REGULATION 2021 ACADEMIC YEAR 2023-2024

Loyola ICAM College of Engineering and Technology(LICET)

QUESTION BANK

III YEAR – 05TH SEMESTER

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE

CCS335 CLOUD COMPUTING

TABLE OF CONTENT

CCS335 CLOUD COMPUTING L T P C 2 0 2 3

COURSE OBJECTIVES:

To understand the principles of cloud architecture, models and infrastructure.
To understand the concepts of virtualization and virtual machines.
To gain knowledge about virtualization Infrastructure.
To explore and experiment with various Cloud deployment environments.
To learn about the security issues in the cloud environment.

UNIT I CLOUD ARCHITECTURE MODELS AND INFRASTRUCTURE 6

Cloud Architecture: System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture – Cloud deployment models – Cloud service models; Cloud Infrastructure: Architectural Design of Compute and Storage Clouds – Design Challenges

UNIT II VIRTUALIZATION BASICS 6

Virtual Machine Basics – Taxonomy of Virtual Machines – Hypervisor – Key Concepts – Virtualization structure – Implementation levels of virtualization – Virtualization Types: Full Virtualization – Para Virtualization – Hardware Virtualization – Virtualization of CPU, Memory and I/O devices.

UNIT III VIRTUALIZATION INFRASTRUCTURE AND DOCKER 7

Desktop Virtualization – Network Virtualization – Storage Virtualization – System-level of Operating Virtualization – Application Virtualization – Virtual clusters and Resource Management – Containers vs. Virtual Machines – Introduction to Docker – Docker Components – Docker Container – Docker Images and Repositories.

UNIT IV CLOUD DEPLOYMENT ENVIRONMENT 6

Google App Engine – Amazon AWS – Microsoft Azure; Cloud Software Environments – Eucalyptus – OpenStack.

UNIT V CLOUD SECURITY 5

Virtualization System-Specific Attacks: Guest hopping – VM migration attack – hyperjacking. Data Security and Storage; Identity and Access Management (IAM) - IAM Challenges - IAM Architecture and Practice. 30PERIODS

PRACTICAL EXERCISES: 30 PERIODS

1. Install Virtualbox/VMware/ Equivalent open source cloud Workstation with different flavours of Linux or Windows OS on top of windows 8 and above.

2. Install a C compiler in the virtual machine created using a virtual box and execute Simple Programs

3. Install Google App Engine. Create a hello world app and other simple web applications using python/java.

4. Use the GAE launcher to launch the web applications.

5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

6. Find a procedure to transfer the files from one virtual machine to another virtual machine.

7. Install Hadoop single node cluster and run simple applications like wordcount.

8. Creating and Executing Your First Container Using Docker.

9. Run a Container from Docker Hub

COURSE OUTCOMES:

CO1: Understand the design challenges in the cloud.

CO2: Apply the concept of virtualization and its types.

CO3: Experiment with virtualization of hardware resources and Docker.

CO4: Develop and deploy services on the cloud and set up a cloud environment.

CO5: Explain security challenges in the cloud environment. TOTAL:60 PERIODS

TEXT BOOKS

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.

2. James Turnbull, "The Docker Book", O'Reilly Publishers, 2014.

3. Krutz, R. L., Vines, R. D, "Cloud security. A Comprehensive Guide to Secure Cloud Computing", Wiley Publishing, 2010.

REFERENCES

1. James E. Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.

2. Tim Mather, Subra Kumaraswamy, and Shahed Latif, "Cloud Security and Privacy: an enterprise perspective on risks and compliance", O'Reilly Media, In

CCS335 CLOUD COMPUTING

Question Bank

UNIT 1

CLOUD ARCHITECTURE MODELS AND INFRASTRUCTURE

SYLLABUS:Cloud Architecture: System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture – Cloud deployment models – Cloud service models; Cloud Infrastructure: Architectural Design of Compute and Storage Clouds – Design Challenges

PART A

2 Marks

1. What is Cloud Computing? BTL 1

Cloud Computing is defined as storing and accessing of data and computing services over the internet. It doesn't store any data on your personal computer. It is the on-demand availability of computer services like servers, data storage, networking, databases, etc. The main purpose of cloud computing is to give access to data centers to many users. Users can also access data from a remote server.

Examples of Cloud Computing Services: AWS, Azure,

2. Write down characteristic of cloud computing? BTL 1

The National Institute of Standards Technology (NIST) lists five essential characteristics of cloud computing:on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service.

3. What are all Cloud Computing Services BTL1

The three major Cloud Computing Offerings are

- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

4.Descirbe the type of cloud computing?BTL1

There Are Four Main Types Of Cloud Computing:

- Private Clouds,
- Public Clouds,
- Hybrid Clouds,
- Multiclouds.

5.Write down advandages of cloud computing? Advantages (or) Pros of Cloud Computing?BTL1

1. Improved Performance
2. Lower IT Infrastructure Costs
3. Fewer Maintenance Issues
4. Lower Software Costs
5. Instant Software Updates
6. Increased Computing Power

6. Write down disadvantage of cloud computing?BTL1

1. Requires a constant Internet connection
2. Does not work well with low-speed connections
3. Can be slow
4. Stored data might not be secure
5. Stored data can be lost

7. What are the computing Paradigm Distinctions?BTL1

- Centralized computing
- Parallel Computing
- Distributed Computing
- Cloud Computing

8.What are the differences between Grid computing and cloud computing?BTL 2

	Grid computing	Cloud computing
What?	Grids enable access to shared computing power and storage capacity from your desktop	Clouds enable access to leased computing power and storage capacity from your desktop
Who provides the service?	Research institutes and universities federate their services around the world.	Large individual companies e.g. Amazon and Microsoft.
Who uses the service?	Research collaborations, called "Virtual Organizations", which bring together researchers around the world working in the same field.	Small to medium commercial businesses or researchers with generic IT needs
Who pays for the service?	Governments - providers and users are usually publicly funded research organizations.	The cloud provider pays for the computing resources; the user pays to use them

9.Difference between Cloud Computing and Distributed Computing :BTL2

S.No.	CLOUD COMPUTING	DISTRIBUTED COMPUTING

01.	Cloud computing refers to providing on demand IT resources/services like server, storage, database, networking, analytics, software etc. over internet.	Distributed computing refers to solve a problem over distributed autonomous computers and they communicate between them over a network.
02.	In simple cloud computing can be said as a computing technique that delivers hosted services over the internet to its users/customers.	In simple distributed computing can be said as a computing technique which allows to multiple computers to communicate and work to solve a single problem.
03.	It is classified into 4 different types such as Public Cloud, Private Cloud, Community Cloud and Hybrid Cloud.	It is classified into 3 different types such as Distributed Computing Systems, Distributed Information Systems and Distributed PervasiveSystems.
04.	There are many benefits of cloud computing like cost effective, elasticity and reliable, economies of Scale, access to the global market etc.	There are many benefits of distributed computing like flexibility, reliability, improved performance etc.
05.	Cloud computing provides services such as hardware, software, networking resources through internet.	Distributed computing helps to achieve computational tasks more faster than using a single computer asit takes a lot of time.
06.	The goal of cloud computing is to provide on demand computing services over internet on pay peruse model.	The goal of distributed computing is to distribute a single task among multiple computers and to solve it quickly by maintaining coordinationbetween them.
07.	Some characteristics of cloud computing are providing shared pool of configurable computing resources, on-demand service, pay per use, provisioned by the Service Providers etc.	Some characteristics of distributed computing are distributing a single task among computers to progress the work at same time, Remote Procedure calls and Remote Method Invocation for distributed computations.

08.	Some disadvantage of cloud computing includes less control especially in the case of public clouds, restrictions on available services may be faced and cloud security.	Some disadvantage of distributed computing includes chances of failure of nodes, slow network may create problem in communicati

10. lists the actors defined in the NIST cloud computing reference architecture? BTL1

The NIST cloud computing reference architecture defines five major actors:
cloud consumer, cloud provider, cloud carrier, cloud auditor and cloud broker. Each actor is an entity (a person or an organization) that participates in a transaction or process and/or performs tasks in cloud computing.

11. Discuss general activity of actors in NIST architecture?BTL2

Actor	Definition
Cloud Consumer	A person or organization that maintains a business relationship with, and uses service from, *Cloud Providers*.
Cloud Provider	A person, organization, or entity responsible for making a service available to interested parties.
Cloud Auditor	A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation.
Cloud Broker	An entity that manages the use, performance and delivery of cloud services, and negotiates relationships between *Cloud Providers* and *Cloud Consumers*.
Cloud Carrier	An intermediary that provides connectivity and transport of cloud services from *Cloud Providers* to *Cloud Consumers*.

12. What is a Cloud Deployment Model?BTL1

Cloud Deployment Model functions as a virtual computing environment with a deployment architecture that varies depending on the amount of data you want to store and who has access to the infrastructure.

13. What is the Right Choice for Cloud Deployment Model?BTL1

- Cost: Cost is an important factor for the cloud deployment model as it tells how much amount you want to pay for these things.

- Scalability: Scalability tells about the current activity status and how much we can scale it.
- Easy to use: It tells how much your resources are trained and how easily can you manage these models.
- Compliance: Compliance tells about the laws and regulations which impact the implementation of the model.
- Privacy: Privacy tells about what data you gather for the model.

14. What are different Models of Cloud Computing?BTL1

Cloud Computing helps in rendering several services according to roles, companies, etc. Cloud computing models are explained below.

- Infrastructure as a service (IaaS)
- Platform as a service (PaaS)
- Software as a service (SaaS)

15. Define Infrastructure as a service (IaaS)?BTL1

Infrastructure as a Service (IaaS) helps in delivering computer infrastructure on an external basis for supporting operations. Generally, IaaS provides services to networking equipment, devices, databases, and web servers.

Infrastructure as a Service (IaaS) helps large organizations, and large enterprises in managing and building their IT platforms. This infrastructure is flexible according to the needs of the client.

16. Define Platform as a service (PaaS)?BTL1

Platform as a Service (PaaS) is a type of cloud computing that helps developers to build applications and services over the Internet by providing them with a platform.PaaS helps in maintaining control over their business applications.

17. Define Software as a service (SaaS)?BTL1

Software as a Service (SaaS) is a type of cloud computing model that is the work of delivering services and applications over the Internet. The SaaS applications are called Web-Based Software or Hosted Software.

18. List the disadvantages of the public cloud model?BTL1

The disadvantages of the public cloud model are:

- Data Security and Privacy Concerns: Because it is open to the public, it does not provide complete protection against cyber-attacks and may expose weaknesses.
- Issues with Reliability: Because the same server network is accessible to a wide range of users, it is susceptible to failure and outages.
- Limitation on Service/License: While there are numerous resources that you may share with renters, there is a limit on how much you can use.

19. List the disadvantages of the hybrid cloud model?BTL1

The disadvantages of the hybrid cloud model are:

- Maintenance: A hybrid cloud computing strategy may necessitate additional maintenance, resulting in a greater operational expense for your company.
- Difficult Integration: When constructing a hybrid cloud, data, and application integration might be difficult. It's also true that combining two or more infrastructures will offset a significant upfront cost.

20. List the disadvantages of the private cloud model?BTL1

The disadvantages of the private cloud model are

- Restricted Scalability: Private clouds have restricted scalability because they are scaled within the confines of internally hosted resources. The choice of underlying hardware has an impact on scalability.
- Higher Cost: Due to the benefits you would receive, your investment will be higher than the public cloud(pay for software, hardware, staffing, etc).

21. What are the Cloud infrastructure components ?BTL1

Different components of cloud infrastructure supports the computing requirements of a cloud computing model. Cloud infrastructure has number of key components but not limited to only server, software, network and storage devices. Still cloud infrastructure is categorized into three parts in general i.e.

1. Computing
2. Networking
3. Storage

PART B

13 Marks

1.Explain in details about architecture of cloud computing ?BTL4

(Definition:2 marks,Diagram:4 marks,Explanation:7 marks)

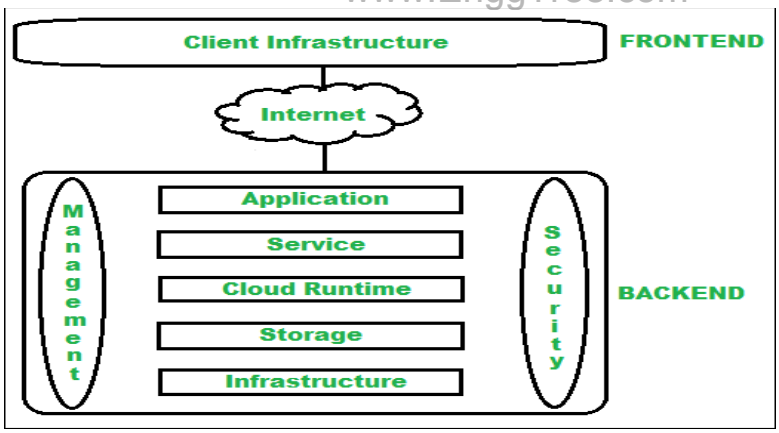
Cloud Computing , which is one of the demanding technology of the current time and which is giving a new shape to every organization by providing on demand virtualized services/resources. Starting from small to medium and medium to large, every organization use cloud computing services for storing information and accessing it from anywhere and any time only with the help of internet. In this article, we will know more about the internal architectureof cloud computing. Transparency, scalability, security and intelligent monitoring are some of the most important constraints which every cloud infrastructure should experience. Current research on other important constraints is helping cloud computing system to come up with new features and strategies with a great capability of providing more advanced cloud solutions.

Cloud Computing Architecture :

The cloud architecture is divided into 2 parts i.e.

1. Frontend
2. Backend

The below figure represents an internal architectural view of cloud computing.



Architecture of Cloud Computing

Architecture of cloud computing is the combination of both SOA (Service Oriented Architecture) and EDA (Event Driven Architecture). Client infrastructure, application, service, runtime cloud, storage, infrastructure, management and security all these are the components of cloud computing architecture.

1. Frontend :

Frontend of the cloud architecture refers to the client side of cloud computing system. Means it contains all the user interfaces and applications which are used by the client to access the cloud computing services/resources. For example, use of a web browser to access the cloud platform.

- **Client Infrastructure –** Client Infrastructure is a part of the frontend component. It contains the applications and user interfaces which are required to access the cloud platform.
- In other words, it provides a GUI(Graphical User Interface) to interact with the cloud.

2. Backend :

Backend refers to the cloud itself which is used by the service provider. It contains the resources as well as manages the resources and provides security mechanisms. Along with this, it includes huge storage, virtual applications, virtual machines, traffic control mechanisms, deployment models, etc.

1. **Application –**
   Application in backend refers to a software or platform to which client accesses. Means it provides the service in backend as per the client requirement.
2. **Service –**
   Service in backend refers to the major three types of cloud based services like SaaS, PaaS and IaaS. Also manages which type of service the user accesses.
3. **RuntimeCloud-**
   Runtime cloud in backend provides the execution and Runtime platform/environment to the Virtual machine.
4. **Storage –**

   Storage in backend provides flexible and scalable storage service and management of stored data.
5. **Infrastructure –**
   Cloud Infrastructure in backend refers to the hardware and software components of cloud like it includes servers, storage, network devices, virtualization software etc.
6. **Management –**
   Management in backend refers to management of backend components like application, service, runtime cloud, storage, infrastructure, and other security mechanisms etc.
7. **Security –**
   Security in backend refers to implementation of different security mechanisms in the backend for secure cloud resources, systems, files, and infrastructure to end-users.
8. **Internet –**
   Internet connection acts as the medium or a bridge between frontend and backend and establishes the interaction and communication between frontend and backend.
9. **Database**– Database in backend refers to provide database for storing structured data, such as SQL and NOSQL databases. Example of Databases services include Amazon RDS, Microsoft Azure SQL database and Google CLoud SQL.
10. **Networking**– Networking in backend services that provide networking infrastructure for application in the cloud, such as load balancing, DNS and virtual private networks.

11. **Analytics**– Analytics in backend service that provides analytics capabillities for data in the cloud, such as warehousing, bussness intellegence and machine learning.

Benefits of Cloud Computing Architecture :

- Makes overall cloud computing system simpler.
- Improves data processing requirements.
- Helps in providing high security.
- Makes it more modularized.
- Results in better disaster recovery.
- Gives good user accessibility.
- Reduces IT operating costs.
- Provides high level reliability.
- Scalability.

2. Discuss about system models for distributed and cloud computing?BTL2

(Definition:2 marks,Diagram:3 marks,Tabular column:3 marks,Explanation:5 marks)

Distributed and cloud computing systems are built over a large number of autonomous computer nodes. These node machines are interconnected by SANs, LANs, or WANs in a hierarchical manner. With today's networking technology, a few LAN switches can easily connect hundreds of machines as a working cluster. A WAN can connect many local clusters to form a very large cluster of clusters. In this sense, one can build a massive system with millions of computers connected to edge networks.

Massive systems are considered highly scalable, and can reach web-scale connectivity, either physically or logically. In Table 1.2, massive systems are classified into four groups: clusters, P2P networks, computing grids, and Internet clouds over huge data centers. In terms of node number, these four system classes may involve hundreds, thousands, or even millions of computers as participating nodes. These machines work collectively, cooperatively, or collaboratively at various levels. The table entries characterize these four system classes in various technical and application aspects.

Table 1.2 Classification of Parallel and Distributed Computing Systems

Functionality, Applications	Computer Clusters [10,28,38]	Peer-to-Peer Networks [34,46]	Data/ Computational Grids [6,18,51]	Cloud Platforms [1,9,11,12,30]
Architecture, Network Connectivity, and Size	Network of compute nodes interconnected by SAN, LAN, or WAN hierarchically	Flexible network of client machines logically connected by an overlay network	Heterogeneous clusters interconnected by high-speed network links over selected resource sites	Virtualized cluster of servers over data centers via SLA
Control and Resources Management	Homogeneous nodes with distributed control, running UNIX or Linux	Autonomous client nodes, free in and out, with self-organization	Centralized control, server-oriented with authenticated security	Dynamic resource provisioning of servers, storage, and networks
Applications and Network-centric Services	High-performance computing, search engines, and web services, etc.	Most appealing to business file sharing, content delivery, and social networking	Distributed supercomputing, global problem solving, and data center services	Upgraded web search, utility computing, and outsourced computing services
Representative Operational Systems	Google search engine, SunBlade, IBM Road Runner, Cray XT4, etc.	Gnutella, eMule, BitTorrent, Napster, KaZaA, Skype, JXTA	TeraGrid, GriPhyN, UK EGEE, D-Grid, ChinaGrid, etc.	Google App Engine, IBM Bluecloud, AWS, and Microsoft Azure

1. Clusters of Cooperative Computers

A computing cluster consists of interconnected stand-alone computers which work cooperatively as a single integrated computing resource. In the past, clustered computer systems have demonstrated impressive results in handling heavy workloads with large data sets.

1.1 Cluster Architecture

Figure 1.15 shows the architecture of a typical server cluster built around a low-latency, high-bandwidth interconnection network. This network can be as simple as a SAN (e.g., Myrinet)

or a LAN (e.g., Ethernet). To build a larger cluster with more nodes, the interconnection network can be built with multiple levels of Gigabit Ethernet, Myrinet, or InfiniBand switches. Through hierarchical construction using a SAN, LAN, or WAN, one can build scalable clusters with an increasing number of nodes. The cluster is connected to the Internet via a virtual private network (VPN) gateway. The gateway IP address locates the cluster. The system image of a computer is decided by the way the OS manages the shared cluster resources. Most clusters have loosely coupled node computers. All resources of a server node are managed by their own OS. Thus, most clusters have multiple system images as a result of having many autonomous nodes under different OS control.

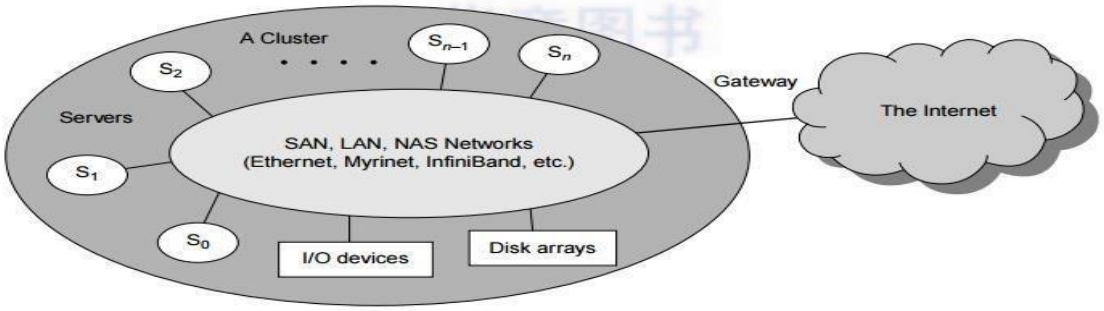


FIGURE 1.15
A cluster of servers interconnected by a high-bandwidth SAN or LAN with shared I/O devices and disk arrays; the cluster acts as a single computer attached to the Internet.

1.2 Single-System Image

Greg Pfister [38] has indicated that an ideal cluster should merge multiple system images into a single-system image (SSI). Cluster designers desire a cluster operating system or some middleware to support SSI at various levels, including the sharing of CPUs, memory, and I/O across all cluster nodes. An SSI is an illusion created by software or hardware that presents a collection of resources as one integrated, powerful resource. SSI makes the cluster appear like a single machine to the user. A cluster with multiple system images is nothing but a collection of independent computers.

1.3 Hardware, Software, and Middleware Support

In Chapter 2, we will discuss cluster design principles for both small and large clusters. Clusters exploring massive parallelism are commonly known as MPPs. Almost all HPC clusters in the Top 500 list are also MPPs. The building blocks are computer nodes (PCs, workstations, servers, or SMP), special communication software such as PVM or MPI, and a network interface card in each computer node. Most clusters run under the Linux OS. The computer nodes are interconnected by a high-bandwidth network (such as Gigabit Ethernet, Myrinet, InfiniBand, etc.).

Special cluster middleware supports are needed to create SSI or high availability (HA). Both sequential and parallel applications can run on the cluster, and special parallel environments are

needed to facilitate use of the cluster resources. For example, distributed memory has multiple images. Users may want all distributed memory to be shared by all servers by forming distributed shared memory (DSM). Many SSI features are expensive or difficult to achieve at various cluster operational levels. Instead of achieving SSI, many clusters are loosely coupled machines. Using virtualization, one can build many virtual clusters dynamically, upon user demand. We will discuss virtual clusters in Chapter 3 and the use of virtual clusters for cloud computing in Chapters 4, 5, 6, and 9.

1.4 Major Cluster Design Issues

Unfortunately, a cluster-wide OS for complete resource sharing is not available yet. Middleware or OS extensions were developed at the user space to achieve SSI at selected functional levels. Without this middleware, cluster nodes cannot work together effectively to achieve cooperative computing. The software environments and applications must rely on the middleware to achieve high performance. The cluster benefits come from scalable performance, efficient message passing, high system availability, seamless fault tolerance, and cluster-wide job management, as summarized in Table 1.3. We will address these issues in Chapter 2.

2. Grid Computing Infrastructures

In the past 30 years, users have experienced a natural growth path from Internet to web and grid computing services. Internet services such as the Telnet command enables a local computer to connect to a remote computer. A web service such as HTTP enables remote access of remote web pages. Grid computing is envisioned to allow close interaction among applications running on distant computers simultaneously. Forbes Magazine has projected the global growth of the IT-based economy from $1 trillion in 2001 to $20 trillion by 2015. The evolution from Internet to web and grid services is certainly playing a major role in this growth.

2.1 Computational Grids

Like an electric utility power grid, a computing grid offers an infrastructure that couples computers, software/middleware, special instruments, and people and sensors together. The grid is often con-structed across LAN, WAN, or Internet backbone networks at a regional, national, or global scale. Enterprises or organizations present grids as integrated computing resources. They can also be viewed as virtual platforms to support virtual organizations. The computers used in a grid are pri-marily workstations, servers, clusters, and supercomputers. Personal computers, laptops, and PDAs can be used as access devices to a grid system.

Figure 1.16 shows an example computational grid built over multiple resource sites owned by different organizations. The resource sites offer complementary computing resources, including workstations, large servers, a mesh of processors, and Linux clusters to satisfy a chain of computational needs. The grid is built across various IP broadband networks including LANs and WANs already used by enterprises or organizations over the Internet. The grid is presented to users as an integrated resource pool as shown in the upper half of the figure.

Many national and international grids will be reported in Chapter 7, the NSF TeraGrid in US, EGEE in Europe, and ChinaGrid in China for various distributed scientific grid applications.

2.2 Grid Families

Grid technology demands new distributed computing models, software/middleware support, network protocols, and hardware infrastructures. National grid projects are followed by industrial grid plat-form development by IBM, Microsoft, Sun, HP, Dell, Cisco, EMC, Platform Computing, and others. New grid service providers (GSPs) and new grid applications have emerged rapidly, similar to the growth of Internet and web services in the past two decades. In Table 1.4, grid systems are classified in essentially two categories: computational or data grids and P2P grids. Computing or data grids are built primarily at the national level. In Chapter 7, we will cover grid applications and lessons learned.

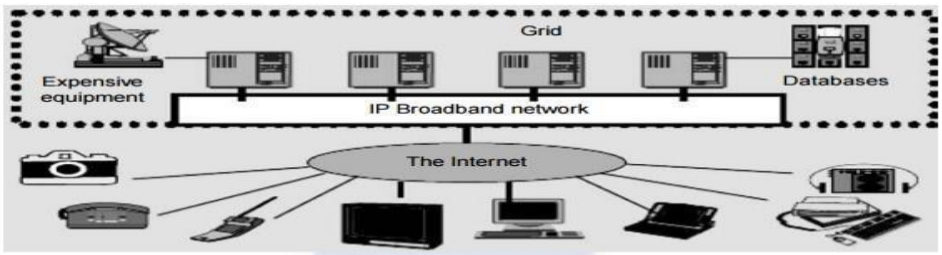


FIGURE 1.16
Computational grid or data grid providing computing utility, data, and information services through resource sharing and cooperation among participating organizations.

3. Peer-to-Peer Network Families

An example of a well-established distributed system is the client-server architecture. In this scenario, client machines (PCs and workstations) are connected to a central server for compute, e-mail, file access, and database applications. The P2P architecture offers a distributed model of networked systems. First, a P2P network is client-oriented instead of server-oriented. In this section, P2P sys-tems are introduced at the physical level and overlay networks at the logical level.

3.1 P2P Systems

In a P2P system, every node acts as both a client and a server, providing part of the system resources. Peer machines are simply client computers connected to the Internet. All client machines act autonomously to join or leave the system freely. This implies that no master-slave relationship exists among the peers. No central coordination or central database is needed. In other words, no peer machine has a global view of the entire P2P system. The system is self-organizing with distributed control.

Figure 1.17 shows the architecture of a P2P network at two abstraction levels. Initially, the peers are totally unrelated. Each peer machine joins or leaves the P2P network voluntarily. Only the participating peers form the physical network at any time. Unlike the cluster or grid, a P2P networkdoes not use a dedicated interconnection network. The physical network is simply an ad hoc network formed at various Internet domains randomly using the TCP/IP and NAI protocols. Thus, the physical network varies in size and topology dynamically due to the free membership in the P2P network.

3.2 Overlay Networks

Data items or files are distributed in the participating peers. Based on communication or file-sharing needs, the peer IDs form an overlay network at the logical level. This overlay is a virtual network

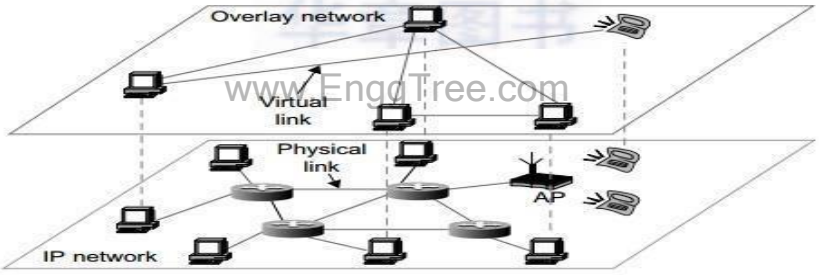


FIGURE 1.17
The structure of a P2P system by mapping a physical IP network to an overlay network built with virtual links.

formed by mapping each physical machine with its ID, logically, through a virtual mapping as shown in Figure 1.17. When a new peer joins the system, its peer ID is added as a node in the overlay network. When an existing peer leaves the system, its peer ID is removed from the overlay network automatically. Therefore, it is the P2P overlay network that characterizes the logical connectivity among the peers.

There are two types of overlay networks: unstructured and structured. An unstructured overlay network is characterized by a random graph. There is no fixed route to send messages or files among the nodes. Often, flooding is applied to send a query to all nodes in an unstructured overlay, thus resulting in heavy network traffic and nondeterministic search results. Structured overlay net-works follow certain connectivity topology and rules for inserting and removing nodes (peer IDs) from the overlay graph. Routing mechanisms are developed to take advantage of the structured overlays.

3.3 P2P Application Families

Based on application, P2P networks are classified into four groups, as shown in Table 1.5. The first family is for distributed file sharing of digital contents (music, videos, etc.) on the P2P network. This includes many popular P2P networks such as Gnutella, Napster, and BitTorrent, among others. Colla-boration P2P networks include MSN or Skype chatting, instant messaging, and collaborative design, among others. The third family is for distributed P2P computing in specific applications. For example, SETI@home provides 25 Tflops of distributed computing power, collectively, over 3 million Internet host machines. Other P2P platforms, such as JXTA,
.NET, and FightingAID@home, support naming, discovery, communication, security, and resource aggregation in some P2P applications. We will dis-cuss these topics in more detail in Chapters 8 and 9.

3.4 P2P Computing Challenges

P2P computing faces three types of heterogeneity problems in hardware, software, and network requirements. There are too many hardware models and architectures to select from; incompatibility exists between software and the OS; and different network connections and protocols

Table 1.5 Major Categories of P2P Network Families [46]

System Features	Distributed File Sharing	Collaborative Platform	Distributed P2P Computing	P2P Platform
Attractive Applications	Content distribution of MP3 music, video, open software, etc.	Instant messaging, collaborative design and gaming	Scientific exploration and social networking	Open networks for public resources
Operational Problems	Loose security and serious online copyright violations	Lack of trust, disturbed by spam, privacy, and peer collusion	Security holes, selfish partners, and peer collusion	Lack of standards or protection protocols
Example Systems	Gnutella, Napster, eMule, BitTorrent, Aimster, KaZaA, etc.	ICQ, AIM, Groove, Magi, Multiplayer Games, Skype, etc.	SETI@home, Geonome@home, etc.	JXTA, .NET, FightingAid@home, etc.

make it too complex to apply in real applications. We need system scalability as the workload increases. System scaling is directly related to performance and bandwidth. P2P networks do have these properties. Data location is also important to affect collective performance. Data locality, network proximity, and interoperability are three design objectives in distributed P2P applications. P2P performance is affected by routing efficiency and self-organization by participating peers.

