution

Workload distribution describes how incoming work is divided among available resources.

The number of requests is one useful measure, but it should not be considered in isolation.

Consider:

Server	Requests	CPU Utilization
A	300	85%
B	290	35%
C	310	30%

The number of requests is approximately equal, but CPU utilization differs considerably.

Server A may be processing more computationally intensive requests, or it may have less processing capacity than the other servers.

This example demonstrates why workload analysis should consider several indicators.

Useful indicators include:

SGOU

- number of requests;
- response time;
- throughput;
- CPU utilization;
- memory utilization;
- error rate.

Comparing these indicators provides a more complete picture of how workloads are distributed.

3.4.4.2 Identifying Performance Issues

A performance issue may exist when one or more resources consistently experience excessive workload or when application performance deteriorates.

For example, consider:

Table 3.4.2 Comparison of Server Performance Metrics for Load Balancing

Server	Requests	CPU	Average Response Time
A	400	90%	2.8 s
B	390	40%	0.9 s
C	410	35%	0.8 s

The request counts are similar, but Server A has substantially higher CPU utilization and response time.

Several explanations are possible:

- Server A may have lower capacity.
- Server A may be processing more demanding requests.
- The selected load-balancing technique may not be appropriate.
- Another component of the application may be affecting Server A.
- The workload may not be homogeneous.

The data therefore indicates a performance issue, but additional investigation is necessary to determine its exact cause.

Activity 2: Analyze the Distribution

Examine the following data:

Server	Requests	CPU Utilization
A	500	92%
B	490	42%

SGOU

C	510	38%

Answer the following:

1. Are the request counts approximately balanced?
2. Is CPU utilization balanced?
3. Which server appears to be under greater pressure?
4. What additional information would help identify the reason?
5. Would changing the load-balancing technique necessarily solve the problem?

The activity emphasizes that performance analysis should be based on multiple measurements rather than a single value.

3.4.4.3 Improving Load Distribution

When an imbalance is identified, the underlying cause should first be examined.

If servers have similar capacities but connections have very different durations, a least-connections approach may be more suitable than simple round-robin. If servers have different capacities, weighted round-robin may provide a more appropriate distribution. If request counts appear balanced but response time remains high, the cause may lie elsewhere in the application. For example, a database or another shared resource may be creating a bottleneck.

Therefore, the process of performance analysis can be represented as:

Measure → Analyze → Identify the problem → Select an appropriate improvement → Monitor again

The objective is not simply to make request numbers equal. The objective is to achieve effective workload distribution and acceptable application performance.

The discussion of performance focuses on how cloud resources are used. Cloud environments must also ensure that resources and activities are managed according to defined requirements. This leads to the concept of compliance.

3.4.5 Compliance in Cloud Environments

Cloud services may handle organizational data, applications, and computing resources. Their use may therefore be subject to various requirements.

Compliance refers to conforming to applicable laws, regulations, standards, contractual requirements, and organizational policies.

For example, an organization may establish a policy stating that only authorized personnel can modify production resources. It may also have requirements concerning data protection, access control, record retention, and monitoring.

Compliance ensures that such requirements are translated into appropriate controls

SGOU

and practices.

Cloud environments are dynamic. Resources can be created, modified, accessed, and removed frequently. Users may access services from different locations, and automated processes may perform activities without direct human interaction.

Without appropriate controls, it may become difficult to ensure that cloud activities follow organizational and external requirements.

Compliance is important because it helps organizations:

- protect sensitive information;
- control access to cloud resources;
- maintain required records;
- enforce organizational policies;
- demonstrate adherence to applicable requirements;
- reduce the risk of unauthorized or inappropriate activities.

Consider the requirement:

Only authorized administrators may modify production resources.

This requirement should be supported by appropriate access controls. It should also be possible to determine whether production modifications actually occurred and who performed them.

This creates a connection between compliance and auditability.

3.4.5.1 Compliance Requirements and Controls

A compliance requirement specifies what must be achieved or followed.

A control is a measure implemented to satisfy or support that requirement.

Compliance requirements can be addressed through appropriate controls. If only authorized users can access a resource, access control can be implemented. To ensure that administrative activities are traceable, activity logging can be used. Required records can be preserved through retention controls, while important changes can be managed through change-management controls.

Compliance controls may be technical, procedural, or organizational. Technical controls can include access permissions, authentication mechanisms, logging, and configuration settings. Procedural controls can include approval processes and documented change procedures.

The effectiveness of a control must be evaluated using appropriate evidence. This is where auditability becomes important.

SGOU

3.4.6 Auditability in Cloud Environments

Suppose a production cloud resource is modified unexpectedly. Several questions may arise:

- Who performed the modification?
- When did it occur?
- What resource was affected?
- What action was performed?
- Was the action authorized?

Without suitable records, answering these questions may be difficult.

Auditability is the ability to trace, review, and verify activities using reliable records and evidence.

Auditability does not simply mean keeping large quantities of logs. The records should contain useful information, be appropriately protected, and be available for authorized examination.

3.4.6.1 Audit Trails and Audit Records

An audit trail is a chronological record of relevant activities.

A simplified audit record may contain:

Table 3.4.3 Example of an Audit Trail Record and its key fields

Field	Example
Time	10:30 AM
Identity	Admin-A
Activity	Configuration change
Resource	Production server
Result	Successful

Such a record provides information about what happened and provides a basis for further investigation.

Compare these two statements:

"A production server was modified."

"Admin-A modified the production server configuration at 10:30 AM, and the operation was successful."

The second statement provides greater traceability.

SGOU

Audit records may be used to examine activities such as access attempts, configuration changes, administrative operations, and other events relevant to the organization's requirements.

Auditability supports several important activities.

- Accountability: It helps associate activities with identifiable users, roles, or services.
- Compliance evaluation: It provides evidence that can be compared with compliance requirements.
- Investigation: It helps reconstruct events when an unusual or unauthorized activity is suspected.
- Troubleshooting: It can help identify changes that may have affected an application's operation.

Audit records must themselves be protected. If unauthorized individuals can modify or delete them, their value as evidence is reduced.

Access to audit information should therefore be restricted appropriately, and the records should be maintained according to applicable requirements.

The purpose of maintaining audit records becomes clearer when they are used together with compliance requirements.

Recap

- Load balancing distributes workloads across multiple resources.
- Load balancing helps improve resource utilization and application performance.
- Response time, throughput, resource utilization, and error rate are key performance indicators.
- Round robin distributes requests sequentially among servers.
- Weighted round robin considers the capacity of different servers.
- Least connections directs requests to the server with fewer active connections.
- The suitable load balancing technique depends on workload and server characteristics.
- Workload distribution should be monitored using multiple performance indicators.
- Compliance means following applicable laws, regulations, standards, and organizational policies.

SGOU

- Compliance requirements are supported through appropriate controls.
- Access control restricts resources to authorized users.
- Activity logging helps make cloud activities traceable.
- Auditability enables activities to be traced, reviewed, and verified.
- Audit trails provide chronological records of important activities.
- Audit records support accountability, compliance evaluation, investigation, and troubleshooting.

Objective Type Questions

1. What technique distributes incoming workloads across multiple servers?
2. What component receives requests and forwards them to backend resources?
3. What metric measures the time taken to respond to a request?
4. What metric measures the amount of work completed within a given period?
5. Which technique distributes requests sequentially among servers?
6. Which technique assigns different weights to servers?
7. Which technique directs requests to the server with the fewest active connections?
8. What term refers to the process of conforming to laws, regulations, standards, and policies?
9. What specifies what must be achieved or followed for compliance?
10. What is a measure implemented to satisfy a compliance requirement called?
11. What control restricts access to authorized users?
12. What control records administrative activities?
13. What enables cloud activities to be traced, reviewed, and verified?
14. What is a chronological record of relevant activities called?
15. What supports accountability and compliance evaluation through recorded evidence?

SGOU

Answers to Objective Type Questions

1. Load balancing
2. Load balancer
3. Response time
4. Throughput
5. Round robin
6. Weighted round robin
7. Least connections
8. Compliance
9. Requirement
10. Control
11. Access control
12. Logging
13. Auditability
14. Audit trail
15. Auditability

Assignments

1. Explain the concept of load balancing in cloud applications and describe the major load balancing techniques with suitable examples.
2. Explain the importance of compliance in cloud environments. Discuss compliance requirements, controls, and their role in effective cloud management.
3. Explain auditability in cloud environments. Describe audit trails and explain how audit records support accountability, compliance evaluation, investigation, and troubleshooting.
4. A cloud application uses three servers. Server A has 90% CPU utilization, while Servers B and C have 40% and 35% respectively, although all three receive a similar number of requests. Analyze the situation and identify possible reasons for the performance imbalance. Suggest suitable measures to improve the situation.

SGOU

5. A company requires that only authorized administrators can modify production resources and that every modification must be traceable. Analyze the requirements and identify the appropriate controls and audit evidence needed to demonstrate compliance.

Reference

1. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2011). *Cloud computing: Principles and paradigms*. Wiley.
2. Sehgal, N. K., Bhatt, P. C. P., & Acken, J. M. (2020). *Cloud computing with security*. Springer.
3. Greer, M. B., Jr., & Jackson, K. L. (2016). *Practical cloud security*. CRC Press.
4. Majumdar, S., Madi, T., Wang, Y., et al. (2019). *Cloud security auditing*. Springer.

Suggested Reading

1. Buyya, R., Broberg, J., & Goscinski, A. M. (2011). *Cloud computing: Principles and paradigms*. Wiley.
2. Majumdar, S., Madi, T., Wang, Y., et al. (2019). *Cloud security auditing*. Springer.
3. Vacca, J. R. (Ed.). (2016). *Cloud computing security*. CRC Press.
4. Sun, X., Chao, H. C., You, X., & Bertino, E. (Eds.). (2017). *Cloud computing and security*. Springer.

SGOU



Cloud Migration and Associated Technologies

SGOU

Unit 1
Introduction to Cloud Migration

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the concept, purpose, and importance of cloud migration
- describe and differentiate major cloud migration strategies such as Lift and Shift, Re-platforming, and Refactoring
- analyze the major benefits and challenges associated with migrating applications and data to the cloud
- evaluate cloud migration requirements and identify suitable approaches for reducing migration risks and improving cloud adoption.

Prerequisite

Organizations traditionally manage applications, data, databases, servers, storage, backup, and other IT resources using their own physical infrastructure. This approach requires investment in hardware, electricity, cooling, maintenance, backup systems, and IT management. For example, a college maintaining its student management system on its own physical server must purchase and maintain the required server and storage capacity. When the number of users or data increases, additional infrastructure may be required, making the traditional approach less flexible and scalable.

These limitations create the need to study Cloud Migration. Instead of continuously purchasing and maintaining physical infrastructure, organizations can move their applications and data to cloud environments and obtain resources according to their requirements. For example, an online shopping company may experience very high traffic during a festival sale. A traditional infrastructure may require additional servers to handle the temporary increase, whereas cloud resources can be increased during high demand and reduced when demand falls. Thus, cloud migration helps address limitations related to scalability, infrastructure maintenance, availability, backup, and resource utilization.

Keywords

Availability, Disaster Recovery, Data Security, Vendor Lock-in, Network Dependency, Latency, Legacy Systems, Data Migration, Cost Management

SGOU

Discussion

4.1.1 Overview of Cloud Migration

Cloud migration is the process of moving data, applications, software, and other IT resources from an organization's own servers or traditional data centers to a cloud environment. Cloud platforms such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud provide computing resources through the Internet. In simple terms, cloud migration means moving from managing IT resources on physical servers to using computing resources provided by a cloud service provider. It helps organizations make their systems more flexible, scalable, accessible, and easier to manage.

Cloud migration can involve moving different types of resources, such as data, databases, applications, servers, storage, and IT infrastructure. For example, a college may store student details, examination results, and learning materials on its own servers. If the college moves these applications and data to the cloud, students and teachers can access the required services through the Internet without depending completely on the college's physical servers. Similarly, an online shopping company may move its website and customer database to the cloud so that it can handle a large number of customers during festival sales or other periods of high demand.

Cloud migration is also an important step in modernizing IT systems. After migration, organizations can use cloud features such as on-demand resources, automatic scaling, cloud storage, backup, disaster recovery, real-time processing, and data analytics. For example, an online education platform may need more computing resources when thousands of students access online classes at the same time. Cloud resources can be increased during periods of high usage and reduced when the demand becomes low.

Cloud migration can also reduce the need for organizations to purchase and maintain large amounts of physical hardware. Instead of buying a new server whenever the workload increases, an organization can obtain additional computing resources from the cloud when required. Many cloud services also follow a pay-as-you-go model, where organizations pay according to the resources they use.

Another important feature of cloud migration is improved accessibility and availability. Applications and data hosted in the cloud can generally be accessed from different locations through the Internet, subject to appropriate security controls. Cloud providers also offer services for backup, monitoring, high availability, and disaster recovery, which can help organizations protect their data and maintain services during system failures.

Cloud migration also enables organizations to use modern technologies such as Artificial Intelligence (AI), Machine Learning (ML), Big Data Analytics, and Internet of Things (IoT) through cloud-based services. For example, a business can use cloud-based AI services to analyze customer information without having to build and maintain its own AI infrastructure.

SGOU

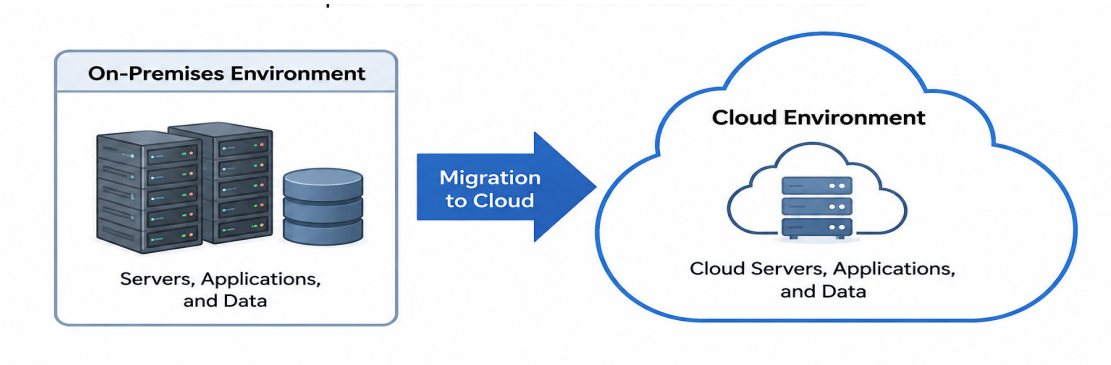


Fig 4.1.1 Cloud migration

4.1.2 Types of Cloud Migration Strategies

When an organization decides to move its applications and data to the cloud, it does not always move everything in the same way. The method used to move an existing application to the cloud is called a cloud migration strategy. The choice of strategy depends on the type of application, cost, time, technical requirements, and how much the organization wants to modify the existing system.

Three common cloud migration strategies are Lift and Shift, Re-platforming, and Refactoring.

1. Lift and Shift

Lift and Shift is the simplest cloud migration strategy. In this method, an existing application is moved from the organization's local server to the cloud with little or no modification to the application. The application is basically transferred from one environment to another.

For example, suppose a college has a student management application running on a physical server. If the college moves the same application to a cloud-based virtual server without changing its program code, it is using the Lift and Shift strategy.

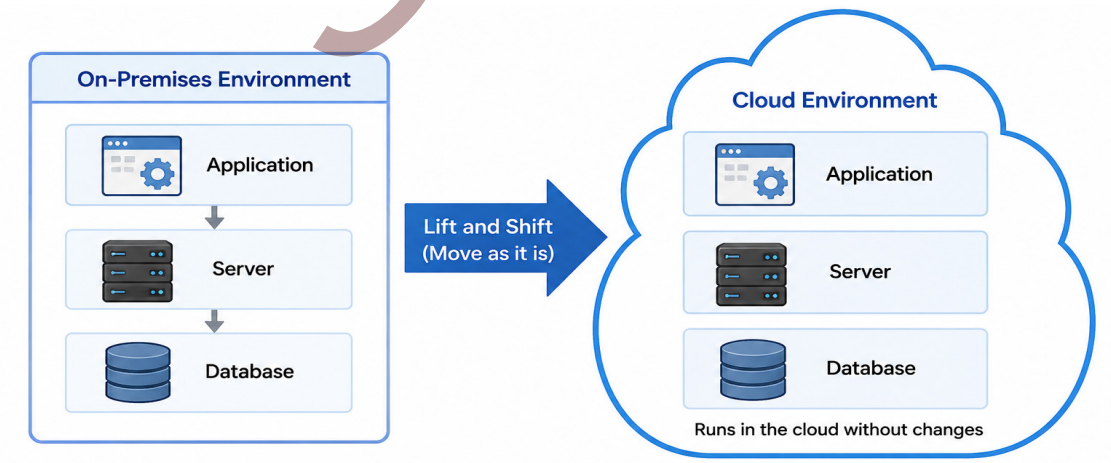


Fig 4.1.2 Lift and shift

SGOU

Example: Local Server → Cloud Server

The main advantage of Lift and Shift is that migration can be completed relatively quickly because developers do not need to redesign the application. It is useful when an organization wants to move its systems to the cloud quickly or when an old application is difficult to modify.

However, the application may not take full advantage of cloud features because it is moved almost in its original form. Therefore, although the application runs in the cloud, it may not achieve the maximum benefits of cloud computing.

2. Re-platforming

Re-platforming means moving an application to the cloud while making some changes or improvements to take advantage of selected cloud services. The basic structure of the application is retained, but certain components are modified.

For example, a company may move its application from an on-premises server to the cloud but replace its locally managed database with a cloud-managed database service. The application itself remains mostly unchanged, but it can benefit from cloud-based database management.

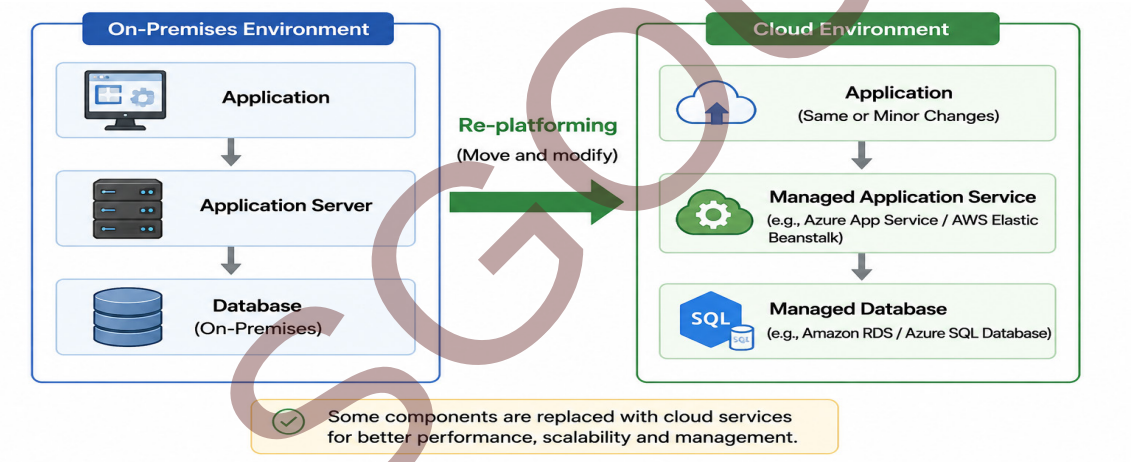


Fig 4.1.3 Replatforming

Example: Existing Application → Minor Modification → Cloud Platform

Re-platforming provides a balance between Lift and Shift and Refactoring. It requires more work than Lift and Shift but usually provides better cloud benefits. It can improve performance, management, scalability, and reliability.

However, some development and testing are required because the application needs to work correctly with the new cloud services.

3. Refactoring

Refactoring is a more advanced migration strategy in which an application is significantly modified or redesigned to take full advantage of cloud computing. Instead of simply moving the existing application, developers change its architecture to make

SGOU

it more suitable for the cloud.

For example, an online shopping application may originally be developed as one large application. During refactoring, developers can divide it into smaller independent services such as Customer Service, Product Service, Order Service, and Payment Service. These services can then be deployed and scaled independently in the cloud.

Example: Traditional Application → Redesign → Cloud-Native Application

Refactoring can provide better scalability, flexibility, performance, and maintainability. It is especially useful when an organization wants to modernize an old application and make full use of cloud-native technologies.

However, refactoring requires more time, cost, technical knowledge, development, and testing compared with the other two strategies.

Table 4.1.1 Comparison of Lift and Shift,Re-platforming and Refactoring

Strategy	Main Idea	Changes Required	Effort
Lift and Shift	Move as it is	Very little or none	Low
Re-platforming	Move and make some improvements	Moderate	Medium
Refactoring	Redesign for the cloud	Major changes	High

4.1.3 Benefits and Challenges of Migrating to the Cloud

Cloud migration provides organizations with the ability to move applications, data, databases, IT infrastructure, and other computing resources from traditional on-premises systems to a cloud environment. It can improve flexibility, scalability, availability, and operational efficiency. However, migration also involves technical, financial, security, and organizational challenges.

4.1.3.1 Benefits of Cloud Migration

1. Cost Reduction

One of the major benefits of cloud migration is the reduction in infrastructure and maintenance costs. In a traditional data center, an organization must purchase servers, storage devices, networking equipment, backup systems, and other hardware. It also needs to spend money on electricity, cooling, maintenance, and physical security. With cloud computing, organizations can use resources provided by cloud service providers and generally pay according to their usage.

Example:

A small company that needs additional storage can obtain cloud storage instead of purchasing and maintaining a new physical storage server.

Benefits include:

- Reduced hardware investment.

SGOU

- Lower maintenance costs.
- Reduced electricity and cooling expenses.
- Pay-as-you-use pricing in many cloud services.
- Better utilization of computing resources.

2. Scalability

Cloud platforms allow organizations to increase or decrease computing resources according to demand. In traditional systems, an organization may need to purchase additional servers when the number of users increases. This can take considerable time and money. Cloud resources can usually be provisioned much more quickly.

Example:

An online shopping website may receive very high traffic during a festival sale. Cloud resources can be increased during the sale and reduced after the demand returns to normal.

Types of scaling:

- **Vertical scaling:** Increasing the capacity of an existing server.
- **Horizontal scaling:** Adding more servers or computing instances.

3. Flexibility and Accessibility

Cloud applications and data can be accessed from different locations through an internet connection, subject to the organization's security policies.

This supports:

- Remote work.
- Online learning.
- Distributed teams.
- Collaboration between different offices.
- Access to applications from multiple devices.

Example:

Employees working from different cities can access the same cloud-based business application without connecting directly to the organization's physical office network.

4. Improved Availability

Cloud providers generally operate large-scale infrastructure with mechanisms such as redundancy, load balancing, and multiple availability zones or data centers.

SGOU

This can help applications remain available even if an individual server fails.

Example:

If one physical server supporting an application fails, the workload may be transferred to another available resource, depending on the cloud architecture.

Thus, cloud migration can help organizations achieve higher availability and reduced service interruption when the system is properly designed.

5. Disaster Recovery and Backup

Cloud platforms can provide backup and disaster recovery capabilities.

Traditional backup systems may require organizations to maintain separate physical locations and hardware. Cloud-based solutions can store backup copies in geographically separated locations.

Example:

If an organization's local server is damaged because of fire or flooding, a backup stored in the cloud can help restore important data and services.

Advantages:

- Automated backup options.
- Remote storage of backup copies.
- Faster recovery.
- Improved business continuity.
- Protection against hardware failures.

6. Faster Deployment

Cloud environments allow computing resources and services to be provisioned quickly.

Developers can create virtual machines, databases, storage, and development environments without waiting for physical hardware to be purchased and installed.

Example:

A software development team can create a testing environment in the cloud within a short period instead of purchasing and configuring a new physical server.

This helps organizations:

- Develop applications faster.
- Test new ideas quickly.
- Release software more frequently.

SGOU

- Respond rapidly to business requirements.

7. Automatic Maintenance and Updates

Many cloud services are maintained by the cloud provider. Depending on the service model, the provider may handle tasks such as infrastructure maintenance, software updates, hardware replacement, and security patches.

This reduces the burden on an organization's internal IT team.

For example, when using a managed cloud database service, many infrastructure-level maintenance activities are handled by the provider.

8. Enhanced Collaboration

Cloud-based applications allow multiple authorized users to work with shared resources.

Example:

A team working on a project can store documents in a cloud-based collaboration platform and access updated versions from different locations.

This can improve:

- Team communication.
- Document sharing.
- Real-time collaboration.
- Productivity.
- Coordination between departments.

9. Better Resource Utilization

Traditional IT infrastructure may remain underutilized for long periods. For example, a server purchased for peak demand may use only a small portion of its capacity during normal periods.

Cloud environments allow resources to be dynamically allocated according to workload.

Therefore, organizations can make more efficient use of computing, storage, and networking resources.

10. Support for Innovation

Cloud platforms provide access to modern technologies such as:

- Artificial Intelligence (AI)
- Machine Learning (ML)

SGOU

- Big Data Analytics
- Internet of Things (IoT)
- Serverless computing
- Containers
- Data analytics

Organizations can use these services without building all the underlying infrastructure themselves.

Example:

A company can use a cloud-based machine learning service to develop a recommendation system without purchasing specialized hardware for the entire ML infrastructure.

4.1.3.2 Challenges of Cloud Migration

Although cloud migration provides many benefits, the migration process must be carefully planned.

1. Initial Migration Cost

Cloud migration does not always reduce costs immediately. Organizations may need to spend money on:

- Migration tools.
- Consultants.
- Employee training.
- Application modification.
- Data transfer.
- Testing.
- Temporary infrastructure.

Therefore, the organization should calculate both migration costs and long-term operational costs before migration.

2. Data Security and Privacy

Moving data from local systems to the cloud introduces security and privacy concerns.

Sensitive information such as customer records, financial data, and business information must be protected from unauthorized access.

Security risks may result from:

- Weak passwords.
- Improper access permissions.

SGOU

- Misconfigured cloud services.
- Insecure APIs.
- Data breaches.

Organizations should implement appropriate authentication, authorization, encryption, monitoring, and access-control mechanisms.

3. Compliance and Legal Requirements

Organizations operating in regulated industries may have to follow specific laws and standards regarding data storage and processing.

For example, an organization may need to know:

- Where its data is physically stored.
- Who can access the data.
- How long data is retained.
- How data is protected.
- Whether the cloud provider satisfies required compliance standards.

Therefore, compliance requirements must be considered before selecting a cloud provider and migration strategy.

4. Downtime During Migration

Poorly planned migration can interrupt business operations.

For example, moving a large database from an on-premises server to the cloud may require synchronization and careful transition planning.

Possible problems include:

- Application downtime.
- Data synchronization errors.
- Network interruptions.
- Service unavailability.

Organizations can reduce these risks through testing, phased migration, backup, and carefully planned cutover procedures.

5. Vendor Lock-In

Vendor lock-in occurs when an organization becomes highly dependent on one cloud provider and finds it difficult or expensive to move its applications and data to another provider.

SGOU

For example, an application may use proprietary cloud databases, APIs, or serverless services. Moving such an application to another platform may require significant redesign.

Ways to reduce vendor lock-in:

- Use standard technologies where practical.
- Use containers where appropriate.
- Maintain portable data formats.
- Develop a multi-cloud or hybrid-cloud strategy when justified.
- Document dependencies.

6. Lack of Skilled Professionals

Cloud environments require specialized knowledge.

Organizations may need employees with skills in:

- Cloud architecture.
- Cloud security.
- Networking.
- Virtualization.
- Containers.
- DevOps.
- Cloud databases.
- Monitoring and cost management.

If employees do not have sufficient skills, migration can become difficult and errors may occur.

Solution:

Organizations can provide cloud training, certification programs, workshops, and practical experience to employees.

7. Network Dependency

Cloud services usually depend heavily on network connectivity. If the internet connection is slow or unavailable, users may experience difficulties accessing cloud applications and data. This is particularly important for organizations that depend on cloud-based applications for critical operations.

Example:

SGOU

If a company's internet connection fails, employees may temporarily lose access to cloud-hosted applications even though the cloud provider's infrastructure is functioning normally.

8. Performance and Latency Issues

Applications may experience delays when data has to travel between users and cloud data centers. This is known as network latency.

Latency can be especially important for:

- Real-time applications.
- Online gaming.
- Financial systems.
- Video processing.
- Industrial control systems.