Fault tolerance, failure management, and load balancing are other important issues in using overlay networks. Lack of trust among peers poses another problem. Peers are strangers to one another. Security, privacy, and copyright violations are major worries by those in the industry in terms of applying P2P technology in business applications [35]. In a P2P network, all clients provide resources including computing power, storage space, and I/O bandwidth. The distributed nature of P2P net-works also increases robustness, because limited peer failures do not form a single point of failure.

By replicating data in multiple peers, one can easily lose data in failed nodes. On the other hand, disadvantages of P2P networks do exist. Because the system is not centralized, managing it is difficult. In addition, the system lacks security. Anyone can log on to the system and cause damage or abuse. Further, all client computers connected to a P2P network cannot be considered reliable or virus-free. In summary, P2P networks are reliable for a small number of peer nodes. They are only useful for applications that require a low level of security and have no concern for data sensitivity. We will discuss P2P networks in Chapter 8, and extending P2P technology to social networking in Chapter 9.

4. Cloud Computing over the Internet

Gordon Bell, Jim Gray, and Alex Szalay [5] have advocated: "Computational science is changing to be data-intensive. Supercomputers must be balanced systems, not just CPU farms but also petascale I/O and networking arrays." In the future, working with large data sets will typically mean sending the computations (programs) to the data, rather than copying the data to the workstations. This reflects the trend in IT of moving computing and data from desktops to large data centers, where there is on-demand provision of software, hardware, and data as a service. This data explosion has promoted the idea of cloud computing.

Cloud computing has been defined differently by many users and designers. For example, IBM, a major player in cloud computing, has defined it as follows: "A cloud is a pool of virtualized computer resources. A cloud can host a variety of different workloads, including batch-style backend jobs and interactive and user-facing applications." Based on this definition, a cloud allows workloads to be deployed and scaled out quickly through rapid provisioning of virtual or physical machines. The cloud supports redundant, self-recovering, highly scalable programming models that allow workloads to recover from many unavoidable hardware/software failures. Finally, the cloud system should be able to monitor resource use in real time to enable rebalancing of allocations when needed.

4.1 Internet Clouds

Cloud computing applies a virtualized platform with elastic resources on demand by provisioning hardware, software, and data sets dynamically (see Figure 1.18). The idea is to move desktop computing to a service-oriented platform using server clusters and huge databases at data centers. Cloud computing leverages its low cost and simplicity to benefit both users and providers. Machine virtualization has enabled such cost-effectiveness. Cloud computing intends to satisfy many user

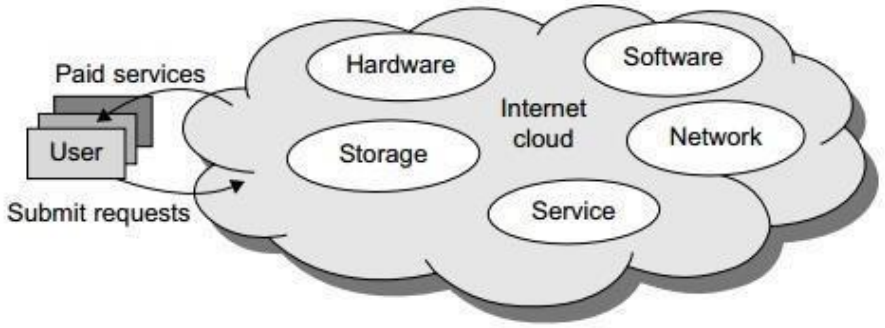


FIGURE 1.18

Virtualized resources from data centers to form an Internet cloud, provisioned with hardware, software, storage, network, and services for paid users to run their applications.

applications simultaneously. The cloud ecosystem must be designed to be secure, trustworthy, and dependable. Some computer users think of the cloud as a centralized resource pool. Others consider the cloud to be a server cluster which practices distributed computing over all the servers used.

3. Explain in detail about Layered Cloud Architecture Design ?BTL4

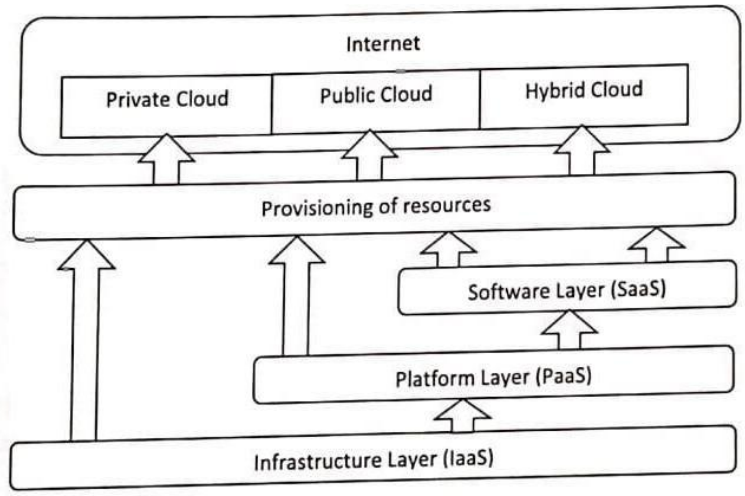
(Definition:2 marks,Explanation 8 marks,Diagram 3 marks)

●◆ The architecture of a cloud is developed at three layers: infrastructure,

platform and application as demonstrated in Figure 1.15. These three development layers are implemented with virtualizationand standardization of hardware and software resources provisioned in the cloud.The services to public, private and hybrid clouds are conveyed to users through networking support over the Internet and intranets involved.

●◆ It is clear that the infrastructure layer is deployed first to support laaS services.
he platform layer is for general purpose and repeated usage of the collection of software resources.This layer provides users with an environment to develop their applications, to test operation flows and to monitor execution results and performance.

●◆ The platform should be able to assure users that they have scalability,
dependability, and security protection.In a way, the virtualized cloud platform serves as a "system middleware" between the infrastructure and application layers of the cloud. The application layer is formed with a collection of all needed software modules for SaaS applications.



Service applications in this layer include daily office management work such as information retrieval, document processing and calendar and authentication services.

●◆ The application layer is also heavily used by enterprises in businesmarketing and sales, consumer relationship management (CRM),financial transactions and supply chain management. From the provider's perspective, the services at various layers demand different amounts of functionality support and resource management by providers.
In general, SaaS demands the most work from the provider, PaaS is in the middle, and laaS demands the least.For example, Amazon EC2 provides not only virtualized CPU resources to users but also management of these provisioned resources.
Services at the application layer demand more work from providers.

◆● The best example of this is the Salesforce.com CRM service in which the provider supplies not only the hardware at the bottom layer and the software at the top layer but also the platform and software tools for user application development and monitoring.

• In Market Oriented Cloud Architecture, as consumers rely on cloud providers to meet more of their computing needs, they will require a specific level of QoS to be maintained by their providers, in order to meet their objectives and sustain their operations. Market-oriented resource management is necessary to regulate the supply and demand of cloud resources to achieve market equilibrium between supply and demand.

●◆ This cloud is basically built with the following entities:
Users or brokers acting on user's behalf submit service requests from anywhere in the world to the data center and cloud to be processed.The request examiner ensures that there is no overloading of resources whereby many service requests cannot be fulfilled successfully due to limited resources.

o The Pricing mechanism decides how service requests are charged. For instance, requests can be charged based on Submission time (peak/off-peak), pricing Rates fixed/changing),(supply/demand) of availability Of resources

• The VM Monitor mechanism keeps track of the availability of VMs and their resource entitlements.
The Accounting mechanism maintains the actual usage of resources by requests so that the final cost can be computed and charged to users.
In addition, the maintained historical usage information can be utilized by the Service Request Examiner and Admission Control mechanism to improve resource allocation decisions.

The Dispatcher mechanism starts the execution of accepted service requests on allocated VMs. The Service Request Monitor mechanism keeps track of the execution progress of service requests.

4. Explain in detail about architectural design challenges of
(i)service availability and data lock in problem
(ii)Data Privacy and Security Concerns?BTL4
(Concept Explanation (i) 7marks, Concept Explanation (ii)6marks)

Challenge 1: Service Availability and Data Lock-in Problem
The management of a cloud service by a single company is often the source of single points of failure.

• To achieve HA, one can consider using multiple cloud providers. Even if a company has multiple data centers located in different geographic regions, it may have common software infrastructure and

accounting systems.

●◆ Therefore, using multiple cloud providers may provide more protection from failures. Another availability obstacle is distributed denial of service (DDoS)

attacks. ◆● Criminals threaten to cut off the incomes of SaaS providers by

making their services unavailable. Some utility computing services offer SaaS providers the opportunity

to defend against DDoS attacks by using quick scale ups. • Software stacks have improved interoperability among different cloud platforms, but the APIs itself are still proprietary. Thus, customers cannot easily extract their data and programs from one site to run on another.

The obvious solution is to standardize the APIs so that a SaaS developer can deploy services and data across multiple cloud providers. ◆● This will rescue the loss of all data due to the failure of a single company. In addition to mitigating data lock-in concerns, standardization of

APIs enables a new usage model in which the same software

infrastructure can be used in both public and private clouds.

Such an option could enable surge computing, in which the public cloud is used to capture the extra tasks that cannot be easily run in the data center of a private cloud.

Challenge 2:
Data Privacy and Security Concerns

Current cloud offerings are essentially public (rather than private) networks, exposing the system to more attacks.
Many obstacles can be overcome immediately with well understood technologies such as encrypted storage, virtual LANs, and network middle boxes (e.g., firewalls, packet filters).

●◆ For example, the end user could encrypt data before placing it in a cloud. Many nations have laws requiring SaaS providers to keep

customer data and copyrighted material within national boundaries. ◆● Traditional network attacks include buffer overflows, DoS attacks,

spyware, malware, rootkits, Trojan horses, and worms. ◆● In a cloud environment, newer attacks may result from hypervisor

malware, guest hopping and hijacking or VM rootkits.

Another type of attack is the man-in-the-middle attack for VM migrations.
In general, passive attacks steal sensitive data or passwords. On the other hand, Active attacks may manipulate kernel data structures which will cause major damage to cloud servers.

5. Explain in detail about architectural design challenges of
(i) Unpredictable Performance and Bottlenecks
(ii) Distributed Storage and Widespread Software Bugs?BTL4

(Concept Explanation,Concept (i) 7marks,Concept Explanation (ii)6marks)

Challenge 3: Unpredictable Performance and Bottlenecks

●◆ Multiple VMs can share CPUs and main memory in cloud

computing, but I/O sharing is problematic.

●◆ For example, to run 75 EC2 instances with the STREAM benchmark requires a mean bandwidth of 1,355 MB/second.

However, for each of the 75 EC2 instances to write 1 GB files to the local disk requires a mean

disk write bandwidth of only 55

MB/second. ◆● This demonstrates the problem of I/O interference between VMs.
One solution is to improve I/O architectures and operating systems to efficiently virtualize interrupts and I/O channels.

●◆ Internet applications continue to become more data intensive. ●◆ If we assume applications to be pulled apart across the boundaries

of clouds, this may complicate data placement and transport. ◆● Cloud users and providers have to think about the implications of

placement and traffic at every level of the system, if they want to minimize costs.

●◆ This kind of reasoning can be seen in Amazon's development of its new CloudFront service.
◆● Therefore, data transfer bottlenecks must be removed, bottlenecklinks must be widened and weak servers should be removed.

Challenge 4: Distributed Storage and Widespread Software Bugs
The database is always growing in cloud applications.

●◆ The opportunity is to create a storage system that will not only meet this growth but also

combine it with the cloud advantage of scaling arbitrarily up and down on demand.

●◆ This demands the design of efficient distributed SANS. ●◆ Data centers must meet
programmer's expectations in terms of scalability, data durability and HA.
Data consistence checking in SAN connected data centers is a major
challenge in cloud computing. Large scale distributed bugs cannot be reproduced, so thedebugging must occur at a scale in the production data centers.No data center will provide such a convenience. One solution may be a reliance on using VMs in cloud computing.The level of virtualization may make it possible to capture valuable information in ways that are impossible without using VMs.

●◆ Debugging over simulators is another approach to attacking

the problem, if the simulator is well designed.

6.Explain in detail about architectural design challenges of
(i) Cloud Scalability, Interoperability
(ii) Software Licensing and Reputation?BTL4

(Concept Explanation (i) 8marks,Concept Explanation (ii)5marks)

Challenge 5: Cloud Scalability, Interoperability,

Standardization

●◆ The pay as you go model applies to storage and network bandwidth; both are counted in terms of the number of bytes used.

●◆ Computation is different depending on virtualization level.

●◆ GAE automatically scales in response to load increases or

decreases and the users are charged by the cycles used.

●◆ AWS charges by the hour for the number of VM instances used, even

if the machine is idle.The opportunity here is to scale quickly up and down in response to load variation, in order to save money, but without violating SLAS. Open Virtualization Format (OVF) describes an open, secure, portable, efficient and extensible format for the packaging anddistribution of VMs. ◆● It also defines a format for distributing software to be deployed in VMs.

●◆ This VM format does not rely on the use of a specific host platform, virtualization platform or guest operating system.

●◆ The approach is to address virtual platform is agnostic packaging with certification and integrity of packaged software.The package supports virtual appliances to span more than one VM.

●◆ OVF also defines a transport mechanism for VM templates and the format can apply to different virtualization platforms with different levels of virtualization.

●◆ In terms of cloud standardization, the ability for virtual appliances to run on any virtual platform.The user is also need to enable VMs to run on heterogeneous hardware platform hypervisors.

◆● This requires hypervisor-agnostic VMs. And also the user need to realize cross platform live migration between x86 Intel and AMD technologies and support legacy hardware for load balancing..

●◆ All these issues are wide open for further research.

Challenge 6: Software Licensing and Reputation SharinMany cloud computing providers originally relied on open source software because the licensing model for commercial software is not ideal for utility computing.
The primary opportunity is either for open source to remain popular or simply for commercial software companies to change their licensing structure to better fit cloud computing. • One can consider using both pay for use and bulk use licensing schemes to widen the business coverage.

PART C
15 Marks

1. Explain in details about Models of Cloud Computing?BTL4
(Definition:2marks,Diagram
:3marks,Concept,Explanation:6marks,Advantages:2marks,Disadvantages:2 marks)

Cloud Computing helps in rendering several services according to roles, companies, etc. Cloud computing models are explained below.

- Infrastructure as a service (IaaS)
- Platform as a service (PaaS)
- Software as a service (SaaS)

1. Infrastructure as a service (IaaS)

Infrastructure as a Service (IaaS) helps in delivering computer infrastructure on an external basis for supporting operations. Generally, IaaS provides services to networking equipment, devices, databases, and web servers.
Infrastructure as a Service (IaaS) helps large organizations, and large enterprises in managing and building their IT platforms. This infrastructure is flexible according to the needs of the client.

Advantages of IaaS

- IaaS is cost-effective as it eliminates capital expenses.
- IaaS cloud provider provides better security than any other software.
- IaaS provides remote access.

Disadvantages of IaaS

- In IaaS, users have to secure their own data and applications.
- Cloud computing is not accessible in some regions of the World.

2.Platform as a service (PaaS)

Platform as a Service (PaaS) is a type of cloud computing that helps developers to build applications and services over the Internet by providing them with a platform.
PaaS helps in maintaining control over their business applications.

Advantages of PaaS

- PaaS is simple and very much convenient for the user as it can be accessed via a web browser.
- PaaS has the capabilities to efficiently manage the lifecycle.

Disadvantages of PaaS

- PaaS has limited control over infrastructure as they have less control over the environment and are not able to make some customizations.
- PaaS has a high dependence on the provider.

3. Software as a service (SaaS)

Software as a Service (SaaS) is a type of cloud computing model that is the work of delivering services and applications over the Internet. The SaaS applications are called Web-Based Software or Hosted Software.
SaaS has around 60 percent of cloud solutions and due to this, it is mostly preferred by companies.

Advantages of SaaS

- SaaS can access app data from anywhere on the Internet.
- SaaS provides easy access to features and services.

Disadvantages of SaaS

- SaaS solutions have limited customization, which means they have some restrictions within the platform.
- SaaS has little control over the data of the user.
- SaaS are generally cloud-based, they require a stable internet connection for proper working.

Cloud infrastructure

Cloud Computing which is one of the demanding technology of current scenario and which has been proved as a revolutionary technology trend for businesses of all sizes. It manages a broad and complex infrastructure setup to provide cloud services and resources to the customers. Cloud Infrastructure which comes under the backend part of cloud architecture represents the hardware and software component such as server, storage, networking, management software, deployment software and virtualization software etc. In backend, cloud infrastructure enables the complete cloud computing system.

Why Cloud Computing Infrastructure :
Cloud computing refers to providing on demand services to the customer anywhere and anytime irrespective of everything where the cloud infrastructure represents the one who activates the

complete cloud computing system. Cloud infrastructure has more capabilities of providing the same services as the physical infrastructure to the customers. It is available for private cloud, public cloud, and hybrid cloud systems with low cost, greater flexibility and scalability.

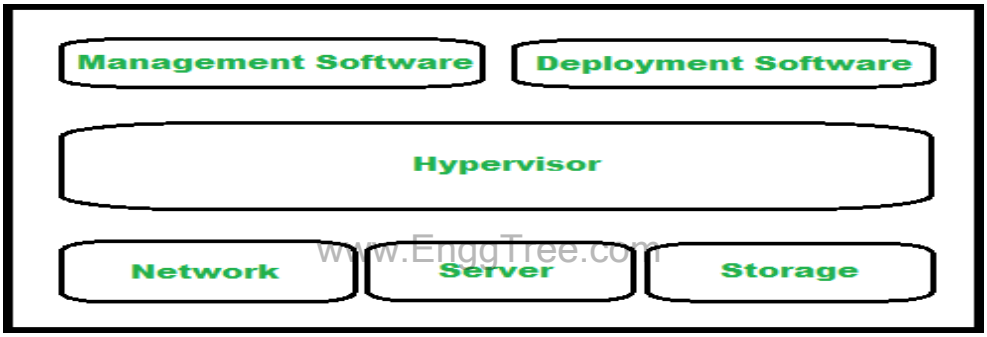
Cloud infrastructure components :

Different components of cloud infrastructure supports the computing requirements of a cloud computing model. Cloud infrastructure has number of key components but not limited to only server, software, network and storage devices. Still cloud infrastructure is categorized into three parts in general i.e.

1. Computing
2. Networking
3. Storage

The most important point is that cloud infrastructure should have some basic infrastructural constraints like transparency, scalability, security and intelligent monitoring etc.

The below figure **represents components of cloud infrastructure**



Components of Cloud Infrastructure

1. Hypervisor :

Hypervisor is a firmware or a low level program which is a key to enable virtualization. It is used to divide and allocate cloud resources between several customers. As it monitors and manages cloud services/resources that's why hypervisor is called as VMM (Virtual Machine Monitor) or (Virtual Machine Manager).

2. Management Software :

Management software helps in maintaining and configuring the infrastructure. Cloud management software monitors and optimizes resources, data, applications and services.

3. Deployment Software :

Deployment software helps in deploying and integrating the application on the cloud. So, typically it helps in building a virtual computing environment.

4. Network :

It is one of the key component of cloud infrastructure which is responsible for connecting cloud services over the internet. For the transmission of data and resources externally and internally network is must required.

5. Server :

Server which represents the computing portion of the cloud infrastructure is responsible for managing and delivering cloud services for various services and partners, maintaining security etc.

6. Storage :

Storage represents the storage facility which is provided to different organizations for storing and managing data. It provides a facility of extracting another resource if one of the resource fails as it keeps many copies of storage.

Along with this, virtualization is also considered as one of important component of cloud infrastructure. Because it abstracts the available data storage and computing power away from the actual hardware and the users interact with their cloud infrastructure through GUI (Graphical User Interface).

2. Explain about NIST reference architecture?BTL4

(Definition:2 marks,Diagram:4 marks,Explanation:9 marks)

NIST stands for National Institute of Standards and Technology

The goal is to achieve effective and secure cloud computing to reduce cost and improve services

• NIST composed for six major workgroups specific to cloud

computing o Cloud computing target business use cases work group

o Cloud computing Reference architecture and Taxonomy work

group

o Cloud computing standards roadmap work group

o Cloud computing SAJACC (Standards Acceleration to Jumpstart Adoption of Cloud Computing) work group

o Cloud Computing security work group

• Objectives of NIST Cloud Computing reference

architecture Illustrate and understand the various level of

services

o To provide technical reference

o Categorize and compare services of cloud computing

o Analysis of security, interoperatability and portability

●◆ In general, NIST generates report for future reference which includes survey, analysis of
existing cloud computing reference model, vendors and federal agencies.

The conceptual reference architecture shown in figure 1.4 involves five actors. Each actor as entity participates in cloud computing
Cloud consumer: A person or an organization that maintains a business relationship with and uses a services from cloud providers

Cloud provider: A person, organization or entity responsible for making a service available to interested parties

Cloud auditor: A party that conduct independent assessment of cloud services, information system operation, performance and security of cloud implementation

●◆ Cloud broker: An entity that manages the performance and delivery of cloud services and negotiates relationship between cloud provider and consumer.

●◆ Cloud carrier: An intermediary that provides connectivity and transport of cloud services from cloud providers to consumers.

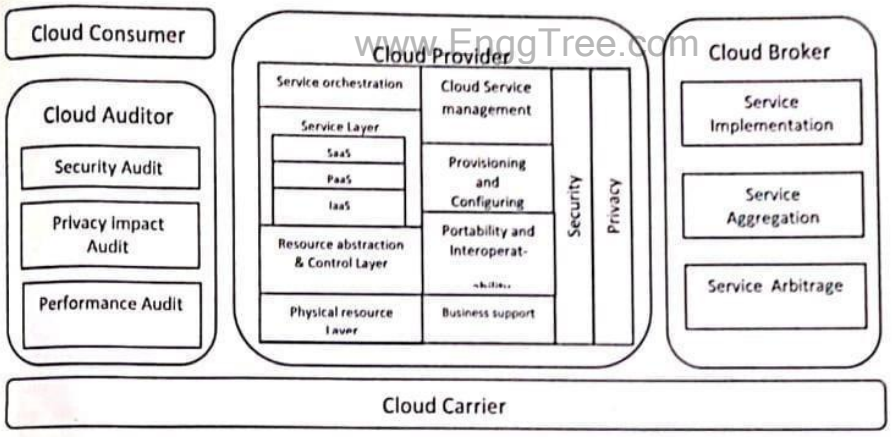


Figure 1.5 illustrates the common interaction exist in between cloud consumer and provider where as the broker used to provide service to consumer and auditor collects the audit information.

The interaction between the actors may lead to different use case scenario.

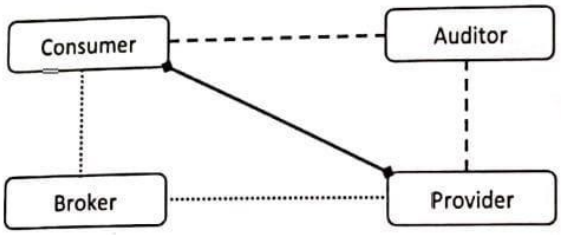


Figure 1.5 Interaction between actors

Figure 1.6 shows one kind of scenario in which the Cloud consumer may request service from a cloud broker instead of contacting service provider directly. In this case, a cloud broker can create a new service by combining multiple services

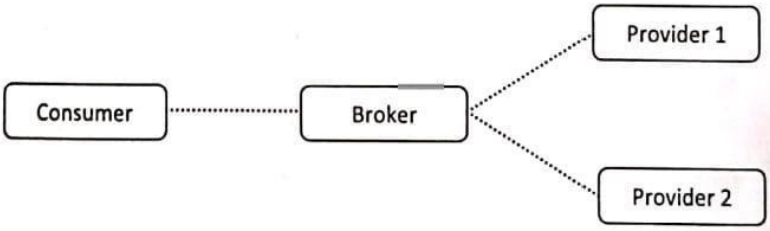
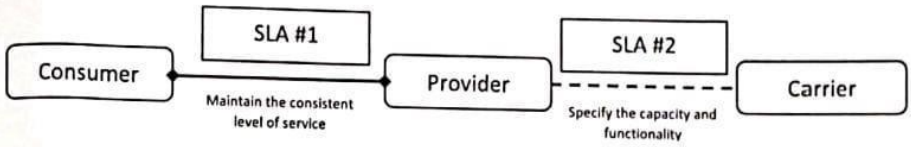


Figure 1.6 Service from Cloud Broker

Figure 1.7 illustrates the usage of different kind of Service Level Agreement (SLA) between consumer, provider and carrier.



Cloud consumer is a principal stake holder for the cloud computing service and requires service level agreements to specify the performance requirements fulfilled by a cloud provider.

●◆ The service level agreement covers Quality of Service and Security aspects. Consumers have limited rights to access the software applications.

There are three kinds of cloud consumers: SaaS consumers, PaaS Consumers and IaaS consumers.

◆● SaaS consumers are members directly access the software application. For example, document management, content management, social networks, financial billing and so on.

PaaS consumers are used to deploy, test, develop and manage applications hosted in cloud environment. Database application deployment, development and testing is an example for these kind of consumer.

●◆ laaS Consumer can access the virtual computer, storage and network infrastructure. For example, usage of Amazon EC2 instance to deploy the web application.

On the other hand, Cloud Providers have complete rights to access software applications. In Software as a Service model, cloud provider is allowed to configure, maintain and update the operations of software application.

• Management process is done by Integrated Development environment and Software Development Kit in Platform as a Service model.

Infrastructure as a Service model covers Operating System and Networks.

●◆ Normally, the service layer defines the interfaces for cloud consumers to access the computing services.

• Resource abstraction and control layer contains the system components that cloud provider use to provide and mange access to the physical computing resources through software abstraction.
• Resource abstraction covers virtual machine management and virtual storage management. Control layer focus on resource allocation, access control and usage monitoring.
• Physical resource layer includes physical computing resources such as CPU, Memory, Router, Switch, Firewalls and Hard Disk Drive.
Service orchestration describes the automated arrangement, coordination and management of complex computing system
• In cloud service management, business support entails the set of business related services dealing with consumer and supporting services which includes content management, contract management, inventory management, accounting service, reporting service and rating service.

• Provisioning of equipments, wiring and transmission is mandatory to setup a new service that provides a specific application to cloud consumer. Those details are described in Provisioning and Configuring management.

Portability enforces the ability to work in more than one computing environment without major task. Similarly, Interoperatability means the ability of the system work with other system.

• Security factor is applicable to enterprise and Government. It may include privacy.
Privacy is one applies to a cloud consumer's rights to safe guard his information from other consumers are parties.

3.Explain in details about Cloud Deployment Models?BTL4

(Diagram 3 marks,Explanation:6 marks,Advantages 2 marks,Disadvantages 2 marks,Tabular column 2 marks)

In cloud computing, we have access to a shared pool of computer resources (servers, storage, programs, and so on) in the cloud. You simply need to request additional resources when you require them. Getting resources up and running quickly is a breeze thanks to the clouds. It is possible to release resources that are no longer necessary. This method allows you to just pay for what you use. Your cloud provider is in charge of all upkeep.

Cloud Deployment Model

Cloud Deployment Model functions as a virtual computing environment with a deployment architecture that varies depending on the amount of data you want to store and who has access to

the infrastructure

Types of Cloud Computing Deployment Models

The cloud deployment model identifies the specific type of cloud environment based on ownership, scale, and access, as well as the cloud's nature and purpose. The location of the servers you're utilizing and who controls them are defined by a cloud deployment model. It specifies how your cloud infrastructure will look, what you can change, and whether you will be given services or will have to create everything yourself. Relationships between the infrastructure and your users are also defined by cloud deployment types. Different types of cloud computing deployment models are described below.

- Public Cloud
- Private Cloud
- Hybrid Cloud
- Community Cloud
- Multi-Cloud

Public Cloud

The public cloud makes it possible for anybody to access systems and services. The public cloud may be less secure as it is open to everyone. The public cloud is one in which cloud infrastructure services are provided over the internet to the general people or major industry groups. The infrastructure in this cloud model is owned by the entity that delivers the cloud services, not by the consumer. It is a type of cloud hosting that allows customers and users to easily access systems and services. This form of cloud computing is an excellent example of cloud hosting, in which service providers supply services to a variety of customers. In this arrangement, storage backup and retrieval services are given for free, as a subscription, or on a per-user basis. For example, Google App Engine etc.

Public Cloud

Advantages of the Public Cloud Model

- **Minimal Investment:** Because it is a pay-per-use service, there is no substantial upfront fee, making it excellent for enterprises that require immediate access to resources.
- **No setup cost:** The entire infrastructure is fully subsidized by the cloud service providers, thus there is no need to set up any hardware.
- **Infrastructure Management is not required:** Using the public cloud does not necessitate infrastructure management.
- **No maintenance:** The maintenance work is done by the service provider (not users).
- **Dynamic Scalability:** *To fulfill your company's needs, on-demand resources are accessible.*

Disadvantages of the Public Cloud Model

- **Less secure:** Public cloud is less secure as resources are public so there is no guarantee of high-level security.
- **Low customization:** It is accessed by many public so it can't be customized according to personal requirements.

Private Cloud

The private cloud deployment model is the exact opposite of the public cloud deployment model. It's a one-on-one environment for a single user (customer). There is no need to share your hardware with anyone else. The distinction between private and public clouds is in how you handle all of the hardware. It is also called the "internal cloud" & it refers to the ability to access systems and services within a given border or organization. The cloud platform is implemented in a cloud- based secure environment that is protected by powerful firewalls and under the supervision of an organization's IT department. The private cloud gives greater flexibility of control over cloud resources.

Private Cloud

Advantages of the Private Cloud Model

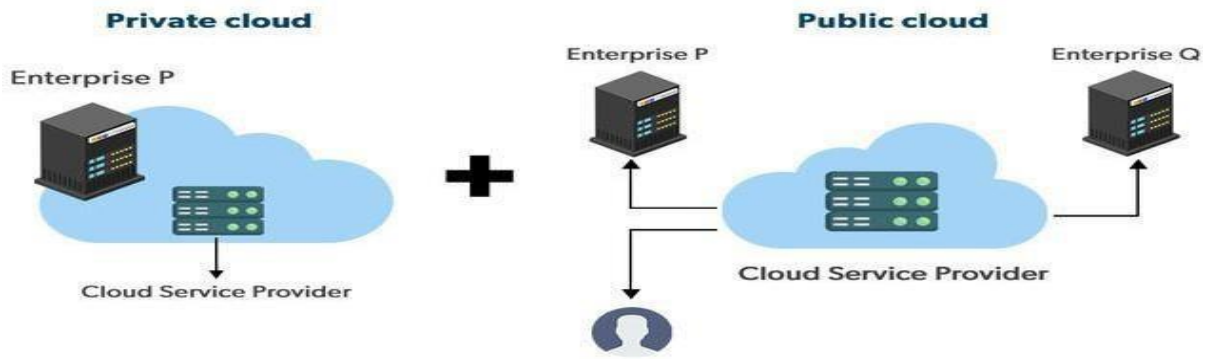
- **Better Control:** You are the sole owner of the property. You gain complete command over service integration, IT operations, policies, and user behavior.
- **Data Security and Privacy:** It's suitable for storing corporate information to which only authorized staff have access. By segmenting resources within the same infrastructure, improved access and security can be achieved.
- **Supports Legacy Systems:** This approach is designed to work with legacy systems that are unable to access the public cloud.
- **Customization:** Unlike a public cloud deployment, a private cloud allows a company to tailor its solution to meet its specific needs.

Disadvantages of the Private Cloud Model

- **Less scalable:** Private clouds are scaled within a certain range as there is less number of clients.
- **Costly:** Private clouds are more costly as they provide personalized facilities.

Hybrid Cloud

By bridging the public and private worlds with a layer of proprietary software, hybrid cloud computing gives the best of both worlds. With a hybrid solution, you may host the app in a safe environment while taking advantage of the public cloud's cost savings. Organizations can move data and applications between different clouds using a combination of two or more cloud deployment methods, depending on their needs.



Hybrid Cloud

Advantages of the Hybrid Cloud Model

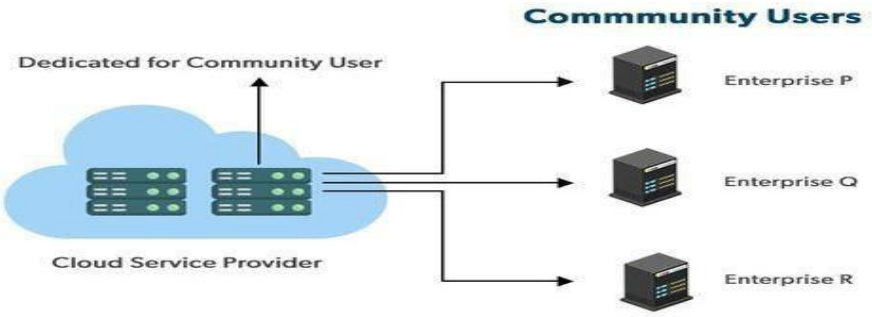
- **Flexibility and control:** Businesses with more flexibility can design personalized solutions that meet their particular needs.
- **Cost:** Because public clouds provide scalability, you'll only be responsible for paying for the extra capacity if you require it.
- **Security:** Because data is properly separated, the chances of data theft by attackers are considerably reduced.

Disadvantages of the Hybrid Cloud Model

- **Difficult to manage:** Hybrid clouds are difficult to manage as it is a combination of both public and private cloud. So, it is complex.
- **Slow data transmission:** Data transmission in the hybrid cloud takes place through the public cloud so latency occurs.

Community Cloud

It allows systems and services to be accessible by a group of organizations. It is a distributed system that is created by integrating the services of different clouds to address the specific needs of a community, industry, or business. The infrastructure of the community could be shared between the organization which has shared concerns or tasks. It is generally managed by a third party or by the combination of one or more organizations in the community.



Community Cloud

Advantages of the Community Cloud Model

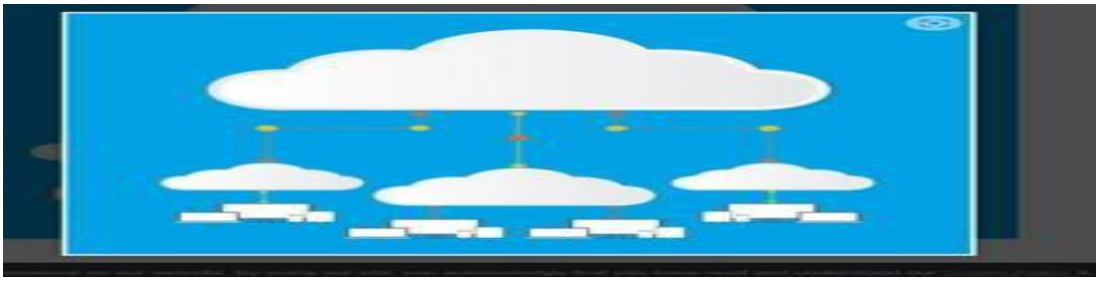
- **Cost Effective:** It is cost-effective because the cloud is shared by multiple organizations or communities.
- **Security:** Community cloud provides better security.
- **Shared resources:** It allows you to share resources, infrastructure, etc. with multiple organizations.
- **Collaboration and data sharing:** It is suitable for both collaboration and data sharing.

Disadvantages of the Community Cloud Model

- **Limited Scalability:** Community cloud is relatively less scalable as many organizations share the same resources according to their collaborative interests.
- **Rigid in customization:** As the data and resources are shared among different organizations according to their mutual interests if an organization wants some changes according to their needs they cannot do so because it will have an impact on other organizations.

Multi-Cloud

We're talking about employing multiple cloud providers at the same time under this paradigm, as the name implies. It's similar to the hybrid cloud deployment approach, which combines public and private cloud resources. Instead of merging private and public clouds, multi- cloud uses many public clouds. Although public cloud providers provide numerous tools to improve the reliability of their services, mishaps still occur. It's quite rare that two distinct clouds would have an incident at the same moment. As a result, multi-cloud deployment improves the high availability of your services even more.



Multi-Cloud

Advantages of the Multi-Cloud Model

- You can mix and match the best features of each cloud provider's services to suit the demands of your apps, workloads, and business by choosing different cloud providers.
- **Reduced Latency:** To reduce latency and improve user experience, you can choose cloud regions and zones that are close to your clients.
- **High availability of service:** It's quite rare that two distinct clouds would have an incident at the same moment. So, the multi-cloud deployment improves the high availability of your services.

Disadvantages of the Multi-Cloud Model

- **Complex:** The combination of many clouds makes the system complex and bottlenecks may occur.
- **Security issue:** Due to the complex structure, there may be loopholes to which a hacker can take advantage hence, makes the data insecure.

Overall Analysis of Cloud Deployment Models

The overall Analysis of these models with respect to different factors is described below.

Factors	Public Cloud	Private Cloud	Community Cloud	Hybrid Cloud
Initial Setup	Easy	Complex, requires a professional team to setup	Complex, requires a professional team to setup	Complex, requires a professional team to setup
Scalability and Flexibility	High	High	Fixed	High

Factors	Public Cloud	Private Cloud	Community Cloud	Hybrid Cloud
Cost-Comparison	Cost-Effective	Costly	Distributed cost among members	Between public and private cloud
Reliability	Low	Low	High	High
Data Security	Low	High	High	High
Data Privacy	Low	High	High	High

UNIT 2
VIRTUALIZATION BASICS

SYLLABUS:Virtualization basicsTaxonomy of virtual machines-Hypervisor-keyconcepts-virtualisation structure-Implementation levels of virtualization-virtualization types-full virtualization-para virtualization-Hardware virtualization of CPU,memory,I/Odevices

PART A
2 Marks

1. Define virtual machine?BTL1

A VM is a virtualized instance of a computer that can perform almost all of the same functions as a computer, including running applications and operating systems. Virtual machines run on a physical machine and access computing resources from software called a hypervisor.

2. Define a cloud virtual machine?BTL1

A cloud virtual machine is the digital version of a physical computer that can run in a cloud. Like a physical machine, it can run an operating system, store data, connect to networks, and do all the other computing functions.

3.List the Advantages of cloud virtual machine?BTL1

There are many advantages to using cloud virtual machines instead of physical machines,

including:

- Low cost: It is cheaper to spin off a virtual machine in the clouds than to procure a physical machine.
- Easy scalability: We can easily scale in or scale out the infrastructure of a cloud virtual machine based on load.
- Ease of setup and maintenance: Spinning off virtual machines is very easy as compared to buying actual hardware. This helps us get set up quickly.

- Shared responsibility: Disaster recovery becomes the responsibility of the Cloud provider. We don't need a different disaster recovery site incase our primary site goes down.

4. List the Benefits of Virtualization?BTL1

- More flexible and efficient allocation of resources.
- Enhance development productivity.
- It lowers the cost of IT infrastructure.
- Remote access and rapid scalability.
- High availability and disaster recovery.
- Pay peruse of the IT infrastructure on demand.
- Enables running multiple operating systems.

5. Is there any limit to no. of virtual machines one can install?BTL1

In general there is no limit because it depends on the hardware of your system. As the VM is using hardware of your system, if it goes out of it's capacity then it will limit you not to install further virtual machines.

6. Can one access the files of one VM from another?BTL1

In general No, but as an advanced hardware feature, we can allow the file-sharing for different virtual machines.

7. What are Types of Virtual Machines ?BTL1

we can classify virtual machines into two types:

1. System Virtual Machine
2. Process Virtual Machine

8. What are Types of Virtualization?BTL1

1. Application Virtualization
2. Network Virtualization
3. Desktop Virtualization
4. Storage Virtualization
5. Server Virtualization
6. Data virtualization

9. Define Uses of Virtualization?BTL1

- Data-integration
- Business-integration
- Service-oriented architecture data-services
- Searching organizational data

10. What is mean by hypervisor?BTL1

A hypervisor, also known as a virtual machine monitor or VMM, is software that creates and runs virtual machines (VMs). A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing.

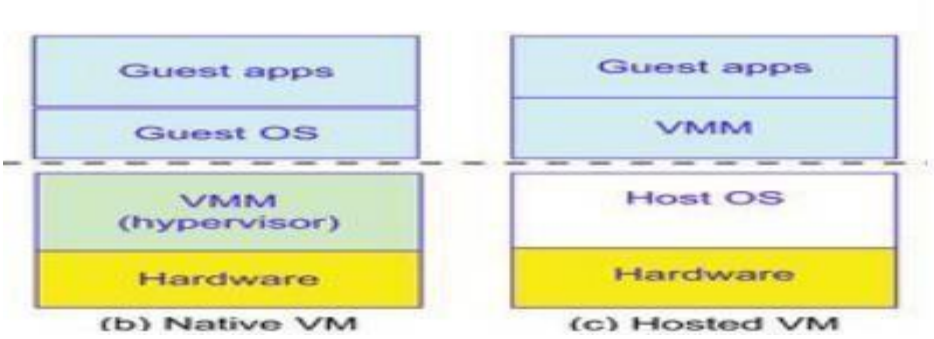
11. List down the different types of VMM?BTL1

- VMWare ESXi
- Xen.
- KVM

12. What are the types of hypervisor?BTL1

Type 1 hypervisors run directly on the system hardware. They are often referred to as a "native"or "bare metal" or "embedded" hypervisors in vendor literature.

Type 2 hypervisors run on a host operating system.



13. What is Virtualized Infrastructure Manager (VIM). ?BTL1

The virtualized infrastructure manager (VIM) in a Network Functions Virtualization (NFV) implementation manages the hardware and software resources that the service provider uses to create service chains and deliver network services to customers.

14. Differentiate between system VM and Process VM?BTL2

A Process virtual machine, sometimes called an application virtual machine, runs as a normal application inside a host OS and supports a single process. It is created when that process is started and destroyed when it exits. Its purpose is to provide a platform-independent programming environment that abstracts away details of the underlying hardware or operating system, and allows a program to executein the same way on any platform.

A System virtual machine provides a complete system platform which supports the execution ofa complete operating system (OS),Just like you said VirtualBox is one example.

15. Mention the signification of Network Virtualization?BTL1

Network virtualization helps organizations achieve major advances in speed, agility, and security by automating and simplifying many of the processes that go into running a data center network and managing networking and security in the cloud. Here are some of the key benefits of network virtualization: Reduce network provisioning time from weeks to minutes

Achieve greater operational efficiency by automating manual processes Place and move workloads independently of physical topology Improve network security within the data center

16. List the implementation levels of virtualization [R]?BTL1

- Instruction set architecture(ISA) level
- Hardware abstraction
- layer(HAL) level Operating
- System Level Library(user-level
- API) level Application level

17. Explain hypervisor architecture ?BTL1

A hypervisor or virtual machine monitor (VMM) is a piece of computer software, firmware orhardware that creates and runs virtual machines.

18. Define para-virtualization?BTL1

Para-virtualization is a virtualization technique that presents a software interface to virtual machines that is similar, but not identical to that of the underlying hardware.

19. What are the two types of hypervisor ?BTL1

Micro-kernel architecture Monolithic hypervisor architecture

20. Define Application virtualization?BTL1

Application-level virtualization is a technique allowing applications to be run in runtime environments that do not natively support all the features required by such applications. These techniques are mostly concerned with partial file systems, libraries, and operating system componentemulation.

21. Define server virtualization?BTL1

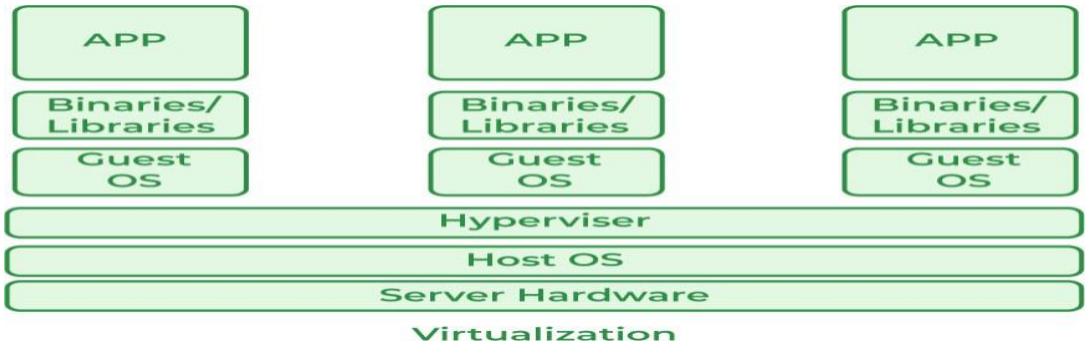
Server virtualization is the process of dividing a physical server into multiple unique and isolated virtual servers by means of a software application. Each virtual server can run its own operating systemsindependently.

PART B
13 Marks

1. Explain in details about Virtualization in Cloud Computing andTypes?BTL4
(Definition:2 marks,Diagram:3 marks,Explanation:8 marks)

Virtualization is a technique how to separate a service from the underlying physical delivery of that service. It is the process of creating a virtual version of something like computer hardware. It was initially developed during the mainframe era. It involves using specialized software to create a virtual or software-created version of a computing resource rather than the actual version of the same resource. With the help of Virtualization, multiple operating systems and applications can run on the same machine and its same hardware at the same time, increasing the utilization and flexibility of hardww `q1are.In other words, one of the main cost-effective, hardware-reducing, and energy-saving techniques used by cloud providers is Virtualization. Virtualization allows

sharing of a single physical instance of a resource or an application among multiple customers and organizations at one time. It does this by assigning a logical name to physical storage and providing a pointer to that physical resource on demand. The term virtualization is often synonymous with hardware virtualization, which plays a fundamental role in efficiently delivering Infrastructure-as- a-Service (IaaS) solutions for cloud computing. Moreover, virtualization technologies provide a virtual environment for not only executing applications but also for storage, memory, and networking.



Virtualization

- Host Machine: The machine on which the virtual machine is going to be built is known as Host Machine.
- Guest Machine: The virtual machine is referred to as a Guest Machine.

Work of Virtualization in Cloud Computing

Virtualization has a prominent impact on Cloud Computing. In the case of cloud computing, users but with the help of Virtualization, users have the extra benefit of sharing the infrastructure. Cloud Vendors take care of the required physical resources, but these cloud providers charge a huge amount for these services which impacts every user or organization. Virtualization helps Users or Organisations in maintaining those services which are required by a company through external (third-party) people, which helps in reducing costs to the company. This is the way through which Virtualization works in Cloud Computing.

Benefits of Virtualization

- More flexible and efficient allocation of resources.
- Enhance development productivity.
- It lowers the cost of IT infrastructure.
- Remote access and rapid scalability.
- High availability and disaster recovery.
- Pay peruse of the IT infrastructure on demand.
- Enables running multiple operating

systems. Drawback of Virtualization

- High Initial Investment: Clouds have a very high initial investment, but it is also true that it will help in reducing the cost of companies.
- Learning New Infrastructure: As the companies shifted from Servers to Cloud, it requires highly skilled staff who have skills to work with the cloud easily, and for this, you have to hire new staff or provide training to current staff.
- Risk of Data: Hosting data on third-party resources can lead to putting the data at risk, it has the chance of getting attacked by any hacker or cracker very easily.

For more benefits and drawbacks, you can refer to the Pros and Cons of Virtualization. Characteristics of Virtualization

- Increased Security: The ability to control the execution of a guest program in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment. All the operations of the guest programs are

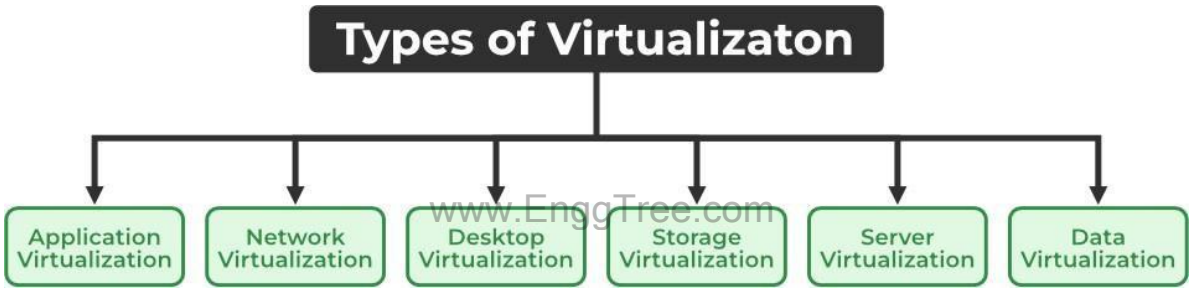
generally performed against the virtual machine, which then translates and applies them to the host programs.

- Managed Execution: In particular, sharing, aggregation, emulation, and isolation are the most relevant features.
- Sharing: Virtualization allows the creation of a separate computing environment within the same host.
- Aggregation: It is possible to share physical resources among several guests, but virtualization also allows aggregation, which is the opposite process.

For more characteristics, you can refer to Characteristics of Virtualization.

Types of Virtualization

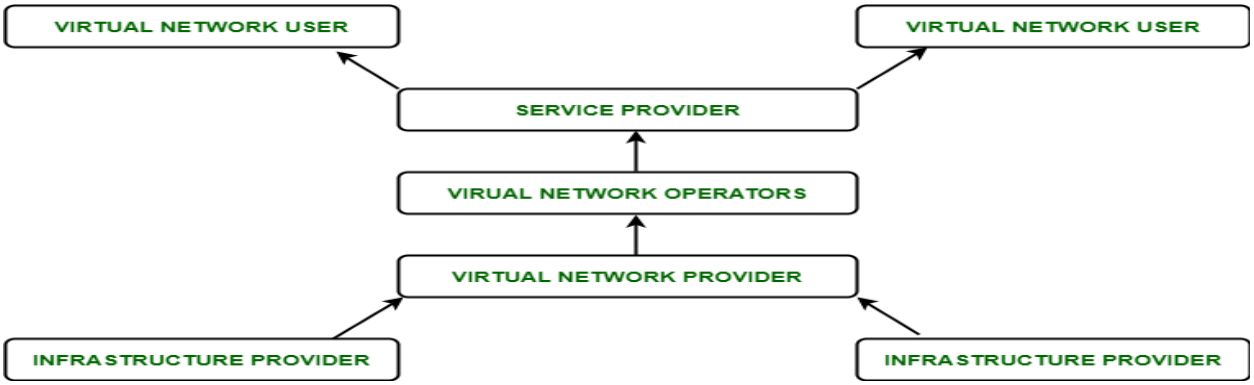
1. Application Virtualization
2. Network Virtualization
3. Desktop Virtualization
4. Storage Virtualization
5. Server Virtualization
6. Data virtualization



Types of Virtualization

1. Application Virtualization: Application virtualization helps a user to have remote access to an application from a server. The server stores all personal information and other characteristics of the application but can still run on a local workstation through the internet. An example of this would be a user who needs to run two different versions of the same software. Technologies that use application virtualization are hosted applications and packaged applications.

2. Network Virtualization: The ability to run multiple virtual networks with each having a separate control and data plan. It co-exists together on top of one physical network. It can be managed by individual parties that are potentially confidential to each other. Network virtualization provides a facility to create and provision virtual networks, logical switches, routers, firewalls, load balancers, Virtual Private Networks (VPN), and workload security within days or even weeks.

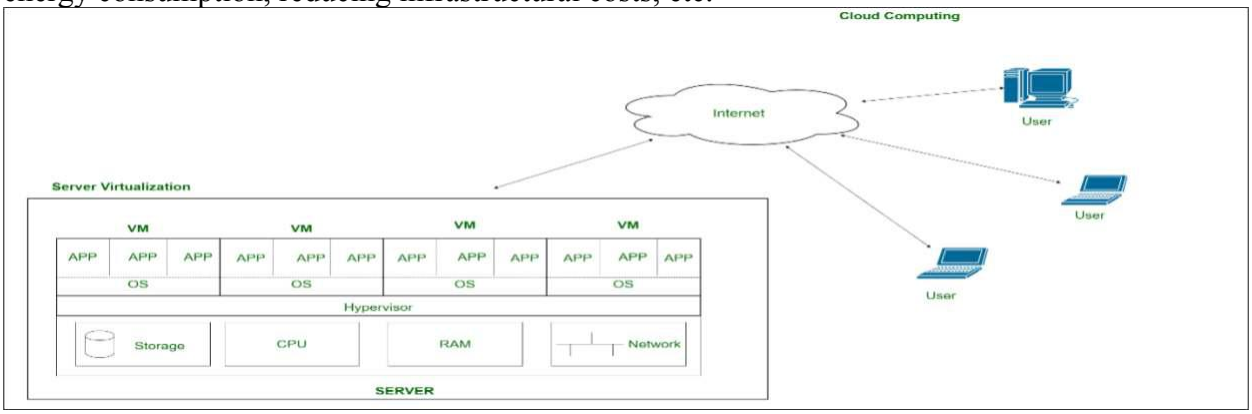


Network Virtualization

3. Desktop Virtualization: Desktop virtualization allows the users' OS to be remotely stored on a server in the data center. It allows the user to access their desktop virtually, from any location by a different machine. Users who want specific operating systems other than Windows Server will need to have a virtual desktop. The main benefits of desktop virtualization are user mobility, portability, and easy management of software installation, updates, and patches.

4. Storage Virtualization: Storage virtualization is an array of servers that are managed by a virtual storage system. The servers aren't aware of exactly where their data is stored and instead function more like worker bees in a hive. It makes managing storage from multiple sources be managed and utilized as a single repository. storage virtualization software maintains smooth operations, consistent performance, and a continuous suite of advanced functions despite changes, breaks down, and differences in the underlying equipment.

5. Server Virtualization: This is a kind of virtualization in which the masking of server resources takes place. Here, the central server (physical server) is divided into multiple different virtual servers by changing the identity number, and processors. So, each system can operate its operating systems in an isolated manner. Where each sub-server knows the identity of the central server. It causes an increase in performance and reduces the operating cost by the deployment of main server resources into a sub-server resource. It's beneficial in virtual migration, reducing energy consumption, reducing infrastructural costs, etc.



Server Virtualization

6. Data Virtualization: This is the kind of virtualization in which the data is collected from various sources and managed at a single place without knowing more about the technical information like how data is collected, stored & formatted then arranged that data logically so that its virtual view can be accessed by its interested people and stakeholders, and users through the various cloud services remotely. Many big giant companies are providing their services like Oracle, IBM, At scale, Cdata, etc.

Uses of Virtualization

- Data-integration
- Business-integration
- Service-oriented architecture data-services
- Searching organizational data

2. **What are the difference between Cloud computing and Virtualization:-BTL2**

(Comparison:13 marks)

S.NO	Cloud Computing	Virtualization
1.	Cloud computing is used to provide pools and automated resources that can be accessed on-demand.	While It is used to make various simulated environments through a physical hardware system.
2.	Cloud computing setup is tedious, complicated.	While virtualization setup is simple as compared to cloud computing.
3.	Cloud computing is high scalable.	While virtualization is low scalable compared to cloud computing.
4.	Cloud computing is Very flexible.	While virtualization is less flexible than cloud computing.
5.	In the condition of disaster recovery, cloud computing relies on multiple machines.	While it relies on single peripheral device.
6.	In cloud computing, the workload is stateless.	In virtualization, the workload is stateful.
7.	The total cost of cloud computing is higher than virtualization.	The total cost of virtualization is lower than Cloud Computing.

S.NO	Cloud Computing	Virtualization
8.	Cloud computing requires many dedicated hardware.	While single dedicated hardware can do a great job in it.
9.	Cloud computing provides unlimited storage space.	While storage space depends on physical server capacity in virtualization.
10.	Cloud computing is of two types : Public cloud and Private cloud.	Virtualization is of two types : Hardware virtualization and Application virtualization.
11.	In Cloud Computing, Configuration is image based.	In Virtualization, Configuration is template based.
12.	In cloud computing, we utilize the entire server capacity and the entire servers are consolidated.	In Virtualization, the entire servers are on-demand.

3. **Explain in details about hypervisor and it is types?BTL4**

(Definition:2 marks,Concept Explanation:8marks,Diagram:3 marks)

Hypervisor

●◆ Hardware level virtualization is a virtualization technique that provides an abstract execution environment in terms of computer hardware on top of which a guest operating system can be run.

●◆ In this model, the guest is represented by the operating system, the host by the physical computer hardware, the virtual machine by its emulation and the virtual machine manager by the hypervisor.

●◆ The hypervisor is generally a program or a combination of software and hardware that allows the abstraction of the underlying physicalhardware.

Hardware level virtualization is also called system virtualization, since it provides ISA to virtual machines, which is the representation of the hardware interface of a system.
This is to differentiate it from process virtual machines, which expose ABI to virtual machines.

●◆ Hypervisors is a fundamental element of hardware virtualization is the hypervisor, or virtual machine manager (VMM).

●◆ It recreates a hardware environment in which guest operating systems are installed. ●◆ There are two major types of hypervisor: Type I and Type II. Figure

2.3 shows different type of hypervisors.

o Type I hypervisors run directly on top of the hardware.

■Type I hypervisor take the place of the operating systems and interact directly with the ISA interface exposed by the underlying hardware and they emulate this interface in order to allow the management of guest operating systems.

This type of hypervisor is also called a native virtual machine since it runs natively on hardware. o Type II hypervisors require the support of an operating system to provide virtualization services.

This means that they are programs managed by the

operating system, which interact with it through the ABI and emulate the ISA of virtual hardware for guest operating systems.

■This type of hypervisor is also called a hosted virtual

machine since it is hosted within an operating system.

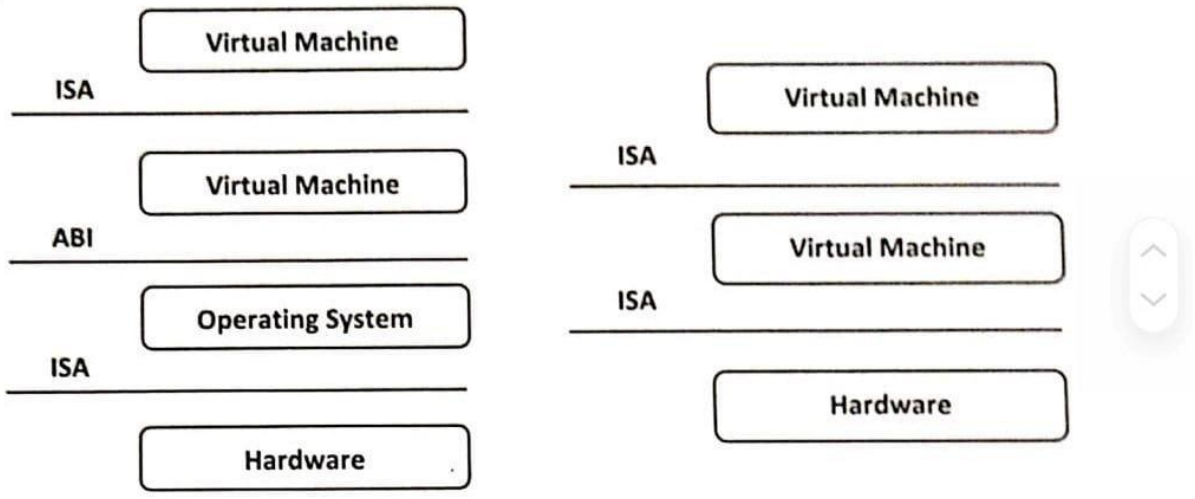


Figure 2.3 Hosted virtual machine and native virtual machine

4. **What are the Taxonomy of virtual machines?BTL1**
(Concept explanation:10 marks,Diagram:3 marks)

Virtualization is mainly used to emulate execution environments,storage and networks.
Execution virtualization techniques into two major categories by considering the type of host they require.
Process level techniques are implemented on top of an existing

operating system, which has full control of the hardware. System level techniques are implemented directly on hardware and do not require or require a minimum of support from existing operating system.

Within these two categories we can list various techniques that offer the guest a different type of virtual computation environment:

◆● Bare hardware

o Operating system resources o Low level programming language

Application libraries

• Execution virtualization includes all techniques that aim to emulate an execution environment that is separate from the one hosting the virtualization layer.
All these techniques concentrate their interest on providing support for the execution of programs, whether these are the operating system, a binary specification of a program compiled against an abstract machine model or an application.

●◆ Therefore, execution virtualization can be implemented directly on top of the hardware by the operating system, an application and libraries (dynamically or statically) linked to an application image. ◆● Modern computing systems can be expressed in terms of the

reference model described in Figure 2.1.