Organizations should select suitable cloud regions and design applications to minimize unnecessary network communication.

9. Integration with Legacy Systems

Many organizations still use older applications and databases. These legacy systems may not be designed to operate efficiently with modern cloud services. Integration can therefore become complex.

Example:

A company may move its customer management application to the cloud while keeping an old billing system on-premises. Connecting the two systems securely may require additional APIs, middleware, or application modifications.

10. Data Migration Complexity

Large organizations may have terabytes or petabytes of data. Moving such large volumes of information to the cloud can take considerable time and requires careful planning.

Problems may include:

- Data loss.
- Duplicate data.
- Incompatible formats.
- Data corruption.
- Slow transfer speeds.
- Synchronization problems.

SGOU

Before migration, organizations should identify, clean, classify, and prioritize their data.

11. Cost Management

Although cloud computing can reduce infrastructure costs, uncontrolled cloud usage can result in unexpectedly high bills. For example, unused virtual machines, excessive storage, unnecessary data transfers, or poorly configured services can increase costs.

Organizations therefore need:

- Cost monitoring.
- Resource optimization.
- Usage limits.
- Budget alerts.
- Regular cost reviews.

12. Organizational Resistance to Change

Cloud migration may change the way employees work and how IT resources are managed. Employees who are familiar with traditional infrastructure may initially resist the transition.

Organizations should therefore provide:

- Training.
- Clear communication.
- Technical support.
- Change-management programs.
- Proper documentation.

SGOU

Recap

- Cloud migration is the process of moving applications, data, databases, servers, storage, and other IT resources from traditional infrastructure to a cloud environment.
- Cloud migration helps organizations improve flexibility, scalability, accessibility, availability, and resource utilization.
- Lift and Shift moves an existing application to the cloud with little or no modification.
- Re-platforming involves making limited modifications to take advantage of selected cloud services.

- Refactoring involves significantly redesigning an application to make it cloud-native.
- Cloud migration can provide benefits such as cost reduction, scalability, accessibility, improved availability, backup, disaster recovery, faster deployment, and enhanced collaboration.
- Cloud migration also involves challenges such as initial migration cost, security and privacy concerns, compliance requirements, downtime, vendor lock-in, lack of skilled professionals, and network dependency.
- Performance and latency, legacy-system integration, data migration complexity, cost management, and organizational resistance are also important considerations.
- Proper planning, testing, training, security controls, monitoring, and risk management can help organizations achieve successful cloud migration.

Objective Type Questions

1. What is the process of moving IT resources from traditional infrastructure to a cloud environment called?
2. Which migration strategy moves an application with little or no modification?
3. Which migration strategy involves making some improvements to an existing application?
4. Which migration strategy significantly redesigns an application for the cloud?
5. What type of scaling increases the capacity of an existing server?
6. What type of scaling adds more servers or computing instances?
7. What refers to dependence on a particular cloud provider?
8. What term refers to delays caused by data travelling between users and cloud data centers?
9. What technology can help reduce dependency on a particular cloud platform?
10. What mechanism can help restore data and services after a major failure?

SGOU

Answers

1. Cloud Migration
2. Lift and Shift
3. Re-platforming
4. Refactoring
5. Vertical Scaling
6. Horizontal Scaling
7. Vendor Lock-in
8. Latency
9. Containers
10. Disaster Recovery

Assignments

1. Explain the concept of cloud migration. Discuss its importance and the major resources that can be migrated to the cloud with suitable examples.
2. Explain the three cloud migration strategies: Lift and Shift, Re-platforming, and Refactoring. Compare these strategies based on their main idea, changes required, and effort involved.
3. Explain the major benefits of cloud migration in detail. Discuss cost reduction, scalability, flexibility and accessibility, improved availability, disaster recovery and backup, faster deployment, automatic maintenance and updates, enhanced collaboration, better resource utilization, and support for innovation.
4. Explain the major challenges of cloud migration in detail. Discuss initial migration cost, data security and privacy, compliance and legal requirements, and downtime during migration.
5. Explain vendor lock-in, lack of skilled professionals, and network dependency as challenges of cloud migration. Discuss suitable ways to address these challenges.
6. Explain performance and latency issues, integration with legacy systems, and data migration complexity as challenges of cloud migration. Give suitable examples.
7. Explain cost management and organizational resistance to change as challenges of cloud migration. Discuss the measures that organizations can adopt to manage these challenges.

SGOU

8. Discuss the benefits and challenges of cloud migration. Explain why an organization should consider both the advantages and challenges before migrating to the cloud.
9. Compare Lift and Shift, Re-platforming, and Refactoring with suitable examples. Explain the advantages and limitations of each strategy.
10. Explain how cloud migration supports scalability, availability, disaster recovery, faster deployment, collaboration, resource utilization, and innovation.

Reference

1. Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). National Institute of Standards and Technology.
2. Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud computing: Concepts, technology & architecture*. Prentice Hall.
3. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2011). *Cloud computing: Principles and paradigms*. John Wiley & Sons.
4. Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—The business perspective. *Decision Support Systems,* 51(1), 176–189. https://doi.org/10.1016/j.dss.2010.12.006

Suggested Reading

1. Amazon Web Services. (n.d.). *AWS Prescriptive Guidance: Cloud migration strategies*. Amazon Web Services.
2. Microsoft. (n.d.). *Cloud Adoption Framework for Azure*. Microsoft.
3. Google Cloud. (n.d.). *Cloud migration*. Google Cloud.
4. National Institute of Standards and Technology. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). U.S. Department of Commerce.

SGOU

Unit 2
Cloud Migration Planning and Assessment

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the importance of cloud migration planning and assessment.
- describe the process of assessing existing infrastructure, workloads, resource utilization, dependencies, and migration readiness.
- analyze the costs, benefits, and risks associated with cloud migration.
- explain suitable migration approaches and tools based on organizational requirements.

Prerequisite

Organizations traditionally maintain their applications, databases, servers, storage, backup systems, and other IT resources within their own infrastructure. This requires investment in hardware, electricity, cooling, maintenance, backup, and technical support. For example, a college operating its Student Management System on physical servers must maintain the servers and increase infrastructure when storage or resource requirements grow. Such an environment can make expansion and resource management more difficult.

These limitations create the need to study cloud migration planning and assessment. Before moving systems to the cloud, an organization needs to identify its existing infrastructure and workloads, measure resource utilization, analyze dependencies, estimate costs and benefits, identify risks, and determine migration readiness. For example, a college planning to move its Student Management System to the cloud must first determine what needs to be migrated, what modifications are required, how much the migration may cost, and which migration approach is appropriate.

Keywords

Resource Utilization, Dependencies, Migration Readiness, Cost-Benefit Analysis, Risk Assessment, Migration Approaches, Migration Tools, Organizational Requirements

SGOU

Discussion

4.2.1 Introduction Cloud Migration Planning and Assessment

Imagine that a college has been using its own computer servers for many years. The college stores student information, examination results, attendance records, staff details, and other important data on these servers. The servers are kept in a room on the college campus. The college has to purchase new hardware when storage becomes insufficient, maintain the servers, provide electricity and cooling, and ensure that the data is backed up.

One day, the college decides to move its systems to the cloud. The management expects that cloud computing will reduce infrastructure maintenance, provide better scalability, and make the applications easier to access. But can the college simply take all its files and applications and upload them to the cloud? No.

Before migration, the college needs to understand what it currently has, what should be moved, how much the migration will cost, what problems might occur, and which migration method is suitable. This process is called cloud migration planning and assessment.

Cloud migration is therefore not simply a process of "moving data to the cloud." It is a carefully planned process in which an organization studies its existing systems and decides the best way to move them to a cloud environment.

The major activities involved are:

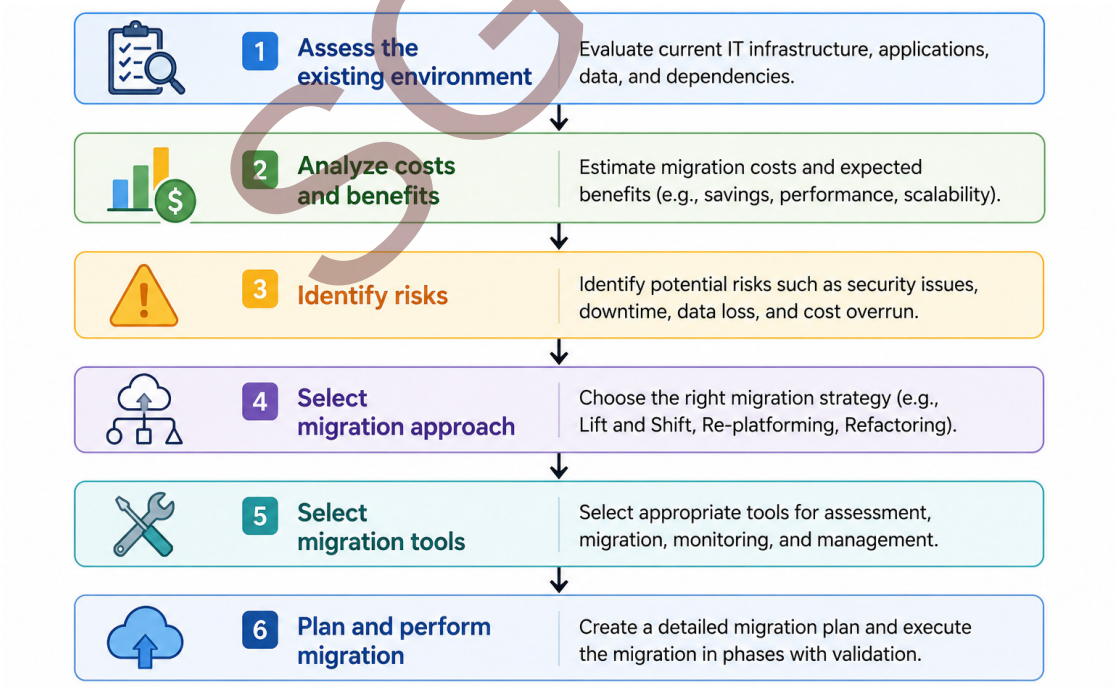


Fig 4.2.1 Cloud migration planning process

SGOU

4.2.2 Cloud Migration Planning and Assessment

What does "planning" mean?

Planning means deciding what needs to be done, how it will be done, who will do it, and when it will be done.

Before constructing a house, we normally prepare a plan. We decide the size of the house, the materials required, the budget, and the construction sequence. We do not start construction without knowing what we are going to build.Cloud migration is similar. Before moving an organization's IT systems to the cloud, we need to prepare a migration plan.

Assessment means examining and understanding the existing situation.

For example, before moving to a new house, we may check how many items we have, which items we want to keep, which items can be discarded, and which items need special care.

Similarly, before cloud migration, an organization examines:

- Servers
- Applications
- Databases
- Storage
- Networks
- Data
- Users
- Dependencies
- Security requirements
- Current costs

This information helps the organization make better migration decisions.

After understanding what planning and assessment mean, the next step is to examine the existing infrastructure and workloads in greater detail. This helps the organization determine what can be migrated, what needs modification, and what should be retained or removed.

4.2.3 Assessing Existing Infrastructure and Workloads

After completing the cloud migration planning stage, the next step is to assess the organization's existing infrastructure and workloads. This assessment helps identify what IT resources are currently available, how they are being used, and whether they are suitable for migration to the cloud.

SGOU

The objective is to collect accurate information about the current environment so that cloud resources can be selected correctly, migration risks can be reduced, and unnecessary applications can be avoided.

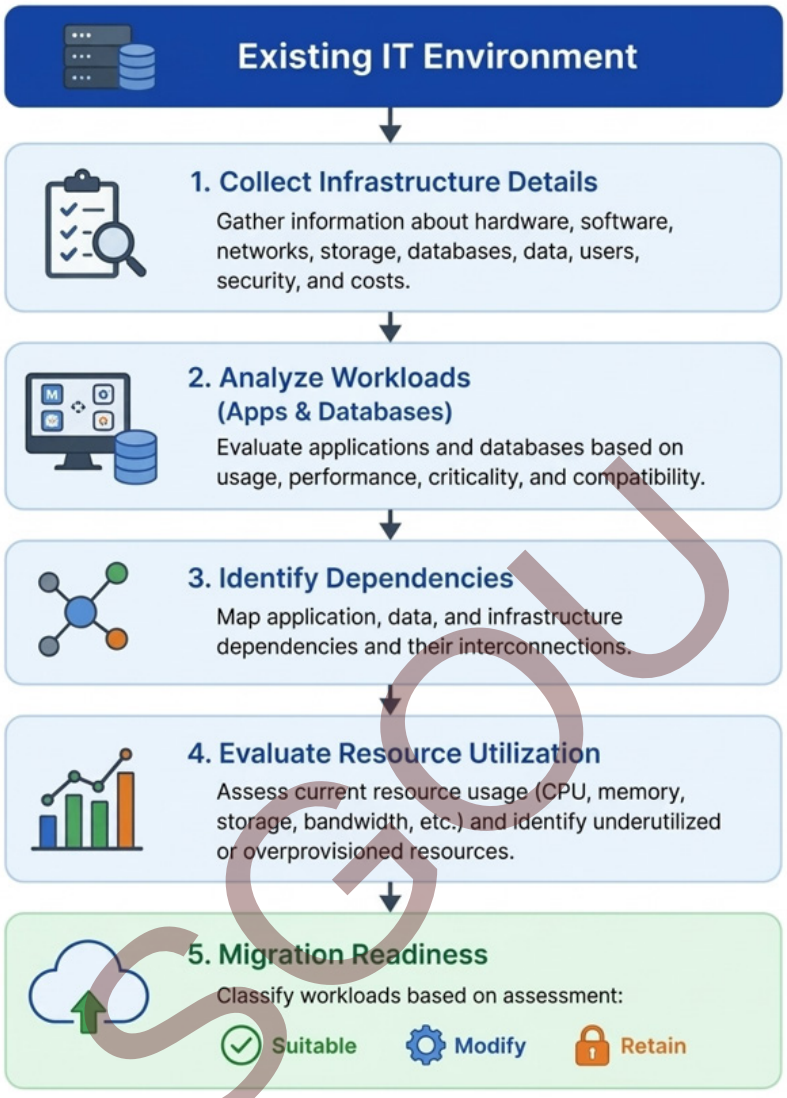


Fig 4.2.2 Steps involved in assessing existing infrastructure and workloads.

1. Understanding the Existing Infrastructure

The first task is to understand the organization's IT infrastructure. Infrastructure includes all the hardware and software resources that support business operations, such as servers, storage devices, networks, databases, and operating systems.

For example, a college may have three physical servers: one for the Student Management System, one for the Examination System, and another for the Library Management System. During assessment, each server is documented with its processor, memory, storage capacity, operating system, and current usage. This information helps the migration team understand the existing computing environment before moving it to the cloud.

SGOU

2. Identifying Workloads

A workload is any application, database, or service that uses computing resources. Every workload should be identified and documented because different workloads may require different migration strategies.

For example, an online shopping company may have separate workloads for product management, customer accounts, payment processing, and order tracking. Each workload has different performance and security requirements, so they cannot all be treated in the same way during migration.

3. Measuring Resource Utilization

After identifying infrastructure and workloads, the organization measures how efficiently its resources are being used. This prevents selecting cloud resources that are either too small or unnecessarily large.

Common utilization measurements include:

- CPU usage
- Memory (RAM) usage
- Storage usage
- Network bandwidth
- Number of active users

Example:

A server may have 64 GB RAM, but monitoring shows that it normally uses only 22 GB. Instead of choosing an expensive 64 GB cloud server, the organization may select a smaller cloud instance that better matches actual usage. This reduces cloud costs without affecting performance.

4. Analyzing Application Dependencies

Most applications do not work independently. They communicate with databases, authentication services, file storage, and other applications. These relationships are called dependencies.

For example, a Student Portal depends on both an application server and a student database.

If the database is migrated but the application server is not configured correctly, the Student Portal will fail to retrieve student information. Therefore, dependency analysis ensures that related systems are migrated in the correct order.

SGOU

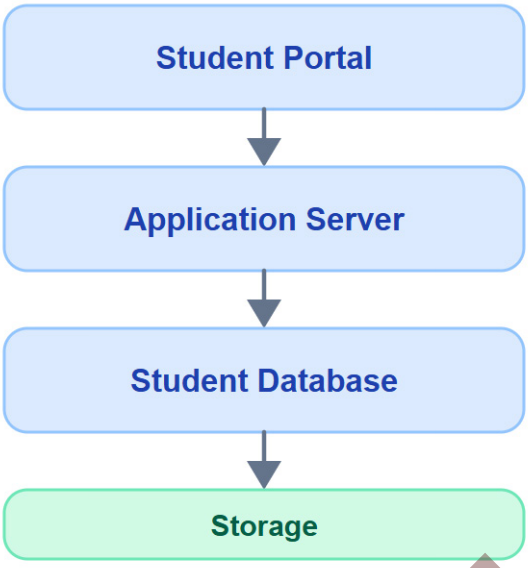


Fig 4.2.3 Dependency between the student portal and its supporting systems.

5. Classifying Workloads by Business Importance

Not every workload is equally important. Organizations classify workloads according to their impact on business operations. This helps determine migration priority.

Table 4.2.1 Migration priority Student Management System

Workload	Business Importance	Migration Priority
Student Management System	Very High	High
Examination System	High	High
Library System	Medium	Medium
Old Reporting System	Low	Low

For example, a college should migrate the Student Management System more carefully than an old reporting application because any failure could affect thousands of students.

6. Assessing Migration Readiness

The final step is to decide whether each workload is ready for cloud migration. Based on technical compatibility, business value, and dependencies, workloads can be categorized as:

Table 4.2.2 Application Migration Readiness Classification Table

Readiness Level	Meaning
Suitable	Can be migrated immediately
Needs Modification	Requires changes before migration
Retain	Should remain on-premises
Retire	No longer needed and can be removed

SGOU

For example, a modern web application may be suitable for immediate migration, while a legacy payroll system may require modification before moving to the cloud.

Assessing existing infrastructure and workloads is a detailed study of the organization's current IT environment. It involves identifying infrastructure components, analyzing workloads, measuring resource utilization, discovering application dependencies, evaluating business importance, and determining migration readiness. The results of this assessment provide the foundation for cost-benefit analysis, risk assessment, and the selection of an appropriate cloud migration strategy.

4.2.4 Cost-Benefit Analysis and Risk Assessment

After assessing the existing infrastructure and workloads, the organization needs to answer two important questions:

"Is migrating to the cloud worth the cost?"

and

"What problems might occur during or after migration?"

Cost-benefit analysis helps the organization compare the expected costs with the expected benefits of cloud migration. Risk assessment helps identify possible problems and prepare ways to reduce their impact. Both are important before making a final migration decision.

4.2.4.1 Cost-Benefit Analysis

Cost-benefit analysis is the process of comparing the costs involved in cloud migration with the benefits that the organization expects to receive.

For example, imagine a college that currently maintains its own servers. The college has to purchase servers, provide electricity and cooling, maintain hardware, perform backups, and employ technical staff. If it moves its systems to the cloud, some of these costs may be reduced, but new costs such as cloud computing, storage, network usage, and support will arise.

Therefore, the organization should not assume that cloud migration is automatically cheaper. It should compare the current cost of maintaining the existing environment with the estimated cost of operating the cloud environment.

Costs Involved in Cloud Migration

Cloud migration may involve costs at different stages.

During the migration process, the organization may need to spend money on migration tools, consultants, employee training, application modification, data transfer, testing, and temporary infrastructure.

After migration, regular costs may include computing resources, storage, databases, network usage, backup, security, monitoring, and technical support.

SGOU

For example, a company may need to modify an old application before moving it to the cloud. Although the cloud may reduce hardware maintenance costs in the long term, the application modification creates an initial migration expense.

Benefits of Cloud Migration

The benefits of migration may be financial as well as operational. An organization may reduce spending on physical hardware and data-center maintenance. It may also gain scalability, improved availability, faster deployment, easier backup and recovery, remote accessibility, and access to modern cloud services.

For example, an online shopping company may normally require ten servers but may need thirty servers during a festival sale. In a properly designed cloud environment, additional resources can be provided during the busy period and reduced afterward. This avoids maintaining expensive infrastructure for peak demand throughout the year.

Comparing Costs and Benefits

The organization can compare the existing environment with the proposed cloud environment.

1. **Existing Environment:** Hardware, Electricity, Cooling, Maintenance, Backup and IT Infrastructure
2. **Cloud Environment:** Compute, Storage, Network, Database, Backup, Security and Support

The organization can then estimate the cost over a suitable period, such as three or five years.

For example, suppose an organization currently spends ₹15 lakh per year maintaining its IT infrastructure. Its estimated cloud operating cost is ₹10 lakh per year. The apparent difference is ₹5 lakh per year. However, the organization must also consider the initial migration cost, training, application changes, and other expenses before determining the actual financial benefit.

Thus, cost-benefit analysis provides a realistic picture of whether migration is economically useful.

4.2.4.2 Risk Assessment

Even when cloud migration appears beneficial, it may involve several risks. Risk assessment is the process of identifying possible problems, understanding their likelihood and impact, and deciding how to reduce them.

1. **Data Loss**

Data may be accidentally deleted, corrupted, or incompletely transferred during migration. For example, if a college transfers its student database to the cloud and some records are lost during the transfer, it could seriously affect students and administrative activities. To reduce this risk, the organization should maintain backups, test the migration process, verify the migrated data, and prepare a recovery plan.

SGOU

2. Security and Privacy Risks

Moving data to the cloud requires careful attention to security. Incorrect permissions, weak authentication, or improper configuration may expose sensitive information. For example, if confidential student records are stored in cloud storage with incorrect access permissions, unauthorized users may be able to view them. Security risks can be reduced through strong authentication, access control, encryption, monitoring, and regular security checks.

3. Downtime

Migration may temporarily interrupt an application or service. For example, if an online examination system becomes unavailable during migration, students may not be able to access their examinations. Critical systems should therefore be migrated carefully, preferably using testing, phased migration, backup systems, and suitable rollback procedures.

4. Compatibility Risk

An existing application may not work correctly in the selected cloud environment. For example, an old payroll application may depend on an outdated operating system or database. Simply moving it to the cloud may not make it function correctly. The organization should test compatibility and modify or replace the application when necessary.

5. Performance Risk

An application may perform differently after migration.

For example, an application that works quickly on a local network may become slower when users access it through the internet because of network latency.

Performance testing should therefore be performed before completing the migration.

6. Unexpected Costs

Cloud costs can become higher than expected if resources are not managed properly.

For example, an organization may leave unused cloud servers running continuously or store unnecessary data. These resources can continue generating charges.

Cost monitoring, resource optimization, and budget alerts can help control this risk.

7. Vendor Lock-In

An organization may become heavily dependent on a particular cloud provider's services. Moving to another provider later may then become difficult or expensive. For example, an application that depends heavily on proprietary cloud services may require significant modification if the organization later wants to change providers. Therefore, organizations should consider portability and long-term cloud strategy while designing their systems.

Consider a college migrating its Student Management System.

SGOU

Table 4.2.3 Risk assessment example

Risk	Possible Impact	Risk Reduction
Data loss	Loss of student records	Backup and data verification
Security breach	Exposure of personal information	Encryption and access control
Downtime	Students cannot access the system	Planned migration and rollback
Compatibility issue	Application failure	Testing before migration
High cloud cost	Increased expenditure	Cost monitoring
Poor performance	Slow response	Performance testing

This table shows that risk assessment is not about avoiding every possible risk. Instead, it is about identifying risks early and preparing suitable measures to control them.



Fig 4.2.4 Cost-Benefit and Risk Assessment Process

Cost-benefit analysis and risk assessment should be considered together.Suppose a company expects a cloud migration to save ₹5 lakh per year. However, the migration may involve a significant risk of downtime for a critical business application. The company cannot make its decision based only on the expected savings. It must also consider the possible business impact of the downtime and the cost of reducing that risk. Therefore, the final decision should be based on the costs, benefits, risks, and business

requirements. Suppose an online shopping company is considering cloud migration. The assessment shows that cloud migration can reduce hardware maintenance and provide automatic scalability. However, the company identifies data security, downtime, and unexpected cloud expenditure as important risks.The company can therefore decide to proceed with migration, but only after implementing suitable security controls, backup mechanisms, monitoring, testing, and cost-management practices.

Thus, cost-benefit analysis determines whether the expected value of migration justifies the investment, while risk assessment determines what could go wrong and how those risks can be controlled. Together, they provide the information needed to make a well-informed cloud migration decision.

4.2.5 Selecting the Right Migration Approach and Tools

After assessing the existing infrastructure and workloads and analyzing the costs, benefits, and risks, the organization needs to decide how each workload should be moved to the cloud. Different applications have different requirements, so a single migration method may not be suitable for every workload.

For example, a modern web application may be moved to the cloud with very few changes, while an old application may need significant modification. Some applications may no longer be required and can be removed instead of being migrated. Therefore, the migration approach should be selected based on the technical condition, business importance, cost, risk, and future requirements of each workload.

4.2.5.1 Rehosting

Rehosting means moving an application from the existing environment to the cloud with little or no modification. It is often described as “lift and shift.”

For example, suppose a company has an application running on a virtual machine in its data center. The organization can move that workload to a cloud virtual machine without making major changes to the application.

Rehosting is useful when the organization wants to migrate quickly and does not need to redesign the application immediately.(Refer Block 4-Unit 1)

4.2.5.2 Re-platforming

Re-platforming means moving an application to the cloud while making some limited improvements.

For example, an organization may move its application to the cloud but replace its locally managed database with a managed cloud database service. The main application remains largely unchanged, but it can benefit from some cloud features.

Re-platforming requires more effort than rehosting but can provide better cloud benefits.(Refer Block 4-Unit 1)

SGOU

4.2.5.3 Refactoring

Refactoring means significantly modifying or redesigning an application so that it can make better use of cloud capabilities.

For example, an old application may be redesigned as several smaller services that can be independently deployed and scaled.

Refactoring can provide better scalability, flexibility, and performance, but it usually requires more time, technical expertise, and investment.(Refer Block 4-Unit 1)

4.2.5.4 Repurchasing

Repurchasing means replacing an existing application with a cloud-based software service.

For example, instead of maintaining an organization's own email application and servers, it may adopt a cloud-based email service.

This approach can reduce the need to manage the underlying infrastructure.

4.2.5.5 Retiring

Sometimes an organization discovers that an application is no longer useful or has been replaced by another system. In such cases, the application can be retired instead of migrated.

For example, a company may have an old reporting application that has not been used for several years. Migrating it would create unnecessary cost and effort, so the organization may simply remove it.

4.2.5.6 Retaining

Some workloads may not be suitable for immediate migration. They may need to remain in the existing environment because of technical, legal, security, or business requirements.

For example, an application that depends on specialized local hardware may be retained on-premises until a suitable alternative becomes available.

The choice of migration approach depends on several factors.

- **Application conditions are** important because modern applications may be easier to migrate than old applications.
- **Business importance** must also be considered. A critical application requires more careful planning and testing than a non-critical application.
- **Cost** is another factor. A complete application redesign may provide long-term benefits but can require significant initial investment.
- **Time requirements** also influence the decision. If an organization needs to move out of a data center quickly, rehosting may be more appropriate than refactoring.

SGOU

- **Performance and scalability requirements** should also be examined. Applications that require frequent scaling may benefit from refactoring or re-platforming.

Selecting a migration approach is only part of the process. The organization also needs suitable migration tools to carry out the migration.

Migration tools are software or services that help organizations discover, assess, transfer, transform, test, and monitor workloads during cloud migration.

For example, one tool may help identify existing servers and their resource usage, another may help move virtual machines, while another may help transfer databases.

Types of Migration Tools

1. Infrastructure Discovery and Assessment Tools

These tools help collect information about the existing environment.

They can identify:

- Servers
- Virtual machines
- Applications
- Resource utilization
- Dependencies
- Storage
- Network usage

This information helps the organization decide which workloads are suitable for migration.

2. Server Migration Tools

These tools help move physical or virtual servers to cloud infrastructure.

They may support features such as:

- Server replication
- Data synchronization
- Automated migration
- Testing before final migration
- Reduced downtime

SGOU

3. Database Migration Tools

Database migration tools are used to move databases from the existing environment to cloud database services.