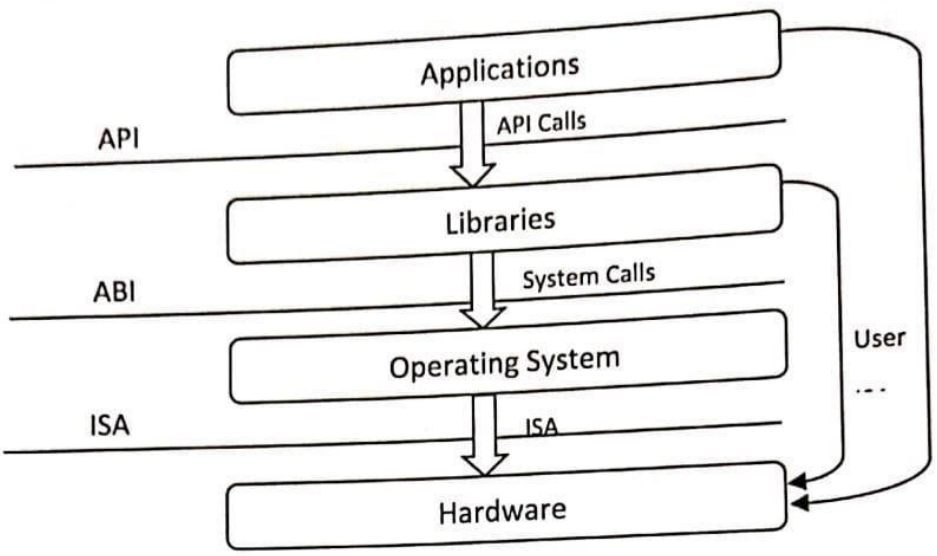


Figure 2.1 Machine reference model

At the bottom layer, the model for the hardware is expressed in terms of the Instruction Set Architecture (ISA), which defines the instruction set for the processor, registers, memory and an interrupt management.

●◆ISA is the interface between hardware and software.

●◆ ISA is important to the operating system (OS) developer (System ISA) and developers of applications that directly manage the underlying hardware (User ISA).

●◆ The application binary interface (ABI) separates the operating system layer from the applications and libraries, which are managed by the

OS. • ABI covers details such as low level data types, alignment, call

conventions and defines a format for executable programs. ◆● System calls are defined at this level. This interface allows portability of applications and libraries across operating systems that implement the same ABI.

●◆ The highest level of abstraction is represented by the application programming interface (API), which interfaces applications to libraries and the underlying operating system.

●◆ For this purpose, the instruction set exposed by the hardware has been

divided into different security classes that define who can operate with them. The first distinction can be made between privileged and non

privileged instructions.

o Non privileged instructions are those instructions that can be used without interfering with other tasks because they do not access shared resources.
This category contains all the floating, fixed-point, and arithmetic instructions.

• Privileged instructions are those that are executed under specific restrictions and are mostly used for sensitive operations, which expose (behavior-sensitive) or modify (control-sensitive) the privileged state.

●◆. Some types of architecture feature more than one class of privileged instructions and implement a finer control of how these instructions can be accessed.

For instance, a possible implementation features a hierarchy of privileges illustrate in the figure 2.2 in the form of ring-based security: Ring 0, Ring 1, Ring 2, and Ring 3;
Ring 0 is in the most privileged level and Ring 3 in the least privileged level.

Ring 0 is used by the kernel of the OS, rings 1 and 2 are used by the OS level services, and Ring 3 is used by the user.

Recent systems support only two levels, with Ring 0 for

supervisor mode and Ring 3 for user mode.

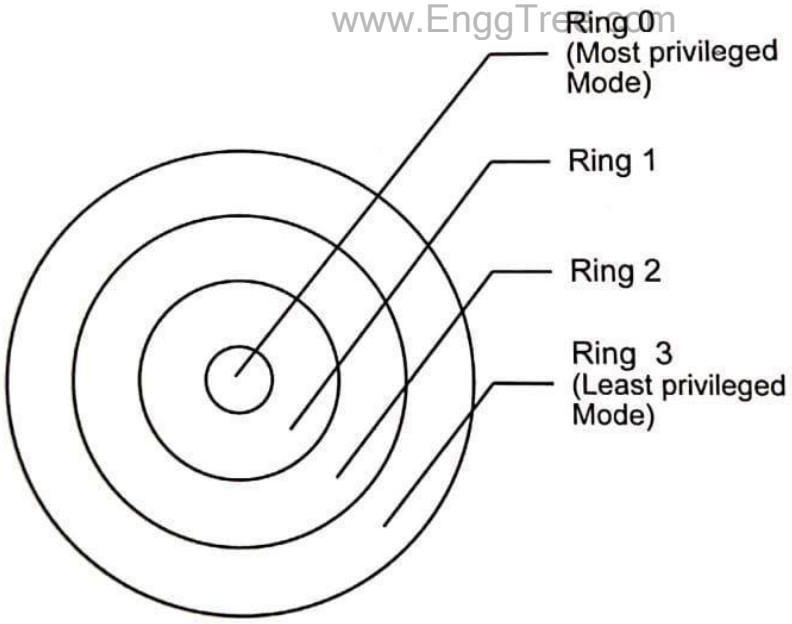


Figure 2.2 Security rings

All the current systems support at least two different execution

modes: supervisor mode and user mode.

o The supervisor mode denotes an execution mode in which all the instructions (privileged and non privileged) can be executed without any restriction.This mode, also called master mode or kernel mode, is generally used by the operating system (or the hypervisor) to perform sensitive operations on hardware level resources.

o In user mode, there are restrictions to control the machine

level resources The distinction between user and supervisor mode allows us to

understand the role of the hypervisor and why it is called that. • Conceptually, the hypervisor runs above the supervisor mode and from here the prefix "hyper" is used.

●◆ In reality, hypervisors are run in supervisor mode and the division between privileged and non privileged instructions has posed challenges in designing virtual machine managers.

5. **Write the Key Concepts of virtualization?BTL1**
(Concept Explanation:13 marks)

●◆ Hypervisor vs Increased security

o The ability to control the execution of a guest in a completely transparent manner opens new possibilities for delivering a

secure, controlled execution environment.

o The virtual machine represents an emulated environment

in which the guest is executed.

o This level of indirection allows the virtual machine manager to control and filter the activity of the guest, thus preventing some harmful operations from being performed.

• Managed execution Virtualization of the execution environment not only allows increased security, but a wider range of features also can be implemented.

In particular, sharing, aggregation, emulation, and isolation are the most relevant feature.
Sharing

o Virtualization allows the creation of a separate computing environment within the same host.

In this way it is possible to fully exploit the capabilities of a powerful guest, which would otherwise be underutilized.

• Aggregation o Not only is it possible to share physical resource among several guests but

virtualization also allows aggregation, which is the opposite process.

。 A group of separate hosts can be tied together and represented to guests as a single virtual host.

Emulation

o Guest programs are executed within an environment that is controlled by the virtualization layer, which ultimately is a program.

• This allows for controlling and tuning the environment that is exposed to guests.

Isolation

o Virtualization allows providing guests whether they are

operating systems, applications, or other entities with a completely separate environment, in which they are executed. • The guest program performs its activity by interacting with an abstraction layer, which provides access to the underlying resources.

o Benefits of Isolation
First it allows multiple guests to run on the same host

without interfering with each other. ■Second, it provides a separation between the host and

the guest.

• Another important capability Enabled by virtualization is performance tuning. ◆● This feature is a reality at present, given the considerable advances in hardware and software supporting virtualization.

◆● It becomes easier to control the performance of the guest by finely tuning the properties of the resources exposed through the virtual environment.

●◆ This capability provides a means to effectively implement a quality of service (QoS) infrastructure that more easily fulfills the service level agreement (SLA) established for the guest.
VM Portability

o The concept of portability applies in different ways according to the specific type of virtualization considered.

In the case of a hardware virtualization solution, the guest is packaged into a virtual image that, in most cases, can be safely moved and executed on top of different virtual machines

6. **Explain in detail about virtualization structures?BTL4**
(Concept explanation 10 marks,Diagram:3 marks)

Virtualization layer is responsible for converting portions of the real hardware into virtual machine

●◆ Therefore, different operating systems such as Linux and
Windows can run on the same physical machine, simultaneously.

• Depending on the position of the virtualization layer, there are several classes of VM architectures, namely the hypervisor architecture, paravirtualization and host based virtualization. The hypervisor is also known as the VMM (Virtual Machine

Monitor). They both perform the same virtualization operations.

Hypervisor and Xen architecture

●◆ The hypervisor supports hardware level virtualization on bare

metal devices like CPU, memory, disk and network interfaces.

●◆ The hypervisor software sits directly between the physical hardware

and its OS. This virtualization layer is referred to as either the VMM or the hypervisor.

The hypervisor provides hypercalls for the guest OSes and applications. Depending on the functionality, a hypervisor can assume microkernel architecture like the Microsoft Hyper-V.

●◆ It can assume monolithic hypervisor architecture like the

VMware ESX for server virtualization.

●◆ A micro kernel hypervisor includes only the basic and unchanging functions (such as physical memory management and processor scheduling).

●◆ The device drivers and other changeable components are outside the hypervisor.

●◆ The hypervisor supports hardware level virtualization on bare

metal devices like CPU, memory, disk and network interfaces.

●◆ The hypervisor software sits directly between the physical

hardware and its OS. This virtualization layer is referred to as either

the VMM

or the hypervisor.

The hypervisor provides hypercalls for the guest OSes and applications.

Depending on the functionality, a hypervisor can assume micro

kernel architecture like the Microsoft Hyper-V.

● It can assume monolithic hypervisor architecture like the

VMware ESX for server virtualization.

● A micro kernel hypervisor includes only the basic and unchanging functions (such as physical memory management and processor scheduling).

● The device drivers and other changeable components are outside the hypervisor.
A monolithic hypervisor implements all the aforementioned functions, including those of the device drivers. Therefore, the size of the hypervisor code of a micro-kernel hypervisor is smaller than that of a monolithic hypervisor.

Essentially, a hypervisor must be able to convert physical devices into

virtual resources dedicated for the deployed VM to use.

Xen architecture

• Xen is an open source hypervisor program developed by Cambridge

University. • Xen is a microkernel hypervisor, which separates the policy from the

mechanism.

• The Xen hypervisor implements all the mechanisms, leaving the policy to be handled by Domain 0. Figure 2.4 shows architecture of Xen hypervisor.

Xen does not include any device drivers natively. It just provides a mechanism by which a guest OS can have direct access to the physical devices.

● As a result, the size of the Xen hypervisor is kept rather small.

• Xen provides a virtual environment located between the hardware and the OS.

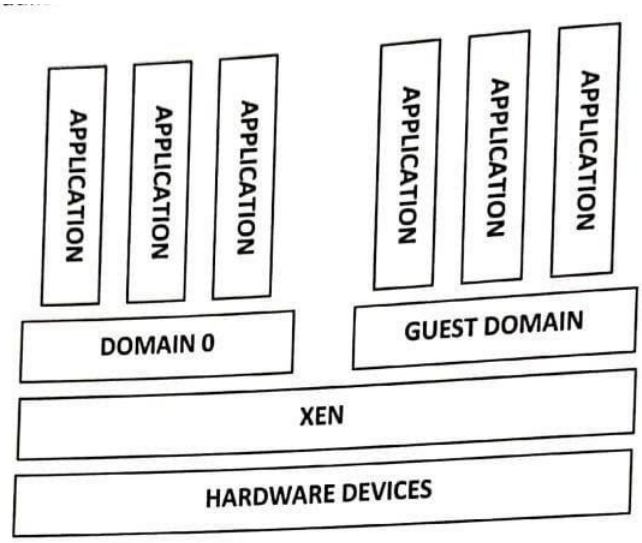


Figure 2.4 Xen domain 0 for control and I/O & guest domain for user applications.

he core components of a Xen system are the hypervisor, kernel, and

applications. ◆● The organization of the three components is

important.

◆● Like other virtualization systems, many guest OSes can run on top of

the hypervisor.

●◆ However, not all guest OSes are created equal, and one in

particular controls the others.

●◆ The guest OS, which has control ability, is called Domain 0, and

the others are called Domain U.

●◆ Domain 0 is a privileged guest OS of Xen. It is first loaded when Xen

boots without any file system drivers being available.Domain 0 is designed to access hardware directly and manage devices. Therefore, one of the responsibilities of Domain 0 is to allocate and map hardware resources for the guest domains (the Domain U domains).

●◆ For example, Xen is based on Linux and its security level is C2. Its management VM is named Domain 0 which has the privilege to manage other VMs implemented on the same host.

●◆ If Domain 0 is compromised, the hacker can control the entire system. So, in the VM system, security policies are needed to improve the security of Domain 0.

●◆ Domain 0, behaving as a VMM, allows users to create, copy, save, read, modify, share, migrate and roll back VMs as easily as manipulating a file, which flexibly provides tremendous benefits for users.

Binary translation with full virtualization

●◆ Depending on implementation technologies, hardware virtualization can be classified into two categories: full virtualization and host based virtualization.

●◆ Full virtualization does not need to modify the host OS. It relies on binary translation to trap and to virtualize the execution of certain

sensitive, non virtualizable instructions. The guest OSes and their applications consist of noncritical and critical instructions.

●◆ In a host-based system, both a host OS and a guest OS are used.
A virtualization software layer is built between the host OS and guest OS.

With full virtualization, noncritical instructions run on the hardware directly while critical instructions are discovered and replaced with

traps into the VMM to be emulated by software. ●◆ Both the hypervisor and VMM approaches are considered full

virtualization.

• The VMM scans the instruction stream and identifies the privileged, control and behavior sensitive instructions. When these instructions are identified, they are trapped into the VMM, which emulates the behavior of these instructions.

• The method used in this emulation is called binary translation. ◆● Full virtualization combines binary translation and direct execution.

●◆ An alternative VM architecture is to install a virtualization layer

on top of the host OS.

◆● This host OS is still responsible for managing the hardware.

●◆ The guest OSes are installed and run on top of the virtualization layer.

Dedicated applications may run on the VMs. Certainly, some other applications can also run with the host OS directly.

●◆ Host based architecture has some distinct advantages, as enumerated

next. o First, the user can install this VM architecture without modifying the host

OS. o Second, the host-based approach appeals to many host machine

configurations.

Paravirtualization with compiler support

●◆ When x86 processor is virtualized, a virtualization layer between the hardware and the OS.

◆● According to the x86 ring definitions, the virtualization layer should also be installed at Ring 0.
Different instructions at Ring 0 may cause

some problems.

Although paravirtualization reduces the overhead, it has incurred other problems.

First, its compatibility and portability may be in doubt, because it must support the unmodified OS as well.

Second, the cost of maintaining paravirtualized OSes is high, because they may require deep OS kernel modifications.

Finally, the performance advantage of paravirtualization varies greatly due to workload variations.

Compared with full virtualization, paravirtualization is relatively

easy and more practical. The main problem in full virtualization is its

low

performance in binary translation. ●◆ KVM is a Linux paravirtualization system. It is a part of the Linux

version 2.6.20 kernel. ◆● In KVM, Memory management and scheduling activities are

carried out by the existing Linux kernel.

●◆ The KVM does the rest, which makes it simpler than the hypervisor that controls the entire machine.
KVM is a hardware assisted and paravirtualization tool, which improves performance and supports unmodified guest OSes such as Windows, Linux, Solaris, and other UNIX variants.

Unlike the full virtualization architecture which intercepts and emulates privileged and sensitive instructions at runtime, paravirtualization handles these instructions at compile time.

The guest OS kernel is modified to replace the privileged and sensitive instructions with hypercalls to the hypervisor or VMM. Xen assumes such paravirtualization architecture

●◆ The guest OS running in a guest domain may run at Ring 1 instead of at Ring 0. This implies that the guest OS may not be able to execute some privileged and sensitive instructions. The privileged instructions are implemented by hypercalls to the hypervisor.

7. **What are the Types of Virtualization?BTL 1**
(Definition:3 marks,Concept explanation:10 marks)

• Hardware virtualization provides an abstract execution environment by Hardware assisted virtualization, Full virtualization, Paravirtualization and Partial virtualization techniques.

1 Full virtualization

●◆ Full virtualization refers to the ability to run a program, most likely an operating system, directly on top of a virtual machine and without any modification, as though it were run on the raw hardware. To make this possible, virtual machine managers are required to

provide a complete emulation of the entire underlying hardware. ◆● The principal advantage of full virtualization is complete isolation, which leads to enhanced security, ease of emulation of different architectures and coexistence of different systems on the same platform.

●◆ Whereas it is a desired goal for many virtualization solutions, full virtualization poses important concerns related to performance and technical implementation.

●◆ A key challenge is the interception of privileged instructions such as

I/O instructions: Since they change the state of the resources exposed
by the host, they have to be contained within the virtual machine manager.

●◆ A simple solution to achieve full virtualization is to provide a virtual environment for all the instructions, thus posing some limits on performance.

• A successful and efficient implementation of full virtualization is obtained with a combination of hardware and software, not allowing potentially harmful instructions to be executed directly on the host.

2.Paravirtualization

• Paravirtualization is a not transparent virtualization solution

that allows implementing thin virtual machine managers.

●◆ Paravirtualization techniques expose a software interface to the virtual machine that is slightly

modified from the host and, as a consequence, guests need to be modified.

●◆ The aim of paravirtualization is to provide the capability to

demand the execution of performance critical operations directly on

the host,

thus preventing performance losses that would otherwise be experienced in managed execution.
This allows a simpler implementation of virtual machine managers

that have to simply transfer the execution of these operations,

which

were hard to virtualize, directly to the host.
To take advantage of such an opportunity, guest operating systems need to be modified and explicitly ported by remapping the performance critical operations through the virtual machine software interface.
This is possible when the source code of the operating system is available, and this is the reason that paravirtualization was mostly explored in the opensource and academic environment.

This technique has been successfully used by Xen for providing virtualization solutions for Linux-based operating systems specifically ported to run on Xen hypervisors.

• Operating systems that cannot be ported can still take advantage of para virtualization by using ad hoc device drivers that remap the execution of critical instructions to the paravirtualization APIs exposed by the hypervisor. Xen provides this solution for running Windows based operating systems on x86 architectures.

●◆ Other solutions using paravirtualization include VMWare, Parallels, and some solutions for embedded and real-time environments such as TRANGO, Wind River, and XtratuM.

3.Hardware assisted virtualization

◆● Hardware assisted virtualization refers to a scenario in which the hardware provides architectural support for building a virtual machine manager able to run a guest operating system in complete isolation.

●◆ This technique was originally introduced in the IBM System/370. . At present, examples of hardware assisted virtualization are the extensions to the x86 architecture introduced with Intel-VT (formerly known as Vanderpool) and AMD-V (formerly known as Pacifica).These extensions, which differ between the two vendors, are meant to reduce the performance penalties experienced by emulating x86
hardware with hypervisors.

●◆ Before the introduction of hardware assisted virtualization, software emulation of x86 hardware was significantly costly from the performance point of view.

● ◆ The reason for this is that by design the x86 architecture did not meet the formal requirements introduced by Popek and Goldberg and early products were using binary translation to trap some

sensitive

instructions and provide an emulated version. • Products such as VMware Virtual Platform, introduced in 1999 by

VMware, which pioneered the field of x86 virtualization, were based on this technique.
. After 2006, Intel and AMD introduced processor extensions and a wide range of virtualization solutions took advantage of them: Kernel-based Virtual Machine (KVM), VirtualBox, Xen, VMware,

Hyper-V, Sun XVM, Parallels, and others.

4. Partial virtualization

●◆ Partial virtualization provides a partial emulation of the underlying hardware, thus not allowing the complete execution of the guest operating system in complete isolation.

●◆ Partial virtualization allows many applications to run transparently, but not all the features of the operating system can be supported as happens with full virtualization.An example of partial virtualization is address space virtualization used in time sharing systems; this allows multiple applications and users to run concurrently in a separate memory space, but they still share the same hardware resources (disk, processor, and network).

●◆ Historically, partial virtualization has been an important milestone for achieving full virtualization, and it was implemented on the experimental IBM M44/44X.

• Address space virtualization is a common feature of contemporary operating systems.

PART C
15 marks

1. What are the implementation levels of virtualization?BTL1
(Definition:2 marks,Diagram:5 marks,Concept Explanation:8 marks)

Virtualization is a computer architecture technology by which multiple virtual machines (VMs) are multiplexed in the same hardware machine. The idea of VMs can be dated back to the 1960s [53]. The purpose of a VM is to enhance resource sharing by many users and improve computer performance in terms of resource utilization and application flexibility. Hardware resources (CPU, memory, I/O devices, etc.) or software resources (operating system and software libraries) can be virtualized in various functional layers. This virtualization technology has been revitalized as the demand for distributed and cloud computing increased sharply in recent years.The idea is to separate the hardware from the software to yield better system efficiency. For example, computer users gained access to much enlarged memory space when the concept of virtual memory was introduced. Similarly, virtualization techniques can be applied to enhance the use of compute engines, networks, and storage. In this chapter we will discuss VMs and their

applications for building distributed systems. According to a 2009 Gartner Report, virtualization was the top strategic technology poised to change the computer industry. With sufficient storage, any computer platform can be installed in another host computer, even if they use processors with different instruction sets and run with distinct operating systems on the same hardware.

1. Levels of Virtualization Implementation

A traditional computer runs with a host operating system specially tailored for its hardware architecture, as shown in Figure 3.1(a). After virtualization, different user applications managed by their own operating systems (guest OS) can run on the same hardware, independent of the host OS. This is often done by adding additional software, called a virtualization layer as shown in Figure 3.1(b). This virtualization layer is known as hypervisor or virtual machine monitor (VMM) [54]. The VMs are shown in the upper boxes, where applications run with their own guest OS over the virtualized CPU, memory, and I/O resources.

The main function of the software layer for virtualization is to virtualize the physical hardware of a host machine into virtual resources to be used by the VMs, exclusively. This can be implemented at various operational levels, as we will discuss shortly. The virtualization software creates the abstraction of VMs by interposing a virtualization layer at various levels of a computer system. Common virtualization layers include the instruction set architecture (ISA) level, hardware level, operating system level, library support level, and application level (see Figure 3.2).

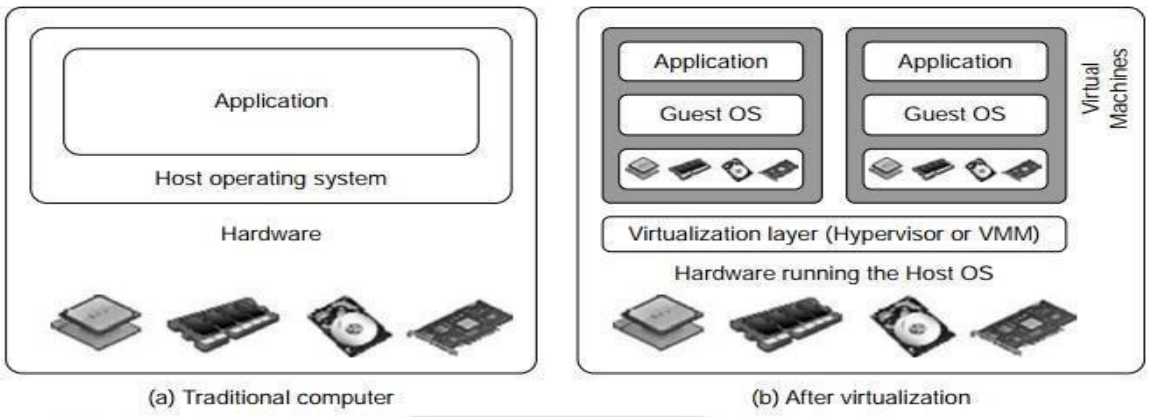


FIGURE 3.1

The architecture of a computer system before and after virtualization, where VMM stands for virtual machine monitor.

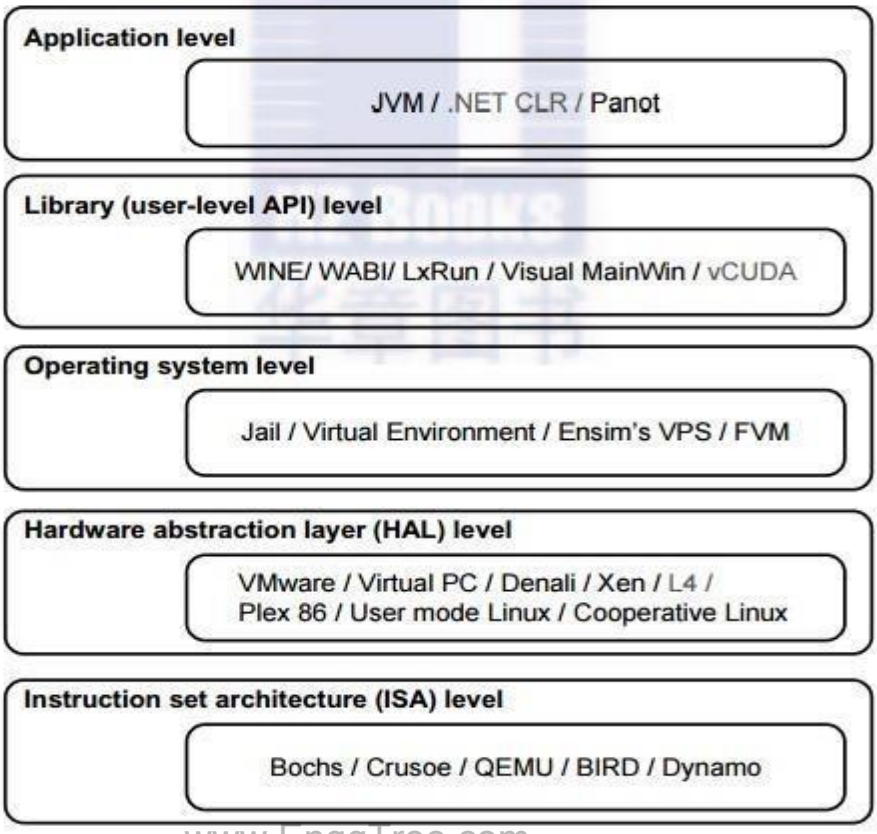


FIGURE 3.2

Virtualization ranging from hardware to applications in five abstraction levels.

1.1 Instruction Set Architecture Level

At the ISA level, virtualization is performed by emulating a given ISA by the ISA of the host machine. For example, MIPS binary code can run on an x86-based host machine with the help of ISA emulation. With this approach, it is possible to run a large amount of legacy binary code writ- ten for various processors on any given new hardware host machine. Instruction set emulation leads to virtual ISAs created on any hardware machine.

The basic emulation method is through code interpretation. An interpreter program interprets the source instructions to target instructions one by one. One source instruction may require tens or hundreds of native target instructions to perform its function. Obviously, this process is relatively slow. For better performance, dynamic binary translation is desired. This approach translates basic blocks of dynamic source instructions to target instructions. The basic blocks can also be extended to program traces or super blocks to increase translation efficiency. Instruction set emulation requires binary translation and optimization. A virtual instruction set architecture (V-ISA) thus requires adding a processor-specific software translation layer to the compiler.

1.2 Hardware Abstraction Level

Hardware-level virtualization is performed right on top of the bare hardware. On the one hand, this approach generates a virtual hardware environment for a VM. On the other hand, the process manages the underlying hardware through virtualization. The idea is to virtualize a computer's resources, such as its processors, memory, and I/O devices. The intention is to upgrade the hardware utilization rate by multiple users concurrently. The idea was implemented in the IBM VM/370 in the 1960s. More recently, the Xen hypervisor has been applied to virtualize x86-based machines to run Linux or other guest OS applications. We will discuss hardware virtualization approaches in more detail in Section 3.3.

1.3 Operating System Level

This refers to an abstraction layer between traditional OS and user applications. OS-level virtualiza-tion creates isolated containers on a single physical server and the OS instances to utilize the hard-ware and software in data centers. The containers behave like real servers. OS-level virtualization is commonly used in creating virtual hosting environments to allocate hardware resources among a large number of mutually distrusting users. It is also used, to a lesser extent, in consolidating server hardware by moving services on separate hosts into containers or VMs on one server. OS-level virtualization is depicted in Section 3.1.3.

1.4 Library Support Level

Most applications use APIs exported by user-level libraries rather than using lengthy system calls by the OS. Since most systems provide well-documented APIs, such an interface becomes another candidate for virtualization. Virtualization with library interfaces is possible by controlling the communication link between applications and the rest of a system through API hooks. The software tool WINE has implemented this approach to support Windows applications on top of UNIX hosts. Another example is the vCUDA which allows applications executing within VMs to leverage GPU hardware acceleration. This approach is detailed in Section 3.1.4.

1.5 User-Application Level

Virtualization at the application level virtualizes an application as a VM. On a traditional OS, an application often runs as a process. Therefore, application-level virtualization is also known as process-level virtualization. The most popular approach is to deploy high level language (HLL)

VMs. In this scenario, the virtualization layer sits as an application program on top of the operating system, and the layer exports an abstraction of a VM that can run programs written and compiled to a particular abstract machine definition. Any program written in the HLL and compiled for this VM will be able to run on it. The Microsoft .NET CLR and Java Virtual Machine (JVM) are two good examples of this class of VM.

Other forms of application-level virtualization are known as application isolation, application sandboxing, or application streaming. The process involves wrapping the application in a layer that is isolated from the host OS and other applications. The result is an application that is much easier to distribute and remove from user workstations. An example is the LANDesk application virtuali-zation platform which deploys software applications as self-contained, executable files in an isolated environment without requiring installation, system modifications, or elevated security privileges.

2. Explain in detail about virtualization of cpu,memory and I/O devices?BTL4

(Definition:2 marks,Diagram:5 marks,Concept Explanation:8 marks)

To support virtualization, processors such as the x86 employ a special running mode and instructions, known as hardware-assisted virtualization. In this way, the VMM and guest OS run in different modes and all sensitive instructions of the guest OS and its applications are trapped in the VMM. To save processor states, mode switching is completed by hardware. For the x86 architecture, Intel and AMD have proprietary technologies for hardware-assisted virtualization.

1. **Hardware Support for Virtualization**

Modern operating systems and processors permit multiple processes to run simultaneously. If there is no protection mechanism in a processor, all instructions from different processes will access the hardware directly and cause a system crash. Therefore, all processors have at least two modes, user mode and supervisor mode, to ensure controlled access of critical hardware. Instructions running in supervisor mode are called privileged instructions. Other instructions are unprivileged instructions. In a virtualized environment, it is more difficult to make OSes and

applications run correctly because there are more layers in the machine stack. Example 3.4 discusses Intel's hardware support approach.