They may support:

- Data transfer
- Database replication
- Schema conversion
- Data validation
- Synchronization

For example, a college migrating its Student Management System may use a database migration tool to transfer its student records while maintaining data accuracy.

4. Data Transfer Tools

Large amounts of data may require specialized tools for efficient and secure transfer.

For example, an organization with several terabytes of files may use a suitable data-transfer service instead of manually uploading every file.

5. Monitoring and Management Tools

Migration does not end when the application reaches the cloud. The migrated environment must be monitored to ensure that it performs correctly.

Monitoring tools can help track:

- CPU usage
- Memory usage
- Storage
- Network traffic
- Application performance
- Availability
- Security events
- Cloud expenditure

Factors to Consider When Selecting Migration Tools

The organization should not select a tool simply because it is popular. The tool should match the requirements of the migration project.

- **Compatibility:** The tool should support the existing source environment and the selected cloud environment.

SGOU

- **Security:** The tool should provide appropriate protection for data during migration.
- **Scalability:** The tool should be capable of handling the amount and number of workloads being migrated.
- **Automation:** Automation can reduce manual effort and minimize human errors.
- **Downtime:** For critical applications, tools that support synchronization and replication can help reduce service interruption.
- **Cost:** The cost of using the tool should be reasonable compared with the expected benefits.
- **Ease of Use:** The migration team should be able to configure, operate, and monitor the tool effectively.

Recap

- Cloud migration planning and assessment helps an organization understand its existing environment before moving workloads to the cloud.
- Planning determines what needs to be done, how it will be done, who will do it, and when it will be done.
- Assessment examines servers, applications, databases, storage, networks, data, users, dependencies, security requirements, and current costs.
- Infrastructure and workload assessment includes identifying workloads, measuring resource utilization, analyzing dependencies, classifying business importance, and assessing migration readiness.
- Cost-benefit analysis compares the existing environment with the proposed cloud environment to determine whether migration is economically useful.
- Risk assessment identifies possible problems such as data loss, security risks, downtime, compatibility issues, performance problems, unexpected costs, and vendor lock-in.
- Migration approaches include Rehosting, Re-platforming, Refactoring, Repurchasing, Retiring, and Retaining.
- Migration tools support activities such as infrastructure discovery, server migration, database migration, data transfer, monitoring, and management.
- The final migration approach and tools should be selected according to compatibility, security, scalability, automation, downtime, cost, and ease of use.

SGOU

Objective Type Questions

1. What is the process of deciding what needs to be done, how it will be done, who will do it, and when it will be done?
2. What is the process of examining and understanding the existing situation?
3. What refers to an application, database, or service that uses computing resources?
4. Which resource utilization measure indicates processor usage?
5. What are the relationships between an application and the systems on which it depends?
6. Which migration readiness level indicates that a workload can be migrated immediately?
7. What analysis compares the costs of migration with its expected benefits?
8. Which risk involves accidental deletion, corruption, or incomplete transfer of data?
9. Which migration approach is also called "lift and shift"?
10. Which approach replaces an existing application with a cloud-based software service?
11. Which approach involves significantly modifying or redesigning an application?
12. Which approach removes an application that is no longer required?
13. Which approach keeps a workload in the existing environment?
14. Which migration tools are used to move databases to cloud database services?
15. Which tools help track CPU usage, memory usage, network traffic, application performance, and cloud expenditure?

SGOU

Answers

1. Planning
2. Assessment
3. Workload
4. CPU Usage
5. Dependencies
6. Suitable

7. Cost-Benefit Analysis
8. Data Loss
9. Rehosting
10. Repurchasing
11. Refactoring
12. Retiring
13. Retaining
14. Database Migration Tools
15. Monitoring and Management Tools

Assignments

1. Explain the process of cloud migration planning and assessment. Discuss the importance of planning and assessment before moving organizational IT resources to the cloud.
2. Explain the steps involved in assessing existing infrastructure and workloads. Discuss infrastructure identification, workload identification, resource utilization, application dependencies, business importance, and migration readiness.
3. Explain cost-benefit analysis in cloud migration. Discuss the costs involved during and after migration and explain how an organization can compare the existing environment with the proposed cloud environment.
4. Explain risk assessment in cloud migration. Discuss data loss, security and privacy risks, downtime, compatibility risk, performance risk, unexpected costs, and vendor lock-in with suitable examples.
5. Explain the different cloud migration approaches: Rehosting, Re-platforming, Refactoring, Repurchasing, Retiring, and Retaining. Give a suitable example for each approach.
6. Discuss the factors that should be considered when selecting an appropriate cloud migration approach for a workload.
7. Explain the different types of migration tools used in cloud migration. Discuss infrastructure discovery and assessment tools, server migration tools, database migration tools, data transfer tools, and monitoring and management tools.
8. Explain the factors to be considered when selecting cloud migration tools.

SGOU

Discuss compatibility, security, scalability, automation, downtime, cost, and ease of use.

9. Explain how cost-benefit analysis and risk assessment together support the final decision on cloud migration. Illustrate your answer with suitable examples.

Reference

1. Amazon Web Services. (n.d.). *AWS migration strategies*. Amazon Web Services.
2. Google Cloud. (n.d.). *Cloud migration*. Google Cloud.
3. Microsoft. (n.d.). *Cloud Adoption Framework for Azure*. Microsoft.
4. National Institute of Standards and Technology. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). U.S. Department of Commerce.

Suggested Reading

1. Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). National Institute of Standards and Technology.
2. Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud computing: Concepts, technology & architecture*. Pearson.
3. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2011). *Cloud computing: Principles and paradigms*. John Wiley & Sons.
4. Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—The business perspective. *Decision Support Systems, 51*(1), 176–189. https://doi.org/10.1016/j.dss.2010.12.006

SGOU

Unit 3
Cloud Migration Tools and Techniques

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the concept and importance of cloud migration
- describe the major cloud migration tools provided by AWS, Azure, and Google Cloud
- apply appropriate data migration methods based on different migration requirements
- compare AWS Migration Hub, Azure Migrate, and Google Cloud migration tools based on their features and applications
- analyze data migration requirements and select suitable migration practices for secure and reliable migration

Prerequisite

Cloud migration tools and techniques are important for organizations that move their applications, servers, databases, and data from traditional on-premises environments to cloud platforms. Cloud migration requires proper planning because an unplanned migration may result in data loss, security issues, application failure, or service interruption. For example, a university may migrate its student information system, examination system, learning management system, and databases from its own data centre to a cloud platform. Tools such as AWS Migration Hub, Azure Migrate, and Google Cloud migration tools help organizations discover, assess, plan, migrate, and monitor their workloads, while suitable data migration methods help transfer data safely and efficiently.

This topic provides knowledge about different cloud migration tools, data migration methods, and important practices such as data cleaning, backup, security, testing, validation, monitoring, and rollback. It helps learners understand and compare different migration tools, select suitable migration methods according to specific requirements, identify possible migration risks, and apply appropriate practices for secure and reliable data migration. Thus, the topic provides practical knowledge of how cloud migration activities are planned, performed, tested, and managed in modern organizations.

SGOU

Keywords

Cloud Migration, AWS Migration Hub, Azure Migrate, Google Cloud Migration, Data Migration, Big Bang Migration, Phased Migration, ETL (Extract, Transform, Load), Data Validation, Migration Best Practices.

Discussion

4.3.1 Introduction

Cloud computing has changed the way organizations store data, run applications, and manage computing resources. Instead of maintaining all hardware and software within their own data centres, organizations can use cloud platforms to access computing resources through the Internet. As businesses increasingly adopt cloud computing, many existing applications, databases, and IT systems need to be moved from traditional on-premises environments to cloud environments. This process is known as cloud migration.

Cloud migration is not simply the process of copying data or applications from one location to another. It involves planning, assessing existing systems, selecting suitable cloud services, transferring workloads, testing the migrated systems, and optimizing them after migration. A well-planned migration can reduce infrastructure costs, improve scalability, increase availability, and provide greater flexibility to organizations.

Cloud migration tools support organizations throughout this process. These tools can discover existing infrastructure, assess workloads, identify dependencies, estimate costs, migrate servers and databases, and monitor the migration process. Major cloud service providers such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud provide specialized migration tools for moving different types of workloads to their platforms.

For example, AWS Application Migration Service can be used to migrate servers to AWS, while AWS Database Migration Service supports database migration. Azure Migrate provides services for discovering, assessing, and migrating workloads to Microsoft Azure. Similarly, Google Cloud Migrate provides tools and services for moving workloads to Google Cloud.

The selection of an appropriate migration tool depends on several factors, such as the type of workload, source and target environments, amount of data, required downtime, security requirements, migration cost, and technical complexity. Therefore, understanding cloud migration tools and techniques is important for planning and executing a successful cloud migration.

In this unit, we will explore the major cloud migration tools, their features, applications, and limitations. We will also examine different data migration methods and best practices that help organizations move their data securely, efficiently, and with minimum disruption.

SGOU

4.3.2 Cloud Migration Tools

Cloud migration tools are software platforms or services that help organizations move their applications, servers, databases, data, and other IT resources from an existing environment to a cloud environment. The existing environment may be a physical data centre, a virtualized environment such as VMware or Hyper-V, another cloud platform, or even a hosted infrastructure.

Migration can be a complex activity because an organization may have hundreds or thousands of servers, databases, applications, and interconnected services. Manually moving each component can be time-consuming and may result in data loss, application failure, or extended downtime. Cloud migration tools automate many of these activities and provide facilities for discovery, assessment, planning, migration, testing, monitoring, and optimization.

4.3.2.1 AWS Migration Hub

AWS Migration Hub is a centralized service that helps organizations discover existing servers, plan application migrations, and track migration progress in one place. It provides visibility into the organization's application portfolio and helps migration teams manage the movement of applications to AWS more efficiently.

Migration Hub allows administrators to view the servers and databases associated with each application and monitor their migration status. This provides an application-level view of the migration process rather than requiring administrators to track every server separately.

Organizations can either begin migrating servers immediately and group them into applications during the migration process, or first discover the existing servers, organize them into applications, and then begin migration. This provides flexibility in planning and executing migration projects.

AWS Migration Hub can receive migration status information from different migration tools. Two important AWS services that integrate with Migration Hub are:

- AWS Application Migration Service (AWS MGN): It is the primary AWS service recommended for lift-and-shift migration of servers and workloads to AWS. Migration Hub can display the migration status of workloads being migrated using AWS MGN.
- AWS Database Migration Service (AWS DMS): It is used for migrating databases to AWS and can provide migration status information to Migration Hub.

Therefore, Migration Hub acts as a centralized monitoring and planning point, while specialized services such as AWS MGN and AWS DMS perform the actual server and database migration activities.

SGOU

Advantages of AWS Migration Hub

1. **Centralized migration tracking:**

AWS Migration Hub provides a single location for monitoring the progress of application migrations. This makes it easier for migration teams to track multiple workloads without maintaining separate tracking systems.

2. **Improves application visibility:**

Migration Hub provides an application-level view of servers and databases. It helps administrators understand which resources belong to each application and monitor their individual migration status.

3. **Supports migration planning:**

Organizations can discover their existing servers and group related resources into applications before beginning migration. This helps migration teams plan the migration in a systematic manner.

4. **Integrates with migration tools:**

Migration Hub can receive status updates from AWS migration services such as **AWS Application Migration Service and AWS Database Migration Service**. This allows organizations to use specialized tools while maintaining centralized visibility of the migration process.

5. **Useful for large-scale migrations:**

Migration Hub is particularly useful when an organization has many applications, servers, and databases to migrate. Centralized visibility makes it easier to monitor progress, identify incomplete migrations, and coordinate migration activities.

4.3.2.2 Azure Migrate

Azure Migrate is a centralized migration service provided by Microsoft Azure that helps organizations discover, assess, plan, and migrate their existing servers, applications, databases, and other workloads to Azure. It provides a systematic approach to cloud migration by helping organizations understand their existing IT environment before moving workloads to the cloud.

Azure Migrate provides visibility into the existing infrastructure and helps organizations evaluate the readiness, performance requirements, dependencies, and estimated cost of migrating workloads to Azure. Based on this information, organizations can develop a suitable migration plan and select appropriate Azure services.

How Azure Migrate Works

First, Azure Migrate helps discover existing servers, virtual machines, applications, and databases in the organization's IT environment. Information such as server configuration, CPU usage, memory usage, storage, and network details can be collected during this stage.

SGOU

Next, the discovered workloads are assessed for their suitability for migration to Azure. The assessment can identify compatibility issues, estimate Azure costs, and help determine suitable Azure resource sizes.

After assessment, organizations can plan the migration by identifying application dependencies and deciding which workloads should be migrated first. Related servers and applications can be considered together to reduce the risk of migration problems.

The selected workloads can then be migrated to Azure using appropriate Azure migration services. After migration, the applications and data should be tested to ensure that they are functioning correctly in the new environment.

Features

- **Discovery of existing workloads:** Azure Migrate helps identify servers, virtual machines, applications, and databases in the existing IT environment.
- **Migration assessment:** It helps evaluate workload readiness, compatibility, resource requirements, and other factors before migration.
- **Application dependency analysis:** It helps identify relationships between servers and applications, which is useful for preparing a suitable migration sequence.
- **Cost estimation:** Azure Migrate can provide estimated costs for running workloads in Azure, helping organizations prepare their cloud budget.
- **Right-sizing:** It helps identify appropriate Azure resources based on workload requirements, which can help avoid unnecessary resource allocation.
- **Migration planning:** Organizations can use the assessment results to create a structured plan for migrating workloads to Azure.
- **Migration support:** Azure Migrate supports different migration scenarios and works with appropriate Azure migration services for moving workloads.
- **Post-migration optimization:** After migration, organizations can monitor workloads and optimize resources for better performance and cost efficiency.

Example

Consider a college that has an on-premises data centre containing servers for a Student Management System, Examination System, Learning Management System, and Faculty Portal.

Before migrating these systems to Azure, the college can use Azure Migrate to discover the existing servers and collect information about their configuration and resource usage. The college can then assess the workloads to determine their Azure readiness, estimated cost, and resource requirements.

Azure Migrate can also help identify dependencies between the applications and their databases. Based on the assessment results, the migration team can decide which workloads should be migrated first.

SGOU

After migration, the applications can be tested to ensure that they are working correctly in Azure. The resources can then be monitored and optimized according to performance and cost requirements.

Advantages of Azure Migrate

1. **Provides centralized migration management:**

Azure Migrate provides a centralized platform for discovering, assessing, planning, and supporting the migration of workloads to Azure. This makes it easier for organizations to manage different migration activities systematically.

2. **Helps assess migration readiness:**

Azure Migrate helps organizations evaluate their existing workloads before migration. This allows administrators to identify compatibility issues, resource requirements, and other potential problems before moving workloads to Azure.

3. **Supports cost estimation and right-sizing:**

Azure Migrate helps estimate the expected cost of running workloads in Azure and identify suitable resource sizes. This helps organizations control cloud expenditure and avoid unnecessary resource allocation.

4. **Helps identify application dependencies:**

Azure Migrate can provide information about relationships between servers and applications. Understanding these dependencies helps migration teams decide the appropriate order for migrating related workloads.

5. **Supports systematic migration planning:**

Azure Migrate provides a structured approach to migration by supporting discovery, assessment, planning, migration, and validation. This helps organizations reduce migration risks and carry out the migration process in an organized manner.

Limitation

Azure Migrate is primarily designed to support migrations to Microsoft Azure. It is therefore most suitable for organizations that have selected Azure as their target cloud platform. In addition, assessment results do not guarantee that every application will work without modification after migration. Applications may have specific configurations, dependencies, or compatibility requirements that need additional testing and changes.

4.3.2.3 Google Cloud Migrate

Google Cloud Migrate refers to the migration services and tools provided by Google Cloud to help organizations move their servers, virtual machines, applications, databases, and other workloads to the Google Cloud environment. These services support organizations throughout different stages of migration, including discovery, assessment, planning, migration, validation, and optimization.

Google Cloud provides different migration solutions for different types of workloads.

SGOU

For example, virtual machines can be migrated using suitable VM migration services, while databases can be migrated using database-specific migration services. This allows organizations to select an appropriate migration approach based on the type of workload and the target Google Cloud service.

How Google Cloud Migration Works

First, the organization identifies the servers, virtual machines, applications, databases, and other resources that need to be migrated. Information about the existing infrastructure is collected to understand the current workload environment.

Next, the workloads are assessed to determine their suitability for migration to Google Cloud. The assessment may consider resource requirements, application dependencies, performance, compatibility, and expected costs.

After assessment, a migration plan is prepared. Related applications and resources are identified, and the migration team determines which workloads should be migrated first. This helps reduce dependencies and migration-related risks.

The selected workloads are then migrated to Google Cloud using appropriate migration services. Depending on the workload, the migration may involve moving virtual machines, databases, applications, or data to suitable Google Cloud services.

After migration, the workloads are tested to ensure that applications, databases, networks, and other components are functioning correctly. Finally, the migrated resources can be monitored and optimized for better performance, availability, security, and cost efficiency.

Features

- **Workload discovery and assessment:** Google Cloud migration services help organizations understand their existing workloads and evaluate their suitability for migration.
- **Virtual machine migration:** Supported VM migration services help organizations move virtual machines from existing environments to Google Cloud.
- **Database migration:** Google Cloud provides database migration services for moving supported databases to Google Cloud database platforms.
- **Application migration:** Applications can be migrated to suitable Google Cloud services based on their architecture and requirements.
- **Data migration:** Organizations can transfer large volumes of data from on-premises or other environments to Google Cloud storage services.
- **Migration planning:** Assessment information can be used to prepare migration plans and identify dependencies between workloads.
- **Testing and validation:** Migrated workloads can be tested before they are fully switched to the Google Cloud environment.

SGOU

- **Modernization support:** After migration, organizations can modernize applications using cloud-native technologies such as containers, Kubernetes, managed databases, and serverless services.

Example

Consider an organization that operates an on-premises data centre containing web servers, application servers, databases, and file storage systems.

The organization decides to move its infrastructure to Google Cloud. First, the existing workloads are identified and assessed to understand their resource requirements, dependencies, and migration requirements.

The migration team then prepares a migration plan. Virtual machines can be moved to suitable Google Cloud compute services, while databases can be migrated to appropriate managed database services. Large amounts of data can be transferred to Google Cloud storage services.

After migration, the organization tests the applications and verifies that the data has been transferred correctly. Once the workloads are validated, they can be made available to users in Google Cloud. The organization can then optimize the cloud resources according to performance and cost requirements.

Advantages of Google Cloud Migration Tools

1. **Supports different types of workloads:**

Google Cloud provides migration solutions for virtual machines, databases, applications, and data. This allows organizations to select suitable migration services according to their specific workload requirements.

2. **Provides a structured migration process:**

Google Cloud migration services support important stages such as discovery, assessment, planning, migration, validation, and optimization. This helps organizations perform migration in a systematic manner.

3. **Supports workload modernization:**

After migrating workloads to Google Cloud, organizations can modernize their applications using cloud-native technologies such as containers, Kubernetes, managed databases, and serverless services. This can improve scalability and flexibility.

4. **Helps reduce migration risks:**

Assessment, dependency analysis, testing, and validation help organizations identify potential issues before completing the migration. This can reduce application failures and unexpected downtime.

5. **Supports scalability and optimization:**

Once workloads are migrated to Google Cloud, organizations can use cloud resources according to their changing requirements. Resources can be monitored and optimized to improve performance and control costs.

SGOU

Limitation

Google Cloud migration services are primarily intended for workloads being migrated to Google Cloud. Organizations with highly complex or heterogeneous multi-cloud environments may require additional third-party tools or services.

Migration may also require application changes when the existing application depends on specific hardware, operating-system configurations, database features, or legacy technologies. Therefore, proper assessment and testing are necessary before completing the migration.

4.3.3 Comparison of Cloud Migration Tools

Table 4.3.1: Comparison of Cloud Migration Tools

Feature	AWS Migration Hub	Azure Migrate	Google Cloud Migration Tools
Cloud Provider	Amazon Web Services (AWS)	Microsoft Azure	Google Cloud
Main Purpose	Discover, plan, and track migration activities	Discover, assess, plan, and migrate workloads	Assess, migrate, and modernize workloads
Server Migration	Tracks server migration using AWS migration services	Supports server discovery, assessment, and migration	Supports virtual machine and workload migration
Database Migration	Supports tracking of database migration using AWS DMS	Supports database assessment and migration	Provides database migration services
Migration Assessment	Provides application-level migration visibility	Provides workload assessment and dependency analysis	Supports workload assessment and migration planning
Primary use	Managing and tracking AWS migration projects	Planning and migrating workloads to Azure	Migrating and modernizing workloads on Google Cloud

4.3.4 Data Migration Methods and Best Practices

Data migration is the process of moving data from one storage system, database, application, or computing environment to another.

In cloud migration, data may be moved from an on-premises data centre to AWS, Microsoft Azure, Google Cloud, or another cloud environment.

For example, an organization may have a student database stored on a local server.

SGOU

When the application is migrated to the cloud, the student database must also be transferred to an appropriate cloud database.

A successful data migration should ensure that the data remains complete, accurate, consistent, secure, and accessible.

4.3.4.1 Data Migration Methods

The major data migration methods include:

1. **Big Bang Migration**

In Big Bang Migration, the complete dataset is transferred from the source system to the target system in a single migration operation.

The source system is usually stopped during migration so that data does not change while the transfer is taking place.

Example:

A small organization may stop its application during a scheduled maintenance period, transfer the complete database to a cloud database, validate it, and then restart the application.

Advantage: It is relatively simple and can complete migration quickly.

Limitation: It may require significant downtime and can involve greater risk if the migration fails.

2. **Trickle or Phased Migration**

In Trickle Migration, data is transferred in smaller phases or batches rather than all at once.

The source system can continue operating during most of the migration process, while data is gradually synchronized with the target system.

Example:

A university can migrate student records year by year instead of transferring the entire database at once.

Advantage: It reduces downtime and allows problems to be identified in individual migration stages.

Limitation: It requires more planning, monitoring, and synchronization.

3. **Online Migration**

Online Migration transfers data while the source system continues to operate.

Users can continue to access the application during migration, and changes made to the source system can be synchronized with the target system.

SGOU

Advantage: It minimizes service interruption.

Limitation: It requires more complex synchronization and monitoring.

4. Offline Migration

In Offline Migration, the source system is temporarily stopped during the migration.

The data is transferred to the target system and validated before the application is made available again.

Advantage: It is relatively simple to manage and helps maintain data consistency during transfer.

Limitation: Users cannot access the system during migration, resulting in downtime.

5. ETL-Based Migration

ETL stands for Extract, Transform, and Load.

In this method, data is first extracted from the source system. It is then transformed into the format required by the target system and finally loaded into the target environment.

Extract → Transform → Load

ETL-based migration is particularly useful when the source and target systems have different data structures or formats.

6. Cloud-Based Migration

Cloud-based migration involves transferring data from an on-premises system or another environment to a cloud platform.

This method can support the migration of large volumes of data and allows organizations to take advantage of cloud scalability, flexibility, and storage services.

4.3.4.2 Data Migration Best Practices

Following best practices helps organizations perform data migration accurately, securely, and efficiently.

1. Create a Dedicated Migration Team

A migration team should include people with appropriate technical and business knowledge. Responsibilities should be clearly assigned to avoid confusion during migration.

2. Profile the Data Before Migration

The source data should be examined before migration. Data profiling helps identify its structure, size, quality, relationships, duplicates, missing values, and inconsistencies.

3. Clean the Data

Duplicate, obsolete, invalid, or unnecessary data should be removed before migration

SGOU

whenever appropriate. Migrating poor-quality data can transfer existing problems to the new environment.

4. **Migrate Only Necessary Data**

Organizations should determine which data is actually required in the target environment. Removing unnecessary data can reduce migration time, storage requirements, and cloud costs.

5. **Back Up Data Before Migration**

A reliable backup should be created before migration. The backup provides a recovery option if the migration fails or data is lost or corrupted.

6. **Test the Migration Repeatedly**

The migration process should be tested before the final production migration. Trial migrations help identify problems related to data transformation, compatibility, performance, and configuration.

7. **Validate Migrated Data**

After migration, the target data should be compared with the source data. The organization should verify record counts, completeness, accuracy, relationships, constraints, and application functionality.

8. **Maintain a Rollback Plan**

A rollback plan should define how the organization can return to the previous system if serious problems occur after migration.

9. **Ensure Data Security**

Data should be protected during migration using appropriate security mechanisms such as **encryption, authentication, access control, secure network connections, monitoring, and auditing**.

10. **Monitor After Migration**

The migrated environment should be monitored after migration to identify performance problems, errors, security events, storage issues, and other unexpected problems.

SGOU

Recap

- Cloud migration is the process of moving applications, data, servers, and IT resources from traditional environments to cloud platforms.
- Cloud migration involves several stages such as discovery, assessment, planning, migration, validation, and optimization.
- Cloud migration tools help organizations discover workloads, assess requirements, plan migration, transfer resources, and monitor migration activities.

- AWS Migration Hub provides a centralized platform for discovering applications and tracking migration progress across AWS migration services.
- AWS Application Migration Service (AWS MGN) is mainly used for migrating servers and workloads to AWS.
- Azure Migrate helps organizations discover, assess, plan, and migrate servers, applications, databases, and other workloads to Microsoft Azure.
- Google Cloud migration tools support the migration of virtual machines, applications, databases, and data to Google Cloud.
- Big Bang migration transfers the complete dataset in a single migration operation and may require significant downtime.
- Phased or Trickle migration moves data in smaller batches or stages, reducing migration risks and downtime.
- Online migration allows the source system to remain operational during migration, while offline migration temporarily stops the source system.
- ETL-based migration uses the three stages—Extract, Transform, and Load—to move data between systems with different structures or formats.
- Data migration best practices include data profiling, data cleaning, backup, testing, validation, security, rollback planning, and post-migration monitoring.

Objective Type Questions

1. What is the process of moving applications, data, and IT resources from an existing environment to a cloud environment called?
2. Which AWS service provides centralized tracking of application migration activities?
3. Which AWS service is primarily used for lift-and-shift migration of servers and workloads?
4. Which AWS service is commonly used for database migration?
5. Which Azure service helps organizations discover, assess, and migrate workloads to Microsoft Azure?
6. Where can organizations use migration tools to access computing resources without maintaining all infrastructure locally?
7. What migration method transfers the complete dataset in a single operation?
8. How is data migrated when it is transferred gradually in smaller batches or phases?

SGOU

9. What type of migration allows the source system to continue operating during data transfer?
10. What does ETL stand for in data migration?
11. Which activity identifies duplicates, missing values, and inconsistencies in source data?
12. Why is a backup created before data migration?
13. How is migrated data checked for completeness, accuracy, and consistency after migration?

Answers

1. Cloud Migration
2. AWS Migration Hub
3. AWS Application Migration Service
4. AWS DMS
5. Azure Migrate
6. Cloud Environment
7. Big Bang
8. Phased Migration
9. Online Migration
10. Extract Transform Load
11. Data Profiling
12. Data Backup
13. Data Validation

SGOU

Assignments

1. Explain the concept of cloud migration and discuss its major stages.
2. Explain AWS Migration Hub and its major advantages.
3. Describe Azure Migrate and explain how it supports cloud migration.

4. Explain Google Cloud migration tools and their major features.
5. Compare AWS Migration Hub, Azure Migrate, and Google Cloud migration tools.
6. Explain different data migration methods with suitable examples.
7. Discuss the major best practices that should be followed during data migration.
8. Differentiate between Big Bang Migration and Trickle Migration.

Reference

1. Gleb, T. (2021). *Systematic Cloud Migration: A Hands-On Guide to Architecture, Design, and Technical Implementation*. Apress.
2. Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—The business perspective. *Decision Support Systems, 51*(1), 176–189.
3. Sosinsky, B. (2011). *Cloud Computing Bible*. Wiley.
4. Armbrust, M., et al. (2010). A view of cloud computing. *Communications of the ACM, 53*(4), 50–58.

Suggested Reading

1. Erl, T., & Barceló Monroy, E. (2023). *Cloud Computing: Concepts, Technology, Security & Architecture* (2nd ed.). Pearson.
2. Rittinghouse, J. W., & Ransome, J. F. (2016). *Cloud Computing: Implementation, Management, and Security* (2nd ed.). CRC Press.
3. Erl, T., Puttini, R., & Mahmood, Z. (2013). *Cloud Computing: Concepts, Technology & Architecture*. Pearson.
4. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2011). *Cloud Computing: Principles and Paradigms*. Wiley.
5. Hurwitz, J., Bloor, R., Kaufman, M., & Halper, F. (2010). *Cloud Computing for Dummies*. Wiley.

SGOU

Unit 4
Planning and Managing Cloud Migration

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the concept and process of cloud migration
- describe the steps involved in planning and managing cloud migration
- apply the 7 'R's to different cloud migration situations
- differentiate between the major cloud migration approaches
- analyze a cloud migration scenario and select a suitable migration approach

Prerequisite

Cloud migration has become an important approach for organizations that want to modernize their IT infrastructure and improve the way applications and data are managed. For instance, a university may operate its student information system, examination database, learning platform, and administrative applications on servers located in its own data centre. As the number of learners and digital services increases, the university may require additional storage, computing capacity, availability, and secure access. Moving suitable workloads to a cloud environment can provide greater scalability, flexibility, and operational efficiency. However, the university must first understand its existing infrastructure, application dependencies, databases, networking, security requirements, costs, and business needs before deciding which workloads should be migrated, retained, replaced, or retired. This illustrates the relevance of cloud migration planning and management, the Plan–Execute–Monitor process, and the 7 Rs of cloud migration in real-world situations.

Learners should have a basic understanding of cloud computing, cloud service models, cloud deployment models, computer networks, databases, operating systems, virtualization, storage, and application architecture before studying this topic. This background helps learners understand how existing IT resources are assessed and transferred to suitable cloud environments. Studying cloud migration provides several advantages, including an understanding of how organizations can improve scalability, flexibility, availability, reliability, resource utilization, cost management, security, and operational efficiency. Through this topic, learners will be able to understand the major stages involved in planning and managing cloud migration, identify workload depen-

SCOU

dencies and migration requirements, evaluate different migration approaches, select suitable strategies using the 7 Rs, recognize migration risks, and understand the importance of testing, monitoring, security, cost optimization, and continuous improvement after migration.

Keywords

Cloud Migration, Migration Strategy, Cloud Computing, Migration Planning, Migration Lifecycle, Workload Assessment, Application Dependency, Plan–Execute–Monitor, 7 Rs of Cloud Migration, Rehosting, Replatforming, Refactoring

Discussion

4.4.1 Introduction

Cloud migration is the process of moving an organization's digital assets, such as applications, data, workloads, databases, servers, storage, and other IT resources, from on-premises or traditional computing environments to cloud-based platforms. An organization may migrate its complete IT environment or only selected workloads to a public, private, hybrid, or multi-cloud environment. The main objectives of cloud migration include improving scalability, flexibility, reliability, availability, and operational efficiency while potentially reducing infrastructure and operational costs.

Cloud migration is not simply a matter of transferring existing servers, applications, or data from one environment to another. A successful migration requires a clear understanding of business objectives, existing IT infrastructure, application dependencies, data characteristics, security requirements, performance expectations, costs, and operational needs. Organizations must also evaluate the suitability of individual workloads for cloud adoption and determine whether they should be migrated, modified, replaced, retained, or retired.

For example, an organization may have several applications running in its existing data centre. Some applications may be suitable for direct migration with minimal modification, while others may require changes to take advantage of cloud services. Certain applications may require complete redesign to achieve better scalability and flexibility. Some applications may no longer be useful and can therefore be retired, whereas other applications may need to remain in the existing environment because of legal, privacy, technical, licensing, or business requirements. Therefore, each workload needs to be evaluated separately before selecting an appropriate migration strategy.

Cloud migration should therefore be treated as a planned and continuously managed lifecycle rather than as a one-time technical activity. Proper planning helps minimize business disruption, protect data, control costs, manage risks, and ensure that the migrated environment provides the expected benefits.

SGOU

4.4.2 Planning and Managing Cloud Migration

A cloud migration strategy is a structured plan that describes how an organization will move its applications, data, infrastructure, and other IT resources from an existing environment to a cloud environment. It provides a systematic approach for assessing the current environment, identifying migration requirements, selecting appropriate cloud services, managing risks, transferring data, migrating applications, testing the new environment, training users, and preparing for possible rollback.

Planning is one of the most important stages of cloud migration because poor planning can lead to data loss, application failure, unexpected costs, security problems, performance issues, and business disruption. A well-designed migration plan should therefore consider both technical and business requirements.

Cloud migration planning begins with a clear understanding of why the organization wants to migrate to the cloud. Organizations may migrate to reduce infrastructure and operational costs, improve scalability, increase availability, support remote access, modernize legacy applications, strengthen disaster recovery, or obtain access to advanced cloud services. Clearly defined objectives help the organization determine which workloads should be migrated and provide measurable criteria for evaluating migration success.

The next step is to assess the existing IT environment. This assessment includes physical and virtual servers, operating systems, applications, databases, storage systems, networks, security mechanisms, software licenses, backup systems, and disaster recovery facilities. The assessment helps determine the current condition of the infrastructure and identifies applications that may be outdated, unsupported, underutilized, or technically difficult to migrate.

Another important activity is application and workload discovery. Organizations should identify all relevant applications, databases, virtual machines, services, and supporting resources. Each workload should be examined in terms of its business importance, technical requirements, data sensitivity, performance requirements, dependencies, and migration feasibility.

Understanding application dependencies is particularly important because applications rarely operate independently. For example, a student information system may depend on web servers, application servers, databases, authentication services, file storage, and external APIs. If these dependencies are not identified, migrating one component without its supporting components may result in application failure. Dependency mapping therefore helps determine the correct migration sequence and reduces unexpected interruptions.

The organization must also select an appropriate cloud deployment model. Depending on its requirements, it may choose a public cloud, private cloud, hybrid cloud, or multi-cloud environment. The selection should consider security, compliance, performance, scalability, cost, geographical requirements, and organizational policies.

The appropriate cloud service model should also be considered. Infrastructure as a Service provides infrastructure resources such as virtual machines, storage, and

SGOU

networking. Platform as a Service provides managed platforms for developing and deploying applications, while Software as a Service provides ready-to-use software applications through the internet. Different workloads within the same organization may use different service models according to their requirements.

After workloads and the target environment have been assessed, an appropriate migration approach should be selected for each workload. The 7 Rs—Retire, Retain, Rehost, Relocate, Repurchase, Replatform, and Refactor—provide a useful framework for making this decision. Not every application should be migrated using the same approach. Some applications may be retired, some may be retained, some may be moved with minimal changes, and others may require modernization or complete redesign.

Cost estimation is another essential part of migration planning. Cloud migration involves both one-time migration costs and continuing operational costs. Organizations should consider computing resources, storage, databases, network and data-transfer charges, software licenses, backup, disaster recovery, security services, migration tools, training, and human resources. Comparing expected cloud expenditure with the cost of maintaining the existing environment helps the organization determine whether the migration provides sufficient financial value.

Security and compliance requirements must be considered from the beginning of the migration process. Sensitive data should be identified and appropriate controls should be planned for authentication, authorization, encryption, network security, logging, monitoring, backup, and access management. Organizations operating in regulated environments should also consider applicable legal, regulatory, and industry requirements when selecting cloud services and designing the migration architecture.

Risk assessment is another important component of migration planning. Migration risks may include data loss, data corruption, application incompatibility, service interruption, security vulnerabilities, network failures, performance degradation, unexpected costs, and lack of skilled personnel. Significant risks should be evaluated according to their likelihood and potential impact, and suitable mitigation and contingency measures should be prepared. A reliable backup and a well-defined rollback strategy are especially important for critical workloads.

The final outcome of the planning stage is a migration roadmap. The roadmap specifies what will be migrated, when it will be migrated, how it will be migrated, who will be responsible, what resources are required, and what testing and validation activities must be completed. Large organizations generally follow a phased migration approach rather than migrating all workloads at the same time. Less critical workloads may be migrated first so that the organization can gain experience and identify problems before moving business-critical applications.

Thus, effective cloud migration planning establishes the foundation for controlled execution and continuous monitoring.

SGOU

4.4.3 Cloud Migration Process: Plan, Execute, Monitor

Once the organization has established its migration objectives, assessed the existing environment, identified workloads and dependencies, estimated costs, evaluated risks, and selected suitable migration approaches, the migration can proceed through three major stages: Plan, Execute, and Monitor.

The Plan–Execute–Monitor model provides a structured way to manage cloud migration. Although the stages are presented sequentially, cloud migration should be viewed as a continuous lifecycle. Information obtained during execution and monitoring may require the organization to revisit its plans and make further improvements.

1. Plan

The Plan stage forms the foundation of the cloud migration process. It determines what needs to be migrated, why it needs to be migrated, how it should be migrated, and what resources and controls are required.

The planning process begins by identifying the organization's business objectives. These objectives may include reducing infrastructure costs, improving scalability, increasing availability, modernizing applications, improving disaster recovery, or supporting changing business requirements. The objectives should be measurable so that the organization can evaluate whether the migration has achieved the expected results.

The organization then performs a detailed assessment of its existing IT environment. Applications, databases, servers, storage, operating systems, networks, security controls, and supporting services are examined. Workloads are classified according to their importance, complexity, performance requirements, security requirements, and dependencies.

Dependency mapping is particularly important because one application may rely on several other services. Understanding these relationships helps determine the correct order in which workloads should be migrated.

The target cloud environment is then selected and designed. The organization determines the appropriate deployment model and service model and prepares the required architecture. Security, networking, identity management, storage, computing, databases, backup, and disaster recovery requirements are considered at this stage.

The organization also selects a suitable migration approach for each workload. The decision may involve retiring unnecessary applications, retaining workloads that cannot currently be migrated, rehosting applications with minimal modification, relocating workloads, repurchasing software, replatforming applications, or refactoring applications for cloud-native operation.

Cost, risk, security, compliance, resource requirements, and migration timelines are also considered. The result is a detailed migration roadmap that guides the execution stage.

SGOU

2. Execute

The Execute stage converts the migration plan into actual migration activities. Execution should be carefully controlled because errors during this stage can directly affect business operations.

Before applications and data are transferred, the target cloud environment must be prepared. This involves configuring computing resources, storage, databases, networks, identity and access management, security controls, monitoring services, backup facilities, and disaster recovery mechanisms. The target environment should satisfy the organization's performance, security, availability, scalability, and compliance requirements.

Security controls should be implemented before workloads are introduced into the cloud. Authentication and authorization mechanisms should ensure that only authorized users and services can access cloud resources. Encryption should be applied where required to protect data during transmission and while stored. Network security mechanisms, firewalls, security groups, logging, auditing, and security monitoring should also be established.

Before data is transferred, it should be prepared for migration. Unnecessary information can be removed, duplicate data can be identified, and required data formats can be converted. Sensitive information should be appropriately protected. Data should also be validated to ensure completeness and consistency.

A complete and verified backup of the existing environment should be maintained before major migration activities begin. The backup provides a recovery mechanism if migration fails or data becomes corrupted. Backup restoration should also be tested because a backup is useful only when it can be successfully restored.

For complex environments, a pilot migration can be performed before large-scale migration. A small number of applications or workloads are migrated as a trial. This allows the organization to identify technical problems, validate application compatibility, test security controls, evaluate performance, and estimate the actual effort required. The lessons learned from the pilot can then be applied to subsequent migration activities.

The next activity is data migration. Data may be transferred through online migration, database migration tools, backup and restore mechanisms, replication, synchronization, or specialized data-transfer solutions. The method depends on the amount of data, network bandwidth, application requirements, security considerations, and acceptable downtime. Data integrity should be verified after migration.

Applications and workloads are then migrated according to their selected strategies. A rehosted application may require minimal changes, whereas a replatformed application may use managed cloud services. Applications selected for refactoring may require significant architectural changes. Migration should follow the dependency sequence established during planning.

After migration, comprehensive testing and validation must be performed. Functional testing verifies that application functions operate correctly. Performance testing evaluates response time, throughput, and resource utilization. Integration testing

SGOU

checks communication between application components and external services. Security testing evaluates access controls and vulnerabilities. Data integrity testing confirms that migrated data is complete and accurate. User acceptance testing determines whether the migrated system satisfies user and business requirements.

After successful testing, the organization performs cutover. Cutover is the point at which users and business operations begin using the cloud-based system as the primary environment. The cutover plan should include final data synchronization, user communication, validation procedures, monitoring arrangements, and rollback procedures. For critical systems, cutover is often scheduled during periods of low business activity to minimize disruption.

A rollback plan provides protection if serious problems occur during or after cutover. If critical functions fail, data becomes inconsistent, performance is unacceptable, or major security problems are identified, the organization should be able to return to the previous environment or activate an alternative recovery mechanism. The rollback procedure should be defined and tested before final migration.

3. Monitor

The Monitor stage begins once workloads have been migrated to the cloud. However, successful migration does not mean that the process ends when applications start running in the new environment. Continuous monitoring is necessary to determine whether applications are performing correctly, resources are being used efficiently, security controls are functioning properly, and cloud expenditure remains within acceptable limits.

Application performance monitoring helps organizations determine whether the migrated applications are providing the expected level of service. Important indicators include response time, throughput, error rates, availability, CPU utilization, memory utilization, network performance, and storage utilization. Monitoring these indicators allows administrators to identify performance degradation and take corrective action before users are significantly affected.

Resource monitoring is equally important because cloud resources are generally provisioned according to usage requirements. If resources are significantly over-provisioned, the organization may pay for capacity that is not being used. If resources are under-provisioned, applications may experience poor performance. Monitoring therefore supports right-sizing, in which cloud resources are adjusted according to actual workload requirements.

Continuous security monitoring is required after migration because threats and vulnerabilities can occur at any stage of cloud operations. Organizations should monitor authentication attempts, access activities, configuration changes, security alerts, network activity, vulnerabilities, and audit logs. Security monitoring helps identify unauthorized access and suspicious activities and supports timely incident response.

Cost monitoring and optimization are also essential in cloud environments. Organizations should regularly review resource consumption and identify unused or unnecessary resources. Unexpected increases in cloud expenditure may occur because

SGOU

of excessive computing resources, unnecessary storage, high data-transfer activity, or resources that were created for testing but were never removed. Continuous cost monitoring helps organizations maintain financial control and achieve the expected economic benefits of cloud migration.

The organization should also monitor availability and reliability. Cloud applications may depend on multiple services, networks, databases, and computing resources. Monitoring helps identify server failures, network problems, database issues, service interruptions, and other operational problems. High-availability and fault-tolerance mechanisms can be used where necessary to reduce service disruption.

Data and backup monitoring should continue after migration. Backup completion, data replication, storage utilization, data integrity, retention policies, and restoration capabilities should be checked regularly. Periodic restoration testing is particularly important because a backup is useful only when it can be successfully restored when required.

Monitoring should also be connected with service-level objectives and business requirements. The organization should compare actual performance with the targets defined during planning. For example, if improving application availability was one of the migration objectives, actual uptime should be measured against the required availability level. Similarly, if cost reduction was a major objective, pre-migration and post-migration costs should be compared.

4. Continuous Optimization after Migration

The Plan–Execute–Monitor model should be viewed as a continuous lifecycle rather than a one-time sequence. Information obtained through monitoring may reveal opportunities for further improvement. Therefore, organizations should continuously optimize their cloud environment after migration.

Optimization may involve improving application performance, reducing unnecessary costs, strengthening security, right-sizing resources, automating routine operations, improving backup and disaster recovery, and adopting additional cloud-native services.

For example, an organization may initially rehost an application because rapid migration is the immediate priority. After the application has stabilized in the cloud, the organization may gradually replatform or refactor it to take advantage of managed databases, containers, serverless computing, automated scaling, or other cloud-native technologies. In this way, migration can become the first stage of a broader application modernization process.

4.4.3.1 Cloud Migration Lifecycle

The cloud migration lifecycle is an end-to-end operational framework structured into three core phases: Plan, Execute, and Monitor.

SGOU

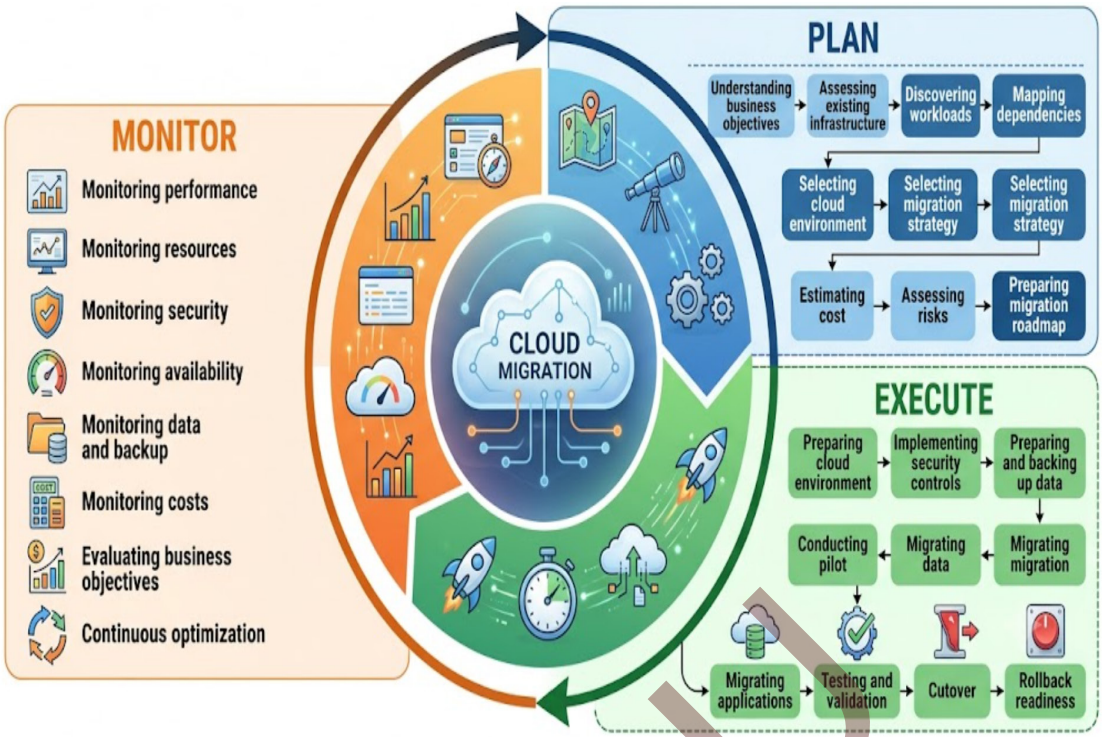


Fig 4.3.1 Cloud Migration Lifecycle Architecture (Plan, Execute, Monitor)

Phase 1: PLAN

Focuses on strategy, assessment, risk management, and roadmap definition before initiating any data movement.

1. **Understanding Business Objectives:** Aligning cloud adoption goals with organizational strategy (e.g., scalability, cost reduction, agility).
2. **Assessing Existing Infrastructure:** Evaluating current legacy systems, server configurations, and performance metrics.
3. **Discovering Workloads:** Cataloging all active workloads, applications, and databases across the environment.
4. **Mapping Dependencies:** Identifying complex application, database, and system interdependencies to prevent service disruptions.
5. **Selecting Cloud Environment:** Choosing the optimal cloud platform (Public, Private, Hybrid, or Multi-Cloud) and deployment model (IaaS, PaaS, SaaS).
6. **Selecting Migration Strategy:** Deciding the approach for each workload (e.g., the 6 Rs: Rehost, Replatform, Refactor, Repurchase, Retire, Retain).
7. **Estimating Cost:** Projecting cloud operational costs (OpEx), total cost of ownership (TCO), and migration expenses.
8. **Assessing Risks:** Identifying potential risks related to data security, downtime, regulatory compliance, and business continuity.

SGOU

9. **Preparing Migration Roadmap:** Creating a structured timeline detailing execution schedules, milestones, and resource assignments.

Phase 2: EXECUTE

Focuses on environment setup, data backup, application migration, testing, and system cutover.

1. **Preparing Cloud Environment:** Setting up cloud infrastructure, Virtual Private Clouds (VPCs), subnets, and identity management rules.
2. **Implementing Security Controls:** Configuring firewalls, encryption keys (in-transit and at-rest), and compliance standards.
3. **Preparing and Backing Up Data:** Creating full system backups and data snapshots to guarantee data safety.
4. **Conducting Pilot Migration:** Executing a small-scale trial migration with non-critical workloads to validate execution paths.
5. **Migrating Data:** Transferring primary datasets, databases, and file stores using batch or continuous replication.
6. **Migrating Applications:** Deploying applications into the prepared cloud environment according to the designated migration strategy.
7. **Testing and Validation:** Performing integration, performance, security, and user acceptance testing (UAT).
8. **Cutover:** Redirecting production traffic and switching active operational systems to the cloud environment.
9. **Rollback Readiness:** Maintaining an active contingency and rollback procedure in case post-cutover issues arise.

Phase 3: MONITOR

Focuses on continuous post-migration tracking, security oversight, cost control, and optimization.

1. **Monitoring Performance:** Tracking system latency, execution speeds, and throughput across services.
2. **Monitoring Resources:** Overseeing CPU, memory, storage, and network utilization rates.
3. **Monitoring Security:** Enforcing real-time threat detection, access audits, and compliance enforcement.
4. **Monitoring Availability:** Ensuring uptime compliance and Service Level Agreements (SLAs).
5. **Monitoring Data and Backup:** Auditing automated backup schedules, disaster recovery readiness, and data integrity.

SGOU

6. **Monitoring Costs:** Tracking real-time cloud expenditure, identifying unallocated resources, and preventing budget overruns.

7. **Evaluating Business Objectives:** Assessing post-migration outcomes against initial business goals set during planning.

8. **Continuous Optimization:** Fine-tuning architecture, adjusting auto-scaling rules, and rightsizing resources to improve performance and cost efficiency.

4.4.4 Cloud Migration Approaches: 7 'R's in Cloud Migration

Cloud migration does not follow a single approach for all applications and workloads. Different applications have different business values, technical architectures, dependencies, security requirements, performance characteristics, and future development needs. Therefore, an organization must evaluate each workload carefully and select an appropriate migration approach.

The 7 'R's of cloud migration provide a structured framework for deciding what should happen to an existing application or workload during cloud adoption. The seven approaches are Retire, Retain, Rehost, Relocate, Repurchase, Replatform, and Refactor. These approaches help organizations determine whether an application should be removed, kept in its current environment, moved with minimal changes, transferred to another cloud environment, replaced with a different product, modified to use cloud capabilities, or completely redesigned for the cloud.

The selection of a particular approach depends on factors such as business value, application complexity, migration cost, technical feasibility, security requirements, performance requirements, time available for migration, and long-term modernization objectives.

1. Retire

Retire means removing or decommissioning an application, system, server, or workload because it is no longer required by the organization.

During migration assessment, organizations often discover applications that are rarely used, duplicate the functionality of other systems, have become obsolete, or no longer provide sufficient business value. Migrating such applications to the cloud would create unnecessary costs and consume technical resources. Instead of migrating them, the organization can safely discontinue them.

For example, a university may have an old student reporting application that has been replaced by a newer integrated student management system. If the old application is no longer required and its historical data has been appropriately preserved, the organization may decide to retire it rather than migrate it.

Before retirement, the organization should verify that the application is not required by users or other systems and that important records have been preserved according to organizational and regulatory requirements.

SGOU

2. Retain

Retain means keeping an application or workload in its existing environment instead of migrating it to the cloud at the present time.

Not every workload is immediately suitable for cloud migration. Some applications may have strict regulatory requirements, specialized hardware dependencies, technical limitations, licensing restrictions, or low migration priority. In such situations, the organization may decide to retain the workload temporarily or permanently.

For example, an organization may have a legacy application that depends on specialized hardware that cannot easily be reproduced in a cloud environment. The organization may continue operating the application in its existing data centre while migrating other applications.

Retention does not necessarily mean that the application will never be migrated. The organization may reconsider migration when technology, business requirements, or application architecture changes. A retained workload should continue to be monitored and maintained, and the reasons for retention should be periodically reviewed.

3. Rehost

Rehost is commonly known as "lift and shift." It involves moving an application or workload from the existing environment to the cloud with minimal or no modification to the application architecture.

For example, a physical or virtual server running an application may be migrated to a cloud-based virtual machine while retaining the existing operating system, application, and configuration.

Rehosting is often selected when an organization wants to migrate a large number of workloads within a relatively short period. Since major application changes are not required, the migration process can be simpler and faster.

The main advantage of rehosting is that it enables relatively rapid migration with limited application modification. However, the migrated application may not fully utilize cloud-native capabilities such as automated scaling, managed services, containers, or serverless computing.

Therefore, an organization may initially rehost an application and later replatform or refactor it after the workload becomes stable in the cloud.

4. Relocate

Relocate involves moving an application or workload from one infrastructure environment to another without making significant changes to the application architecture.

Relocation is useful when an organization wants to move workloads between compatible infrastructure platforms while preserving the existing application architecture. It may involve moving workloads between data centres, virtualization platforms, or cloud environments.

SGOU

For example, an organization may move a group of virtual machines from an existing virtualization environment to a compatible cloud infrastructure platform without redesigning the applications running on those virtual machines.

The main distinction between Rehost and Relocate is that rehosting generally refers to moving the application or workload with minimal modification to cloud infrastructure, whereas relocation emphasizes moving an existing workload between compatible infrastructure or platform environments while preserving its architecture.

Before relocation, the organization should verify networking, storage, virtualization, licensing, security, and operational compatibility between the source and target environments.

5. Repurchase

Repurchase means replacing an existing application with a different commercially available product or cloud-based solution.

Instead of migrating the existing application, the organization purchases or subscribes to another solution that provides similar or improved functionality. This approach is sometimes described as "drop and shop."

For example, an organization may currently operate its own locally installed customer relationship management system. Instead of migrating that application to the cloud, it may replace it with a cloud-based SaaS CRM solution.

Repurchasing can be beneficial when an existing application is expensive to maintain, difficult to modernize, approaching the end of its useful life, or no longer meeting current business requirements.

The approach can reduce infrastructure management and application maintenance while providing access to modern functionality and regular software updates. Cloud-based solutions may also provide scalability and vendor-managed security and maintenance capabilities.

However, repurchasing requires careful evaluation of data migration, vendor dependency, licensing, security, integration, customization, and long-term costs. The organization should verify that the replacement solution can meet its functional and technical requirements.

Integration with existing systems is another important consideration. The new application may need to communicate with existing databases, authentication systems, reporting systems, payment systems, or other business applications.

6. Replatform

Replatform involves making limited modifications to an application so that it can take advantage of cloud capabilities without completely redesigning its architecture.

It is sometimes described as "lift, tinker, and shift." The application is not fundamentally rewritten, but selected components are modified or moved to managed cloud services.

SGOU

For example, an organization may migrate an application to the cloud while moving its database from a self-managed database server to a managed cloud database service. The application itself may require only limited changes to communicate with the new database service.

Another example is moving an application from a traditional server environment to a managed application platform while making relatively small configuration changes.

Replatforming provides a balance between rapid migration and application modernization. It generally requires more effort than rehosting but less effort than complete refactoring.

Replatforming can improve performance, scalability, reliability, and operational efficiency while reducing infrastructure management. It also allows organizations to take advantage of selected managed cloud services without completely redesigning the application.

However, even limited modifications can affect application functionality, compatibility, security, and performance. Therefore, the replatformed application should be thoroughly tested before and after migration.

7. Refactor

Refactor is the most extensive modernization-oriented approach among the 7 Rs. It involves redesigning or re-architecting an application specifically to take advantage of cloud-native capabilities.

Unlike rehosting, where the existing architecture is largely preserved, refactoring may involve significant changes to the application's architecture, code, databases, deployment model, and supporting services.

For example, a traditional monolithic application may be redesigned as a collection of microservices. Some components may use containers, serverless functions, managed databases, message queues, and automated scaling mechanisms.

Refactoring is generally considered when an application is strategically important and requires substantial modernization to meet future business requirements. It may be particularly suitable when the existing architecture limits scalability, performance, maintainability, or integration with modern services.

Refactoring can provide significant benefits, including improved scalability, greater flexibility, better performance, improved resilience, easier deployment, greater automation, and better use of cloud-native services.

However, refactoring usually requires more time, technical expertise, testing, and investment than simpler migration approaches. It may also introduce additional development and project-management risks. Therefore, the organization should evaluate whether the expected long-term benefits justify the additional effort.

A refactored application should be thoroughly tested for functionality, performance, security, scalability, integration, and reliability before it becomes the primary production system.