At the time of this writing, many hardware virtualization products were available. The VMware Workstation is a VM software suite for x86 and x86-64 computers. This software suite allows users to set up multiple x86 and x86-64 virtual computers and to use one or more of these VMs simultaneously with the host operating system. The VMware Workstation assumes the host-based virtualization. Xen is a hypervisor for use in IA-32, x86-64, Itanium, and PowerPC 970 hosts. Actually, Xen modifies Linux as the lowest and most privileged layer, or a hypervisor.

One or more guest OS can run on top of the hypervisor. KVM (Kernel-based Virtual Machine) is a Linux kernel virtualization infrastructure. KVM can support hardware-assisted virtualization and paravirtualization by using the Intel VT-x or AMD-v and VirtIO framework, respectively. The VirtIO framework includes a paravirtual Ethernet card, a disk I/O controller, a balloon device for adjusting guest memory usage, and a VGA graphics interface using VMware drivers.

Example 3.4 Hardware Support for Virtualization in the Intel x86 Processor

Since software-based virtualization techniques are complicated and incur performance overhead, Intel provides a hardware-assist technique to make virtualization easy and improve performance. Figure 3.10 provides an overview of Intel's full virtualization techniques. For processor virtualization, Intel offers the VT-x or VT-i technique. VT-x adds a privileged mode (VMX Root Mode) and some instructions to processors. This enhancement traps all sensitive instructions in the VMM automatically. For memory virtualization, Intel offers the EPT, which translates the virtual address to the machine's physical addresses to improve performance. For I/O virtualization, Intel implements VT-d and VT-c to support this.

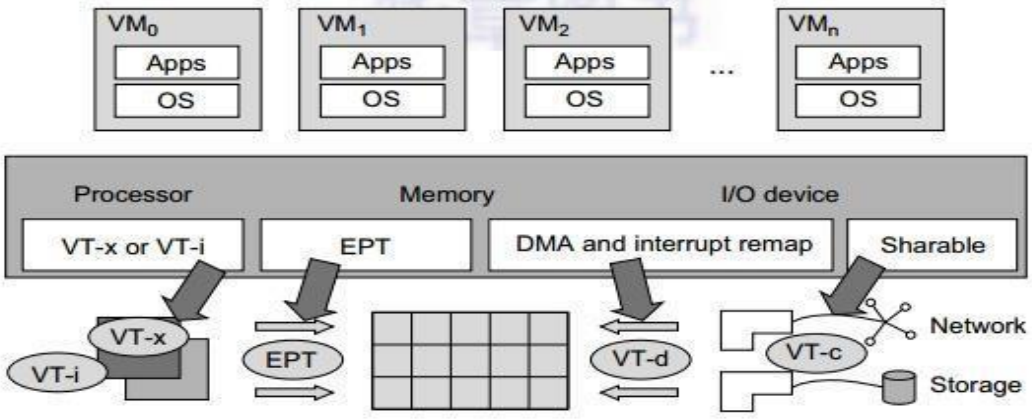


FIGURE 3.10

Intel hardware support for virtualization of processor, memory, and I/O devices.

2. CPU Virtualization

A VM is a duplicate of an existing computer system in which a majority of the VM instructions are executed on the host processor in native mode. Thus, unprivileged instructions of VMs run directly on the host machine for higher efficiency. Other critical instructions should be handled carefully for correctness and stability. The critical instructions are divided into three categories: privileged instructions, control-sensitive instructions, and behavior-sensitive instructions. Privileged instructions execute in a privileged mode and will be trapped if executed outside this mode. Control-sensitive instructions attempt to change the configuration of resources used. Behavior-sensitive instructions have different behaviors depending on the configuration of resources, including the load and store operations over the virtual memory.

A CPU architecture is virtualizable if it supports the ability to run the VM's privileged and unprivileged instructions in the CPU's user mode while the VMM runs in supervisor mode. When the privileged instructions including control- and behavior-sensitive instructions of a VM are exe- cuted, they are trapped in the VMM. In this case, the VMM acts as a unified mediator for hardware access from different VMs to guarantee the correctness and stability of the whole system. However, not all CPU architectures are virtualizable. RISC CPU architectures can be naturally virtualized because all control- and behavior-sensitive instructions are privileged instructions. On the contrary, x86 CPU architectures are not primarily designed to support virtualization. This is because about 10 sensitive instructions, such as SGDT and SMSW, are not privileged instructions. When these instruc-tions execute in virtualization, they cannot be trapped in the VMM.

On a native UNIX-like system, a system call triggers the 80h interrupt and passes control to the OS kernel. The interrupt handler in the kernel is then invoked to process the system call. On a para-virtualization system such as Xen, a system call in the guest OS first triggers the 80h interrupt nor-mally. Almost at the same time, the 82h interrupt in the hypervisor is triggered. Incidentally, control is passed on to the hypervisor as well. When the hypervisor completes its task for the guest OS system call, it passes control back to the guest OS kernel. Certainly, the guest OS kernel may also invoke the hypercall while it's running. Although paravirtualization of a CPU lets unmodified applications run in the VM, it causes a small performance penalty.

2.1 Hardware-Assisted CPU Virtualization

This technique attempts to simplify virtualization because full or paravirtualization is complicated. Intel and AMD add an additional mode called privilege mode level (some people call it Ring-1) to x86 processors. Therefore, operating systems can still run at Ring 0 and the hypervisor can run at Ring -1. All the privileged and sensitive instructions are trapped in the hypervisor automatically. This technique removes the difficulty of implementing binary translation of full virtualization. It also lets the operating system run in VMs without modification.

Example 3.5 Intel Hardware-Assisted CPU Virtualization

Although x86 processors are not virtualizable primarily, great effort is taken to virtualize them. They are used widely in comparing RISC processors that the bulk of x86-based legacy systems cannot discard easily. Virtuali-zation of x86 processors is detailed in the following sections. Intel's VT-x technology is an example of hardware-assisted virtualization, as shown in Figure 3.11. Intel calls the privilege level of x86 processors the VMX Root Mode. In order to control the start and stop of a VM and allocate a memory page to maintain the

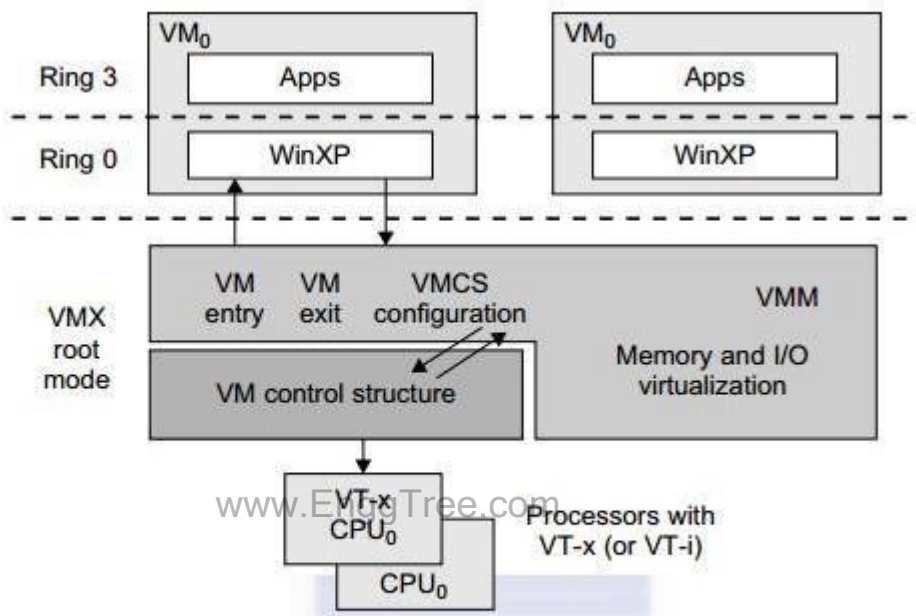


FIGURE 3.11
Intel hardware-assisted CPU virtualization.

CPU state for VMs, a set of additional instructions is added. At the time of this writing, Xen, VMware, and the Microsoft Virtual PC all implement their hypervisors by using the VT-x technology.

Generally, hardware-assisted virtualization should have high efficiency. However, since the transition from the hypervisor to the guest OS incurs high overhead switches between processor modes, it sometimes cannot outperform binary translation. Hence, virtualization systems such as VMware now use a hybrid approach, in which a few tasks are offloaded to the hardware but the rest is still done in software. In addition, para-virtualization and hardware-assisted virtualization can be combined to improve the performance further.

3. **Memory Virtualization**

Virtual memory virtualization is similar to the virtual memory support provided by modern operat- ing systems. In a traditional execution environment, the operating system maintains mappings of virtual memory to machine memory using page tables, which is a one-stage mapping from virtual memory to machine memory. All modern x86 CPUs include a memory management unit (MMU) and a translation lookaside buffer (TLB) to optimize virtual memory performance. However, in a virtual execution environment, virtual memory virtualization involves sharing the physical system memory in RAM and dynamically allocating it to the physical memory of the VMs.

That means a two-stage mapping process should be maintained by the guest OS and the VMM, respectively: virtual memory to physical memory and physical memory to machine memory. Furthermore, MMU virtualization should be supported, which is transparent to the guest OS. The guest OS continues to control the mapping of virtual addresses to the physical memory addresses of VMs. But the guest OS cannot directly access the actual machine memory. The VMM is responsible for mapping the guest physical memory to the actual machine memory. Figure 3.12 shows the two-level memory mapping procedure.

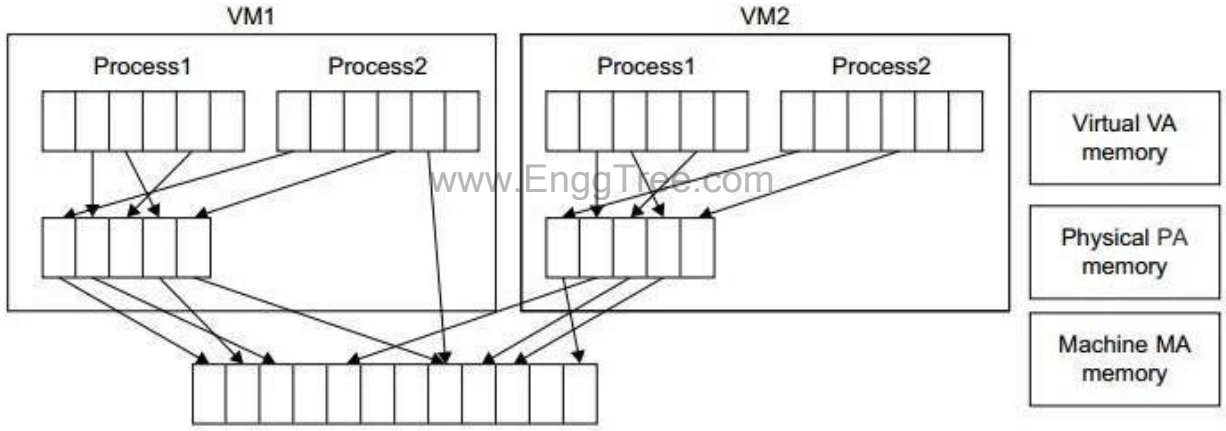


FIGURE 3.12

Two-level memory mapping procedure.

Since each page table of the guest OSes has a separate page table in the VMM corresponding to it, the VMM page table is called the shadow page table. Nested page tables add another layer of indirection to virtual memory. The MMU already handles virtual-to-physical translations as defined by the OS. Then the physical memory addresses are translated to machine addresses using another set of page tables defined by the hypervisor. Since modern operating systems maintain a set of page tables for every process, the shadow page tables will get flooded. Consequently, the perfor-mance overhead and cost of memory will be very high.

VMware uses shadow page tables to perform virtual-memory-to-machine-memory address translation. Processors use TLB hardware to map the virtual memory directly to the machine

memory to avoid the two levels of translation on every access. When the guest OS changes the virtual memory to a physical memory mapping, the VMM updates the shadow page tables to enable a direct lookup. The AMD Barcelona processor has featured hardware-assisted memory virtualization since 2007. It provides hardware assistance to the two-stage address translation in a virtual execution environment by using a technology called nested paging.

Example 3.6 Extended Page Table by Intel for Memory Virtualization

Since the efficiency of the software shadow page table technique was too low, Intel developed a hardware-based EPT technique to improve it, as illustrated in Figure 3.13. In addition, Intel offers a Virtual Processor ID (VPID) to improve use of the TLB. Therefore, the performance of memory virtualization is greatly improved. In Figure 3.13, the page tables of the guest OS and EPT are all four-level.

When a virtual address needs to be translated, the CPU will first look for the L4 page table pointed to by Guest CR3. Since the address in Guest CR3 is a physical address in the guest OS, the CPU needs to convert the Guest CR3 GPA to the host physical address (HPA) using EPT. In this procedure, the CPU will check the EPT TLB to see if the translation is there. If there is no required translation in the EPT TLB, the CPU will look for it in the EPT. If the CPU cannot find the translation in the EPT, an EPT violation exception will be raised.

When the GPA of the L4 page table is obtained, the CPU will calculate the GPA of the L3 page table by using the GVA and the content of the L4 page table. If the entry corresponding to the GVA in the L4

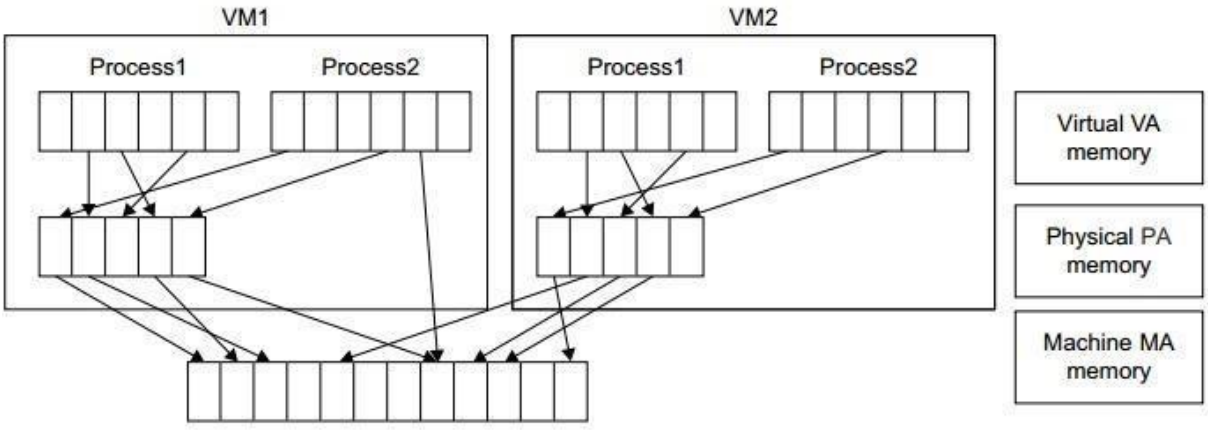


FIGURE 3.12

Two-level memory mapping procedure.

page table is a page fault, the CPU will generate a page fault interrupt and will let the guest OS kernel handle the interrupt. When the PGA of the L3 page table is obtained, the CPU will look for the EPT to get the HPA of the L3 page table, as described earlier. To get the HPA corresponding

to a GVA, the CPU needs to look for the EPT five times, and each time, the memory needs to be accessed four times. There-fore, there are 20 memory accesses in the worst case, which is still very slow. To overcome this short-coming, Intel increased the size of the EPT TLB to decrease the number of memory accesses.

4. I/O Virtualization

I/O virtualization involves managing the routing of I/O requests between virtual devices and the shared physical hardware. At the time of this writing, there are three ways to implement I/O virtualization: full device emulation, para-virtualization, and direct I/O. Full device emulation is the first approach for I/O virtualization. Generally, this approach emulates well-known, real-world devices.

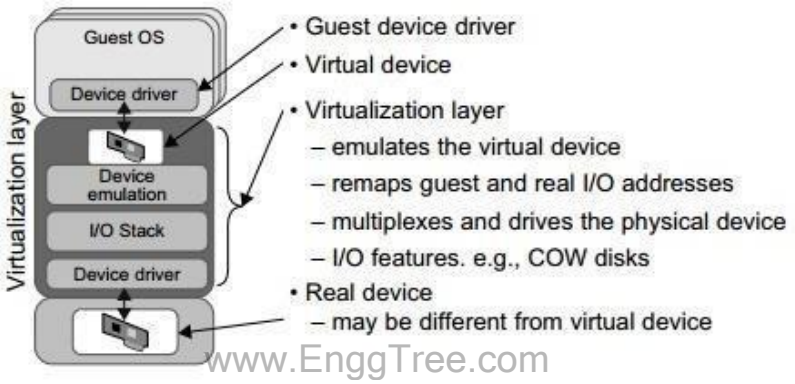


FIGURE 3.14

Device emulation for I/O virtualization implemented inside the middle layer that maps real I/O devices into the virtual devices for the guest device driver to use.

All the functions of a device or bus infrastructure, such as device enumeration, identification, interrupts, and DMA, are replicated in software. This software is located in the VMM and acts as a virtual device. The I/O access requests of the guest OS are trapped in the VMM which interacts with the I/O devices. The full device emulation approach is shown in Figure 3.14.

A single hardware device can be shared by multiple VMs that run concurrently. However, software emulation runs much slower than the hardware it emulates [10,15]. The para-virtualization method of I/O virtualization is typically used in Xen. It is also known as the split driver model consisting of a frontend driver and a backend driver. The frontend driver is running in Domain U and the backend dri-ver is running in Domain 0. They interact with each other via a block of shared memory. The frontend driver manages the I/O requests of the guest OSes and the backend driver is responsible for managing the real I/O devices and multiplexing the I/O data of different VMs. Although para-I/O-virtualization achieves better device performance than full device emulation, it comes with a higher CPU overhead.

Direct I/O virtualization lets the VM access devices directly. It can achieve close-to-native performance without high CPU costs. However, current direct I/O virtualization implementations focus on networking for mainframes. There are a lot of challenges for commodity hardware devices. For example, when a physical device is reclaimed (required by workload migration) for later reassign-ment, it may have been set to an arbitrary state (e.g., DMA to some arbitrary memory locations) that can function incorrectly or even crash the whole system. Since software-based I/O virtualization requires a very high overhead of device emulation, hardware-assisted I/O virtualization is critical. Intel VT-d supports the remapping of I/O DMA transfers and device-generated interrupts. The architecture of VT-d provides the flexibility to support multiple usage models that may run unmodified, special-purpose, or "virtualization-aware" guest OSes.

Another way to help I/O virtualization is via self-virtualized I/O (SV-IO) [47]. The key idea of SV-IO is to harness the rich resources of a multicore processor. All tasks associated with virtualizing an I/O device are encapsulated in SV-IO. It provides virtual devices and an associated access API to VMs and a management API to the VMM. SV-IO defines one virtual interface (VIF) for every kind of virtua-lized I/O device, such as virtual network interfaces, virtual block devices (disk), virtual camera devices, and others. The guest OS interacts with the VIFs via VIF device drivers. Each VIF consists of two mes-sage queues. One is for outgoing messages to the devices and the other is for incoming messages from the devices. In addition, each VIF has a unique ID for identifying it in SV-IO.

3. Explain in detail about Hypervisor and Xen architecture?BTL 4
(Definition:2 marks,Diagram:3 marks,Concept explanation:10 marks)

●◆ The hypervisor supports hardware level virtualization on bare

metal devices like CPU, memory, disk and network interfaces.

●◆ The hypervisor software sits directly between the physical hardware

and its OS. This virtualization layer is referred to as either the VMM or the hypervisor.

The hypervisor provides hypercalls for the guest OSes and applications. Depending on the functionality, a hypervisor can assume microkernel architecture like the Microsoft Hyper-V.

●◆ It can assume monolithic hypervisor architecture like the

VMware ESX for server virtualization.

●◆ A micro kernel hypervisor includes only the basic and unchanging functions (such as physical memory management and processor scheduling).

●◆ The device drivers and other changeable components are outside the hypervisor.

●◆ The hypervisor supports hardware level virtualization on bare

metal devices like CPU, memory, disk and network interfaces.

●◆ The hypervisor software sits directly between the physical

hardware and its OS. This virtualization layer is referred to as either

the VMM

or the hypervisor.

The hypervisor provides hypercalls for the guest OSes and applications.

Depending on the functionality, a hypervisor can assume micro

kernel architecture like the Microsoft Hyper-V.

●◆ It can assume monolithic hypervisor architecture like the

VMware ESX for server virtualization.

●◆ A micro kernel hypervisor includes only the basic and unchanging functions (such as physical memory management and processor scheduling).

●◆ The device drivers and other changeable components are outside the hypervisor.
A monolithic hypervisor implements all the aforementioned functions, including those of the device drivers. Therefore, the size of the hypervisor code of a micro-kernel hypervisor is smaller than that of a monolithic hypervisor.

Essentially, a hypervisor must be able to convert physical devices into

virtual resources dedicated for the deployed VM to use.

Xen architecture

• Xen is an open source hypervisor program developed by Cambridge

University. • Xen is a microkernel hypervisor, which separates the policy from the

mechanism.

• The Xen hypervisor implements all the mechanisms, leaving the policy to be handled by Domain 1. Figure 2.4 shows architecture of Xen hypervisor.

Xen does not include any device drivers natively. It just provides a mechanism by which a guest OS can have direct access to the physical devices.

●◆ As a result, the size of the Xen hypervisor is kept rather small.

• Xen provides a virtual environment located between the hardware and the OS.

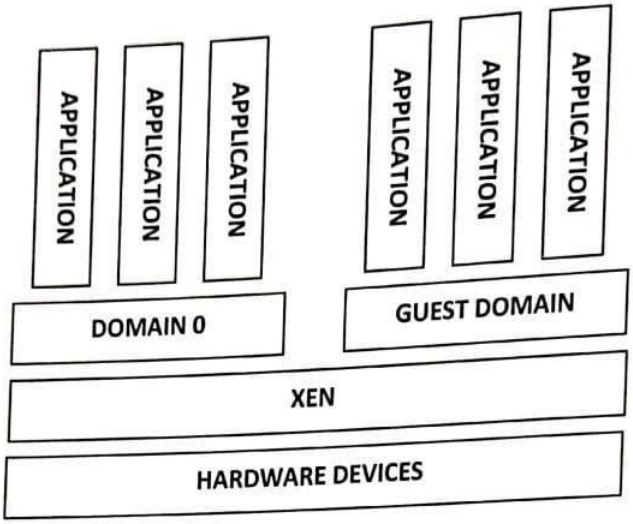


Figure 2.4 Xen domain 0 for control and I/O & guest domain for user applications.

he core components of a Xen system are the hypervisor, kernel, and

applications. ◆● The organization of the three components is

important.

●◆ Like other virtualization systems, many guest OSes can run on top of

the hypervisor.

●◆ However, not all guest OSes are created equal, and one in

particular controls the others.

●◆ The guest OS, which has control ability, is called Domain 0, and

the others are called Domain U.

●◆ Domain 0 is a privileged guest OS of Xen. It is first loaded when Xen

boots without any file system drivers being available.Domain 0 is designed to access hardware directly and manage devices. Therefore, one of the responsibilities of Domain 0 is to allocate and map hardware resources for the guest domains (the Domain U domains).

- For example, Xen is based on Linux and its security level is C2. Its management VM is named Domain 0 which has the privilege to manage other VMs implemented on the same host.

- If Domain 0 is compromised, the hacker can control the entire system. So, in the VM system, security policies are needed to improve the security of Domain 0.

- Domain 0, behaving as a VMM, allows users to create, copy, save, read, modify, share, migrate and roll back VMs as easily as manipulating a file, which flexibly provides tremendous benefits for users.

UNIT III
VIRTUALIZATION INFRASTRUCTURE AND DOCKER

SYLLABUS:Desktop Virtualization – Network Virtualization – Storage Virtualization – System-level of Operating Virtualization – Application Virtualization – Virtual clusters and Resource Management – Containers vs. Virtual Machines – Introduction to Docker – Docker Components – Docker Container – Docker Images and Repositories.

PART A
2 Marks

2. How to implement internal network virtualization?BTL1

The guest can share the same network interface of the host and use Network Address Translation (NAT) to access the network;The virtual machine manager can emulate, and install on the host, an additional network device, together with the driver.The guest can have a private network only with the guest.

3. What is Hardware-level virtualization?BTL1

Hardware-level virtualization is a virtualization technique that provides an abstract execution environment in terms of computer hardware on top of which a guest operating system can be run.

4. Define hypervisor?BTL1

The hypervisor is generally a program or a combination of software and hardware that allows the abstraction of the underlying physical hardware.
Hypervisors is a fundamental element of hardware virtualization is the hypervisor, or virtual machine manager (VMM).

5. Mention the advantages of SAN?BTL1

There are different techniques for storage virtualization, one of the most popular being network based virtualization by means of storage area networks (SANs).SANS use a network accessible device through a large bandwidth connection to provide storage facilities.

6. What is Operating system-level virtualization?BTL1

• Operating system-level virtualization offers the opportunity to create different and separated execution environments for applications that are managed concurrently.

• Differently from hardware virtualization, there is no virtual machine manager or hypervisor, and the virtualization is done within a single operating system, where the OS kernel allows for multiple isolated user space instances.

7. What is storage virtualization?BTL1

• Storage virtualization is a system administration practice that allows decoupling the physical organization of the hardware from its logical representation.Using this

technique, users do not have to be worried about the specific location of their data, which can be identified using a logical path.

8. Define Desktop virtualization?BTL1

●◆ Desktop virtualization abstracts the desktop environment available on a personal computer in order to provide access to it using a client/server approach.

• Desktop virtualization provides the same outcome of hardware virtualization but serves a different purpose.

9. What is Network Migration?BTL1

A migrating VM should maintain all open network connections without relying on forwarding mechanisms on the original host or on support from mobility or redirection mechanisms.
To enable remote systems to locate and communicate with a VM, each VM must be assigned a virtual IP address known to other entities.

10. Differentiate between physical and virtual clusterBTL2

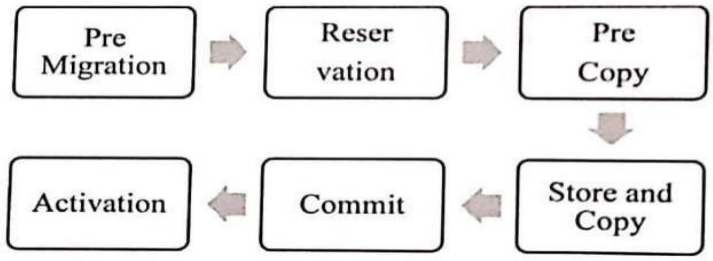
A physical cluster is a collection of physical servers / machines interconnected by a physical network such as a LAN. On the other hand, A virtual cluster is a collection of virtual servers / machines interconnected by a virtual network

11. List the issues in migration process?BTL1
Memory Migration

File System Migration

Network Migration

11. List six steps in live migration.



12. How to manage a virtual cluster?BTL1
cluster manager resides on a guest system Cluster manager resides on the host systems. The host-based manager supervises the guest systems and can restart the guest system on another physical machine.
Use an independent cluster manager on both the host and guest systems Use an integrated cluster on the guest and host systems. This means the manager must be designed to distinguish between virtualized resources and physical resources.

13. Differentiate between Containers and virtual machines?BTL2

Containers and virtual machines are two types of virtualization technologies that share many similarities.

Virtualization is a process that allows a single resource, such as RAM, CPU, Disk, or Networking, to be virtualized and represented as multiple resources.

However, the main difference between containers and virtual machines is that virtual machines virtualize the entire machine, including the hardware layer, while containers only virtualize software layers above the operating system level.

14. List the different types of Docker networks?BTL1

Bridge: This is the default network driver and is suitable for different containers that need to communicate with the same Docker host.

Host: This network is used when there is no need for isolation between the container and the host.

Overlay: This network allows swarm services to communicate with each other.
None: This network disables all networking.

Macvlan: This assigns a Media Access Control (MAC) address to containers, which looks like a physical address.

15. What is the purpose of Docker Hub?BTL1

The Docker Hub is a cloud-based repository service where users can push their Docker Container Images and access them from anywhere via the internet.It offers the option to push images as private or public and is primarily used by DevOps teams.

The Docker Hub is an open-source tool that is available for all operating systems. It functions as a storage system for Docker images and allows users to pull the required images when needed.

PART B
13 Marks

1. Write a short notes on Desktop virtualization?BTL1
(Definition:2 marks,Concept explanation:11 marks)

Desktop virtualization abstracts the desktop environment available on a personal computer in order to provide access to it using a client/server approach.
Desktop virtualization provides the same outcome of hardware virtualization but serves a different purpose.Similarly to hardware virtualization, desktop virtualization makes accessible a different system as though it were natively installed on the host but this system is remotely stored on a different host and accessed through a network connection.
Moreover, desktop virtualization addresses the problem of making the same desktop

environment accessible from everywhere.

Although the term desktop virtualization strictly refers to the ability- environment, generally the desktop

Although the term desktop virtualization strictly refers to the ability to remotely access a desktop environment, generally the desktop environment is stored in a remote server or a data center that provides a high availability infrastructure and ensures the accessibility and persistence of the data.

In this scenario, an infrastructure supporting hardware virtualization is fundamental to provide access to multiple desktop environments hosted on the same server.A specific desktop environment is stored in a virtual machine image that is loaded and started on demand when a client connects to the desktop environment.This is a typical cloud computing scenario in which the user leverages the virtual infrastructure for performing the daily tasks on his computer. The advantages of desktop virtualization are high availability, persistence, accessibility, and ease of management.

The basic services for remotely accessing a desktop environment are implemented in software components such as Windows Remote Services, VNC, and X Server.

Infrastructures for desktop virtualization based on cloud computing solutions include Sun Virtual Desktop Infrastructure (VDI), Parallels Virtual Desktop Infrastructure (VDI), Citrix XenDesktop, and others.

2. Explain in detail about Network virtualization?BTL4

(Definition:2 marks,Concept explanation:11 marks)

●◆ Network virtualization combines hardware appliances and specific software for the creation and management of a virtual network.Network virtualization can aggregate different physical networks into a single logical network (external network virtualization) or provide network like functionality to an operating system partition (internal network virtualization). The result of external network virtualization is generally a virtual LAN (VLAN).