SGOU

4.4.5 Comparison of the 7 Rs

Table 4.4.1: Comparison of Cloud Migration Approaches

Approach	Meaning	Level of Application Change
Retire	Remove an unnecessary application or workload	No migration
Retain	Keep the application in the existing environment	No immediate change
Rehost	Move the application with minimal modification	Very low
Relocate	Move the workload to another compatible infrastructure or cloud environment	Low
Repurchase	Replace the existing application with another solution	Application replacement
Replatform	Modify the application to use selected cloud services	Moderate
Refactor	Redesign the application for cloud-native operation	High

4.4.6 Selecting an Appropriate Migration Strategy

There is no universally best migration approach for every workload. The appropriate strategy depends on the size and complexity of the application, hardware requirements, scalability and availability requirements, technical compatibility, business value, migration cost, security requirements, and organizational objectives.

A workload that needs to be migrated quickly with minimal changes may be rehosted. An application that can benefit from selected cloud services without major redesign may be replatformed. Applications requiring extensive modernization may be refactored. An existing application may be repurchased when a suitable cloud-based replacement is available.

Applications that no longer provide business value can be retired, while workloads that cannot currently be migrated can be retained. Relocation can be considered when the organization wants to move workloads between compatible infrastructure platforms with minimal application changes.

The choice should also consider the application's expected future lifecycle. An organization may decide to rehost a legacy application temporarily because rapid migration is necessary, and later refactor it as part of a modernization programme. Similarly, an application approaching the end of its useful life may be retained temporarily and subsequently replaced through a repurchase strategy.

SGOU

Therefore, an organization may use different 7R strategies for different workloads within the same migration project. This workload-specific approach helps balance migration cost, time, risk, business value, technical requirements, and long-term modernization objectives.

Recap

- Cloud migration is the process of moving applications, data, workloads, databases, servers, and other IT resources to a cloud environment.
- Cloud migration helps organizations improve scalability, flexibility, availability, reliability, and operational efficiency.
- Successful cloud migration requires a clear understanding of business objectives and technical requirements.
- Workload assessment helps organizations determine which applications and resources are suitable for migration.
- Application dependency mapping identifies the relationships between applications, databases, servers, networks, and other supporting services.
- Migration planning considers cost, security, compliance, performance, risks, resources, and business requirements.
- The Plan–Execute–Monitor model provides a structured approach for managing the cloud migration lifecycle.
- The Plan stage includes assessing the existing environment, identifying workloads, selecting migration strategies, estimating costs, and preparing a migration roadmap.
- The Execute stage involves preparing the cloud environment, transferring data, migrating applications, testing workloads, and performing cutover.
- A backup and rollback plan helps protect data and business operations when unexpected problems occur during migration.
- The Monitor stage focuses on application performance, resource utilization, security, availability, backup, and cloud expenditure.
- Cloud migration should be treated as a continuous lifecycle, because monitoring results can lead to further optimization and improvement.
- The 7 Rs of cloud migration are Retire, Retain, Rehost, Relocate, Repurchase, Replatform, and Refactor.
- Rehost and Relocate involve moving workloads with limited changes, while Repurchase replaces an existing application with another suitable solution.
- Replatform makes limited modifications to use cloud services, whereas Refactor significantly redesigns an application to take advantage of cloud-native capabilities.

SGOU

Objective Type Questions

1. What is the process of moving IT resources to a cloud environment called?
2. Which stage involves assessing the existing IT environment before migration?
3. How are relationships between applications and supporting resources identified?
4. Why is a backup created before a major cloud migration?
5. Where are migrated applications and data deployed after migration?
6. When should security requirements be considered during cloud migration?
7. Who evaluates the business requirements for cloud migration?
8. What approach is commonly known as "lift and shift"?
9. Which approach replaces an existing application with a new solution?
10. How can an organization identify dependencies between workloads?
11. Why is continuous monitoring required after migration?
12. Where can an organization keep a workload that is not suitable for immediate migration?
13. When is a rollback strategy used during migration?
14. How many approaches are included in the 7 Rs framework?
15. Name the approach that redesigns an application for cloud-native operation.

SGOU

Answers to Objective Type Questions

1. Cloud migration
2. Plan
3. Dependency mapping
4. Data protection
5. Cloud environment
6. Planning stage
7. Migration team
8. Rehost

9. Repurchase
10. Dependency mapping
11. Performance monitoring
12. Existing environment
13. Migration failure
14. Seven
15. Refactor

Assignments

1. Explain the importance of cloud migration planning and describe the major activities involved in preparing a cloud migration roadmap.
2. Discuss the Plan–Execute–Monitor process of cloud migration and explain the activities performed at each stage.
3. Explain the 7 Rs of cloud migration and describe how each approach can be selected for different applications and workloads.
4. Analyze the factors that an organization should consider when selecting an appropriate cloud migration strategy for its applications and workloads.

Reference

1. Ruparelia, N. B. (2023). *Cloud Computing* (Revised and Updated ed.). MIT Press.
2. Marinescu, D. C. (2017). *Cloud Computing: Theory and Practice*. Morgan Kaufmann.
3. Rittinghouse, J. W., & Ransome, J. F. (2017). *Cloud Computing: Implementation, Management, and Security*. CRC Press.
4. Buyya, R., Broberg, J., & Goscinski, A. M. (2011). *Cloud Computing: Principles and Paradigms*. Wiley.

SGOU

Suggested Reading

1. Surianarayanan, C., & Chelliah, P. R. (2023). *Essentials of Cloud Computing: A Holistic, Cloud-Native Perspective* (2nd ed.). Springer.
2. Sehgal, N. K., Bhatt, P. C. P., & Acken, J. M. (2022). *Cloud Computing with Security and Scalability: Concepts and Practices* (3rd ed.). Springer.
3. Erl, T., Puttini, R., & Mahmood, Z. (2021). *Cloud Computing: Concepts, Technology & Architecture*. Pearson.
4. Kavis, M. J. (2020). *Architecting the Cloud: Design Decisions for Cloud Computing Service Models*. Wiley.
5. Sosinsky, B. (2020). *Cloud Computing Bible*. Wiley.

SGOU



Cloud Automation Tools

SGOU

Unit 1
Introduction to Cloud Automation

Learning Outcomes

Upon completion of the unit, learner will be able to:

- define cloud automation
- identify the different types of cloud automation
- explain the concept, purpose, and benefits of Infrastructure as Code (IaC).
- describe the concepts and basic workflow of Continuous Integration and Continuous Deployment (CI/CD).
- differentiate between Infrastructure as Code and CI/CD in cloud environments

Prerequisite

Cloud computing has made it possible to create servers, databases, networks, and applications within minutes. But as the number of cloud resources grows, managing everything manually becomes difficult. Creating the same resources repeatedly, applying configurations, and deploying application updates can consume a great deal of time. Even a small mistake in one of these activities can affect the entire application.

This is where cloud automation becomes important. Automation allows repetitive cloud tasks to be performed using predefined instructions and software tools. It helps organizations manage resources more quickly, maintain consistency, and respond efficiently when requirements change.

Cloud automation is not limited to creating cloud resources. It can also configure systems, deploy applications, and support frequent software updates. Two important ideas that make this possible are Infrastructure as Code and CI/CD. Infrastructure as Code helps manage cloud infrastructure through code or configuration files, while CI/CD helps move application changes from development towards deployment in a systematic way.

Understanding these concepts will help you see how modern cloud applications can be managed without depending entirely on repetitive manual work. It will also provide a foundation for learning the tools and practices used in cloud and DevOps environments.

Ready to explore how cloud resources and applications can be managed more efficiently? Let us begin with the fundamentals of cloud automation.



SGOU

Keywords

Cloud Automation, Infrastructure as Code, CI/CD, Infrastructure Automation, Configuration Automation, Deployment Automation, Continuous Integration, Continuous Deployment

Discussion

5.1.1 Cloud Automation

Cloud computing provides computing resources and services such as virtual machines, storage, databases, networks, and software through the internet. These resources can be provisioned and managed quickly, but managing a large number of cloud resources manually can become time-consuming and difficult. Repeated manual operations may also result in configuration errors and inconsistencies.

Cloud automation refers to the use of software tools, scripts, and predefined processes to perform cloud-related tasks automatically with minimal manual intervention. It can be used to provision infrastructure, configure systems, deploy applications, manage resources, and perform routine operational tasks.

For example, consider an organization that needs to create several virtual machines with the same operating system, software, and configuration. Creating and configuring each machine manually requires the same steps to be repeated several times. An automated process can perform these operations according to predefined instructions, reducing repetitive work and maintaining greater consistency among the machines.

Cloud automation can operate at different levels. It may involve creating cloud infrastructure, configuring the software environment, deploying an application, or managing resources after deployment. The level of automation depends on the requirements of the organization and the tasks that need to be performed.

5.1.1.1 Need for Cloud Automation

Cloud environments are often dynamic. Resources may need to be created, modified, scaled, updated, or removed as application requirements change. When these activities are performed manually, the time and effort required increase with the number of resources.

For instance, a small application may initially require only one server. As the number of users increases, additional servers may be required. If every server has to be created and configured manually, the process can become slow and prone to errors. Automation makes such repetitive operations easier to manage.

Automation is particularly valuable when the same task has to be performed frequently or across a large number of resources. It also helps organizations establish standardized procedures for managing their cloud environments.

SGOU

5.1.1.2 Significance of Cloud Automation

The importance of cloud automation arises mainly from the need to manage cloud resources efficiently and consistently.

- **Faster operations:** Automated processes can complete repetitive tasks much faster than manual execution. For example, a predefined process can provision several cloud resources without requiring an administrator to configure each resource individually.
- **Reduced manual effort:** Routine activities such as resource provisioning, software installation, configuration, and deployment can be automated. This allows IT personnel to spend more time on tasks that require planning, analysis, and decision-making.
- **Improved consistency:** When the same automated instructions are used repeatedly, resources are more likely to receive the same configuration. This is particularly important when maintaining development, testing, and production environments.
- **Fewer human errors:** Manual configuration may involve errors such as entering incorrect values, skipping a step, or applying different settings to similar resources. Automation can reduce errors associated with repetitive operations. However, the automation process itself must be properly designed and tested.
- **Better scalability:** Cloud environments can expand or shrink according to workload requirements. Automation makes it easier to manage such changes when the number of resources increases or decreases.
- **Efficient resource management:** Automated processes can help organizations manage the lifecycle of cloud resources, including their creation, modification, and removal. This can contribute to better utilization of cloud resources.

Consider an online learning platform hosted in the cloud. During normal usage, it may require a limited number of application servers. During an examination or admission period, the number of users may increase significantly, requiring additional computing resources.

If administrators manually create and configure every additional server, responding to increased demand may take considerable time. With appropriate automation, predefined procedures can be used to provision and configure the required resources more efficiently.

Similarly, when the additional workload is no longer present, automated processes can assist in reducing unnecessary resources.

Cloud automation does not completely replace human involvement. Human decisions are still necessary to determine which resources are required, how they should be

SGOU

configured, what security policies should be followed, and when particular operations should occur.

The role of automation is to execute well-defined tasks reliably and repeatedly.

For example, an administrator may decide that a particular application requires three application servers. The actual creation and configuration of those servers can then be performed automatically using suitable tools.

Thus, effective cloud automation combines appropriate human decisions with automated execution.

5.1.1.3 Areas of Cloud Automation

Cloud automation can be used throughout the lifecycle of cloud resources and applications. Important areas include:

- **Infrastructure provisioning:** Creating virtual machines, networks, storage, databases, and other cloud resources.
- **Configuration management:** Installing software and applying required system and application settings.
- **Application deployment:** Delivering new applications or updated versions to cloud environments.
- **Resource management:** Starting, stopping, modifying, or removing resources according to predefined requirements.
- **Monitoring and operational tasks:** Automating selected responses to changes or events in the cloud environment.

These areas are closely connected. Creating infrastructure, configuring it, and deploying an application are different activities, but they can all be automated as part of a broader cloud management process.

5.1.2 Types of Cloud Automation

Cloud automation can be used for different purposes. As discussed in the previous section, automation can help in creating cloud resources, preparing them for use, deploying applications, and managing resources. Based on the activity being automated, cloud automation can be broadly classified into three types: infrastructure automation, configuration automation, and deployment automation.

These types are closely related, but each one performs a different role in the cloud environment.

5.1.2.1 Infrastructure Automation

Infrastructure automation involves automatically creating, modifying, and managing the basic resources required to run an application in the cloud.

Cloud infrastructure may include:

SGOU

- Virtual machines
- Virtual networks
- Subnets
- Storage
- Databases
- Load balancers

For example, consider an application that requires two virtual machines, one database, and a network. These resources can be created manually through a cloud provider's management interface. However, when the same type of infrastructure is required repeatedly, creating each resource manually takes more time and may result in differences between environments.

Infrastructure automation allows these resources to be created according to predefined requirements.

The main purpose of infrastructure automation is to make the provisioning and management of cloud infrastructure faster, consistent, and repeatable.

Infrastructure automation is closely related to a major cloud automation concept called Infrastructure as Code (IaC). In IaC, infrastructure requirements are described using machine-readable files, which can then be processed by suitable automation tools. IaC is discussed in detail in a later section.

5.1.2.2 Configuration Automation

Creating a cloud resource is only the first step. A newly created virtual machine, for example, may not have all the software and settings required to run an application.

Configuration automation is concerned with automatically setting up and maintaining the required configuration of cloud resources.

Configuration tasks may include:

- Installing software packages
- Configuring a web server
- Setting environment variables
- Creating directories and files
- Applying system settings
- Configuring application dependencies

For example, after a virtual machine is created, a web application may require a

SGOU

web server and certain software libraries. Instead of installing and configuring these components manually on every machine, an automated process can perform the same configuration consistently.

Configuration automation is especially useful when many systems need the same setup. It helps reduce differences between systems and makes it easier to maintain a standard configuration.

5.1.2.3 Deployment Automation

Once the required infrastructure has been created and configured, the application needs to be placed in that environment. This is the role of deployment automation.

Deployment automation uses automated processes to move an application or a new version of an application into a target environment.

A deployment process may involve activities such as:

1. Obtaining the application code from a source repository.
2. Building the application.
3. Performing required tests.
4. Preparing the application for release.
5. Deploying it to the target environment.
6. Verifying the deployment.

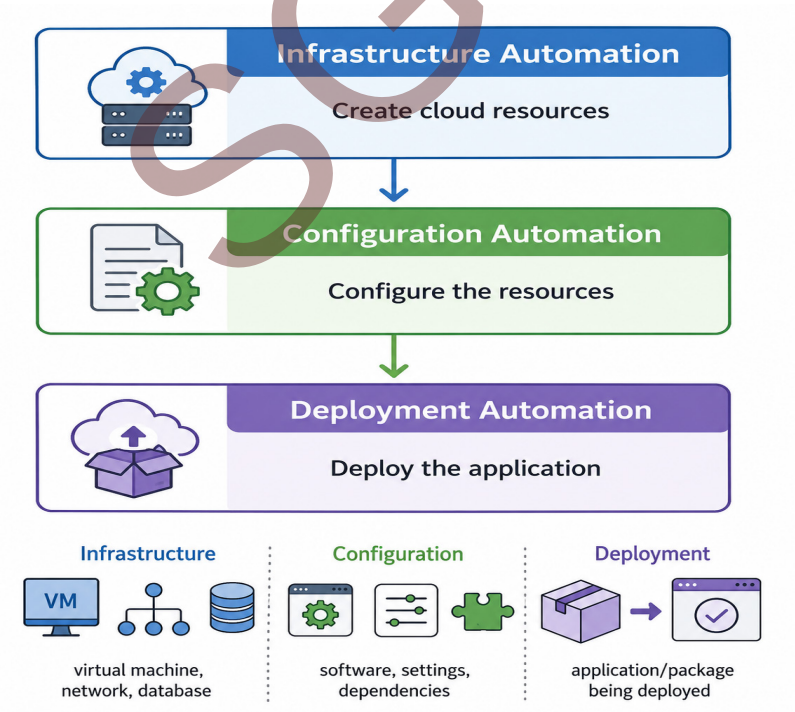


Fig. 5.1.1. Types of Cloud Automation

SGOU

For example, when a new version of a web application is developed, the application files do not have to be copied manually to every server. An automated deployment process can perform the required steps according to predefined rules.

Deployment automation is therefore closely connected with Continuous Integration and Continuous Deployment (CI/CD), which will be discussed later in this unit.

5.1.2.4 Comparing the Types of Cloud Automation

Although infrastructure, configuration, and deployment automation are different, they often work together.

Table 5.1.1 Comparison of different types of Cloud Automation

Type of automation	Main purpose	Example
Infrastructure automation	Create and manage cloud resources	Creating a virtual machine
Configuration automation	Set up and maintain systems	Installing and configuring a web server
Deployment automation	Deliver applications to an environment	Deploying a new application version

Consider a simple web application. First, the required virtual machine, network, and storage are created. Next, the operating system and required software are configured. Finally, the application is deployed.

Thus, a single cloud application may use all three types of automation at different stages.

Understanding these three types makes it easier to understand how infrastructure itself can be managed through code. This leads to the concept of Infrastructure as Code (IaC).

5.1.3 Infrastructure as Code (IaC)

As discussed earlier, infrastructure automation involves creating and managing cloud resources using automated processes. Infrastructure as Code (IaC) takes this idea further by representing infrastructure requirements in machine-readable files and using these definitions to provision and manage cloud resources.

In traditional cloud management, an administrator may use a cloud provider's management console to create a virtual machine, configure a network, or set up a database. When many resources are involved, performing these tasks manually can take considerable time. It can also be difficult to remember and reproduce all the settings used for a particular environment.

With IaC, the required infrastructure is described in configuration files. An appropriate IaC tool reads these definitions and performs the necessary operations to create or modify the cloud resources.

SGOU

For example, an application may require two virtual machines, a database, a network, and storage. Instead of creating each resource manually, these requirements can be specified in an IaC definition and used to provision the required infrastructure.

The important idea is that cloud infrastructure can be defined, stored, reviewed, modified, and reused in a structured manner.

Managing cloud infrastructure manually becomes increasingly difficult when an organization has many resources or needs to create similar environments repeatedly. Infrastructure as Code provides a way to describe these resources in a structured form and use the same description whenever required.

One of the major purposes of IaC is to bring **consistency** to infrastructure management. When the same definition is used to create different environments, the possibility of accidental differences is reduced.

For example, an organization may maintain separate development, testing, and staging environments. These environments may not need exactly the same amount of computing resources, but their basic structure can be based on a common infrastructure definition. This makes the environments easier to create and maintain.

IaC is also useful when an existing environment has to be recreated. Instead of depending on an administrator's memory of the earlier configuration, the infrastructure definition can be used as a reliable reference.

Thus, IaC helps organizations manage infrastructure in a way that is consistent, repeatable, and easier to maintain.

5.1.3.1 How IaC Works

An IaC-based environment begins with a description of the infrastructure that is required. This description contains information about resources and their properties, such as the type of virtual machine, network to be used, storage requirements, and relationships between resources.

An IaC tool processes this description and communicates with the cloud platform to create or modify the required resources.

A simplified workflow is:

Infrastructure definition → Validation and review → Resource provisioning → Verification

The definition is normally checked before changes are made. Depending on the tool, it may also be possible to preview the changes that will be made to the existing infrastructure.

After the changes are applied, the resulting environment can be checked to ensure that it matches the intended configuration.

Suppose an organization specifies that a particular application requires two virtual machines connected to a particular network. The IaC tool uses this information to create

SGOU

or modify the required resources rather than requiring an administrator to perform each operation manually.

The infrastructure definition can later be updated when requirements change. This makes infrastructure management an ongoing and controlled process rather than a one-time activity.

5.1.3.2 Declarative approach to IaC

A key idea in IaC is the declarative approach.

In a declarative approach, the required result is described without specifying every operation needed to achieve it. The automation tool determines the necessary actions.

For example, a requirement may state:

Two application servers should be connected to a specified network.

The definition describes the required condition. The tool determines whether the servers and network already exist and what changes are required to achieve the specified state.

An imperative approach, in contrast, focuses on the sequence of actions:

1. Create the network.
2. Create the first server.
3. Create the second server.
4. Connect the servers to the network.

The difference is therefore:

Declarative Approach - What the infrastructure should be

Imperative Approach - How the infrastructure should be created

Declarative definitions are widely used in IaC because they allow infrastructure requirements to be expressed independently of the individual steps needed to create them.

5.1.3.3 Desired State

The **desired state** refers to the condition in which the infrastructure is expected to be.

Suppose a cloud environment is intended to contain two application servers, one database, and a particular network configuration. These requirements together describe the desired state.

The actual environment may be different. Perhaps only one server currently exists, or a network setting has been changed. IaC tools can identify the difference between the

SGOU

intended configuration and the existing resources and determine the changes required, depending on the tool and its configuration.

This approach is useful because infrastructure requirements are explicitly recorded. The organization does not have to depend entirely on manually maintained instructions to know what the environment is supposed to contain.

The desired-state concept is also important when infrastructure changes over time. Whenever requirements change, the infrastructure definition can be updated to represent the new desired state.

5.1.3.4 IaC Tools

Different tools are available for implementing Infrastructure as Code. Some widely used examples are:

Table 5.1.2 IaC Tools

Tool	Description
Terraform	Used to define and manage infrastructure across multiple cloud and other environments
AWS CloudFormation	Used to define and provision resources on AWS using templates
Azure Resource Manager templates	Used to describe and deploy Azure resources
Pulumi	Allows infrastructure to be defined using general-purpose programming languages

These tools differ in their syntax, supported platforms, and features. However, they all support the broader idea of managing infrastructure through definitions that can be processed by automation systems.

For introductory study, it is more important to understand the role of an IaC tool than to memorize the commands or syntax of individual tools.

A Simple Example

A university is preparing an online examination system. The system requires application servers, a database, storage, and a network.

If the infrastructure is created manually, an administrator has to configure each resource through the cloud platform. If another examination environment is required later, the same work may have to be repeated.

With IaC, the required infrastructure can be described in a configuration file. The file can then be used by an IaC tool to provision the required resources.

If another environment with a similar structure is needed, the same definition can be

SGOU

reused with suitable modifications.

5.1.4 Continuous Integration and Continuous Deployment (CI/CD)

An application does not remain unchanged after it is deployed. Developers continuously add new features, correct errors, improve performance, and fix security issues. As these changes increase, releasing every new version manually becomes difficult.

Consider an application developed by a team of five developers. One developer works on the login page, another works on payment processing, and others work on different features. Each developer may modify the application at different times. If all these changes are combined only at the end, problems may arise when the different parts are brought together.

A better approach is to integrate the changes frequently and check them automatically. This is the idea behind Continuous Integration (CI).

5.1.4.1 Continuous Integration

Continuous Integration is a software development practice in which developers frequently add their changes to a shared code repository. Whenever new code is added, automated processes can build the application and run tests to check whether the changes have introduced problems.

Suppose a developer adds a new feature to an application and uploads the updated code to the project repository. The CI system can automatically:

- obtain the latest code;
- build the application;
- run predefined tests;
- report whether the checks were successful.

If a test fails, the developers can investigate the problem immediately.

This frequent checking is useful because software problems are generally easier to find when they are introduced than after many changes have accumulated.

For example, if a developer changes the payment module and the change causes another part of the application to stop working, automated testing can reveal the problem soon after the change is submitted. The developer can then correct it before the problem spreads further into the development process.

Thus, Continuous Integration encourages developers to integrate small changes frequently and receive quick feedback about those changes.

SGOU

5.1.4.2 From Integration to Delivery

Building and testing the application is not the final goal. Once the software has been successfully tested, it needs to be made available to users.

This raises another question: Can the tested application be prepared for release automatically?

Yes. This is where Continuous Delivery comes in.

Continuous Delivery extends Continuous Integration by automatically taking the successfully built and tested application through the steps required to make it ready for release.

For example, after the application passes its tests, it can be packaged and placed in a release-ready environment. The organization may then require a person to approve the production release.

The important point is that the software is kept ready to be released at any time.

A Continuous Delivery process may therefore look like as in Fig. 5.1.2.

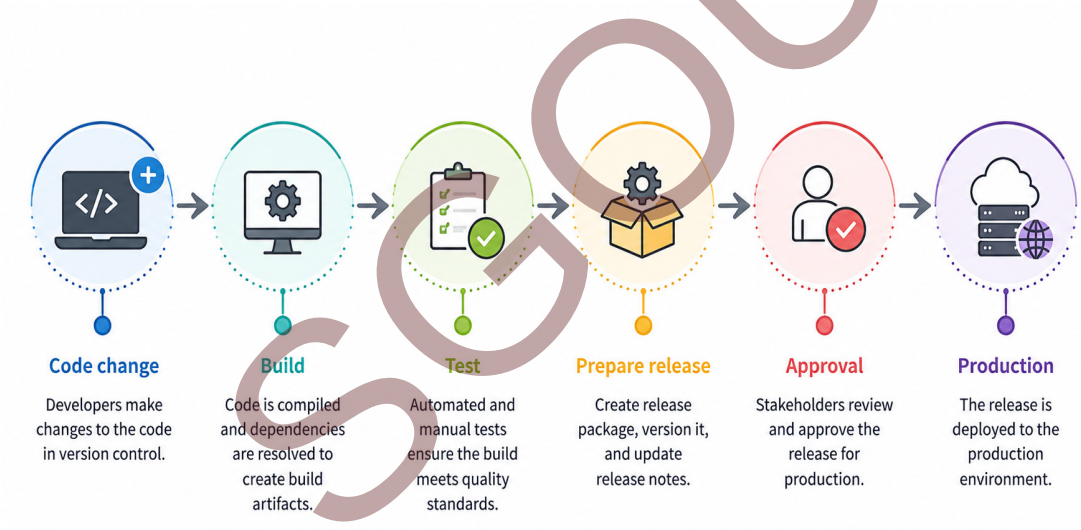


Fig. 5.1.2 CI/CD Workflow

The approval step is optional in some automated workflows, but it is a key distinction when comparing Continuous Delivery with Continuous Deployment.

5.1.4.3 Continuous Deployment

What if the organization does not want a person to approve every successful release?

In Continuous Deployment, a change that passes all required automated checks can be deployed directly to the production environment.

For example, imagine an online shopping application. A developer makes a small improvement to the product-search feature. The code is submitted to the repository. The

SGOU

system builds the application and runs the required tests. If everything passes, the new version is automatically deployed to the live application.

There is no separate manual deployment step for that change. This makes Continuous Deployment particularly useful for applications that need frequent releases. The difference can be understood through a simple example:

Continuous Delivery:

The application is tested and kept ready. A release decision may still be required.

Continuous Deployment:

The application is tested and, if successful, is automatically released.

Thus, Continuous Deployment takes the automation of software delivery one step further.

5.1.4.4 CI/CD in Cloud Applications

CI/CD becomes particularly useful when applications are hosted in the cloud. Cloud applications are updated frequently, and users expect new features and fixes to become available without long delays.

Consider a web application hosted on a cloud platform. A developer corrects an error in the application and submits the updated code. The CI/CD pipeline can automatically build the new version, test it, and deploy it to the cloud environment when the required conditions are satisfied.

The developer therefore does not have to manually copy the updated application to the server every time a change is made.

This also improves consistency. The same deployment procedure can be followed for every release instead of depending on someone remembering and performing a series of manual steps correctly.

The real value of CI/CD is not simply that it makes software deployment faster. It changes the way software changes are handled.

Frequent integration helps identify problems early. Automated testing provides quick feedback. Automated delivery reduces repetitive deployment work. Together, these practices help teams release software in a more controlled and consistent manner.

However, automation must be supported by appropriate testing and security practices. A pipeline that automatically releases poorly tested software can make problems reach users faster rather than preventing them.

There is also an important connection with Infrastructure as Code. IaC prepares and manages the cloud infrastructure, while CI/CD manages the movement of application changes through building, testing, and deployment. Understanding how these two approaches complement each other gives a clearer picture of automation in a complete cloud environment.