◆●A VLAN is an aggregation of hosts that communicate with each other as though they were located under the same broadcasting domain. Internal network virtualization is generally applied together with hardware and operating system-level virtualization, in which the guests obtain a virtual network interface to communicate with. • There are several options for implementing internal network virtualization:

1. The guest can share the same network interface of the host and use Network Address Translation (NAT) to access the network; The virtual machine manager can emulate, and install on the host, an additional network device, together with the driver.
2. The guest can have a private network only with the guest.

3. Write a short notes on Storage virtualization?BTL1

(Definition:2 marks,Concept explanation:11 marks)

• Storage virtualization is a system administration practice that allows decoupling the physical organization of the hardware from its logical representation.

Using this technique, users do not have to be worried about the specific location of their data, which can be identified using a logical path.

Storage virtualization allows us to harness a wide range of storage facilities and represent them under a single logical file system.

There are different techniques for storage virtualization, one of the most popular being network based virtualization by means of storage
area networks (SANS).

SANS use a network accessible device through a large bandwidth connection to provide storage facilities.

4. Explain in detail about Operating system level virtualization?BTL4
(Definition:2 marks,Concept explanation:11 marks)

- Operating system level virtualization offers the opportunity to create different and separated execution environments for applications that are managed concurrently.

Differently from hardware virtualization, there is no virtual machine manager or hypervisor and the virtualization is done within a single operating system where the OS kernel allows for multiple isolated user space instances.

- The kernel is also responsible for sharing the system resources among instances and for limiting the impact of instances on each other.
- A user space instance in general contains a proper view of the file system which is completely isolated and separate IP addresses, software configurations and access to devices
- Operating systems supporting this type of virtualization are general purpose, timeshared operating systems with the capability to provide stronger namespace and resource isolation.
- This virtualization technique can be considered an evolution of the chroot mechanism in Unix systems.
- The chroot operation changes the file system root directory for a process and its children to a specific directory.As a result, the process and its children cannot have access to other portions of the file system than those accessible under the new root directory.
- Because Unix systems also expose devices as parts of the file system, by using this method it is possible to completely isolate a set of processes.
- Following the same principle, operating system level virtualization aims to provide separated and multiple execution containers for running applications.
- This technique is an efficient solution for server consolidation scenarios in which multiple application servers share the same technology: operating system, application server framework, and other components.
- Examples of operating system-level virtualizations are FreeBSD Jails, IBM Logical Partition (LPAR), SolarisZones and Containers, Parallels Virtuozzo Containers, OpenVZ, iCore Virtual Accounts, Free Virtual Private Server (FreeVPS), and other

5. Explain in detail about application level virtualization?BTL4
(Definition:2 marks,Concept explanation:11 marks)

- Application level virtualization is a technique allowing applications to be run in runtime environments that do not natively support all the features required by such applications.
- In this scenario, applications are not installed in the expected runtime environment but are run as though they were.
- In general, these techniques are mostly concerned with partial file systems, libraries, and operating system component emulation. Such emulation is performed by a thin layer called a program or an operating system component that is in charge of executing the application.
- Emulation can also be used to execute program binaries compiled for different hardware architectures.
- In this case, one of the following strategies can be implemented:
- Interpretation: In this technique every source instruction is interpreted by an emulator for executing native ISA instructions, leading to poor performance. Interpretation has a minimal startup cost but a huge overhead, since each instruction is emulated.
- Binary translation: In this technique every source instruction is converted to native instructions with equivalent functions. After a block of instructions is translated, it is cached and reused.

◆● Application virtualization is a good solution in the case of missing libraries in the host operating system

In this case a replacement library can be linked with the application or library calls can be remapped to existing functions available in the host system.

Another advantage is that in this case the virtual machine manager is much lighter since it provides a partial emulation of the runtime environment compared to hardware virtualization.

●◆ Compared to programming level virtualization, which works across all the applications developed for that virtual machine, application level virtualization works for a specific environment.

◆● It supports all the applications that run on top of a specific environment.

One of the most popular solutions implementing application virtualization is Wine, which is a software application allowing Unix-like operating systems to execute programs written for the Microsoft

Windows platform. Wine features a software application acting as a container for the guest application and a set of libraries, called Winelib, that developers can use to compile applications to be ported on Unix systems. ◆● Wine takes its inspiration from a similar product from Sun, WindowsApplication Binary Interface (WABI) which implements the Win 16

API specifications on Solaris.

• A similar solution for the Mac OS X environment is CrossOver, which allows running Windows applications directly on the Mac OS X operating system.

●◆ VMware ThinApp is another product in this area, allows capturing the setup of an installed application and packaging it into an executable image isolated from the

hosting operating system.

6. Explain in detail about Virtual Clusters and Resource Management?BTL4
(Definition:2 marks,Diagram:4 marks,Concept explanation:7 marks)

A physical cluster consists of physical servers interconnected by a physical network, while a virtual cluster comprises virtual servers interconnected by a virtual network.

●◆ Virtual clusters present design challenges such as live migration of virtual
machines, memory and file migrations, and dynamic deployment of virtual clusters.

Virtual clusters are built using virtual machines installed across one or more physical clusters, logically interconnected by a virtual network across several physical networks.

Each virtual cluster is formed by physical machines or a virtual machine hosted by multiple physical clusters, with distinct boundaries shown.
Virtual machines can run with different operating systems and are intended to consolidate multiple functionalities on the same server, enhancing server utilization and application flexibility.

They can also be replicated in multiple servers to promote distributed parallelism, fault tolerance, and disaster recovery.

Virtual cluster sizes can grow or shrink dynamically, similar to overlay networks in peer-to-peer networks.

Physical node failures may disable some virtual machines, but virtual machine failures will not affect the host system

Figure 3.3 illustrates the concepts of virtual clusters

Each virtual cluster is formed with physical machines or a VM hosted by multiple physical clusters.The virtual cluster boundaries are shown as distinct boundaries.
Virtual machines can run with different operating systems and are intended to consolidate multiple functionalities on the same server, enhancing server utilization and application flexibility.
They can also be replicated in multiple servers to promote distributed parallelism, fault tolerance, and disaster recovery.Virtual cluster sizes can grow or shrink dynamically, similar to overlay networks in peer-to-peer networks.Physical node failures may disable some virtual machines, but virtual machine failures will not affect the host system

Figure 3.3 illustrates the concepts of virtual clusters

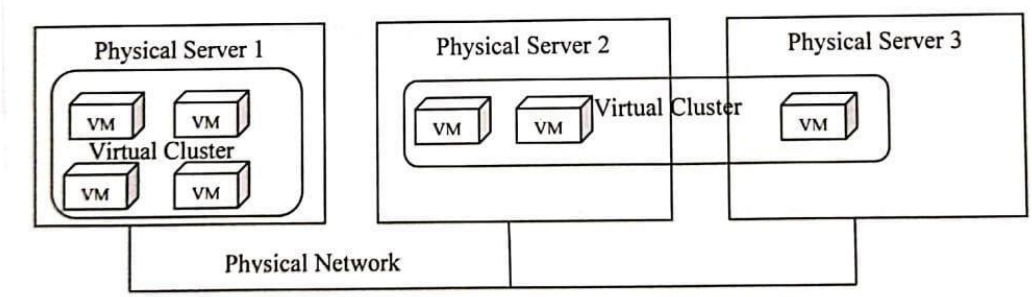


Figure 3.3 virtual clusters

Each virtual cluster is formed with physical machines or a VM hosted by multiple physical clusters.The virtual cluster boundaries are shown as distinct boundaries.

1.Fast Deployment and Effective Scheduling

The system should be capable of quick deployment, which involves creating and distributing software stacks (including the OS, libraries, and applications) to physical nodes within clusters, as well as rapidly
switching runtime environments between virtual clusters for different users.

When a user is finished using their system, the corresponding virtual cluster should be quickly shut down or suspended to free up resources for other users.The concept of "green computing" has gained attention recently, which focuses on reducing energy costs by applying energy-efficient techniques across clusters of homogeneous workstations and specific applications.Live migration of VMs allows workloads to be transferred from one node to another, but designing migration strategies for green computing without compromising cluster performance is a challenge.

Virtualization also enables load balancing of applications within a virtual cluster using the load index and user login frequency.

This load balancing can be used to implement an automatic scale-up and scale-down mechanism for the virtual cluster.

2. High-Performance Virtual Storage

To customize VMs, the template VM can be distributed to multiple physical hosts in the cluster.The process of deploying a group of VMs onto a target cluster involves four key steps: preparing the disk image, configuring the VMs, selecting the destination nodes, and executing the VM deployment command on each host. Each VM is configured with a name, disk image, network settings, as well as a designated amount of CPU and memory, which is then recorded in a file.

Most of the configuration items use identical settings, while some, like UUID, VM name, and IP address, are assigned with automatically calculated values.The primary objective of VM deployment is to meet the VM requirements and balance the workloads across the entire host network.

3.Live VM Migration Steps and Performance

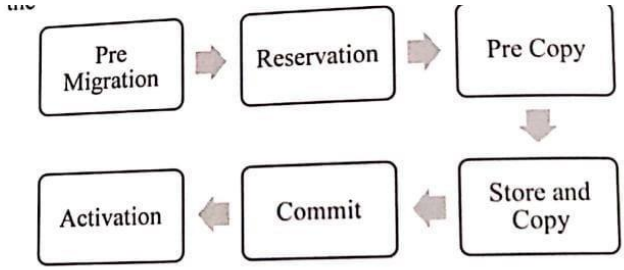
Virtual clustering provides a flexible solution for building clusters consisting of both physical and virtual machines.

It is widely used in various computing systems such as cloud platforms, high-performance computing systems, and computational grids.

Virtual clustering enables the rapid deployment of resources upon user demand or in response to node failures.There are four different ways to manage virtual clusters, including having the cluster manager reside on the guest or host systems, using independent cluster managers, or an integrated cluster manager designed to distinguish between virtualized and physical resources.

A VM can be in one of four states, including an inactive state, an active state, a paused state, and a suspended state.
The live migration of VMs allows for VMs to be moved from one physical machine to another.

In the event of a VM failure, another VM running with the same guest OS can replace it on a different node. During migration, the VM state file is copied from the storage area to the host machine



machine.

Figure 3.4 Live migration steps

- Figure 3.4 shows the six steps of life migration of a VM from host A to host B.

4.Migration of Memory, Files, and Network Resources
Since clusters have a high initial cost of ownership which includes space, power conditioning, and cooling equipment
When one system migrates to another physical node, consider the following issues.

Memory Migration

File System Migration

Network Migration

4.1Memory Migration

One crucial aspect of VM migration is memory migration, which involves moving the memory instance of a VM from one physical host to another.
The efficiency of this process depends on the characteristics of the application/workloads supported by the guest OS. In today's systems, memory migration can range from a few hundred megabytes to several gigabytes.
The Internet Suspend-Resume (ISR) technique takes advantage of temporal locality, where memory states are likely to have significant overlap between the suspended and resumed instances of a VM.The ISR technique represents each file in the file system as a tree of sub files, with a copy existing in both the suspended and resumed VM instances.

By caching only the changed files, this approach minimizes transmission overhead. However, the ISR technique is not suitable for situations where live machine migration is necessary, as it results in high downtime compared to other techniques.

4.2 File System Migration

For a system to support VM migration, it must ensure that each VM has a consistent and location-independent view of the file system that is available on all hosts.

One possible approach is to assign each VM with its own virtual disk and map the file system to it.
However, due to the increasing capacity of disks, it's not feasible to transfer the entire contents of a disk over a network during migration.

Another alternative is to implement a global file system that is accessible across all machines, where a VM can be located without the need to copy files between machines.

4.3 Network Migration
When a VM is migrated to a new physical host, it is important that any open network connections are maintained without relying on forwarding mechanisms or support from mobility or redirection mechanisms on the original host.

To ensure remote systems can locate and communicate with the migrated VM, it must be assigned a virtual IP address that is known to other entities.

This virtual IP address can be different from the IP address of the host machine where the VM is currently located.
Additionally, each VM can have its own virtual MAC address, and the VMM maintains a mapping of these virtual IP and MAC addresses to their corresponding VMs in an ARP table.

7.5 Dynamic Deployment of Virtual Clusters
The Cellular Disco virtual cluster was created at Stanford on a shared- memory multiprocessor system, while the INRIA virtual cluster was built to evaluate the performance of parallel algorithms.

At Duke University, COD was developed to enable dynamic resource allocation with a virtual cluster management system, and at Purdue University, the VIOLIN cluster was constructed to demonstrate the benefits of dynamic adaptation using multiple VM clustering.

7. What is a docker explain its feauters in detail? BTL1
(Definition:2 marks,Concept explanation:11 marks)

Docker is a collection of platforms as a service (PaaS) tools that leverage operating system-level virtualization to distribute software as self-contained packages called containers.

Containers operate in isolation from each other and come bundled
with their own software, libraries, and configuration files. They can communicate

with each other through well-defined channels.
Unlike virtual machines, all containers share a single operating system kernel, which results in lower resource consumption

Docker vs Virtual Machine
Docker containers package an application, its binaries, libraries, and configuration files but do not include a guest OS.

They rely on the underlying OS kernel, which makes them lightweight and they share resources with other containers on the same host OS while providing OS-level process isolation. On the other hand, virtual machines run on hypervisors, which allow multiple VMs to run on a single machine along with its own operating system.
Each VM has its own copy of an operating system along with the application and necessary binaries, making it significantly larger and requiring more resources.

VMs provide hardware-level process isolation, but they are slow to boot.

Key Terminologies

A Docker Image is a file containing multiple layers of instructions used to create and run a Docker container. It provides a portable and reproducible way to package and distribute applications.
A Docker Container is a lightweight and isolated runtime environment created from an image. It encapsulates an application and its dependencies, providing a consistent and predictable environment for running the application.

A Dockerfile is a text file that contains a set of instructions to build a Docker Image. It defines the base image, application code, dependencies, and configuration needed to create a custom Docker Image.
Docker Engine is the software that enables the creation and management of Docker containers. It consists of three main components:

Docker Daemon: It is a server-side component that manages
Docker images, containers, networks, and volumes.
o REST API: It is a set of web services that allows remote clients to interact with Docker Daemon.

O Docker CLI: It is a command-line tool that provides a user- friendly interface to interact with Docker Engine.

Docker Hub is a cloud-based registry that provides a centralized platform for storing, sharing, and discovering Docker Images. It offers a vast collection of pre-built Docker Images that developers can use to build, test, and deploy their applications.

Features of Docker

Open-source platform

An Easy, lightweight, and consistent way of delivery of applications Fast and efficient development life cycle.

Segregation of duties

Service-oriented architecture

Security

Scalability

Reduction in size

Image management

Networking

Volume management

8. What are Docker Components?BTL1
(Definition:2 marks,Diagram:4 marks,Concept explanation:7marks)

Docker implements a client-server model where the Docker client communicates with the Docker daemon to create, manage, and distribute containers.

The Docker client can be installed on the same system as the daemon or connected remotely.
Communication between the client and daemon occurs through a REST API either over a UNIX socket or a network.

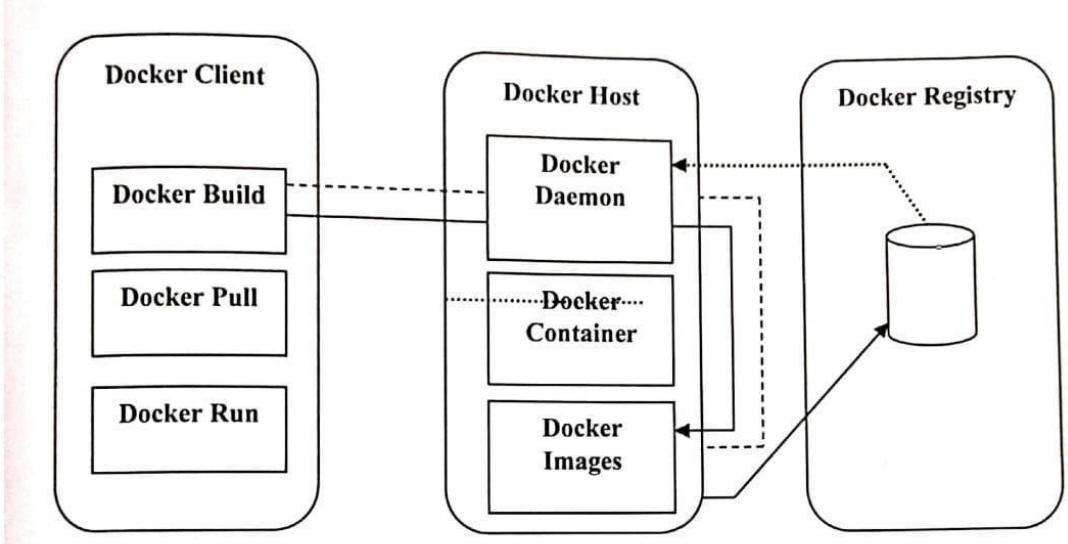


Figure 3.7 Architecture of Docker

The Docker daemon is responsible for managing various Docker services and communicates with other daemons to do so. Using Docker's API requests, the daemon manages Docker objects such as images, containers, networks, and volumes.

Docker Client

The Docker client allows users to interact with Docker and utilize its functionalities. It communicates with the Docker daemon using the Docker API.

The Docker client has the capability to communicate with multiple daemons. When a user runs a Docker command on the terminal, the instructions are sent to the daemon. The Docker daemon receives these instructions in the form of commands and REST API requests from the Docker client.

The primary purpose of the Docker client is to facilitate actions such as pulling images from the Docker registry and running them on the Docker host.

Commonly used commands by Docker clients include docker build, docker pull, and docker run.

Docker Host

A Docker host is a machine that is capable of running multiple containers and is equipped with the Docker daemon, Images, Containers, Networks, and Storage to enable containerization

Docker Registry

Docker images are stored in the Docker registry, which can either be a public registry like Docker Hub, or a private registry that can be set up.

To obtain required images from a configured registry, the 'docker run' or 'docker pull' commands can be used. Conversely, to push images into a configured registry, the 'docker push' command can be used.

Docker Objects

When working with Docker, various objects such as images, containers, volumes, and networks are created and utilized.

Docker Images

A docker image is a set of instructions used to create a container, serving as a read-only template that can store and transport applications.

Images play a critical role in the Docker ecosystem by enabling collaboration among developers in ways that were previously impossible

Docker Storage

Docker storage is responsible for storing data within the writable layer of the container, and this function is carried out by a storage driver.The storage driver is responsible for managing and controlling the images and containers on the Docker host.There are several types of Docker storage.

o Data Volumes, which can be mounted directly into the container's filesystem, are essentially directories or files on the Docker Host filesystem.

o Volume Container is used to maintain the state of the containers' data produced by the running container, where Docker volumes file systems are mounted on Docker containers. These volumes are stored on the host, making it easy for users to exchange file systems among containers and backup data.

O Directory Mounts, where a host directory is mounted as a volume in the container, can also be specified.Finally, Docker volume plugins allow integration with external volumes, such as Amazon EBS, to maintain the state of the container.

Docker networking

Docker networking provides complete isolation for containers, allowing users to link them to multiple networks with minimal OS instances required to run workloads. There are different types of Docker networks available, including:

o Bridge: This is the default network driver and is suitable for different containers that need to communicate with the same Docker host.

o Host: This network is used when there is no need for isolation between the container and the host.

oOverlay: This network allows to communicate with each other. None: This network disables all networking.

Macvlan: This assigns a Media Access Control (MAC)

address to containers, which looks like a physical address.

9. Explain the Docker Containers?BTL4

(Definition:2 marks,Concept explanation:11 marks)

Containers can be connected to one or multiple networks, storage can be attached, and a new image can even be created based on its current state.

By default, a container is isolated from other containers and its host machine. It is possible to control the level of isolation for a container's network, storage or other underlying subsystems from other containers or from the host machine.

A container is defined by its image and configuration options provided during creation or start-up.

Any changes made to a container's state that are not stored in persistent storage will be lost once the container is removed.

Advantages of Docker Containers

Docker provides a consistent environment for running applications from design and development to production and maintenance, which eliminates production issues and allows developers to focus on introducing quality features instead of debugging errors and resolving configuration/compatibility issues.

Docker also allows for instant creation and deployment of containers for every process, without needing to boot the OS, which saves time and increases agility. Creating, destroying, stopping or starting a container can be done with ease, and YAML configuration files can be used to automate deployment and scale the infrastructure.

In multi-cloud environments with different configurations, policies and processes, Docker containers can be easily moved across any environment, providing efficient management. However, it is important to remember that data inside the container is permanently destroyed once the container is destroyed.

Docker environments are highly secure, as applications running in Docker containers are isolated from each other and possess their own resources without interacting with other containers. This allows for better control over traffic flow and easy removal of applications.

Docker enables significant infrastructure cost reduction, with minimal costs for running applications when compared with VMs and other technologies. This can lead to increased ROI and operational cost savings with smaller engineering teams

PART C
15 Marks

1. What are the other types of virtualization?BTL1
(Definition:2 marks,Diagram:5 marks,Concept explanation:8 marks)

Other than execution virtualization, other types of virtualization provide an abstract environment to interact with.

1.Programming language-level virtualization Programming language level virtualization is mostly used to achieve case of deployment of applications, managed execution, portability

across different platforms and operating systems.

●◆ It consists of a virtual machine executing the byte code of a

program which is the result of the compilation process.

• Compilers implemented and used this technology to produce a binary format representing the machine code for an abstract architecture.

●◆ The characteristics of this architecture vary from implementation to implementation.

●◆ Generally these virtual machines constitute a simplification of the underlying
hardware instruction set and provide some high level instructions that map some of the features of the languages compiled for them.

●◆ At runtime, the byte code can be either interpreted or compiled on the fly against the underlying hardware instruction set.

●◆ Programming language level virtualization has a long trail in computer science

history and originally was used in 1966 for the implementation of Basic Combined Programming Language (BCPL), a language for writing compilers and one of the ancestors of the C
programming language.

●◆ Other important examples of the use of this technology have been the UCSD Pascal and Smalltalk

. ●◆ Virtual machine programming languages become popular again with Sun's
introduction of the Java platform in 1996.

The Java virtual machine was originally designed for the execution of programs written in the Java language, but other languages such as

Python, Pascal, Groovy and Ruby were made available.

◆● The ability to support multiple programming languages has been one of the key elements of the Common Language Infrastructure (CLI) which is the specification behind .NET Framework

2.Application server virtualization
Application server virtualization abstracts a collection of application servers that provide the same services as a single virtual application server by using load balancing strategies and providing a high availability infrastructure for the services hosted in the application server.

This is a particular form of virtualization and serves the same purpose of storage virtualization by providing a better quality of service rather than emulating a different environment. 3.6.3 Virtualization Support and Disaster Recover

◆● One very distinguishing feature of cloud computing infrastructure is the use of system virtualization and the modification to provisioning tools.

• Virtualization of servers on a shared cluster can consolidate web services. • In cloud computing, virtualization also means the resources and

fundamental infrastructure are virtualized. • The user will not care about the computing resources that are used for providing the services.

●◆ Cloud users do not need to know and have no way to discover physical resources that are involved while processing a service request. In addition, application developers do not care about some infrastructure issues such as scalability and fault tolerance. Application developers focus on service logic. In many cloud computing systems, virtualization software is used to virtualize the hardware.System virtualization software is a special kind of software which simulates the execution of hardware and runs even unmodified
operating systems.

●◆ Cloud computing systems use virtualization so ware as the running environment for legacy software such as old operating systems and unusual applications.

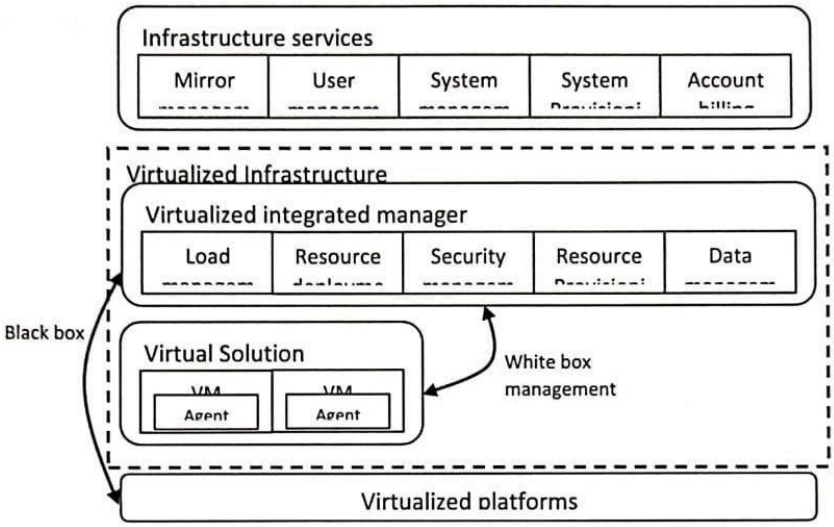


Figure 3.1 Virtualized servers, storage, and network for cloud platform construction

3. Hardware Virtualization

Virtualization software is also used as the platform for developing new cloud applications that enable developers to use any operating systems and programming environments they like.

The development environment and deployment environment can now be the same, which eliminates some runtime problems.

VMs provide flexible runtime services to free users from worrying about the system environment.

●◆ Using VMs in a cloud computing platform ensures extreme flexibility for users. As the computing resources are shared by many users, a method is required to maximize the user's privileges and still keep them separated safely.
Traditional sharing of cluster resources depends on the user agr mechanism on a system.

Such sharing is not flexible.

o Users cannot customize the system for their special purposes.

o Operating systems cannot be changed.

o The separation is not complete.

An environment that meets one user's requirements often cannot satisfy another user.

Virtualization allows us to have full privileges while keeping them separate.

Users have full access to their own VMs, which are completely separate from other user's VMs.

●◆ Multiple VMs can be mounted on the same physical server. Different VMs may run with different OSes.

The virtualized resources form a resource pool.
The virtualization is carried out by special servers dedicated to generating the virtualized resource pool.The virtualized infrastructure (black box in the middle) is built with many virtualizing integration managers.

These managers handle loads, resources, security, data, and provisioning functions. Figure 3.2 shows two VM platforms.

●◆ Each platform carries out a virtual solution to a user job. All cloud services are managed in the boxes at the top.

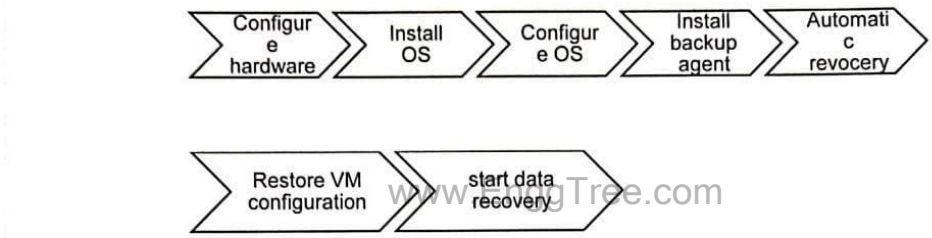


Figure 3.2 Conventional disaster recover scheme versus live migration of VMs

4. Virtualization Support in Public Clouds

AWS provides extreme flexibility (VMS) for users to execute their own applications.

GAE provides limited application level virtualization for users to build applications only based on the services that are created by Google.

Microsoft provides programming level virtualization (.NET virtualization) for users to build their applications.

The VMware tools apply to workstations, servers, and virtual infrastructure.

●◆ The Microsoft tools are used on PCs and some special servers.

• The XenEnterprise tool applies only to Xen-based servers.

5. Virtualization for IaaS

VM technology has increased in ubiquity.

This has enabled users to create customized environments atop physical infrastructure for cloud computing.
Use of VMs in clouds has the following distinct benefits:

o System administrators consolidate workloads of underutilized servers in fewer servers

VMs have the ability to run legacy code without interfering with other APIs VMs can be used to improve security through creation of sandboxes for running applications with questionable reliability

o Virtualized cloud platforms can apply performance isolation, letting providers offer some guarantees and better QoS to customer applications

2. Explain in detail about Containers with advantages and disadvantages?BTL1 (Definition:2marks,Concept Explanation:7 marks,Diagram:2 marks,Advantages:2 marks,Disadvantages:2 marks)

Containers are software packages that are lightweight and self- contained, and they comprise all the necessary dependencies to run an application.
The dependencies include external third-party code packages, system libraries, and other operating system-level applications.
These dependencies are organized in stack levels that are higher than the operating system.

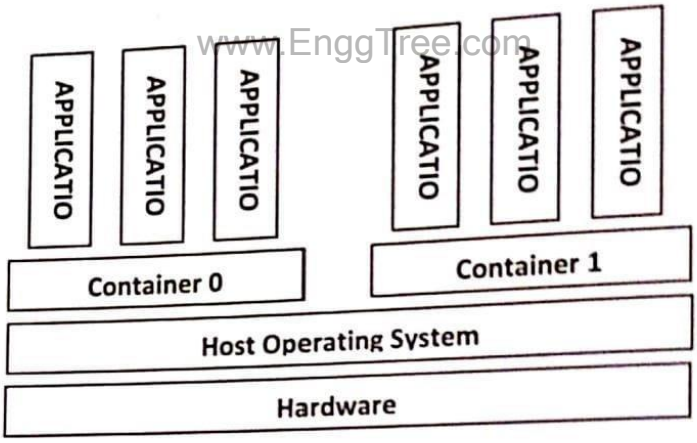


Figure 3.5 Container

Advantages:

One advantage of using containers is their fast iteration speed. Due to their lightweight nature and focus on high-level software, containers can be quickly modified and updated.

o Additionally, container runtime systems often provide a robust ecosystem, including a hosted public repository of pre- made containers.

。 This repository offers popular software applications such as databases and messaging systems that can be easily downloaded and executed, saving valuable time

for development

teams. Disadvantages:

o As containers share the same hardware system beneath the operating system layer, any vulnerability in one container can potentially affect the underlying hardware and break out of the container.
Although many container runtimes offer public repositories of pre-built containers, there is a security risk associated with using these containers as they may contain exploits or be susceptible to hijacking by malicious actors. Examples:

o Docker is the most widely used container runtime that offers Docker Hub, a public repository of containerized applications that can be easily deployed to a local Docker runtime.

o RKT, pronounced "Rocket," is a container system focused on security, ensuring that insecure container functionality is not allowed by default.

o Linux Containers (LXC) is an open-source container runtime system that isolates system-level processes from one another and is utilized by Docker in the background.