SGOU

Recap

- Cloud automation reduces repetitive manual work.
- Automation helps manage cloud resources efficiently.
- Cloud automation can improve speed and consistency.
- Infrastructure automation creates and manages cloud resources.
- Configuration automation sets up systems and software.
- Deployment automation delivers applications to target environments.
- Infrastructure as Code represents infrastructure through machine readable definitions.
- IaC makes infrastructure provisioning more consistent.
- IaC allows infrastructure definitions to be reused.
- IaC supports repeatable infrastructure management.
- A declarative approach describes the desired infrastructure state.
- The desired state specifies how the infrastructure should exist.
- IaC definitions can be maintained using version control.
- Continuous Integration frequently integrates code changes.
- CI automatically builds and tests code changes.
- Continuous Delivery keeps tested software ready for release.
- Continuous Deployment automatically releases successful changes to production.
- A CI/CD pipeline connects code changes with build, testing, and deployment.
- IaC mainly manages infrastructure, while CI/CD mainly manages application delivery.
- IaC and CI/CD can work together to automate cloud operations.

SGOU

Objective Type Questions

1. What refers to the automatic management of cloud resources and operations?
2. Which type of automation is concerned with creating and managing cloud resources?
3. Which type of automation is concerned with setting up software and system settings?
4. Which type of automation is concerned with delivering applications to a target environment?
5. What is the abbreviation for Infrastructure as Code?
6. What approach describes the desired state of infrastructure?
7. What is the term for the condition in which infrastructure is expected to exist?
8. Name one popular IaC tool used across different cloud environments.
9. Which AWS service is used for defining and provisioning AWS resources through templates?
10. Which version control system is commonly used to maintain IaC files?
11. What practice involves frequently integrating code changes into a shared repository?
12. What does CI stand for?
13. What does CD stand for in the context of CI/CD?
14. Which process automatically builds and tests newly integrated code?
15. Which approach automatically deploys successfully tested changes to production?
16. What is the term for the sequence of automated stages used to build, test, and deploy software?
17. Which approach keeps software ready for release but may require approval before production deployment?
18. Which IaC approach focuses on what the infrastructure should look like rather than how to create it?
19. Which type of automation mainly manages application delivery?
20. Which type of automation mainly manages cloud infrastructure?

SGOU

Answers

1. Cloud automation
2. Infrastructure automation
3. Configuration automation
4. Deployment automation
5. IaC
6. Declarative
7. Desired state
8. Terraform
9. CloudFormation
10. Git
11. Continuous Integration
12. Continuous Integration
13. Continuous Deployment
14. Continuous Integration
15. Continuous Deployment
16. Pipeline
17. Continuous Delivery
18. Declarative
19. Deployment automation
20. Infrastructure automation

SGOU

Assignments

1. Explain the concept of cloud automation and discuss its importance and benefits in cloud computing.
2. Explain the three types of cloud automation: infrastructure automation, configuration automation, and deployment automation, with suitable examples.

3. Explain Infrastructure as Code (IaC) and discuss its purpose, working, benefits, and commonly used IaC tools.

4. An organization is manually creating and configuring cloud resources for its development, testing, and production environments. The environments often have configuration differences and application updates take considerable time. Analyse how IaC and CI/CD can be used to improve this situation.

5. A web application receives frequent updates from a development team. Currently, developers manually test the application and deploy each update to the cloud server. Analyse the problems with this approach and propose a suitable CI/CD workflow to make the process more efficient and reliable.

Reference

1. Krief, M. (2022). *Learning DevOps: A comprehensive guide to accelerating DevOps culture adoption with Terraform, Azure DevOps, Kubernetes, and Jenkins* (2nd ed.). Packt Publishing.

2. Dehury, C. K., & Srirama, S. N. (2026). *Mastering DevOps: A cloud engineering and data science perspective*. Elsevier.

3. Morris, K. (2021). *Infrastructure as Code* (2nd ed.). O'Reilly Media.

4. Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (Eds.). (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media.

5. Kim, G., Humble, J., Debois, P., & Willis, J. (2021). *The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations* (2nd ed.). IT Revolution Press.

6. Pandya, S., & Guha Thakurta, R. (2022). *Introduction to infrastructure as code: A brief guide to the future of DevOps*. Apress.

7. Campbell, B. (2020). *The definitive guide to AWS infrastructure automation: Craft infrastructure-as-code solutions*. Apress.

8. Wang, I. (2025). *Terraform made easy: Provisioning, managing and automating cloud infrastructure with Terraform on Google Cloud*. Apress.

SGOU

Suggested Reading

1. Brikman, Y. (2019). *Terraform: Up & running: Writing infrastructure as code* (2nd ed.). O'Reilly Media.
2. Humble, J., & Farley, D. (2010). *Continuous delivery: Reliable software releases through build, test, and deployment automation*. Addison Wesley Professional.
3. Morris, K. (2020). *Infrastructure as code: Managing servers in the cloud* (2nd ed.). O'Reilly Media.
4. Sabharwal, N., Pandey, S., & Pandey, P. (2021). *Infrastructure as code automation using Terraform, Packer, Vault, Nomad and Consul: Hands-on deployment, configuration, and best practices*. Apress.
5. Patil, A., & Soni, M. (2022). *Infrastructure automation with Terraform*. BPB Publications.

SGOU

Unit 2
Infrastructure Automation Tools

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the purpose and role of infrastructure automation tools in cloud computing
- describe the features and working of infrastructure automation tools
- differentiate between major infrastructure automation tools based on their features and usage
- explain the structure and purpose of infrastructure templates
- analyse how automation supports the provisioning and scaling of cloud resources

Prerequisite

Cloud computing makes it easy to create resources such as virtual machines, networks, storage, and databases. But when an application grows, the number of resources also grows. Creating and managing each resource manually can take a lot of time, and even a small configuration mistake can affect the performance or availability of an application.

The situation becomes more challenging when similar environments are needed for development, testing, and production. Creating each environment separately may result in differences between them. An infrastructure automation tool can help by using a structured definition of the required infrastructure and reusing it whenever necessary.

Different cloud environments also bring different choices. Terraform can work with multiple infrastructure providers, while AWS CloudFormation is closely associated with AWS and Azure Resource Manager provides infrastructure management for Azure. Understanding these tools helps in choosing an appropriate approach for different cloud requirements.

But how does a simple configuration file become a complete cloud environment? How can the same infrastructure be created repeatedly? How can resources be increased when demand rises and reduced when demand falls?

SGOU



These questions lead us to the world of **Infrastructure Automation Tools**. Let us explore how Terraform, AWS CloudFormation, and Azure Resource Manager make cloud infrastructure easier to provision, manage, and scale.

Keywords

Terraform, CloudFormation, Azure Resource Manager, Infrastructure Automation, Infrastructure Templates, Resource Provisioning, Resource Scaling, Infrastructure as Code

Discussion

5.2.1 Introduction to Infrastructure Automation Tools

Cloud infrastructure consists of resources such as virtual machines, networks, storage, databases, load balancers, and security components. As cloud applications grow, the number of resources that need to be created and managed also increases. Managing each resource manually can become difficult, particularly when the same type of infrastructure is required repeatedly.

Infrastructure automation tools provide a systematic way to manage these resources. They allow infrastructure requirements to be described in a structured form and use those descriptions to create, modify, and manage cloud resources.

Consider an organization that needs similar infrastructure for development, testing, and production. Creating every resource manually for each environment can take considerable time. There is also a possibility of differences between the environments because of accidental changes or missed configuration steps. Automation tools help reduce such problems by allowing infrastructure requirements to be defined and reused.

Infrastructure automation is therefore concerned not only with creating resources but also with managing changes to those resources throughout their lifecycle.

The need for infrastructure automation becomes clear as cloud environments become larger and change more frequently. Manual infrastructure management may involve several activities, such as creating virtual machines, configuring networks, creating storage, setting up databases, modifying resource configurations, and increasing or decreasing resources according to demand. Performing these activities repeatedly requires considerable time and careful attention. When many resources are involved, even a small mistake can affect the availability or performance of an application. Infrastructure automation tools help address these challenges by making resource management more consistent, repeatable, and efficient.

5.2.2 Overview of Infrastructure Automation Tools

Several tools are available for automating cloud infrastructure. The choice of tool often depends on the cloud platform being used, the organization's requirements, and the preferred method of defining infrastructure.

SCOU

Three important technologies in this area are Terraform, AWS CloudFormation, and Azure Resource Manager.

Terraform is commonly used when infrastructure needs to be managed across multiple cloud platforms. AWS CloudFormation is designed specifically for managing resources within Amazon Web Services. Azure Resource Manager provides the resource management framework for Microsoft Azure.

Before examining each tool separately, it is useful to understand why different tools exist.

Cloud providers offer their own infrastructure management services because each platform has its own resources, APIs, security mechanisms, and management requirements.

At the same time, organizations may use more than one cloud platform. A company might use AWS for one application and Azure for another. In such situations, a tool that supports multiple providers can be useful.

This creates two broad approaches.

A cloud specific approach uses the infrastructure management facilities provided by a particular cloud platform.

A multi cloud approach uses a tool that can work with resources from different providers.

Neither approach is universally better. The appropriate choice depends on factors such as cloud strategy, infrastructure complexity, team expertise, and the resources that need to be managed.

This distinction helps explain the different roles of Terraform, CloudFormation, and Azure Resource Manager.

SGOU

5.2.3 Terraform

Terraform is an infrastructure as code tool developed by HashiCorp. It is used to define, provision, and manage infrastructure using configuration files.

One of Terraform's important characteristics is its ability to work with resources from multiple infrastructure providers. This makes it useful in environments where an organization uses more than one cloud platform or combines cloud services with other infrastructure.

Terraform uses configuration files to describe the infrastructure that is required. These files can contain information about resources, their properties, and relationships between them.

For example, a configuration may describe a virtual network and the virtual machines that should be connected to that network. Terraform processes this configuration and uses the appropriate provider to communicate with the infrastructure platform.

5.2.3.1 Terraform Configuration

Terraform configurations are generally written using **HashiCorp Configuration Language (HCL)**, although Terraform also supports JSON syntax.

A configuration describes resources rather than requiring the user to manually perform each provisioning operation.

A simplified conceptual example might describe:

- a virtual network;
- a subnet;
- a virtual machine;
- storage associated with the machine.

Terraform uses these definitions to determine what infrastructure needs to be created or changed.

The important point is that the configuration represents the intended infrastructure. The actual syntax of Terraform can be learned separately, but understanding the relationship between configuration and infrastructure is more important at this stage.

5.2.3.2 Terraform Providers

Terraform interacts with different platforms through providers.

A provider acts as an interface between Terraform and a particular infrastructure platform or service. For example, Terraform can use providers for major cloud platforms and many other technologies.

The provider enables Terraform to understand how to communicate with the relevant platform and manage its resources.

This provider based architecture is one reason Terraform can be used beyond a single cloud provider.

5.2.3.3 Terraform State

Terraform maintains information about the infrastructure it manages in a **state**.

The state helps Terraform understand the relationship between the configuration and the resources that currently exist.

Suppose a configuration specifies three virtual machines. Terraform needs to know which resources correspond to those definitions. State information helps it track these relationships and determine what changes may be necessary when the configuration is modified.

State management is therefore an important part of Terraform.

SGOU

Because state may contain sensitive information depending on the resources and configuration involved, it should be stored and managed securely, especially when multiple people work on the same infrastructure.

5.2.3.4 Terraform Workflow

A typical Terraform workflow involves several stages.

First, the infrastructure requirements are defined in configuration files. The configuration is then checked for syntax and structural problems.

Terraform can subsequently generate a plan showing the changes it expects to make. This plan can be reviewed before the changes are applied.

After approval, Terraform applies the required changes to the infrastructure.

This workflow provides an opportunity to examine proposed infrastructure changes before they are made. Terraform therefore supports a controlled approach to infrastructure management rather than requiring every change to be performed manually.

5.2.4 AWS CloudFormation

Organizations that use Amazon Web Services can use AWS CloudFormation to define and manage AWS infrastructure.

CloudFormation allows AWS resources to be described using templates. These templates can specify resources such as compute instances, storage, networks, databases, and other AWS services.

Instead of creating each AWS resource separately through the AWS Management Console, a template can describe the required infrastructure. CloudFormation then provisions the resources according to the template.

This approach is especially useful when the same AWS infrastructure needs to be created repeatedly.

5.2.4.1 CloudFormation Templates

A CloudFormation template is a structured description of AWS resources.

Templates can be written using formats such as YAML or JSON.

A template may specify:

- the resources to be created;
- properties of those resources;
- relationships between resources;
- parameters that allow values to be supplied when the template is used;
- outputs that provide information about created resources.

SGOU

For example, a template may define a virtual network together with subnets and computing resources that belong to it.

The template acts as a reusable description of the required AWS environment.

5.2.4.2 CloudFormation Stacks

CloudFormation manages groups of related AWS resources as **stacks**.

A stack can contain all the resources required for a particular application or service.

For example, an application stack may contain:

- networking resources;
- application servers;
- a database;
- storage;
- security related resources.

Managing these resources as a group makes it easier to create, update, and remove an application environment.

This is an important difference from treating every cloud resource as an isolated object. The resources required for an application can be managed as parts of one infrastructure definition.

5.2.4.3 CloudFormation and AWS Resource Management

CloudFormation is closely integrated with AWS services. This allows it to manage many AWS resources through templates.

When a template is updated, CloudFormation can determine the changes required to bring the stack to the new configuration, subject to the capabilities and behaviour of the relevant AWS resources.

This makes CloudFormation useful for maintaining infrastructure throughout its lifecycle rather than only for initial provisioning.

The AWS specific nature of CloudFormation distinguishes it from a multi provider tool such as Terraform.

5.2.5 Azure Resource Manager

Microsoft Azure provides Azure Resource Manager (ARM) as its management and deployment service for Azure resources.

Azure Resource Manager provides a consistent way to create, manage, organize, and control Azure resources.

SGOU

Resources in Azure can include virtual machines, storage accounts, virtual networks, databases, and many other services.

ARM enables these resources to be managed as part of a structured deployment rather than requiring every resource to be handled separately.

5.2.5.1 ARM Templates

Infrastructure can be described using Azure Resource Manager templates.

These templates are based on JSON and specify the Azure resources that should be deployed and their properties.

An ARM template may contain information about:

- resources;
- resource properties;
- parameters;
- variables;
- dependencies;
- outputs.

For example, a template can describe a virtual network and the resources that need to be deployed within that environment.

The template can then be reused to create similar Azure environments.

5.2.5.2 Resource Groups

A key concept in Azure resource management is the **resource group**.

A resource group is a logical container for Azure resources that belong to a particular solution or workload.

For example, an application may have a virtual machine, storage account, network resources, and other related services. These resources can be placed in a resource group and managed together.

Resource groups make it easier to organize resources, apply management policies, monitor usage, and control access.

5.2.5.3 Azure Resource Manager and Deployment

ARM supports the deployment and management of Azure resources according to defined configurations.

When an infrastructure definition is changed, the deployment process can apply the required modifications to the Azure environment.

SGOU

This makes infrastructure management more organized and reduces dependence on repeated manual operations.

Azure Resource Manager therefore provides a platform specific approach to infrastructure automation, just as CloudFormation does for AWS.

5.2.6 Infrastructure Templates

An infrastructure automation tool becomes useful only when it knows what infrastructure has to be created. This information is provided through an infrastructure template. A template is a structured document that describes the cloud resources required for an application and the properties of those resources.

Instead of manually creating a virtual machine, configuring storage, connecting networks, and setting security settings every time, these requirements are written once in a template. The automation tool reads the template and provisions the infrastructure accordingly.

Consider a university that conducts online examinations every semester. The examination platform always requires application servers, a database, storage, and a virtual network. Rather than creating these resources repeatedly for every examination session, the university can maintain a single infrastructure template and reuse it whenever a new examination environment is needed.

An infrastructure template therefore acts as a blueprint for building cloud infrastructure.

Elements of an Infrastructure Template

Although Terraform, CloudFormation, and Azure Resource Manager use different formats, their templates generally contain similar elements.

Table 5.2.1 Purpose of Infrastructure Template elements

Element	Purpose
Resources	Defines the cloud resources to be created
Properties	Specifies characteristics such as size, location, and configuration
Parameters	Accepts values that may change for different deployments
Variables	Stores reusable values within the template
Dependencies	Defines the order in which related resources are created
Outputs	Displays useful information after deployment

For example, a virtual machine cannot be connected to a network that does not yet

SGOU

exist. Dependencies ensure that the network is created before the virtual machine is deployed.

Why Templates are reusable?

One of the greatest advantages of infrastructure templates is reuse. A single template can support multiple environments by changing only a few parameter values.

Suppose an organization needs three environments for the same application.

Table 5.2.2 Sample Environment needs of an organization

Environment	Server Size	Purpose
Development	Small	Software development
Testing	Medium	Functional testing
Production	Large	Live application

The overall infrastructure remains similar, but the resource sizes differ. Instead of writing three separate templates, one template can be reused with different parameters. This reduces duplication and makes infrastructure easier to maintain.

5.2.6.1 Benefits of using Templates

Infrastructure templates offer several practical advantages:

- They provide a standardized method for creating cloud environments.
- Similar infrastructure can be deployed repeatedly with minimal effort.
- Changes can be recorded and reviewed through version control.
- Large infrastructures become easier to manage than manual configurations.
- New environments can be created more quickly and consistently.

A well-designed template therefore becomes a valuable organizational asset that can be reused across projects.

5.2.7 Automation of Provisioning Resources

Once an infrastructure template has been prepared, the next step is provisioning. Provisioning is the process of creating and preparing cloud resources so that they are ready for use.

In cloud computing, provisioning does not refer only to creating virtual machines. It includes allocating storage, configuring networks, assigning security rules, creating databases, and connecting these resources into a complete working environment.

For example, when a new web application is deployed, the cloud platform may need to create a virtual network, application servers, storage, and a database before the application becomes operational. Performing these tasks manually can take considerable

SGOU

time, especially when the same infrastructure is required repeatedly.

Automation tools perform provisioning by reading the infrastructure template and creating the required resources according to the defined configuration.

5.2.7.1 Provisioning Workflow

The provisioning process follows a logical sequence.

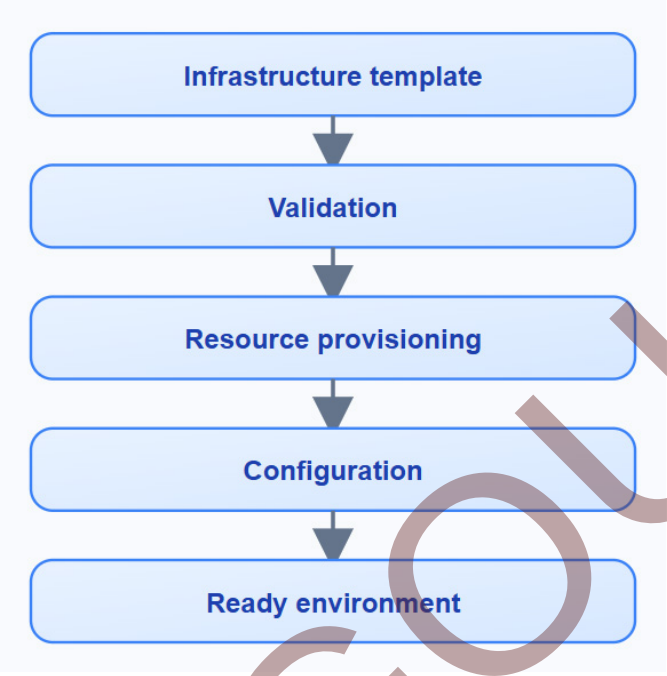


Fig. 5.2.1. Automated Resource Provisioning Workflow

The process begins with an infrastructure template. Before deployment, the configuration is validated to identify syntax or structural errors. The automation tool then provisions the required cloud resources and applies the necessary configurations. The result is a ready-to-use cloud environment.

5.2.7.2 Advantages of Automated Provisioning

Automated provisioning improves cloud operations in several ways.

- Speed: Resources can be created within minutes instead of requiring repeated manual configuration.
- Consistency: Every deployment follows the same template, reducing accidental differences between environments.
- Repeatability: The same infrastructure can be recreated whenever required.

Reduced Human Error: Automated execution minimizes mistakes caused by repetitive manual operations.

These benefits become even more important in organizations that frequently create temporary cloud environments for testing, training, or development.

SGOU

5.2.8 Automation of Scaling Resources

Creating infrastructure is only the beginning of cloud resource management. Workloads rarely remain constant. A cloud application may serve hundreds of users during normal operation and thousands during peak periods.

For example, an online admission portal may receive a sudden increase in visitors on the last day of application submission. If the available computing resources remain unchanged, users may experience slow response times or service interruptions.

To address this challenge, cloud platforms support resource scaling.

Scaling is the process of increasing or decreasing computing resources according to workload requirements. Automation allows this adjustment to happen with minimal manual intervention.

5.2.8.1 Vertical and Horizontal Scaling

Cloud platforms commonly support two approaches to scaling.

Vertical Scaling

Increases the capacity of an existing resource by adding more CPU, memory, or storage to the same virtual machine.

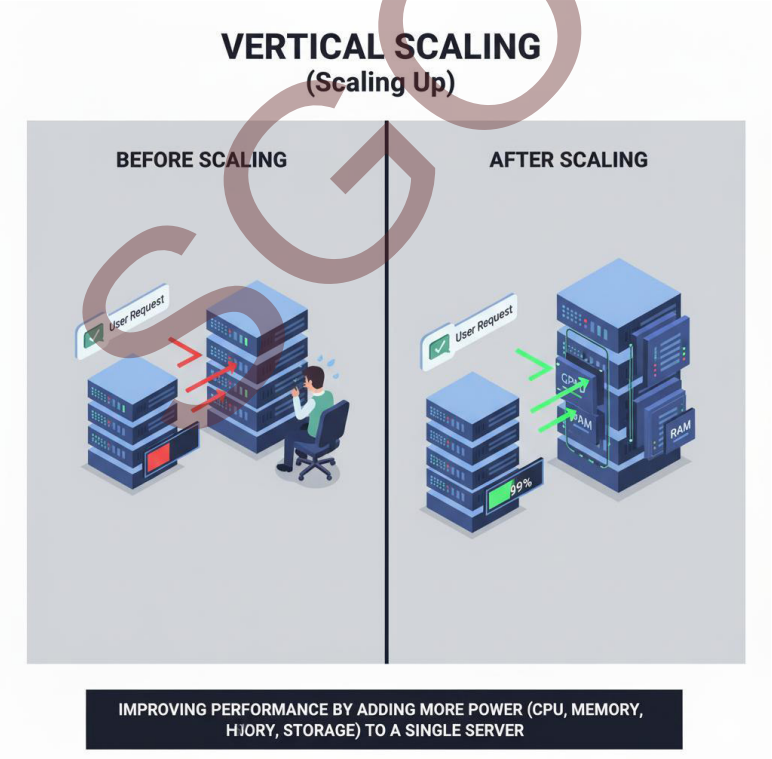


Fig.5.2.2 Vertical Scaling

Horizontal Scaling

Increases the number of resource instances by adding additional servers that share the workload.

SGOU

Vertical scaling strengthens a single resource, whereas horizontal scaling distributes the workload across multiple resources.

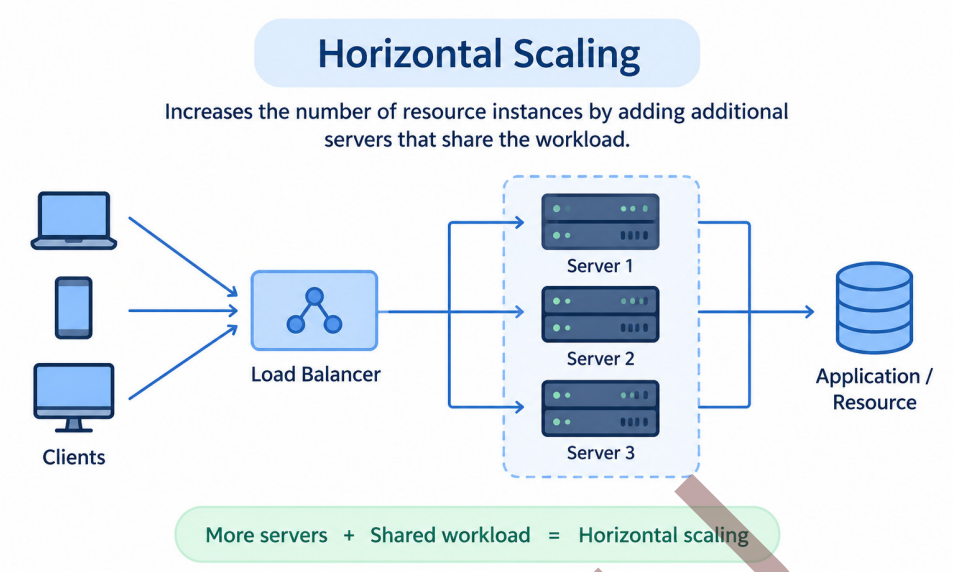


Fig. 5.2.3 Horizontal Scaling

5.2.8.2 How Automated Scaling Works

Automated scaling depends on predefined rules. These rules continuously monitor resource utilization and determine whether additional resources should be added or removed.

For example, an organization may define the following policy:

Table 5.2.3. Sample table

CPU Utilization	Action
Above 75%	Add one application server
40%–75%	Maintain current resources
Below 30%	Remove one application server

When resource utilization crosses the defined threshold, the cloud platform automatically adjusts the infrastructure. This allows applications to respond more effectively to changing demand while avoiding unnecessary resource usage.

SGOU

Recap

- Infrastructure automation reduces manual cloud management.
- Automation makes infrastructure management more consistent.
- Terraform supports infrastructure across multiple providers.
- AWS CloudFormation is designed for AWS resources.
- Azure Resource Manager manages Azure resources.
- Infrastructure templates describe required cloud resources.
- Templates can be reused for similar environments.
- Terraform uses configuration files to define infrastructure.
- Terraform uses providers to communicate with infrastructure platforms.
- Terraform maintains state information about managed resources.
- Terraform can show proposed changes before applying them.
- CloudFormation uses templates to define AWS infrastructure.
- CloudFormation manages related AWS resources as stacks.
- Azure Resource Manager uses templates for Azure deployments.
- Resource groups organize related Azure resources.
- Provisioning creates and prepares cloud resources for use.
- Automated provisioning improves speed and repeatability.
- Scaling adjusts resources according to workload requirements.
- Vertical scaling increases the capacity of an existing resource.
- Horizontal scaling increases the number of resource instances.
- Automated scaling can respond to changing resource demand.

SGOU

Objective Type Questions

1. Which tool is developed by HashiCorp for infrastructure automation?
2. Which AWS service is used to define and manage AWS infrastructure through templates?

3. What is the infrastructure management service provided by Microsoft Azure?
4. Which language is commonly used to write Terraform configurations?
5. What acts as an interface between Terraform and an infrastructure platform?
6. What does Terraform maintain to track the infrastructure it manages?
7. What is the AWS CloudFormation group of related resources called?
8. Which format is commonly used for CloudFormation templates?
9. What is the logical container for related Azure resources called?
10. Which format is used for Azure Resource Manager templates?
11. What is a structured description of required cloud infrastructure called?
12. What is the process of creating and preparing cloud resources called?
13. What type of scaling increases the capacity of an existing resource?
14. What type of scaling increases the number of resource instances?
15. What process automatically adjusts resources according to workload?
16. Which Terraform stage shows the proposed infrastructure changes?
17. Which Terraform stage applies the proposed changes?
18. What allows an infrastructure template to be used for different environments?
19. What helps track changes to infrastructure definitions?
20. What approach manages infrastructure using structured definitions?

SGOU

Answers

1. Terraform
2. CloudFormation
3. Azure Resource Manager
4. HCL
5. Provider
6. State
7. Stack
8. YAML

9. Resource Group
10. JSON
11. Template
12. Provisioning
13. Vertical
14. Horizontal
15. Scaling
16. Plan
17. Apply
18. Parameters
19. Version Control
20. IaC

Assignments

1. Explain the purpose of infrastructure automation tools and discuss their importance in cloud resource management.
2. Describe the features and working of Terraform, AWS CloudFormation, and Azure Resource Manager.
3. Explain infrastructure templates and discuss their main elements, reuse, and benefits in cloud infrastructure management.
4. An organization uses AWS for its production applications and Azure for some of its development workloads. Analyse the factors that should be considered when selecting between Terraform, AWS CloudFormation, and Azure Resource Manager for managing its infrastructure.
5. A university's online examination system experiences a large increase in users during examinations and much lower demand at other times. Analyse how automated provisioning and resource scaling can be used to maintain application performance while avoiding unnecessary resource usage.

SGOU

Reference

1. Humble, J., & Farley, D. (2010). *Continuous delivery: Reliable software releases through build, test, and deployment automation*. Addison Wesley Professional.
2. Kim, G., Humble, J., Debois, P., & Willis, J. (2021). *The DevOps handbook: How to create world class agility, reliability, & security in technology organizations* (2nd ed.). IT Revolution Press.
3. Beyer, B., Jones, C., Petoff, J., & Murphy, N. R. (Eds.). (2016). *Site reliability engineering: How Google runs production systems*. O'Reilly Media.

Suggested Reading

1. Brikman, Y. (2019). *Terraform: Up & running* (2nd ed.). O'Reilly Media.
2. Krief, M. (2022). *Learning DevOps: A comprehensive guide to accelerating DevOps culture adoption with Terraform, Azure DevOps, Kubernetes, and Jenkins* (2nd ed.). Packt Publishing.
3. Morris, K. (2020). *Infrastructure as Code* (2nd ed.). O'Reilly Media.

SGOU

Unit 3
Configuration Management Tools

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the concept, importance, and components of configuration management.
- describe the architecture and major features of Ansible, Chef, and Puppet.
- apply configuration management tools to maintain consistent configurations across multiple systems.
- demonstrate how version control, automation, and monitoring can be used to manage configuration changes.
- analyze the causes and effects of configuration drift and select suitable methods to detect and correct it.

Prerequisite

Imagine that a university operates several servers for its online learning platform. Initially, all the servers are configured with the same operating system, software versions, security settings, network configurations, and application requirements. As time passes, administrators may make different changes to individual servers. One server may receive a software update, another may have a modified configuration file, while another may have a different security setting. Gradually, the servers may become different from their original configuration, resulting in application failures, security issues, inconsistent performance, and difficulty in troubleshooting. Managing such configurations manually becomes increasingly difficult as the number of systems grows. Configuration management tools such as Ansible, Chef, and Puppet help administrators automate configuration tasks and maintain systems according to a defined desired state. Therefore, studying configuration management and configuration drift is important for understanding how modern IT infrastructure can be managed efficiently and consistently.

To study this topic effectively, learners should have a basic understanding of computer systems, operating systems, servers, software installation, configuration files, and basic system administration. Familiarity with basic networking concepts such as IP addresses, ports, and client-server communication will also be helpful. A general understanding of automation, cloud computing, version control, and DevOps can further support

SGOU

learning, although advanced knowledge is not required. Studying this topic helps learners understand how Ansible, Chef, and Puppet automate repetitive configuration tasks, maintain consistency across multiple systems, and reduce configuration errors. It also helps them understand how configuration drift occurs and how it can be detected, prevented, and corrected. These concepts are useful for developing skills in cloud computing, DevOps, infrastructure automation, server management, security, scalability, and reliable IT operations.

Keywords

Configuration Management, Configuration Item (CI), Desired State, Ansible, Chef, Puppet, Configuration Drift, Version Control, Infrastructure as Code (IaC), Automation

Discussion

5.3.1 Introduction

Imagine an organization operating 100 servers for an online education platform. All the servers are expected to use the same operating system settings, software versions, security policies, database connections, and application configurations. Initially, the administrator configures all the servers carefully. After a few months, different administrators make changes to individual servers. One server receives a software update, another has a modified configuration file, and a third server has a different security setting.

The servers that were originally identical have now become different. As a result, an application may work correctly on one server but fail on another. Finding the reason for such differences can be difficult and time-consuming. This situation demonstrates the need for configuration management.

Configuration management provides a systematic way to define, apply, monitor, and maintain the desired configuration of IT systems. Instead of depending entirely on manual administration, organizations can use automation tools to configure multiple systems consistently. Ansible, Chef, and Puppet are important tools used for configuration management and infrastructure automation.

Configuration management has become particularly important with the growth of cloud computing, DevOps, Infrastructure as Code, and CI/CD. Modern organizations need infrastructure that can be created quickly, configured consistently, monitored continuously, and recovered easily when problems occur.

5.3.2 Configuration Management

Configuration management is the systematic process of identifying, defining, controlling, monitoring, and maintaining the configuration of IT systems throughout their lifecycle.

SGOU

An IT system consists of many components. These components may include physical servers, virtual machines, operating systems, applications, databases, networks, storage systems, configuration files, and security settings. Each component may have configuration information that affects how the system operates.

A component whose configuration needs to be managed is called a Configuration Item (CI).

For example, in a web application environment, the following can be configuration items:

- Web server
- Application server
- Database server
- Operating system
- Application software
- Configuration files
- Network settings
- Storage resources
- User accounts
- Security policies

Configuration management ensures that these items are configured according to approved requirements.

Configuration management becomes increasingly important as the number of systems and applications increases. Manually configuring a small number of servers may be manageable, but manually maintaining hundreds or thousands of systems is difficult.

One of the major advantages of configuration management is consistency. When the same configuration is automatically applied to multiple systems, the possibility of differences between systems is reduced.

Configuration management also improves efficiency. Administrators do not need to repeat the same configuration activities manually on every server. Automation can perform these activities much faster.

Another important advantage is repeatability. Once a configuration has been properly defined, the same configuration can be used to create or configure another system.

Configuration management also improves reliability because automated and tested configurations reduce the possibility of human errors.

It supports security and compliance by helping organizations maintain approved security settings and configuration standards.

SGOU

Configuration management also makes troubleshooting and recovery easier. When configuration changes are tracked, administrators can identify what changed and, when necessary, return to a previously working configuration.

5.3.2.1 Desired State

The concept of desired state is central to configuration management.

The desired state describes how a system is expected to be configured and operate. It defines the conditions that the organization wants the system to maintain.

For example, an organization may define the following desired state for a web server:

- Apache web server must be installed.
- Apache service must be running.
- A specific configuration file must exist.
- Only approved users should have administrative access.
- Required security settings must be enabled.

The actual state of the server is then compared with these requirements.

Consider the following situation:

Desired State:

Apache must be installed and running.

Actual State:

Apache is installed but stopped.

There is a difference between the desired state and actual state. A configuration management tool can detect this difference and take corrective action by starting the Apache service.

Thus:

Desired State ≠ Actual State → Configuration Deviation

Configuration Management → Corrective Action → Desired State Restored

This concept helps configuration management tools maintain systems in a consistent condition.

5.3.2.2 Configuration Items

A Configuration Item (CI) is an identifiable component of an IT environment whose configuration needs to be controlled.

Configuration items can exist at different levels of granularity. A CI may be a single software package, a server, a database, or a group of interconnected systems.

SGOU

Common configuration items include:

Hardware

Physical servers, storage devices, network devices, and other infrastructure components may be configuration items.

Software

Operating systems, application software, databases, libraries, and other software components may require configuration management.

Configuration Files

Application and system configuration files contain settings that determine how software operates.

Network Configuration

IP addresses, ports, routing rules, firewall settings, and network connections may need to be controlled.

Security Configuration

User accounts, permissions, authentication settings, encryption settings, and security policies can also be configuration items.

Identifying configuration items helps organizations understand what needs to be managed and how different components depend on one another.

5.3.2.3 Configuration Management and Version Control

Configuration management works closely with version control systems. Version control records changes made to configuration files and allows teams to track, review, and restore previous versions.

For example, suppose an administrator changes the configuration of a production application and the application subsequently stops functioning correctly. If the configuration is stored in a version control system, the team can identify the change that caused the problem and restore the previous stable configuration.

A version control system provides a history of configuration changes. This makes it possible to determine:

- What configuration was changed
- When the change was made
- Who made the change
- What the previous configuration was
- Whether the change was reviewed or approved

SGOU

This improves visibility, accountability, auditing, and recovery.

Version control also supports rollback. If a newly introduced configuration causes unexpected problems, the organization can return to a previously known stable version.

Therefore, configuration management and version control together reduce the risk associated with uncontrolled configuration changes.

5.3.3 Configuration Management Tools

Configuration management tools automate the process of configuring and maintaining IT systems. They allow administrators to define the desired configuration and apply it consistently across multiple systems.

Three important configuration management tools are:

1. Ansible
2. Chef
3. Puppet

Although these tools perform similar functions, they differ in architecture, configuration methods, execution models, and operational approaches.

5.3.3.1 Ansible

Ansible is an open-source automation and configuration management tool used to automate tasks such as server configuration, application deployment, orchestration, and infrastructure management.

One of the major characteristics of Ansible is its agentless architecture. In many common Ansible deployments, managed Linux systems do not require a separate Ansible agent to be installed. Ansible commonly communicates with Linux systems using SSH and with Windows systems using appropriate Windows management mechanisms.

Ansible configurations are commonly written using YAML, which is relatively easy to read and understand. These configuration files are called playbooks.

Ansible Architecture

The main components of an Ansible environment are:

Control Node

The control node is the system from which Ansible automation is executed.

Managed Nodes

Managed nodes are the systems that Ansible configures and manages.

Inventory

SGOU

The inventory identifies the systems that Ansible manages. Systems can be grouped according to their roles, such as web servers, application servers, and database servers.

Playbooks

Playbooks are YAML files that describe the tasks to be performed.

Modules

Modules are reusable components that perform specific operations such as installing packages, managing files, creating users, and controlling services.

Example of Ansible Usage

Suppose a university operates 30 web servers. Every server must have the same web server software, configuration file, security settings, and required packages.

Instead of configuring all 30 servers manually, the administrator can create an Ansible playbook that defines the required configuration and apply it to the servers.

The same configuration can therefore be reproduced across multiple systems.

Advantages of Ansible

Ansible provides several advantages:

- Agentless architecture in common deployments
- Simple YAML-based playbooks
- Easy-to-understand configuration
- Automation of repetitive administrative tasks
- Support for configuration management and application deployment
- Ability to manage multiple systems
- Support for cloud and infrastructure automation
- Reusable playbooks and modules

5.3.3.2 Chef

Chef is an automation and configuration management platform used to define and maintain the desired configuration of infrastructure.

Chef follows an infrastructure as code approach in which infrastructure configurations are expressed using code. Administrators define how systems should be configured, and Chef applies these configurations to managed nodes.

Chef traditionally follows a client-server architecture. A central Chef server stores configuration information, while Chef clients running on managed systems communicate with the Chef server and apply the required configurations.

SGOU

The major components include:

Chef Workstation

The workstation is used by administrators and developers to create and manage Chef configurations.

Chef Server

The Chef Server stores configuration information and related policies.

Chef Client

The Chef Client runs on managed nodes and communicates with the Chef Server to obtain configuration information.

Cookbooks

A cookbook contains configuration information related to a particular application, service, or system component.

Recipes

A recipe contains instructions describing how a system should be configured.

Advantages of Chef

Chef provides:

- Infrastructure as code capabilities
- Automated configuration management
- Reusable cookbooks and recipes
- Centralized configuration management
- Support for complex infrastructure environments
- Repeatable and consistent configurations
- Integration with deployment and automation workflows

5.3.3.3 Puppet

Puppet is a configuration management and automation tool used to define, deploy, and maintain the desired state of IT infrastructure.

Puppet allows administrators to describe how systems should be configured rather than manually specifying every command required to achieve that configuration.

Puppet traditionally uses an agent-server architecture. A Puppet agent runs on managed nodes and communicates with a Puppet server, which provides the configuration information or catalog required by the agent.

SGOU

Puppet Architecture

Puppet traditionally follows an agent-server architecture.

Puppet Server

The Puppet Server manages configuration information and provides configuration data to managed systems.

Puppet Agent

The Puppet Agent runs on a managed system and communicates with the Puppet Server.

Manifests

Puppet configuration definitions are commonly written in files called manifests.

Puppet uses a declarative approach. The administrator describes what the system should look like rather than specifying every individual command that must be executed.

For example, the administrator can define:

A particular package must be installed and a particular service must be running.

Puppet determines the necessary actions to achieve this desired state.

Example of Puppet Usage

Suppose an organization has 100 application servers. All servers must have a particular software package installed and a specific service running.

The required state can be defined using Puppet manifests. Puppet can then ensure that the managed servers maintain that configuration.

If a required service stops or a configuration is changed unexpectedly, Puppet can help restore the desired configuration according to the defined policies.

Advantages of Puppet

Puppet provides:

- Automated configuration management
- Declarative configuration
- Continuous enforcement of desired state
- Centralized management
- Reusable configuration modules
- Support for large infrastructure environments
- Configuration consistency and repeatability

SGOU

Consider an organization that manages 100 Linux servers. All servers need to have the same web server software installed, the same configuration settings applied, and the required services running. Performing these tasks manually on every server would be time-consuming and may introduce errors.

With Ansible, the administrator can define the required configuration in a playbook and apply it to all servers from a central control machine. Ansible uses an agentless approach and commonly communicates with Linux systems through SSH.

With Chef, the administrator defines the desired configuration using recipes and cookbooks. Chef uses a client-server architecture in which Chef Client runs on the managed systems and communicates with the Chef Server.

With Puppet, the administrator describes the desired system state using manifests. Puppet then ensures that the managed systems follow the defined configuration.

Thus, all three tools help automate configuration management, but they differ in their architecture, configuration approach, and method of operation.

Table 5.3.1: Comparison of Ansible, Chef and Puppet

Feature	Ansible	Chef	Puppet
Main focus	Automation and configuration management	Configuration management and infrastructure automation	Configuration management and desired-state enforcement
Common architecture	Agentless	Client-server	Agent-server
Configuration	YAML playbooks	Cookbooks and recipes	Manifests
Managed agent	Generally not required in common SSH-based Linux management	Traditionally Chef Client	Traditionally Puppet Agent
Configuration style	Task-oriented	Code-based	Declarative
Major strength	Simplicity and automation	Infrastructure as Code	Desired-state management
Typical use	Automation, deployment, configuration	Infrastructure automation	Configuration enforcement

5.3.4 Configuration Drift

Configuration drift refers to the gradual difference between the desired configuration of a system and its actual configuration.

In an ideal environment, all systems should remain in the configuration state defined by organizational policies. However, configurations can change over time because

SGOU

of manual modifications, software updates, emergency fixes, unauthorized changes, installation of new software, or differences introduced during deployment.

Example of Configuration Drift

Suppose an organization has ten web servers configured identically. The desired configuration requires all servers to use the same operating system settings and the same version of a web server.

An administrator manually changes the configuration of one server to solve a temporary problem but does not apply the same change to the remaining servers. The modified server now has a different configuration from the other servers.

This difference is an example of configuration drift.

Configuration drift may be small at first, but repeated changes can gradually create significant differences between systems.

5.3.4.1 Causes of Configuration Drift

Configuration drift can occur for several reasons.

Manual changes are one of the most common causes. Administrators may directly modify system configurations without updating the central configuration definitions.

Emergency changes can also create drift. During a critical incident, administrators may make temporary changes to restore a service. If these changes are not documented and incorporated into the desired configuration, the system may remain different from other systems.

Software updates and patches can change configuration files, dependencies, or system settings.

Different deployment procedures can result in servers being configured differently. For example, two administrators may follow slightly different procedures while setting up similar servers.

Unauthorized changes may also cause configuration drift when users modify systems without following approved change-management processes.

Environmental differences can occur when development, testing, and production systems are created at different times or using different configuration procedures.

5.3.4.2 Effects of Configuration Drift

Configuration drift can create several operational and security problems.

It can result in inconsistent system behavior because two servers that are expected to be identical may operate differently.

It may cause application failures when one server has a different software version, configuration parameter, or dependency.

Configuration drift can also create security vulnerabilities. A server may accidentally

SGOU

lose an important security setting or remain unpatched because its configuration differs from the approved standard.

It can make troubleshooting difficult because administrators may not immediately know how one system differs from another.

Drift can also affect compliance. Organizations may be required to maintain specific security and configuration standards. Uncontrolled changes can cause systems to violate these requirements.

5.3.4.3 Managing Configuration Drift

Configuration drift should be detected and corrected continuously.

- **Define the Desired State**

The organization should clearly define how systems are expected to be configured.

- **Use Configuration as Code**

Configuration definitions should be represented using automation code or structured configuration files.

- **Use Version Control**

Configuration files should be maintained in a version control system so that changes can be tracked and reviewed.

- **Automate Configuration**

Tools such as Ansible, Chef, and Puppet can automatically apply approved configurations.

- **Monitor Systems**

Continuous monitoring can identify deviations between the desired and actual configuration.

- **Apply Change Management**

Changes should be properly documented, tested, reviewed, and approved.

- **Perform Regular Audits**

Periodic audits can identify unauthorized or unexpected configuration changes.

- **Restore the Desired State**

When drift is detected, corrective actions should be performed to restore the system to the approved configuration.

SGOU

5.3.4.4 Best Practices for Configuration Management

Effective configuration management requires appropriate tools as well as well-defined processes.

The desired configuration should be clearly defined and documented. Configuration files and automation code should be maintained in version control so that changes can be tracked and previous versions can be restored.

Organizations should minimize direct manual modifications to managed systems. Configuration changes should follow an established review and approval process.

Automation should be designed to be repeatable and idempotent, meaning that applying the same configuration repeatedly should produce a predictable result without causing unnecessary changes.

Sensitive information such as passwords, API keys, and encryption keys should be handled securely rather than being stored directly in publicly accessible configuration files.

Configurations should be tested before being applied to production environments. Monitoring and auditing should also be used to detect configuration drift and unauthorized changes.

5.3.4.5 Practical Scenario

Consider an online university that operates separate web servers for its student portal. Each server requires the same operating system configuration, web server software, security settings, application dependencies, and configuration files.

Initially, administrators manually configure ten servers. After several months, different administrators make changes to individual servers. One server receives a newer software version, another has a modified configuration file, and another has a different security setting. Although the servers were originally identical, their configurations have gradually become different.

This situation represents configuration drift.

The university can use a configuration management tool such as Ansible, Chef, or Puppet to define the desired configuration. The configuration can be stored in version control and applied consistently to all servers. Regular monitoring can identify deviations, and automation can restore systems to the desired state.

This approach improves consistency, reduces manual effort, simplifies troubleshooting, and helps maintain security and compliance.

SGOU

Recap

- Configuration management is the systematic process of identifying, defining, controlling, monitoring, and maintaining the configuration of IT systems.
- A Configuration Item (CI) is an identifiable component of an IT environment whose configuration needs to be controlled.
- The desired state describes how a system is expected to be configured and operate.
- Configuration management helps maintain consistency, reliability, security, efficiency, and repeatability across IT systems.
- Version control records configuration changes and supports tracking, auditing, rollback, and recovery.
- Ansible is an automation and configuration management tool that commonly uses an agentless architecture and YAML-based playbooks.
- Chef uses cookbooks and recipes to define and maintain infrastructure configuration, traditionally using a client-server architecture.
- Puppet uses manifests and a declarative approach to define and maintain the desired state of systems.
- Configuration drift occurs when the actual configuration of a system gradually differs from its desired configuration.
- Configuration drift can be caused by manual changes, emergency fixes, software updates, unauthorized changes, and different deployment procedures.
- Configuration drift can lead to inconsistent system behaviour, application failures, security vulnerabilities, troubleshooting difficulties, and compliance issues.
- Configuration drift can be managed through detection, prevention, correction, continuous monitoring, and regular auditing.
- Configuration as Code and version control help organizations maintain consistent and traceable configurations.
- Ansible, Chef, and Puppet reduce manual effort and help automate configuration across multiple systems.
- Effective configuration management supports modern IT environments such as cloud computing, DevOps, Infrastructure as Code, and automated infrastructure management.

SGOU

Objective Type Questions

1. Define the systematic process of managing the configuration of IT systems.
2. Identify the component whose configuration needs to be controlled.
3. Describe the expected configuration and operation of a system.
4. Name the configuration management tool that commonly uses YAML-based playbooks.
5. Mention the tool that uses cookbooks and recipes for configuration management.
6. Specify the tool that commonly uses manifests to define the desired state.
7. State the condition in which the actual configuration differs from the desired configuration.
8. Recognize the system used to track and manage different versions of configuration files.
9. List the approach in which infrastructure configuration is represented using code.
10. Recall the reusable YAML files used to describe Ansible tasks.
11. Indicate the Chef component that contains instructions for configuring a system.
12. Point out the Puppet files commonly used to define configuration requirements.

Answers

1. Configuration Management
2. Configuration Item (CI)
3. Desired State
4. Ansible
5. Chef
6. Puppet
7. Configuration Drift
8. Version Control

SGOU

9. Infrastructure as Code (IaC)
10. Playbooks
11. Recipes
12. Manifests

Assignments

1. Explain the concept of Configuration Management in detail
2. Describe Ansible, Chef, and Puppet as configuration management tools.
3. What is Configuration Drift? Explain the concept in detail with a suitable example
4. Explain the concept of Desired State in configuration management.
5. An organization manages a large number of servers using manual configuration, and over time the servers develop different software versions, security settings, and configuration files. Analyze the causes and effects of this configuration drift and evaluate how Ansible, Chef, or Puppet can be used to restore and maintain the desired state.

Reference

1. Wojsław, D., & Adamowicz, G. (2023). The Linux DevOps handbook. Packt Publishing.
2. Meijer, B., Hochstein, L., & Moser, R. (2022). Ansible: Up and running (3rd ed.). O'Reilly Media.
3. Smith, S. R., & Membrey, P. (2022). Beginning Ansible concepts and application: Provisioning, configuring, and managing servers, applications, and their dependencies. Apress.

SGOU

Suggested Reading

1. Sesto, V. (2022). Practical Ansible: Configuration management from start to finish. Apress.
2. Morris, K. (2020). Infrastructure as code: Dynamic systems for the cloud age (2nd ed.). O'Reilly Media.
3. Hüttermann, M. (2020). DevOps for developers. Apress.
4. Love, R. (2017). Learning Ansible 2. Packt Publishing.
5. Hochstein, L., & Moser, R. (2017). Ansible: Up and running: Automating configuration management and deployment the easy way (2nd ed.). O'Reilly Media.
6. Turnbull, J. (2014). The Puppet 4.5 type reference. James Turnbull.

SGOU

Unit 4
CI/CD and Orchestration Tools

Learning Outcomes

Upon completion of the unit, learner will be able to:

- explain the concepts of Continuous Integration, Continuous Delivery, and Continuous Deployment.
- describe the stages of a CI/CD pipeline lifecycle.
- identify the major CI/CD tools such as Jenkins, GitLab CI/CD, and AWS CodePipeline.
- apply CI/CD automation for building, testing, and deploying applications.
- analyze how Kubernetes supports the deployment and management of containerized applications.

Prerequisite

Consider a university that provides an online learning platform with services such as student registration, course management, online examinations, assignment submission, fee payment, and result publication. Developers frequently modify these services, and integrating all changes only after several weeks can lead to compatibility problems, testing difficulties, and deployment errors. CI/CD and orchestration help address these challenges by automating the processes of building, testing, releasing, deploying, and monitoring applications. The major advantages include faster software delivery, early detection of errors, reduced manual effort, consistent deployment, and reliable application management. To study this topic effectively, learners should have a basic understanding of software development, source code, version control, application building, software testing, and deployment. Basic knowledge of containers and container images will also help learners understand how Kubernetes works with CI/CD pipelines.

Studying CI/CD and Kubernetes orchestration enables learners to understand how modern software applications can be developed and delivered efficiently. It helps them learn how tools such as Jenkins, GitLab CI/CD, and AWS CodePipeline automate software delivery and how Kubernetes supports the deployment, scaling, management, and monitoring of containerized applications. The knowledge gained from this topic is useful for developing reliable and scalable applications, reducing deployment risks, improving release speed, maintaining consistent environments, and understanding

SGOU

modern DevOps practices. These skills are particularly valuable for learners who wish to work in software development, cloud computing, DevOps, or application deployment.

Keywords

Continuous Integration (CI), Continuous Delivery (CD), Continuous Deployment, CI/CD Pipeline, Pipeline Lifecycle, Jenkins, GitLab CI/CD, AWS CodePipeline, Build Automation, Automated Testing, Deployment Automation, Kubernetes, Container Orchestration

Discussion

5.4.1 Introduction

Consider a university that provides an online learning platform for thousands of students. The platform contains several services such as student registration, course management, online examinations, assignment submission, fee payment, and result publication. A team of developers continuously works on improving these services. One developer may add a new feature, while another may fix an existing problem. At the same time, another developer may update the database component.

Imagine that all these changes are collected and integrated only once every few weeks. The development team may then discover that different parts of the application are incompatible. A change made by one developer may affect another part of the system. Testing the complete application at this stage can become difficult. If the application has to be deployed manually, system administrators may also have to perform many repetitive activities. A small mistake during deployment can make the application unavailable.

Modern software development requires a faster and more reliable approach. Developers need to integrate their work frequently, test changes automatically, and deliver new versions of applications quickly. This requirement has led to the adoption of Continuous Integration (CI) and Continuous Delivery/Deployment (CD).

CI/CD practices use automation to connect different stages of software development, testing, and delivery. When a developer submits a change to a source code repository, an automated process can retrieve the code, build the application, run tests, create a deployable package, and deliver the application to a target environment.

Several tools are available for implementing CI/CD. Jenkins, GitLab CI/CD, and AWS CodePipeline are important examples. These tools can connect source code repositories with build systems, testing frameworks, container technologies, cloud platforms, and deployment environments.

The increasing use of containers has also created a need for container orchestration. Running a few containers manually may be possible, but managing hundreds or

SGOU

thousands of containers requires automation. Kubernetes is a widely used container orchestration platform that helps deploy, scale, manage, and maintain containerized applications.

Therefore, CI/CD and container orchestration work together to support modern cloud-based and cloud-native application development.

5.4.2 Continuous Integration and Continuous Delivery

Before studying CI/CD tools, it is important to understand the ideas behind CI/CD.

5.4.2.1 Continuous Integration

Continuous Integration (CI) is a software development practice in which developers frequently integrate their code changes into a shared repository. Whenever a change is submitted, automated processes can build the application and run automated tests.

The main purpose of Continuous Integration is to identify problems early.

For example, assume that three developers are working on an online examination application. One developer modifies the login module, another modifies the examination module, and the third modifies the database connection. If they work independently for several weeks, combining their changes later may produce several errors.

With Continuous Integration, developers frequently submit their changes to the shared repository. The CI system automatically builds the application and runs appropriate tests. If the build or a test fails, developers can investigate the problem immediately.

Thus, Continuous Integration encourages frequent integration and early detection of defects.

5.4.2.2 Continuous Delivery

Continuous Delivery extends Continuous Integration by automatically preparing software for release. After the application is built and tested successfully, it can be packaged and deployed to environments such as testing or staging.

The application is maintained in a state where it is ready to be released. However, Continuous Delivery does not necessarily mean that every successful change is automatically deployed to the production environment. An organization may require a person to approve the final production release.

Therefore, the key idea of Continuous Delivery is to keep the software continuously ready for release.

5.4.2.3 Continuous Deployment

Continuous Deployment goes one step further than Continuous Delivery. In Continuous Deployment, changes that successfully pass the required automated checks are automatically released to the production environment without requiring manual approval.

SGOU

Continuous Deployment can make software releases extremely fast. However, it requires reliable automated testing, monitoring, security controls, and recovery mechanisms.

5.4.3 CI/CD Pipeline Lifecycle

The CI/CD pipeline lifecycle describes the sequence of activities through which a software application moves from development to deployment and continuous monitoring. It provides an automated and systematic approach to delivering software. Instead of performing each activity manually, a CI/CD pipeline connects different stages so that the completion of one stage can automatically trigger the next stage.