CRI-O, on the other hand, is a lightweight alternative to using Docker as the runtime for Kubernetes, implementing the Kubernetes Container Runtime Interface (CRI) to support Open Container Initiative (OCI)-compatible runtimes.
Virtual Machines

Virtual machines are software packages that contain a complete emulation of low-level hardware devices, such as CPU, disk, and networking devices.They may also include a complementary software stack that can run on the emulated hardware.
Together, these hardware and software packages create a functional snapshot of a computational system.
Advantages:

O Virtual machines provide full isolation security since they operate as standalone systems, which means that they are protected from any interference or exploits from other virtual machines on the same host.

o Though a virtual machine can still be hijacked by an exploit, the affected virtual machine will be isolated and cannot contaminate other adjacent virtual machines.

O On the other hand, virtual machines can be interactively developed, unlike containers, which are usually static definitions of the required dependencies and configuration to run the container.
After defining the basic hardware specifications for a virtual machine, it can be treated as a bare-bones computer.

o One can manually install software to the virtual machine and snapshot the virtual machine to capture the present configuration state.

o The virtual machine snapshots can then be utilized to restore the virtual machine to that particular point in time or create additional virtual machines with that

configuration.

Disadvantages:

o Virtual machines are known for their slow iteration speed due to the fact that they involve a complete system stack.

o Any changes made to a virtual machine snapshot can take a considerable amount of time to rebuild and validate that they function correctly.

o Another issue with virtual machines is that they can occupy a significant amount of storage space, often several gigabytes in size.

o This can lead to disk space constraints on the host machine where the virtual machines are stored.

Examples:

Virtualbox is an open source emulation system that emulates x86 architecture, and is owned by Oracle. It is widely used and has a set of additional tools to help develop and distribute virtual machine images.

oVMware is a publicly traded company that provides a hypervisor along with its virtual machine platform, which allows deployment and management of multiple virtual machines. VMware offers robust UI for managing virtual machines, and is a popular enterprise virtual machine solution with support.

o QEMU is a powerful virtual machine option that can emulate any generic hardware architecture. However, it lacks a graphical user interface for configuration or execution, and is a command line only utility. As a result, QEMU is one of the fastest virtual machine options available.

3.Explain Docker Repositories with its features?BTL1
(Definition:2 marks,Concept explanation:13 marks)

The Docker Hub is a cloud-based repository service where users can push their Docker Container Images and access them from anywhere via the internet.It offers the option to push images as private or public and is primarily used by DevOps teams.

The Docker Hub is an open-source tool that is available for all operating systems. It functions as a storage system for Docker images and allows users to pull the required images when needed.However, it is necessary to have a basic knowledge of Docker to push or pull images from the Docker Hub. If a developer team wants to share a project along with its dependencies for testing, they can push the code to Docker Hub. To do this, the developer must create images and push them to Docker Hub. The testing team can then pull the same image from Docker Hub without needing any files, software, or plugins, as the developer has already shared the image with all dependencies.

Features of Docker Hub

Docker Hub simplifies the storage, management, and sharing of images with others. It provides security checks for images and generates comprehensive reports on any security issues.

Additionally, Docker Hub can automate processes like Continuous Deployment and Continuous Testing by triggering webhooks when a new image is uploaded.

Through Docker Hub, users can manage permissions for teams, users, and organizations.

Moreover, Docker Hub can be integrated with tools like GitHub and Jenkins, streamlining workflows.

Advantages of Docker Hub

Docker Container Images have a lightweight design, which enables us to push images in a matter of minutes using a simple command.

This method is secure and offers the option of pushing private or public images.

Docker Hub is a critical component of industry workflows as its popularity grows, serving as a bridge between developer and testing teams.

Making code, software or any type of file available to the public can be done easily by publishing the images on the Docker Hub as public

UNIT IV
CLOUD DEPLOYMENT ENVIRONMENT

SYLLABUS: Google App Engine – Amazon AWS – Microsoft Azure; Cloud Software Environments – Eucalyptus – OpenStack.

PART A
2 Marks

1. Describe about GAE?BTL1

Google's App Engine (GAE) which offers a PaaS platform supporting various cloud and web applications.This platform specializs in supporting scalable (elastic) web applications.GAE enables users to run their applications on a large number of data centers associated with Google's search engine operations.

2. Mention the components maintained in a node of Google cloud platform?BTL1

GFS is used for storing large amounts of data.

MapReduce is for use in application program development. Chubby is used for distributed application lock services. BigTable offers a storage service for accessing structured data.

3. List the functional modules of GAE?BTL1

Datastore Application runtime environment
Software development kit (SDK) • Administration consoleGAE web service infrastructure

4. List some of the storage tools in Azure?BTL1

Blob, Queue, File, and Disk Storage, Data Lake Store, Backup, and Site Recovery.

5. List the applications of GAE?BTL1

Well-known GAE applications include the Google Search Engine, Google Docs, Google Earth, and Gmail.These applications can support large numbers of users simultaneously.Users can interact with Google applications via the web interface provided by each application.Third-party application providers can use GAE to build cloud applications for providing services.

6. Mention the goals for design and implementation of the BigTable system?BTL1

The applications want asynchronous processes to be continuously updating different pieces of data and want access to the most current data at all times.The database needs to support very high read/write rates and the scale might be millions of operations per second.The application may need to examine data changes over time.

7. Describe about Openstack?BTL1

The OpenStack project is an open source cloud computing platform for all types of clouds, which aims to be simple to implement, massively scalable, and feature rich.Developers and cloud computing technologists from around the world create the OpenStack project.OpenStack provides an Infrastructure-as-a-Service (IaaS) solution through a set of interrelated services.

8. List the key services of OpenStack?BTL1

The OpenStack system consists of several key services that are separately installed.
Compute, Identity, Networking, Image, Block Storage, Object Storage, Telemetry, Orchestration and Database services.

9. Describe about Eucalyptus?BTL1

Eucalyptus is an open-source cloud computing software architecture based on Linux that offers Infrastructure as a Service (IaaS) and a storage platform.
It delivers fast and effective computing services and is designed to be compatible with Amazon's EC2 cloud and Simple Storage Service (S3).
Eucalyptus Command Line Interfaces (CLIS) have the capability to manage both Amazon Web Services and private instances.

10. List different types of computing environment?BTL1

Mainframe

Client-Server

Cloud Computing

Mobile Computing

Grid Computing

11. Write short note on Amazon EC2?BTL1

Amazon Elastic Compute Cloud (Amazon EC2) is a cloud-based web service that offers a secure and scalable computing capacity.It allows organizations to customize virtual compute capacity in the cloud, with the flexibility to choose from a range of operating systems and resource configurations such as CPU, memory, and storage.With Amazon EC2, capacity can be increased or decreased within minutes, and it supports the use of hundreds or thousands of server instances simultaneously.This is all managed through web service APIs, enabling applications to scale themselves up or down as needed.

12. Mention the advantages of Dynamo DB?BTL1

Amazon DynamoDB is a NoSQL database service that offers fast and flexible storage for applications requiring consistent, low-latency access at any scale.It's fully managed and supports both document and key-value data models.

13. What is Microsoft Azure?BTL1

Azure is a cloud platform developed by Microsoft, similar to Google Cloud and Amazon Web Services (AWS).It provides access to Microsoft's resources, such as virtual machines, analytical and monitoring tools, and fast data processing.Azure is a cost-effective platform with simple pricing based on the "Pay As You Go" model, which means the user only pay for the resources the user use.

14. List the three modes of network component in Eucalyptus?BTL1

Static mode, which allocates IP addresses to instances

System mode, which assigns a MAC address and connects the instance's network interface to the physical network via NC Managed mode, which creates a local network of instances.

15. Mention the disadvantages of AWS?BTL1

AWS can present a challenge due to its vast array of services and functionalities, which may be hard to comprehend and utilize, particularly for inexperienced users.The cost of AWS can be high, particularly for high-traffic applications or when operating multiple services.

PART B
13 Marks

1. What is Google App Engine and explain its architecture?BTL1
(Definition:2 marks,Concept explanation:8,Diagram:3 marks)

Google has the world's largest search engine facilities.The company has extensive experience in massive data processing that has led to new
insights into data-center design and novel programming models that scale to incredible sizes.
Google platform is based on its search engine expertise.Google has hundreds of data centers and has installed more than 460,000 servers worldwide.
For example, 200 Google data centers are used at one time for a number of cloud applications.
Data items are stored in text, images, and video and are replicated to tolerate faults or failures.
Google's App Engine (GAE) which offers a PaaS platform supporting various cloud and web applications.Google has pioneered cloud development by leveraging the large number of data centers it operates.
For example, Google pioneered cloud services in Gmail, Google Docs, and Google Earth, among other applications.These applications can support a large number of users simultaneously with HA.
Notable technology achievements include the Google File System (GFS), MapReduce, BigTable, and Chubby.In 2008, Google announced the GAE web application platform which is becoming a common platform for many small cloud service providers.This platform specializes in supporting scalable (elastic) web applications.GAE enables users to run their applications on a large number of data centers associated with Google's search engine operations.

1.1 GAE Architecture
GFS is used for storing large amounts of data.
MapReduce is for use in application program development.Chubby is used for distributed application lock services.BigTable offers a storage service for accessing structured data.
Users can interact with Google applications via the web interface provided by each application.
Third-party application providers can use GAE to build cloud applications for providing services.
The applications all run in data centers under tight management by Google engineers.
Inside each data center, there are thousands of servers forming different clusters

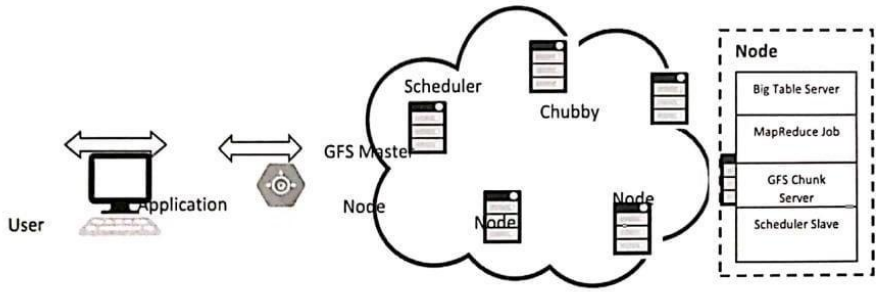


Figure 4.1 Google cloud platform

Google is one of the larger cloud application providers, although it is fundamental service program is private and outside people cannot use the Google infrastructure to build their own service.

The building blocks of Google's cloud computing application include the Google File System for storing large amounts of data, the MapReduce programming framework for application developers, Chubby for distributed application lock services, and BigTable as a storage service for accessing structural or semistructural data.With these building blocks, Google has built many cloud applications.

Figure 4.1 shows the overall architecture of the Google cloud infrastructure.

A typical cluster configuration can run the Google File System, MapReduce jobs and BigTable servers for structure data.

• Extra services such as Chubby for distributed locks can also run in the clusters.

• GAE runs the user program on Gogle's infrastructure. As it is a platform running third-party programs, application developers now do not need to worry about the maintenance of servers.

GAE can be thought of as the combination of several software components.The frontend is an application framework which is similar to other web application frameworks such as ASP, J2EE and JSP.At the time of this writing, GAE supports Python and Java programming environments. The applications can run similar to web application containers.The frontend can be used as the dynamic web serving infrastructure which can provide the full support of common technologies.

2. What are the functional Modules of GAE?BTL1

(Definition:2 marks,Concept explanation:11 marks)

The GAE platform comprises the following five major components.The GAE is not an infrastructure platform, but rather an application development platform for users.The datastore offers object-oriented, distributed, structured data storage services based on BigTable techniques. The datastore secures data management operations.

The application runtime environment offers a platform for scalable web programming and execution. It supports two development languages: Python and Java.

o The software development kit (SDK) is used for local application development. The SDK allows users to execute test runs of local applications and upload application code.

o The administration console is used for easy management of user application development cycles, instead of for physical resource management.

o The GAE web service infrastructure provides special interfaces to guarantee flexible use and management of storage and network resources by GAE.Google offers essentially free GAE services to all Gmail account owners.The user can register for a GAE account or use Gmail account name to sign up for the service.The service is free within a quota.If the user exceeds the quota, the page instructs how to pay for the service. Then the user can download the SDK and read the Python or Java guide to get started.
Note that GAE only accepts Python, Ruby and Java programming languages.
The platform does not provide any IaaS services, unlike Amazon, which offers IaaS and PaaS.
This model allows the user to deploy user-built applications on top of the cloud infrastructure that are built using the programming languages and software tools supported by the provider (e.g., Java, Python).
Azure does this similarly for underlying cloud infrastructure. The cloud provider facilitates support of application development, testing, and operation support on a well-defined service platform.

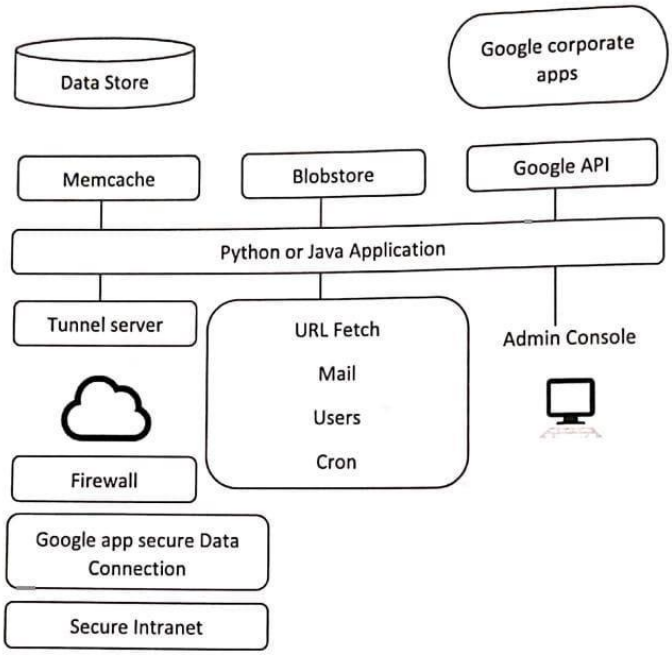
3. Explain the GAE Applications?BTL4
(Definition:2 marks,Concept explanation:8 marks,Diagram:3 marks)

Best-known GAE applications include the Google Search Engine, Google Docs, Google Earth and Gmail.These applications can support large numbers of users simultaneously.Users can interact with Google applications via the web interface provided by each application.Third party application providers can use GAE to build cloud applications for providing services.The applications are all run in the Google data centers. Inside each data center, there might be thousands of server nodes to form different clusters. Each cluster can run multipurpose servers.

GAE supports many web applications.
One is a storage service to store application specific data in the Google infrastructure.
The data can be persistently stored in the backend storage server while still providing the facility for queries, sorting and even transactions similar to traditional database systems.
GAE also provides Google specific services, such as the Gmail account service. This can eliminate the tedious work of building customized user management components in web applications.

1.4 Programming Environment for Google App Engine:
Several web resources (e.g., http://code.google.com/appengine/) and specific books and articles discuss how to program GAE.
Figure 4.2 summarizes some key features of GAE programming model for two supported languages: Java and Python. A client environment that includes an Eclipse plug-in for Java allows
you to debug your GAE on your local machine.
Also, the GWT Google Web Toolkit is available for Java web application developers. Developers can use this, or any other language using a JVM based interpreter or compiler, such as JavaScript or Ruby.Python is often used with frameworks such as Django and CherryPy, but Google also supplies a built in webapp Python environment.
There are several powerful constructs for storing and accessing data.The data store is

a NOSQL data management system for entities that can be, at most, 1 MB in size and are labeled by a set of schema-less properties.Queries can retrieve entities of a given kind filtered and sorted by the values of the properties.Java offers Java Data Object (JDO) and Java Persistence API (JPA) interfaces implemented by the open source Data Nucleus Access platform, while Python has a SQL-like query language called GQL.The data store is strongly consistent and uses optimistic concurrency control.



An update of an entity occurs in a transaction that is retried a fixed number of times if other processes are trying to update the same entity simultaneously.

The user application can execute multiple data store operations in a single transaction which either all succeed or all fail together.

The data store implements transactions across its distributed network using entity groups. A transaction manipulates entities within a single group.Entities of the same group are stored together for efficient execution of transactions.

The user GAE application can assign entities to groups when the entities are created.The performance of the data store can be enhanced by in-memory caching using the memcache, which can also be used independently of the data store.

Recently, Google added the blobstore which is suitable for large files as its size limit is 2 GB.

There are several mechanisms for incorporating external resources.

The Google SDC Secure Data Connection can tunnel through the Internet and link your intranet to an external GAE application. The URL Fetch operation provides the ability for applications to fetch resources and communicate with other hosts over the Internet using HTTP and HTTPS requests.

There is a specialized mail mechanism to send e-mail from your GAE application.

Applications can access resources on the Internet, such as web services or other data, using GAE's URL fetch service.The URL fetch service retrieves web resources using the same high- speed Google infrastructure that retrieves web pages for many other Google products.There are dozens of Google "corporate" facilities including maps, sites, groups, calendar, docs, and YouTube, among others. These support the Google Data API which can be used inside GAE.An application can use Google Accounts for user authentication. Google Accounts handles user account creation and sign-in, and a user that already has a Google account (such as a Gmail account) can use that account

with your app.
GAE provides the ability to manipulate image data using a dedicated Images service which can resize, rotate, flip, crop and enhance images. An application can perform tasks outside of responding to web requests.A GAE application is configured to consume resources up to certain limits or quotas. With quotas, GAE ensures that your application would not exceed your budget and that other applications running on GAE would not impact the performance of your app. In particular, GAE use is free up to certain quotas.GFS was built primarily as the fundamental storage service for Google's search engine.As the size of the web data that was crawled and saved was quite substantial, Google needed a distributed file system to redundantly store massive amounts of data on cheap and unreliable computers.
In addition, GFS was designed for Google applications and Google applications were built for GFS.
In traditional file system design, such a philosophy is not attractive, as there should be a clear interface between applications and the file system such as a POSIX interface. GFS typically will hold a large number of huge files, each 100 MB or larger, with files that are multiple GB in size quite common. Thus, Google has chosen its file data block size to be 64 MB instead of the 4 KB in typical traditional file systems.The I/O pattern in the Google application is also special.Files are typically written once, and the write operations are often the appending data blocks to the end of files.
Multiple appending operations might be concurrent.
BigTable was designed to provide a service for storing and retrieving structured and semi structured data. BigTable applications include storage of web pages, per-user data, and geographic locations.
This is one reason to rebuild the data management system and the resultant system can be applied across many projects for a low incremental cost.

The other motivation for rebuilding the data management system is performance.
Low level storage optimizations help increase performance significantly which is much harder to do when running on top of a traditional database layer.The design and implementation of the BigTable system has the following goals.
The applications want asynchronous processes to be continuously updating different pieces of data and want access to the most current data at all times.The database needs to support very high read/write rates and the scale might be millions of operations per second. The application may need to examine data changes over time. . Thus, BigTable can be viewed as a distributed multilevel map. It provides a fault tolerant and persistent database as in a storage service.
The BigTable system is scalable, which means the system has thousands of servers, terabytes of in-memory data, peta bytes of disk based data, millions of reads/writes per second and efficient scans.BigTable is a self managing system (i.e., servers added/removed dynamically and it features automatic load balancing). can be Chubby, Google's Distributed Lock Service Chubby is intended to provide a coarse-grained locking service.
It can store small files inside Chubby storage which provides a simple namespace as a file system tree.The files stored in Chubby are quite small compared to the huge files in GFS.

4. What are the important AWS Services?BTL1
(Definition:2 marks,Concept explanation:11 marks)

Amazon EC2:
Amazon Elastic Compute Cloud (Amazon EC2) is a cloud- based web service that offers a secure and scalable computing capacity.It allows organizations to customize virtual compute capacity in the cloud, with the flexibility to choose from a range of operating systems and resource configurations such as CPU, memory, and storage.With Amazon EC2 falls under the category of Infrastructure as a Service(IaaS) and provides reliable, cost-effective compute and high-performance infrastructure to meet the demands of businesses.

AWS Lambda:
AWS Lambda is a serverless, event-driven compute service that enables code execution without server management.Compute time consumption is the only factor for payment,and there is no charge when code is not running.AWS Lambda offers the ability to run code for any application type with no need for administration.

AWS Elastic Beanstalk:

AWS Elastic Beanstalk is a cloud-based Platform as a Service that simplifies the process of deploying applications by offering all the necessary application services.It provides a plug-and-play platform that supports a variety of programming languages and environments, including Node.js, Java, PHP, Python, and Ruby. Amazon VPC:
Amazon VPC (Virtual Private Cloud) is a networking service that enables the creation of a private network within the AWS cloud with similar networking concepts and controls as an on- premises network.Users have the ability to configure the network settings, such as IP address ranges, subnets, routing tables, gateways, and security measures.Amazon VPC is an essential AWS service that integrates with many other AWS services.

Amazon Route 53:
Amazon Route 53 is a cloud-based web service that offers a scalable and highly available Domain Name System (DNS) solution. Its primary purpose is to provide businesses and developers with a cost-effective and reliable method of directing end-users to internet applications by converting human-readable domain names into IP addresses that computers can understand.

Amazon S3
Amazon S3(Simple Storage Service) is a web service interface for object storage that enables you to store and retrieve any amount of data from any location on the web. It is designed to provide limitless storage with a 99.999999999% durability guarantee.Amazon S3 can be used as the primary storage solution for cloud-native applications, as well as for backup and recovery and disaster recovery purposes. It delivers unmatched scalability, data availability, security, and performance.

Amazon Glacier:
Amazon Glacier is a highly secure and cost-effective storage service designed for long-term backup and data archiving. It offers reliable durability and ensures the safety of your data. However, since data retrieval may take several hours, Amazon Glacier is primarily intended for archiving purposes.

Amazon RDS

Amazon Relational Database Service (Amazon RDS) simplifies the process of setting up, managing, and scaling a relational database in the cloud. Additionally, it offers resizable and cost-effective capacity and is available on multiple database instance types that are optimized for memory, performance, or I/O.With Amazon RDS, choice of six popular database engines including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, and Microsoft SQL Server.

Amazon DynamoDB
Amazon DynamoDB is a NoSQL database service that offers fast and flexible storage for applications requiring consistent, low-latency access at any scale.It's fully managed and supports both document and key-value
data models.Its versatile data model and dependable performance make it well-suited for various applications such as mobile, web, gaming, Internet of Things (IoT), and more.

5. Explain in detail about Microsoft Azure and its services?BTL4
(Definition:2 marks,Concept explanation:8marks,Advantages:3 marks)

Azure is a cloud platform developed by Microsoft, similar to Google Cloud and Amazon Web Services (AWS). It provides access to Microsoft's resources, such as virtual machines, analytical and monitoring tools, and fast data processing.

Azure is a cost-effective platform with simple pricing based on the "Pay As You G o^ prime prime model, which means the user only pay for the resources the user use. This makes it a convenient option for setting up large servers without requiring significant investments, effort, or physical space.

History

Windows Azure was announced by Microsoft in October 2008 and became available in February 2010.In 2014, Microsoft renamed it as Microsoft Azure.It offered a platform for various services including .NET services, SQL Services, and Live Services.However, some people were uncertain about using cloud technology.Nevertheless, Microsoft Azure is constantly evolving, with new tools and functionalities being added.The platform has two releases: v1 and v2. The earlier version was JSON script-oriented, while the newer version features an interactive UI for easier learning and simplification. Microsoft Azure v2 is still in the preview stage.

Advantages of Azure
Azure offers a cost-effective solution as it eliminates the need for expensive hardware investments.With a pay-as-you-go subscription model, the user can manage their Setting up an Azure account is a simple process through the Azure Portal, where you can choose the desired subscription and begin using the platform.
One of the major advantages of Azure is its low operational cost. Since it operates on dedicated servers specifically designed for cloud functionality, it provides greater reliability compared to on-site servers.By utilizing Azure, the user can eliminate the need for hiring a dedicated technical support team to monitor and troubleshoot servers. This results in significant cost savings for an organization.Azure provides easy backup and recovery options for valuable data. In the event of a disaster, the user can quickly recover the data with a single click, minimizing any impact on end user

business.Cloud-based backup and recovery solutions offer convenience, avoid upfront investments, and provide expertise from third-party providers.Implementing the business models in Azure is straightforward, with intuitive features and user-friendly interfaces. Additionally, there are numerous tutorials available to expedite learning and deployment process

Azure offers robust security measures, ensuring the protection of your critical data and business applications.Even in the face of natural disasters, Azure serves as a reliable safeguard for the resources. The cloud infrastructure remains operational, providing continuous protection.

Azure services

Azure offers a wide range of services and tools for different needs.These include Compute, which includes Virtual Machines, Virtual Machine Scale Sets, Functions for serverless computing, Batch for containerized batch workloads, Service Fabric for microservices and container orchestration, and Cloud Services for building cloud-based apps and APIs.The Networking tools in Azure offer several options like the Virtual Network, Load Balancer, Application Gateway, VPN Gateway, Azure DNS for domain hosting, Content Delivery Network, Traffic Manager, Express Route dedicated private network fiber connections, and Network Watcher monitoring and diagnostics.The Storage tools available in Azure include Blob, Queue, File, and Disk Storage, Data Lake Store, Backup, and Site Recovery, among others. Web + Mobile services make it easy to create and deploy web and mobile applications.Azure also includes tools for Containers, Databases, Data + Analytics, AI + Cognitive Services, Internet of Things, Security + Identity, and Developer Tools, such as Visual Studio Team Services, Azure DevTest Labs, HockeyApp mobile app deployment and monitoring, and Xamarin cross-platform mobile development.

6. Write a short notes on Cloud Software Environments?BTL1

(Definition:2 marks,Concept explanation:11 marks)

Computing environments encompass the technology infrastructure and software platforms utilized for various aspects of software application development, testing, deployment, and execution. These environments come in different types, each serving specific purposes:Mainframe: These are powerful and robust computer systems employed for critical applications and handling extensive data processing tasks.

Client-Server: In this environment, client devices access resources and services from a central server, facilitating the sharing of data and processing capabilities.

Cloud Computing: Cloud computing leverages the Internet to provide resources and services that can be accessed through web browsers or client software. It offers scalability, flexibility, and on-demand availability.

Mobile Computing: This environment revolves around accessing information and applications through handheld devices like smartphones and tablets, allowing users to stay connected on the go.

G**rid Computing:** Grid computing involves the sharing of computing resources and services across multiple computers, enabling large- scale computational tasks and data processing

Embedded Systems: Embedded systems integrate software into devices and products, typically with limited processing power and memory. These systems perform specific functions within various industries, from consumer electronics to automotive and industrial applications.
Each computing environment has its own set of advantages and disadvantages, and the choice of environment depends on the specific requirements of the software application and the available resources Computing has become an integral part of modern life, where computers are utilized extensively to manage, process, and communicate information efficiently.

7.Explain in detail about Eucalyptus and its components?BTL4
(Definition:2 marks,Diagram:3 marks,Concept explanation:6 marks,Advantages:2 marks)

Eucalyptus is an open-source cloud computing software architecture based on Linux that offers Infrastructure as a Service (IaaS) and a storage platform.It delivers fast and effective computing services and is designed to be compatible with Amazon's EC2 cloud and Simple Storage Service (S3).
Eucalyptus Command Line Interfaces (CLIS) have the capability to manage both Amazon Web Services and private instances.This provides clients with the flexibility to migrate instances from Eucalyptus to Amazon Elastic Cloud.The virtualization layer is responsible for managing the network, storage, and computing resources, while hardware virtualization ensures that instances are isolated from each other.

Components:
Eucalyptus has various components that work together to provide efficient cloud computing services.

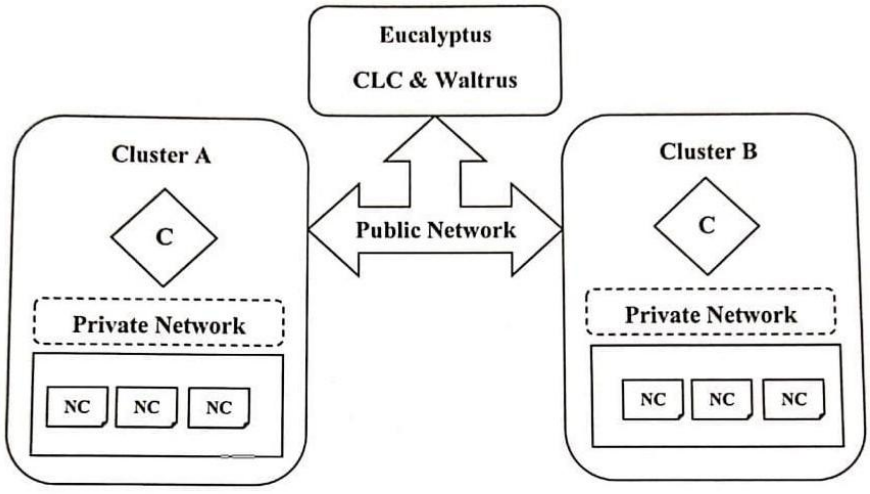


Figure 4.3 Architecture of Eucalyptus

The Node Controller manages the lifecycle of instances and interacts with the operating system, hypervisor, and Cluster Controller.On the other hand, the Cluster Controller manages multiple Node Controllers and the Cloud Controller, which acts as the front-end for the entire architecture.

The Storage Controller, also known as Walrus, allows the creation of snapshots of volumes and persistent block storage over VM instances.
Eucalyptus operates in different modes, each with its own set of features.In Managed Mode, users are assigned security groups that are isolated by VLAN between the Cluster Controller and Node Controller. In Managed (No VLAN) Node mode, however, the root user on the virtual machine can snoop into other virtual machines running on the same network layer.The System Mode is the simplest mode with the least number of features, where a MAC address is assigned to a virtual machine instance and attached to the Node Controller's bridge Ethernet device. Finally, the Static Mode is similar to System Mode but provides more control over the assignment of IP addresses, as a MAC address/IP address pair is mapped to a static entry within the DHCP server.

Features of Eucalyptus

Eucalyptus offers various components to manage and operate cloud infrastructure.The Eucalyptus Machine Image is an example of an image, which is software packaged and uploaded to the cloud, and when it is run, it becomes an instance.

The networking component can be divided into three modes: Static mode, which allocates IP addresses to instances, System mode, which assigns a MAC address and connects the instance's network interface to the physical network via NC, and Managed mode, which creates a local network of instances.Access control is used to limit user permissions. Elastic Block Storage provides block-level storage volumes that can be attached to instances.Auto-scaling and load balancing are used to create or remove instances or services based on demand.