A typical CI/CD lifecycle can be represented as:

Plan → Code → Build → Test → Release → Deploy → Operate → Monitor → Feedback

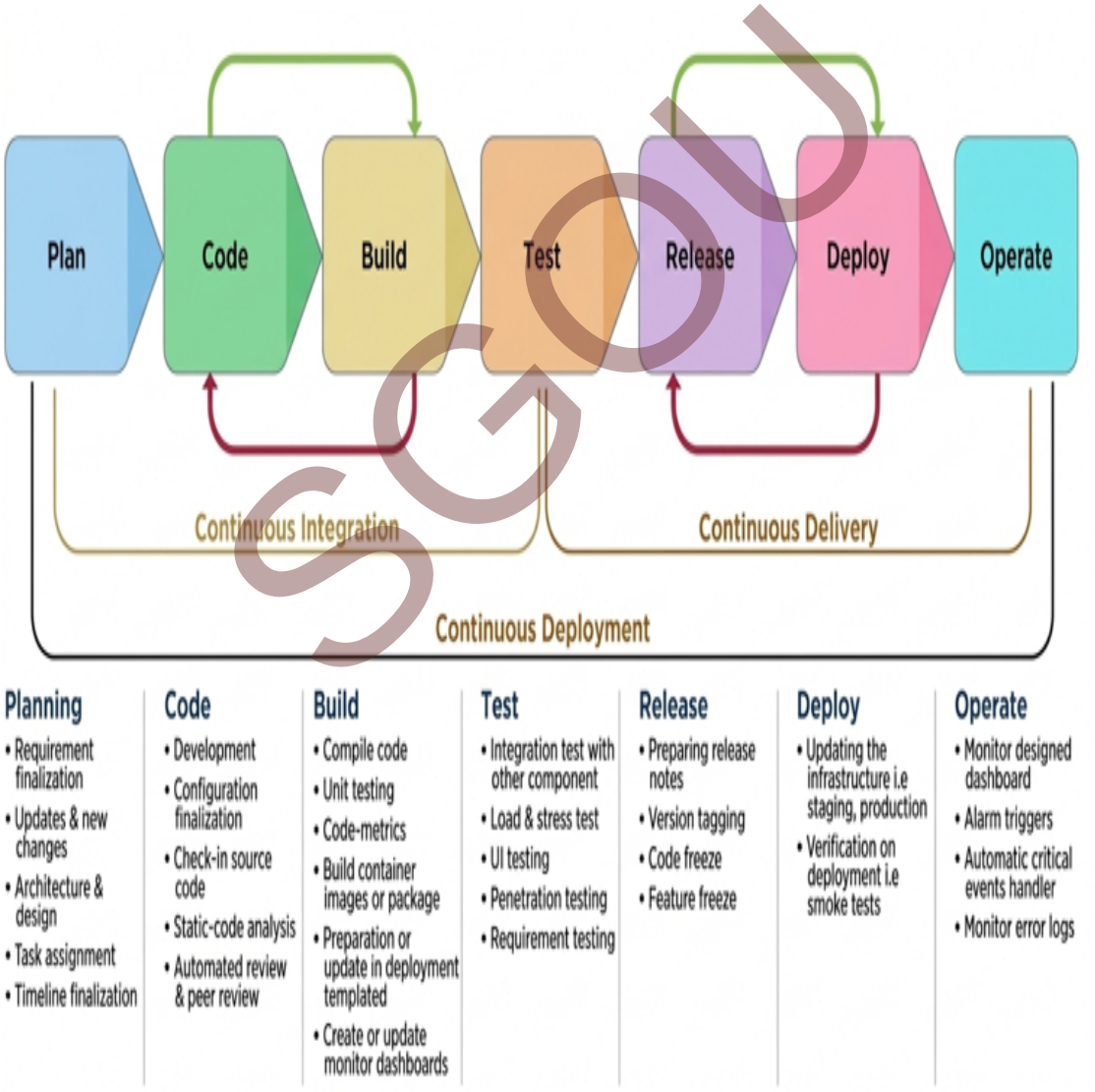


Fig 5.4.1: DevOps CI/CD Pipeline Lifecycle Diagram

SGOU

The diagram outlines the complete DevOps software development lifecycle, dividing the workflow into three primary automation boundaries: Continuous Integration, Continuous Delivery, and Continuous Deployment. Continuous Integration focuses on the initial developer feedback loop, automating the process of checking in code, compiling builds, and running basic unit tests to detect defects early. Continuous Delivery expands this scope across testing and release preparation, ensuring the software remains in a constantly deployable state at any given moment. Continuous Deployment takes automation a step further by automatically releasing fully validated code into production environments without requiring manual approval gates.

The lifecycle begins with the planning phase, where teams finalize project requirements, design system architecture, set release timelines, and assign tasks for new feature requests. Developers then move into the coding phase, writing application logic, managing source control commits, and conducting static code analysis alongside peer reviews. In the build phase, the pipeline compiles the source code into executable artifacts or container images, runs unit tests, measures code metrics, and updates deployment templates to ensure build consistency.

Following the build phase, the application enters comprehensive testing, where it undergoes integration testing with external components, load and stress evaluations, user interface checks, and security penetration scans. Once verified, the release phase stabilizes the software by tagging versions, creating release notes, and enforcing feature freezes. In the deployment phase, the automated pipeline updates infrastructure in staging or production and executes post-deployment smoke tests. Finally, the operating phase maintains application health through real-time telemetry dashboards, error log tracking, and automated alarm triggers for incident response.

Example: CI/CD Lifecycle of an Online University Application

Consider an online university that wants to introduce a new online examination feature.

The team first plans the feature and identifies its requirements. Developers then write the code and store it in a Git repository. The CI/CD system retrieves the code and builds the application. Automated tests verify whether students can log in, attend examinations, submit answers, and receive appropriate responses.

If the tests are successful, the application is released and prepared for deployment. The CI/CD pipeline then deploys the application, possibly as containers running on Kubernetes. The application is operated in the production environment while monitoring tools continuously observe its health and performance.

If monitoring identifies a problem, the information is sent back to the development team as feedback. The team can then improve the application and begin another cycle.

5.4.4 CI/CD and Orchestration Tools

Several tools are available to implement CI/CD pipelines. Among them, Jenkins, GitLab CI/CD, and AWS CodePipeline are widely used. Although all three can automate

SGOU

software delivery, they differ in their architecture, configuration methods, integrations, and deployment environments.

1. **Jenkins**

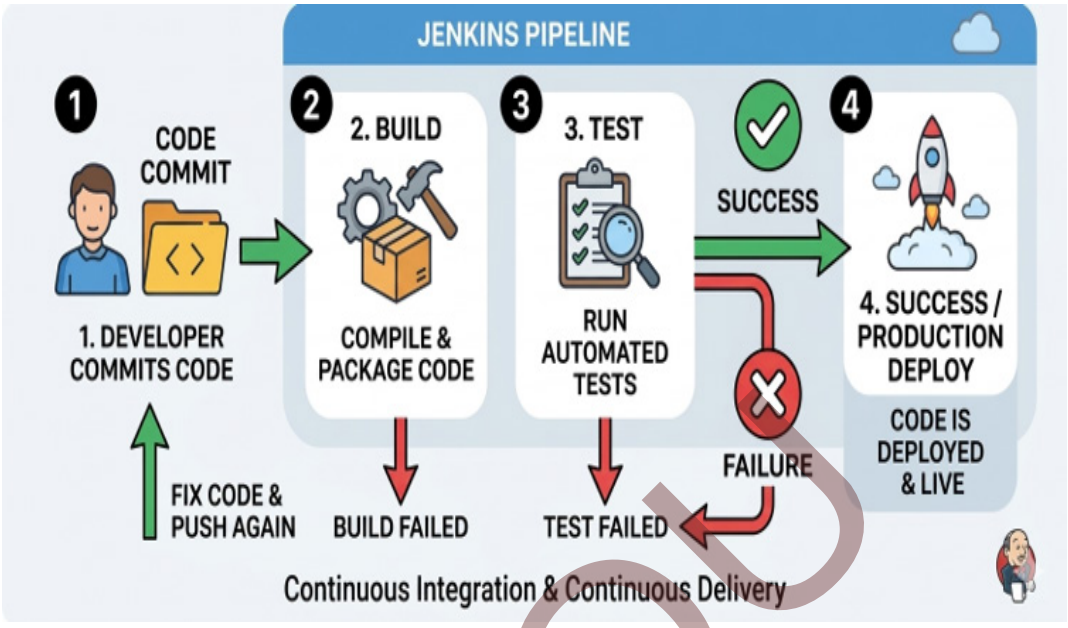


Fig 5.4.2 : JENKINS Pipeline Diagram

Jenkins is an open-source automation server widely used for implementing CI/CD pipelines. It can automatically perform activities such as building source code, executing test cases, generating packages, and deploying applications.

Jenkins works by connecting different stages of the software development process. For example, when a developer commits code to a Git repository, Jenkins can detect the change and start a pipeline. The pipeline may first compile the application, then execute automated tests. If the tests are successful, Jenkins can create a deployable package and deploy it to a development, testing, or production environment.

One important feature of Jenkins is its extensibility through plugins. Plugins allow Jenkins to work with Git repositories, Docker, Kubernetes, cloud platforms, testing frameworks, notification systems, and many other technologies. This makes Jenkins suitable for organizations that need customized CI/CD workflows.

A Jenkins pipeline can also be defined as code using a Jenkinsfile. This allows the pipeline configuration to be stored along with the application source code. As a result, the build and deployment process becomes easier to manage and reproduce.

Example

Suppose a university develops an online examination application. A developer modifies the examination module and pushes the changes to Git. Jenkins detects the change and automatically performs the following activities:

SGOU

Code Commit → Build → Automated Testing → Package → Deploy

If the automated tests fail, Jenkins can stop the pipeline and notify the development team. If all tests pass, the application can proceed to the deployment stage.

2. GitLab CI/CD

GitLab CI/CD is the continuous integration and continuous delivery capability provided by GitLab. Unlike Jenkins, which is primarily an automation server that can be extended through plugins, GitLab provides source code management and CI/CD capabilities within the same platform.

GitLab CI/CD uses a configuration file called .gitlab-ci.yml. This file describes the stages and jobs that should be executed in the pipeline. For example, a pipeline can contain stages such as build, test, and deploy.

When developers push code to a GitLab repository, GitLab CI/CD can automatically start the pipeline. The jobs are executed by GitLab Runner, which is responsible for running the commands defined in the pipeline configuration.

For example, an application may have the following pipeline:

Build → Test → Security Check → Deploy

If the test stage fails, the deployment stage can be prevented. This helps ensure that an application is not deployed before it satisfies the required quality conditions.

GitLab CI/CD is particularly useful when an organization wants source code management, issue tracking, code review, and CI/CD capabilities within a unified platform.

3. AWS CodePipeline

AWS CodePipeline is a fully managed continuous delivery service provided by Amazon Web Services. It is designed to automate the movement of application changes through different stages of the software delivery process.

A CodePipeline workflow generally consists of source, build, test, and deployment stages. The source stage obtains the application code from a supported source repository. The build stage can use services such as AWS CodeBuild to compile and package the application. Testing can then be performed before the application is deployed using services such as Amazon ECS, Amazon EKS, AWS Elastic Beanstalk, or other deployment mechanisms.

The major advantage of AWS CodePipeline is its integration with the AWS ecosystem. Organizations that already use AWS services can create automated delivery workflows without having to manage their own CI/CD server.

Example

Consider an online university that hosts its student portal on AWS. Whenever developers push a new version of the portal, CodePipeline can automatically retrieve

SGOU

the code, build the application, perform tests, and deploy the new version to the AWS environment.

The basic flow can be represented as:

Source Repository → Build → Test → Deploy → Monitor

5.4.5 Automating Build, Test and Deployment Pipeline

Automation is one of the fundamental ideas behind CI/CD. Without automation, developers and system administrators may have to perform the same tasks repeatedly.

For example, consider an application that needs to be deployed every week. If an administrator manually downloads the source code, installs dependencies, builds the application, runs tests, copies files to the server, modifies configuration files, and restarts services every week, the process can become time-consuming and error-prone.

A CI/CD pipeline can perform these activities automatically.

5.4.5.1 Build Automation

Build automation is the automatic process of converting source code into an executable or deployable form.

For example, a Java application may need to be compiled and packaged. A web application may require dependencies to be installed and files to be prepared for deployment.

The CI/CD system performs these activities according to predefined instructions. Build automation provides consistency because the same procedure can be executed whenever the application is built.

Benefits of Build Automation

- Reduces manual effort
- Provides consistent builds
- Reduces human errors
- Saves development time
- Produces deployable artifacts automatically

5.4.5.2 Automated Testing

Testing is an important stage in a CI/CD pipeline. Automated tests can be executed whenever new code changes are introduced.

Depending on the application, the pipeline may include unit tests, integration tests, functional tests, security tests, and other forms of testing.

SGOU

Suppose a developer modifies the login functionality of a university application. The pipeline can automatically test whether users can log in successfully and whether the change has affected other parts of the application.

If a test fails, the pipeline can stop before the application reaches the production environment. Therefore, automated testing acts as an important quality-control mechanism.

Benefits of Automated Testing

- Detects defects early
- Reduces manual testing effort
- Improves software quality
- Provides faster feedback
- Helps prevent faulty releases

5.4.5.3 Deployment Automation

Deployment automation involves automatically delivering an application to a target environment.

A deployment process may involve creating deployment artifacts, transferring them to the target environment, updating configuration settings, starting application services, and verifying that the application is functioning correctly.

Automation makes deployments repeatable. The same deployment procedure can be applied consistently instead of depending on different administrators performing the same task in different ways.

5.4.6 Integration with Container Orchestration Platforms such as Kubernetes

Modern CI/CD pipelines often work together with container orchestration platforms such as Kubernetes to automate the deployment, management, and scaling of containerized applications. While the CI/CD pipeline is responsible for building and testing the application, Kubernetes manages the containers after they are deployed. This integration creates a smooth path from source code changes to running applications.

How CI/CD Integrates with Kubernetes

When a developer commits new code to a Git repository, the CI/CD pipeline is triggered. The pipeline first builds the application and runs automated tests. If the tests are successful, a Docker container image is created and stored in a container registry. The pipeline then deploys the new container image to a Kubernetes cluster.

Kubernetes takes responsibility for running the application containers. It automatically schedules containers on available nodes, maintains the desired number of application instances, and can replace failed containers. This means that developers can deliver

SGOU

new versions of applications quickly while Kubernetes maintains the application's availability and stability.

A typical integration follows these steps:

5.4.6.1 Code Commit

The developer pushes application code to a Git repository such as GitLab or GitHub.

5.4.6.2 Continuous Integration

The CI tool, such as Jenkins or GitLab CI/CD, automatically starts the pipeline. The application is compiled or built, and automated tests are executed.

5.4.6.3 Container Image Creation

After successful testing, the application is packaged into a Docker container image. The image contains the application along with its required libraries and dependencies.

5.4.6.4 Container Registry

The container image is pushed to a container registry such as Docker Hub, Amazon Elastic Container Registry, or GitLab Container Registry.

5.4.6.5 Kubernetes Deployment

The CI/CD pipeline communicates with the Kubernetes cluster and updates the application deployment. Kubernetes pulls the required container image from the registry.

5.4.6.6 Application Management

Kubernetes creates and manages the application containers. It can automatically restart failed containers, distribute workloads, and maintain the desired number of replicas.

5.4.6.7 Continuous Monitoring

The deployed application is monitored for performance, availability, and errors. If problems are detected, the deployment can be rolled back to an earlier stable version.

Example

Consider an online shopping application. A developer modifies the payment module and pushes the changes to the Git repository. The CI/CD pipeline automatically builds the application and performs unit and integration tests. Once all tests pass, a new Docker image is created and stored in a container registry.

The pipeline then instructs Kubernetes to deploy the new image. Kubernetes gradually replaces the old application containers with the new ones. If the new version works correctly, the deployment continues. If a serious problem occurs, the system can roll back to the previous version.

SGOU

Thus, CI/CD provides automation for delivering software, while Kubernetes provides automation for running and managing the containerized software.

The integration of CI/CD with Kubernetes provides several important benefits:

- Automated deployment: Applications can be deployed automatically after successful testing.
- Scalability: Kubernetes can increase or decrease the number of application instances according to workload.
- High availability: Failed containers can be automatically restarted.
- Faster releases: New application versions can be delivered more frequently.
- Consistent environments: Container images provide a consistent application environment.
- Easy rollback: Kubernetes can help restore a previous application version when a deployment fails.
- Reduced manual effort: Deployment and container management require less human intervention.

Recap

- Continuous Integration (CI) enables developers to frequently integrate code changes into a shared repository and automatically test them.
- Continuous Delivery prepares successfully tested applications for release and deployment to testing or staging environments.
- Continuous Deployment automatically releases successfully tested changes to the production environment.
- A CI/CD pipeline automates important stages such as code integration, building, testing, releasing, and deployment.
- The CI/CD pipeline lifecycle includes Plan, Code, Build, Test, Release, Deploy, Operate, Monitor, and Feedback.
- Jenkins is an open-source automation server used to create and manage CI/CD pipelines.
- GitLab CI/CD provides integrated tools for automating software development, testing, security checks, and deployment.
- AWS CodePipeline is a managed AWS service that automates the continuous delivery process.
- Kubernetes is a container orchestration platform that helps deploy, scale, manage, and monitor containerized applications.

SGOU

Objective Type Questions

1. Define the practice of frequently integrating code changes into a shared repository.
2. Identify the process that prepares software for release after successful testing.
3. Name the approach that automatically releases successful changes to production.
4. Mention the automated sequence of stages used to deliver software.
5. Specify the open-source automation server used for CI/CD.
6. Recognize the platform that provides integrated CI/CD capabilities.
7. State the AWS service used to automate continuous delivery.
8. Indicate the process that automatically builds and packages applications.
9. Describe the process used to automatically check applications for errors.
10. List the automation process used to deliver applications to target environments.
11. Identify the platform used for container orchestration.
12. Mention the technology used to package applications into portable units.
13. Name the Kubernetes feature that gradually replaces old application instances with new ones.

SGOU

Answers

1. Continuous Integration
2. Continuous Delivery
3. Continuous Deployment
4. CI/CD Pipeline
5. Jenkins
6. GitLab CI/CD

7. AWS CodePipeline
8. Build Automation
9. Automated Testing
10. Deployment Automation
11. Kubernetes
12. Container
13. Rolling Update

Assignments

1. Describe the different stages of the CI/CD pipeline lifecycle.
2. Discuss Jenkins, GitLab CI/CD, and AWS CodePipeline as CI/CD tools. Explain their major features and their role in automating software delivery.
3. Explain the process of automating build, testing, and deployment pipelines
4. Explain how Kubernetes can be integrated with a CI/CD pipeline for deploying and managing containerized applications.
5. Analyze the benefits of integrating CI/CD pipelines with Kubernetes.

SGOU

Reference

1. Jenkins. (n.d.). *Jenkins documentation. https://www.jenkins.io/doc/*
2. GitLab. (n.d.). *GitLab CI/CD documentation. https://docs.gitlab.com/ci/*
3. Amazon Web Services. (n.d.). *AWS CodePipeline documentation. https://docs.aws.amazon.com/codepipeline/*
4. Kubernetes. (n.d.). *Kubernetes documentation.* https://kubernetes.io/docs/
5. Davis, J. D., & Daniels, K. (2016). *Effective DevOps: How to build a culture of collaboration, affinity, and tooling at scale.* O'Reilly Media.

Suggested Reading

1. Burns, B., Beda, J., & Hightower, K. (2022). *Kubernetes: Up and running (3rd ed.).* O'Reilly Media.

2. Kim, G., Humble, J., Debois, P., & Willis, J. (2021). *The DevOps handbook: How to create world-class agility, reliability, and security in technology organizations* (2nd ed.). IT Revolution Press.

3. Morris, M. (2020). *Infrastructure as code: Dynamic systems for the cloud age.* O'Reilly Media.

4. Humble, J., & Farley, D. (2010). *Continuous delivery: Reliable software releases through build, test, and deployment automation*. Addison-Wesley.

SGOU

Model Question Paper Sets

SGOU

SREENARAYANAGURU OPEN UNIVERSITY

MODEL QUESTION PAPER (SET 1)

QP CODE: ……

Reg. No:.....

Name: ……

BSc DATA SCIENCE AND ANALYTICS, FIFTH SEMESTER
END SEMESTER EXAMINATION

DISCIPLINE SPECIFIC ELECTIVE COURSE

B24DS02DE : Introduction to Cloud Computing

Time: 3 Hours | Max Marks: 70

Section A

Answer any 10 questions. Each carries one mark

(10x1=10)

1. Define cloud computing
2. Which cloud deployment model provides resources to multiple users through a third-party provider?
3. What is the expansion of IaaS ?
4. Who manages the underlying physical infrastructure in IaaS?
5. What does VPC stand for in AWS?
6. Name one popular FaaS provider.
7. Which verification method uses an OTP?
8. Which entity authenticates users in an SSO environment?
9. Which resource management model adjusts resources according to workload changes?
10. Which cloud migration service is provided by Microsoft Azure?
11. Write any four components of an organization that are examined before cloud migration.
12. Write any two advantages of Google Cloud Migration Tools.
13. Which type of automation is concerned with automatically installing and maintaining software settings?
14. What approach manages infrastructure using structured definitions?
15. Point out the Puppet files commonly used to define configuration requirements.

SGOU

Section B

Answer any 5 questions. Each carries two marks

(5x2=10)

16. What is grid computing?
17. What is Hybrid Cloud?
18. Write any 4 advantages of BPaaS.
19. What are the common use cases of Serverless Computing?
20. What is Auditability in Cloud Environments?
21. What is tagging?
22. What is the purpose of measuring resource utilization before cloud migration?
23. What is vendor lock-in in cloud computing? Mention one challenge associated with it.
24. Differentiate between infrastructure automation and configuration automation.
25. What is configuration drift? Mention any two methods for managing configuration drift.

Section C

Answer any 5 questions. Each carries four marks

(5x4=20)

26. Explain the five essential characteristics of cloud computing
27. Explain the role of network infrastructure and virtualization in cloud computing.
28. Discuss the role of Security as a Service (SecaaS) and Identity as a Service (IdMaaS) in enhancing cloud security.
29. Explain the AWS Shared Responsibility Model and distinguish between AWS responsibilities and customer responsibilities with suitable examples.
30. Explain Resource Provisioning and De-provisioning.
31. Explain the AAA model.
32. Explain the factors to be considered when selecting an appropriate cloud migration approach for different workloads. Provide suitable examples.
33. Explain any four advantages of using AWS Migration Hub for managing and monitoring cloud migration activities.

SGOU

34. Explain the process of creating and managing infrastructure templates using Terraform.

35. Compare Jenkins, GitLab CI/CD, and AWS CodePipeline based on their features, working principles, and integration capabilities.

Section D

Answer any 2 questions. Each carries fifteen mark

(2x15=30)

36. Explain the four major cloud deployment models with their features, advantages and disadvantages.

37. Write a detailed account of Google Cloud Platform (GCP), explaining its key features, major services, architecture, advantages, disadvantages, and real-world applications.

38. Explain Resource Management and different types of resources with examples.

39. Explain the different types of cloud migration strategies in detail, highlighting their key characteristics and processes.

SGOU

SREENARAYANAGURU OPEN UNIVERSITY

MODEL QUESTION PAPER (SET 2)

QP CODE: ……

Reg. No:.....

Name: ……

BSc DATA SCIENCE AND ANALYTICS, FIFTH SEMESTER
END SEMESTER EXAMINATION

DISCIPLINE SPECIFIC ELECTIVE COURSE

B24DS02DE : Introduction to Cloud Computing

Time: 3 Hours | Max Marks: 70

Section A

Answer any 10 questions. Each carries one mark

(10x1=10)

1. What is meant by on-demand self-service?
2. Which cloud deployment model is dedicated to a single organization?
3. What is the expansion of SaaS ?
4. Which workload manager is used by AWS Parallel Computing Service (AWS PCS)?
5. What does SaaS stand for?
6. Which Azure tool is used to estimate the expected cost of Azure services before deployment?
7. Which component manages cloud expenditure?
8. What does IAM stand for?
9. What metric measures the amount of work completed within a given period?
10. What is cloud migration?
11. Write any two advantages of Azure Migrate.
12. Define retiring and retaining approaches to cloud migration.
13. Which AWS service is used to define and provision cloud infrastructure using templates?
14. Which format is commonly used for CloudFormation templates?
15. Recognize the system used to track and manage different versions of configuration files.

SGOU

Section B

Answer any 5 questions. Each carries two marks

(5x2=10)

16. What is utility computing?
17. What is a Community Cloud?
18. Write any 4 key features of SaaS.
19. Define IdMaaS.
20. What is Compliance in Cloud Environments?
21. What is Auto Scaling?
22. What is performance risk in cloud migration? Explain with an example.
23. What is cost-benefit analysis in cloud migration?
24. List any four benefits of cloud automation.
25. What is build automation? Mention any two benefits of build automation.

Section C

Answer any 5 questions. Each carries four marks

(5x4=20)

26. Explain how virtualization contributed to the development of cloud computing.
27. Explain the major components of cloud computing.
28. Explain the components and working of IaaS.
29. Describe Serverless Computing, its working principle, and the role of event-driven execution, automatic scaling, and cloud-managed infrastructure.
30. Explain the working of Single Sign-On (SSO).
31. Explain the different Steps in Creating a Cloud Account.
32. Explain the process of assessing an organisation's existing infrastructure and workloads before cloud migration.
33. Explain the three core phases of the cloud migration lifecycle: Plan, Execute and Monitor, with the help of a neat diagram.
34. Describe the role of Infrastructure as Code (IaC) and CI/CD in automating cloud operations.
35. Compare Ansible, Chef, and Puppet based on their features, architecture, working principles, and advantages.

SGOU

Section D

Answer any 2 questions. Each carries fifteen mark

(2x15=30)

36. Explain the different stages in the evolution of cloud computing, starting from mainframe computing to modern cloud computing.

37. Write a brief note on the Microsoft Azure Cloud Computing Platform, explaining its key features, major services, advantages, disadvantages, and real-world applications.

38. Explain the Load Balancing in Cloud Applications. Also explain different techniques in load balancing.

39. Discuss the major infrastructure automation tools and also explain how infrastructure templates are created and managed using these tools.

SGOU

സർവ്വകലാശാലാഗീതം

വിദ്യയാൽ സ്വതന്ത്രരാകണം
വിശ്വപൗരരായി മാറണം
ഗ്രഹപ്രസാദമായ് വിളങ്ങണം
ഗുരുപ്രകാശമേ നയിക്കണേ

കൂരിരുട്ടിൽ നിന്നു ഞങ്ങളെ
സൂര്യവീഥിയിൽ തെളിക്കണം
സ്നേഹദീപ്തിയായ്വിളങ്ങണം
നീതിവൈജയന്തി പാറണം

ശാസ്ത്രവ്യാപ്തിയെന്നുമേകണം
ജാതിഭേദമാകെ മാറണം
ബോധരശ്മിയിൽ തിളങ്ങുവാൻ
ജ്ഞാനകേന്ദ്രമേ ജ്വലിക്കണേ

കുരീപ്പുഴ ശ്രീകുമാർ

SGOU

SREENARAYANAGURU OPEN UNIVERSITY

Regional Centres

Kollam
Sreenarayanaguru Open University
Headquarters Building
BSNL Bhavan, Vellayittambalam,
Kollam, Kerala, Pin : 691012
Ph: 9188912554
email:rckollam@sgou.ac.in

Tripunithura
Govt. College
Tirpunithura, Ernakulam
Kerala, Pin : 682301
Ph: 04842927436
email:rcedirector@sgou.ac.in

Thalassery
Govt. Brennen College
Dharmadam, Thalassery
Kannur, Pin : 670106
Ph: 04952990494
email:rctdirector@sgou.ac.in

Pattambi
Sree Neelakanta Govt. Sanskrit College
Pattambi, Palakkad
Kerala, Pin : 679303
Ph: 04662912009
email:rcpdirector@sgou.ac.in

Kozhikode
Govt. Arts and Science College
Meenchantha, Kozhikode,
Kerala, Pin : 673002
Ph: 04952920228
email:rckdirector@sgou.ac.in

SAY NO TO RAGGING

BEFORE YOU EVEN THINK OF RAGGING



Humiliation

Blacklisting

Ruined Career

Download

ANTI RAGGING

App

Don't just stand and watch. Stop Ragging! Show Character

Visit UGC Website i.e. www.ugc.ac.in & www.antiraging.in to UGC Anti Ragging regulations.

Ministry of Education
Government of India

Are You Being Ragged ?

विश्वविद्यालय अनुदान आयोग
University Grants Commission

SSGOU

DON'T LET IT BE TOO LATE

SAY NO TO DRUGS

LOVE YOURSELF AND ALWAYS BE HEALTHY



SREENARAYANAGURU OPEN UNIVERSITY

The State University for Education, Training and Research in Blended Format, Kerala

Introduction to Cloud Computing

COURSE CODE: B24DS02DE

SGOU

Sreenarayanaguru Open University
Kollam, Kerala Pin- 691601, email: info@sgou.ac.in, www.sgou.ac.in Ph: +91 474 2966841

ISBN 978-81-180900-4-5