Advantages of Eucalyptus
Eucalyptus is a versatile solution that can be used for both private and public cloud computing.
Users can easily run Amazon or Eucalyptus machine images on either type of cloud. Additionally, its API is fully compatible with all Amazon Web Services, making it easy to integrate with other tools like Chef and Puppet for DevOps.
Although it is not as widely known as other cloud computing solutions like OpenStack and CloudStack, Eucalyptus has the potential to become a viable alternative.It enables hybrid cloud computing, allowing users to combine public and private clouds for their needs. With Eucalyptus, users can easily transform their data centers into private clouds and extend their services to other organizations.

PART C
15 Marks

1. Explain in deail about Amazon AWS and its services?BTL4
(Definition:2 marks,Diagram:3 marks,Concept explanation:10 marks)

Amazon Web Services (AWS), a subsidiary of Amazon.com, has invested significant resources in IT infrastructure distributed globally, which is shared among all AWS account holders worldwide.
These accounts are isolated from each other, and on-demand IT resources are provided to account holders on a pay-as-you-go pricing model with no upfront

costs.AWS provides flexibility by allowing users to pay only for the services they need, helping enterprises reduce their capital expenditure on building private IT infrastructure.AWS has a physical fiber network that connects availability zones, regions, and edge locations, with maintenance costs borne by AWS. While cloud security is AWS's responsibility, security in the cloud is the responsibility of the customer.Performance efficiency in the cloud has four main areas: selection, review, monitoring, and tradeoff.

Advantages of AWS

AWS provides the convenience of easily adjusting resource usage based on your changing needs, resulting in cost savings and ensuring that your application always has sufficient resources.
With multiple data centers and a commitment to 99.99 for many of its services, AWS offers a reliable and secure infrastructure.
Its flexible platform includes a variety of services and tools that can be combined to build and deploy various applications.Additionally, AWS's pay-as-you-go pricing model means user only pay for the resource use, eliminating upfront costs and long-term commitments.

Disadvantages:

AWS can present a challenge due to its vast array of services and functionalities, which may be hard to comprehend and utilize, particularly for inexperienced users.The cost of AWS can be high, particularly for high-traffic applications or when operating multiple services.Furthermore, service expenses can escalate over time, necessitating frequent expense monitoring.AWS's management of various infrastructure elements may limit authority over certain parts of your environment and application.

Global infrastructure

The AWS infrastructure spans across the globe and consists of geographical regions, each with multiple availability zones that are physically isolated from each other. When selecting a region, factors such as latency optimization, cost reduction, and government regulations are considered. In case of a failure in one zone, the infrastructure in other availability zones remains operational, ensuring business continuity.AWS's largest region, North Virginia, has six availability zones that are connected by high-speed fiber-optic networking.
To further optimize content delivery, AWS has over 100 edge locations worldwide that support the CloudFront content delivery network.This network caches frequently accessed content, such as images and videos, at these edge locations and distributes them globally for faster delivery and lower latency for end-users. Additionally, CloudFront offers protection against DDoS attacks

AWS Service model

AWS provides three main types of cloud computing services:
Infrastructure as a Service (IaaS): This service gives developers access to basic building blocks such as data storage space, networking features, and virtual or

dedicated computer hardware. It provides a high degree of flexibility and management control over IT resources. Examples of laaS services on AWS include VPC, EC2, and EBS.

Platform as a Service (PaaS): In this service model, AWS manages the underlying infrastructure, including the operating system and hardware. This allows developers to be more efficient and focus on deploying and managing applications rather than managing infrastructure. Examples of PaaS services on AWS include RDS, EMR, and ElasticSearch.

Software as a Service (SaaS): This service model provides complete end-user applications that typically run on a browser. The service provider runs and manages the software, so end-users only need to worry about using the software that suits their needs. Examples of SaaS applications on AWS include Salesforce.com, web-based email, and Office 365.

2.Explain in detail about OpenStack?BTL4

(Definition:2 marks,Diagram:4 marks,Concept explanation: 9 marks)

The OpenStack project is an open source cloud computing platform for all types of clouds, which aims to be simple to implement, massively scalable and feature rich.Developers and cloud computing technologists from around the world create the OpenStack project.OpenStack provides an Infrastructure as a Service (IaaS) solution through a set of interrelated services.Each service offers an application programming interface (API) that facilitates this integration.Depending on their needs, administrator can install some or all services.

OpenStack began in 2010 as a joint project of Rackspace Hosting and NASA.As of 2012, it is managed by the OpenStack Foundation, a non-profit corporate entity established in September 2013 to promote OpenStack software and its community.Now, More than 500 companies have joined the projectThe OpenStack system consists of several key services that are separately installed.

These services work together depending on the user cloud needs and include the Compute, Identity, Networking, Image, Block Storage, Object Storage, Telemetry, Orchestration, and Database services.

The administrator can install any of these projects separately and configure them standalone or as connected entities.

Figure 4.4 shows the relationships among the OpenStack services:

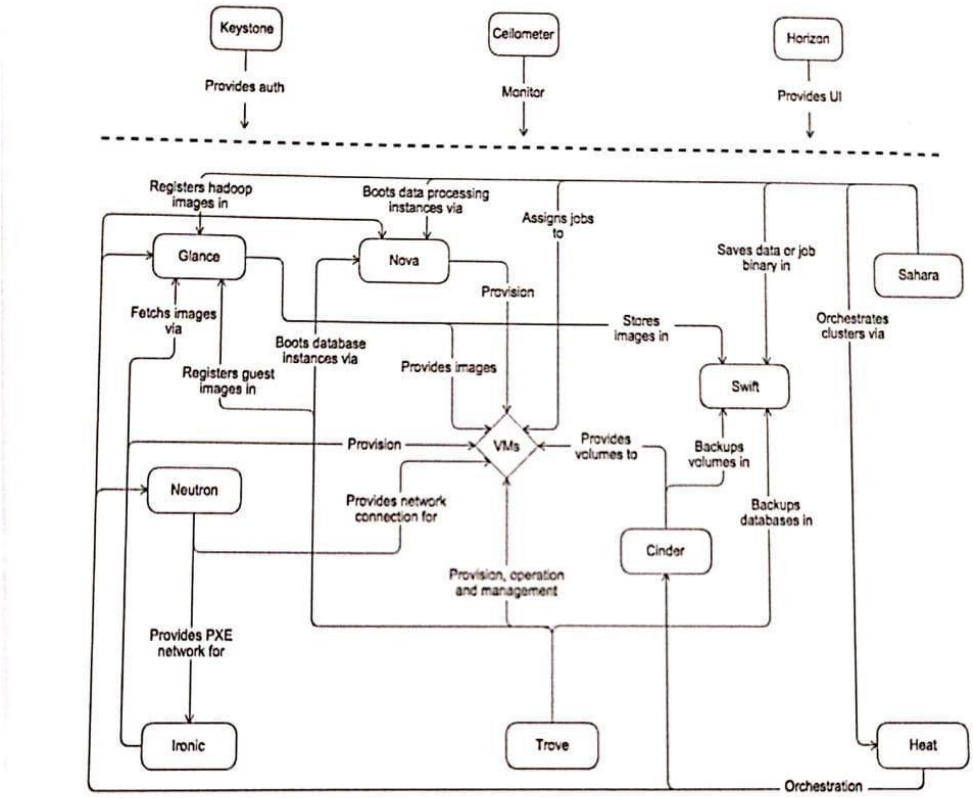


Figure 4.4 Relationship between OpenStack services

To design, deploy, and configure OpenStack, administrators must understand the logical architecture.OpenStack consists of several independent parts, named the OpenStack services. All services authenticate through a common Identity service.Individual services interact with each other through public APIs, except where privileged administrator commands are necessaryInternally, OpenStack services are composed of several processes.

All services have at least one API process, which listens for API requests, preprocesses them and passes them on to other parts of the service.With the exception of the Identity service, the actual work is done by distinct processes.For communication between the processes of one service, an AMQP message broker is used.The service's state is stored in a database.When deploying and configuring the OpenStack cloud, administrator can choose among several message broker and database solutions, such as RabbitMQ, MySQL, MariaDB, and SQLite.Users can access OpenStack via the web-based user interface implemented by the Horizon Dashboard, via command-line clients and by issuing API requests through tools like browser plug-ins or curl.For applications, several SDKs are available. Ultimately, all these access methods issue REST API calls to the various OpenStack services.

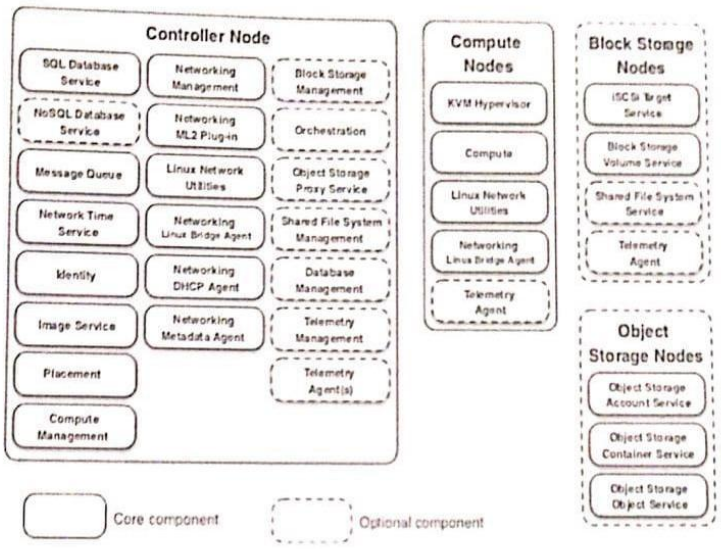


Figure 4.5 Example OpenStack architecture

The controller node runs the Identity service, Image service, Placement service, management portions of Compute, management portion of Networking, various Networking agents, and the Dashboard.It also includes supporting services such as an SQL database, message queue, and NTP.

Optionally, the controller node runs portions of the Block Storage, Object Storage, Orchestration, and Telemetry services.The controller node requires a minimum of two network interfaces.The compute node runs the hypervisor portion of Compute that operates instances. By default, Compute uses the KVM hypervisor. The compute node also runs a Networking service agent that connects instances to virtual networks and provides firewalling services to instances via security groups.

Administrator can deploy more than one compute node. Each node requires a minimum of two network interfaces. The optional Block Storage node contains the disks that the BlockStorage and Shared File System services provision for instances. For simplicity, service traffic between compute nodes and this node uses the management network.

Production environments should implement a separate storage network to increase performance and security. Administrator can deploy more than one block storage node. Each node requires a minimum of one network interface.The optional Object Storage node contains the disks that the Object Storage service uses for storing accounts, containers, and objects.For simplicity, service traffic between compute nodes and this node uses the management network.Production environments should implement a separate storage network to increase performance and security.This service requires two nodes. Each node requires a minimum of one network interface. Administrator can deploy more than two object storage nodes.The provider networks option deploys the OpenStack Networking service in the simplest way possible with primarily layer 2 (bridging/switching) services and VLAN segmentation of networks. Essentially, it bridges virtual networks to physical networks and relies on physical network infrastructure for layer-3 (routing) services.Additionally, a DHCP service provides IP address information to instances.

UNIT V
CLOUD SECURITY

SYLLABUS:Virtualization System-Specific Attacks: Guest hopping – VM migration attack – hyperjacking. Data Security and Storage; Identity and Access Management (IAM) - IAM Challenges - IAM Architecture and Practice.

PART A
2 Marks

1.What is a virtualization attack?BTL1

Virtualization Attacks One of the top cloud computing threats involves one of its core enabling technologies: virtualization. In virtual environments, the attacker can take control of virtual machines installed by compromising the lower layer hypervisor.

2.What are the different types of VM attacks?BTL1

However, virtualization introduces serious threats to service delivery such as Denial of Service (DoS) attacks, Cross-VM Cache Side Channel attacks, Hypervisor Escape and Hyper-jacking. One of the most sophisticated forms of attack is the cross-VM cache side channel attack that exploits shared cache memory between VMs.

3.What is guesthopping?BTL1

Guest-hopping attack: In this type of attack, an attacker will try to get access to one virtual machine by penetrating another virtual machine hosted in the same hardware. One of the possible mitigations of guest hopping attack is the Forensics and VM debugging tools to observe the security of cloud.

4.What is a hyperjacking attack?BTL1

Hyperjacking is an attack in which a hacker takes malicious control over the hypervisor that creates the virtual environment within a virtual machine (VM) host.

5.How does a hyperjacking attack work?BTL1

Hyperjacking is an attack in which an adversary takes malicious control over the hypervisor that creates the virtual environment within a virtual machine (VM) host.

6.What is data security and storage in cloud computing?BTL1

Cloud data security is the practice of protecting data and other digital information assets from security threats, human error, and insider threats. It leverages technology, policies, and processes to keep your data confidential and still accessible to those who need it in cloud-based environments

7.What are the 5 components of data security in cloud computing?BTL1

Visibility.
Exposure Management.
Prevention Controls.
Detection.
Response

8. What is cloud storage and its types?BTL1

What are the types of cloud storage? There are three main cloud storage types: object storage, file storage, and block storage. Each offers its own advantages and has its own use cases.

9. What are the four principles of data security?BTL1

There are many basic principles to protect data in information security. The primary principles are confidentiality, integrity, accountability, availability, least privilege, separation of privilege, and least common mechanisms. The most common security principle is CIA triad with accountability

10. What is the definition if IAM?BTL1

Identity and access management (IAM) ensures that the right people and job roles in your organization (identities) can access the tools they need to do their jobs. Identity management and access systems enable your organization to manage employee apps without logging into each app as an administrator

11. What are the challenges of IAM?BTL1

Lack of centralized view
Difficulties in User Lifecycle Management
Keeping Application Integrations Updated
Compliance Visibility into Third Party SaaS Tools

12. What is the principle of IAM?BTL1

A principal is a human user or workload that can make a request for an action or operation on an AWS resource. After authentication, the principal can be granted either permanent or temporary credentials to make requests to AWS, depending on the principal type.

13. What is IAM tools?BTL1

Identity access management (IAM) or simply put, identity management, is a category of software tools that allows businesses of all sized to generally manage the identities and access rights of all their employees.

14. How many types of IAM are there?BTL1

IAM roles are of 4 types, primarily differentiated by who or what can assume the role: Service Role. Service-Linked Role. Role for Cross-Account Access.

15. What are IAM requirements?BTL1

IAM requirements are organized into four categories: Account Provisioning & De-provisioning, Authentication, Authorization & Role Management, and Session

Management. For each category a general description of goals is provided, followed by a list of specific requirements that will help ensure goals will be met

PART B
13 Marks

1. What is virtual migration attacks?BTL1
(Definition:2 marks,Concept explanation:11 marks)

Virtual Machine Migration

The movement of VMs from one resource to another, such as from one physical host to another physical host, or data store to data store, is known as VM migration. There are two types of VM migration: cold and live. **Cold migration** occurs when the VM is shut down. **Live migration** occurs while the VM is actually running. This amazing new capability is particularly useful if maintenance is required on the part of the physical infrastructure and the application running on that infrastructure is mission-critical. Before the availability of live migration applications, managers were stuck with the choice of either causing a planned outage, which in some global corporations is not always feasible, or waiting and not taking the system down, which risks an unplanned outage in the future. Needless to say, neither of these choices is optimal.With live migration, a running system is copied to another system and when the last bits of the running system's state are copied, the switch is made and the new system becomes the active server. This process can take several minutes to complete, but is a great advantage over the two previous options.

Earlier versions of live migration were limited to moving VMs within the same data centers. That restriction was removed and it is now possible to perform live migrations between different data centers. This capability provides an entirely new set of options and availability, including the ability to move workloads from a data center that may be in the eye of a storm to another data center outside of the target area. Again, these application moves can be accomplished without any application outages. There are several products on the market today that provide some form of live migration. These products and platforms may have some guidelines and requirements to provide the capability. If an organization is considering live migration as an option, it is recommended to check with the virtualization software vendor to understand those requirements, particularly for the data center.

VM Migration attcak

Migration works by sending the state of the guest virtual machine's memory and any virtualized devices to a destination host physical machine. Live Migration has many security vulnerabilities. The security threats could be on the data plane, control plane and migration plane.

Types of migration attacks

virtualization introduces serious threats to service delivery such as Denial of Service (DoS) attacks, Cross-VM Cache Side Channel attacks, Hypervisor Escape and Hyper-jacking. One of the most sophisticated forms of attack is the cross-VM cache side channel attack that exploits shared cache memory between VMs.

2. Write a short notes on guest hopping?BTL1
(Definition:2 marks,Concept explanation:11 marks)

Guest-hopping attack: one of the possible mitigations of guest hopping attack is the Forensics and VM debugging tools toobserve any attempt to compromise VM. Another possible mitigation is using High Assurance Platform (HAP) whichprovides a high degree of isolation between virtual machines.-SQL injection: to mitigate SQL injection attack you should remove all stored procedures that are rarely used. Also,assign the least possible privileges to users who have permissions to access the database-Side channel attack: as a countermeasure, it might be preferable to ensure that none of the legitimate user VMs resides onthe same hardware of other users. This completely eliminates the risk of side-channel attacks in a virtualized cloudenvironment-Malicious Insider: strict privileges' planning, security auditing can minimize this security threat-Data storage security: ensuring data integrity and confidentlyoEnsure limited access to the users' data by the CSP employees.

What Is Hyperjacking?

Hyperjacking involves the compromise and unauthorized control of a virtual machine (VM). So, before we discuss hyperjacking in detail, we'll need to first understand what a virtual machine is.

Virtual Machine

A virtual machine is just that: a non-physical machine that uses virtualization software instead of hardware to function. Though virtual machines must exist on a piece of hardware, they operate using virtual components (such as a virtual CPU).

Hypervisors form the backbone of virtual machines. These are software programs that are responsible for creating, running, and managing VMs. A single hypervisor can host multiple virtual machines, or multiple guest operating systems, at one time, which also gives it the alternative name of virtual machine manager (VMM).

There are two kinds of hypervisors. The first is known as a "bare metal" or "native" hypervisor, with the second being a "host" hypervisor. What you should note is that it is the hypervisors of virtual machines that are the targets of hyperjacking attacks (hence the term "hyper-jacking").

Origins of Hyperjacking

In the mid-2000s, researchers found that hyperjacking was a possibility. At the time, hyperjacking attacks were entirely theoretical, but the threat of one being carried out was always there. As technology advances and cybercriminals become more inventive, the risk of hyperjacking attacks increases by the year.

In fact, in September 2022, warnings of real hyperjacking attacks began to arise. Both Mandiant and VMWare published warnings stating that they found malicious actors using malware to conduct hyperjacking attacks in the wild via a harmful version of VMWare software. In this venture, the threat actors inserted their own malicious code within victims' hypervisors while bypassing the target devices' security measures (similarly to a rootkit).

Through this exploit, the hackers in question were able to run commands on the virtual machines' host devices without detection.

How Does a Hyperjacking Attack Work?

Hypervisors are the key target of hyperjacking attacks. In a typical attack, the original hypervisor will be replaced via the installation of a rogue, malicious hypervisor that the threat actor has control of. By installing a rogue hypervisor under the original, the attacker can therefore gain control of the legitimate hypervisor and exploit the VM.

By having control over the hypervisor of a virtual machine, the attacker can, in turn, gain control of the entire VM server. This means that they can manipulate anything in the virtual machine. In the aforementioned hyperjacking attack announced in September 2022, it was found that hackers were using hyperjacking to spy on victims.

Compared to other hugely popular cybercrime tactics like phishing and ransomware, hyperjacking isn't very common at the moment. But with the first confirmed use of this method, it's important that you know how to keep your devices, and your data, safe.

3. Explain about cloud data security in detail?BTL4

(Definition:2 marks,Concept explanation:11 marks)

Cloud data security is the practice of protecting data and other digital information assets from security threats, human error, and insider threats. It leverages technology, policies, and processes to keep your data confidential and still accessible to those who need it in cloud-based environments. Cloud computing delivers many benefits, allowing you to access data from any device via an internet connection to reduce the chance of data loss during outages or incidents and improve scalability and agility. At the same time, many organizations remain hesitant to migrate sensitive data to the cloud as they struggle to understand their security options and meet regulatory demands.

Understanding how to secure cloud data remains one of the biggest obstacles to overcome as organizations transition from building and managing on-premises data centers. So, what is data security in the cloud? How is your data protected? And what cloud data security best practices should you follow to ensure cloud-based data assets are secure and protected?

Read on to learn more about cloud data security benefits and challenges, how it works, and how Google Cloud enables companies to detect, investigate, and stop threats across cloud, on-premises, and hybrid deployments.

Cloud data security protects data that is stored (at rest) or moving in and out of the cloud (in motion) from security threats, unauthorized access, theft, and corruption. It relies on physical security, technology tools, access management and controls, and organizational policies.

Why companies need cloud security

Today, we're living in the era of big data, with companies generating, collecting, and storing vast amounts of data by the second, ranging from highly confidential business or personal customer data to less sensitive data like behavioral and marketing analytics.

Beyond the growing volumes of data that companies need to be able to access, manage, and analyze, organizations are adopting cloud services to help them achieve more agility and faster times to market, and to support increasingly remote or hybrid workforces. The traditional network perimeter is fast disappearing, and security teams are realizing that they need to rethink current and past approaches when it comes to securing cloud data. With data and applications no longer living inside your data center and more people than ever working outside a physical office, companies must solve how to protect data and manage access to that data as it moves across and through multiple environments.

4. What are the challenges of cloud data security?BTL1
(Definition:2 marks,Concept explanation:11marks)

As more data and applications move out of a central data center and away from traditional security mechanisms and infrastructure, the higher the risk of exposure becomes. While many of the foundational elements of on-premises data security remain, they must be adapted to the cloud.

Common challenges with data protection in cloud or hybrid environments include:

- **Lack of visibility.** Companies don't know where all their data and applications live and what assets are in their inventory.

- **Less control.** Since data and apps are hosted on third-party infrastructure, they have less control over how data is accessed and shared.
- **Confusion over shared responsibility.** Companies and cloud providers share cloud security responsibilities, which can lead to gaps in coverage if duties and tasks are not well understood or defined.
- **Inconsistent coverage.** Many businesses are finding multicloud and hybrid cloud to better suit their business needs, but different providers offer varying levels of coverage and capabilities that can deliver inconsistent protection.
- **Growing cybersecurity threats.** Cloud databases and cloud data storage make ideal targets for online criminals looking for a big payday, especially as companies are still educating themselves about data handling and management in the cloud.
- **Strict compliance requirements.** Organizations are under pressure to comply with stringent data protection and privacy regulations, which require enforcing security policies across multiple environments and demonstrating strong data governance.
- **Distributed data storage.** Storing data on international servers can deliver lower latency and more flexibility. Still, it can also raise data sovereignty issues that might not be problematic if you were operating in your own data center.

5.What are the Benefits of cloud data security?BTL1

(Definition:2 marks,Concept explanation:11marks)

Greater visibility

Strong cloud data security measures allow you to maintain visibility into the inner workings of your cloud, namely what data assets you have and where they live, who is using your cloud services, and the kind of data they are accessing.

Easy backups and recovery

Cloud data security can offer a number of solutions and features to help automate and standardize backups, freeing your teams from monitoring manual backups and troubleshooting problems. Cloud-based disaster recovery also lets you restore and recover data and applications in minutes.

Cloud data compliance

Robust cloud data security programs are designed to meet compliance obligations, including knowing where data is stored, who can access it, how it's processed, and how it's protected. Cloud data loss prevention (DLP) can help you easily discover, classify, and de-identify sensitive data to reduce the risk of violations.

Data encryption

Organizations need to be able to protect sensitive data whenever and wherever it goes. Cloud service providers help you tackle secure cloud data transfer, storage, and sharing by implementing several layers of advanced encryption for securing cloud data, both in transit and at rest.

Lower costs

Cloud data security reduces total cost of ownership (TCO) and the administrative and management burden of cloud data security. In addition, cloud providers offer the latest security features and tools, making it easier for security professionals to do their jobs with automation, streamlined integration, and continuous alerting.

Advanced incident detection and response

An advantage of cloud data security is that providers invest in cutting-edge AI technologies and built-in security analytics that help you automatically scan for suspicious activity to identify and respond to security incidents quickly.

6. Write a short notes on IAM challenges?BTL1

(Definition:2 marks,Concept explanation:11 marks)

IAM Challenges One critical challenge of IAM concerns managing access for diverse user populations (employees, contractors, partners, etc.) accessing internal and externally hosted services. IT is constantly challenged to rapidly provision appropriate access to the users whose roles and responsibilities often change for business reasons. Another issue is the turnover of users within the organization. Turnover varies by industry and function—seasonal staffing fluctuations in finance departments, for example—and can also arise from changes in the business, such as mergers and acquisitions, new product and service releases, business process outsourcing, and changing responsibilities. As a result, sustaining IAM processes can turn into a persistent challenge. Access policies for information are seldom centrally and consistently applied. Organizations can contain disparate directories, creating complex webs of user identities, access rights, and procedures. This has led to inefficiencies in user and access management processes while exposing these organizations to significant security, regulatory compliance, and reputation risks. To address these challenges and risks, many companies have sought technology solutions to enable centralized and automated user access management. Many of these initiatives are entered into with high expectations, which is not surprising given that the problem is often large and complex. Most often those initiatives to improve IAM can span several years and incur considerable cost. Hence, organizations should approach their IAM strategy and architecture with both business and IT drivers that address the core inefficiency issues while preserving the control's efficacy (related to access control). Only then will the organizations have a higher likelihood of success and return on investment.

PART C

15 Marks

1. Explain in detail about IAM architecture?BTL4
(Definition:2 marks,Diagram:4 marks,Concept explanation:9 marks)

Identity Access Management is used by the root user (administrator) of the organization. The users represent one person within the organization, and the users can be grouped in that all the users will have the same privileges to the services.

Shared Responsibility Model Identity Access Management Cloud Service Provider (CSP)

For Infrastructure (Global Security of the Network)

Configuration and Vulnerability Analysis

Compliance Validation Custom Roles,Policies Users, Groups, Management and Monitoring

Use IAM tools to apply for appropriate permissions. Analyze access patterns and review permissions.The Architecture of Identity Access Management

User Management:- It consists of

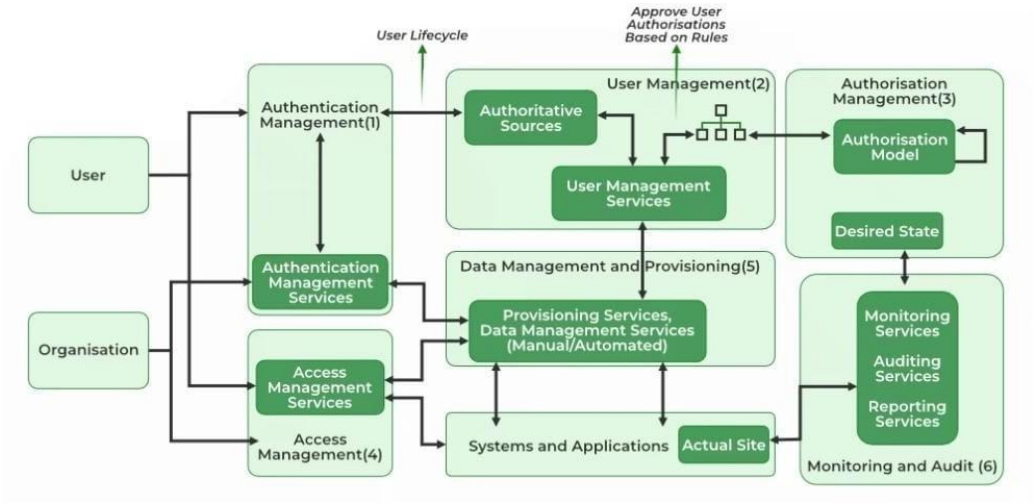
activities for the control and management over the identity life cycles.

Authentication Management:- It consists of activities for effectively controlling and managing the processes for determining which user is trying to access the services and whether those services are relevant to him or not.

Authorization Management:- It consists of activities for effectively controlling and managing the processes for determining which services are allowed to access according to the policies made by the administrator of the organization.

Access Management: It is used in response to a request made by the user wanting to access the resources with the which services are allowed to access according to the policies made by the administrator of the organization.

Access Management: It is used in response to a request made by the user wanting to access the resources with the organization.



Data Management and Provisioning:
The authorization of data and identity are carried towards the IT resource through automated or manual processes.

Monitoring and Auditing:- Based on the defined policies the monitoring, auditing, and reporting are done by the users regarding their access to within the organization. resources

Operational Activities of IAM:- In this process, we onboard the new users on the organization's system and application and provide them with necessary access to the services and data. Deprovisioning works completely opposite in that we delete or deactivate the identity of the user and de-relinquish all the privileges of the user.

Credential and Attribute Management:- Credentials are bound to an individual user and are verified during the authentication process. These processes generally include allotment of username, static or dynamic password, handling the password expiration, encryption management, and access policies of the user.

Entitlement Management:- These are also known as authorization policies in which we address the provisioning and de-provisioning of the privileges provided to the user for accessing the databases, applications, and systems. We provide only the required privileges to the users according to their roles. It can also be used for security purposes.

Identity Federation Management:- In this process, we manage the relationships beyond the internal networks of the organization that is among the different organizations. The federations are the associate of the organization that came together for exchanging information about the user's resources to enable collaboration and transactions.

Centralization of Authentication andAuthorization:- It needs to be developed in

order to build custom authentication and authorization features into their application, it also promotes the loose coupling architecture.

2.What are the IAM Practices in Cloud ?BTL1

(Comparison table:3 marks,Diagram:3 marks,Concept explanation:9 marks)

When compared to the traditional applications deployment model within the enterprise, IAM practices in the cloud are still evolving. In the current state of IAM technology, standards support by CSPs (SaaS, PaaS, and IaaS) is not consistent across providers. Although large providers such as Google, Microsoft, and Salesforce.com seem to demonstrate basic IAM capabilities, our assessment is that they still fall short of enterprise IAM requirements for managing regulatory, privacy, and data protection requirements. Table 5-2 illustrates the current maturity model, based on the authors' assessment, generalized across SPI service delivery models.

TABLE 5-2. Comparison of SPI maturity models in the context of IAM

Level	SaaS	PaaS	IaaS
User Management, New Users	Capable	Immature	Aware
User Management, User Modifications	Capable	Immature	Immature
Authentication Management	Capable	Aware	Capable
Authorization Management	Aware	Immature	Immature

The maturity model takes into account the dynamic nature of IAM users, systems, and applications in the cloud and addresses the four key components of the IAM automation process: • User Management, New Users • User Management, User Modifications • Authentication Management • Authorization Management Table 5-3 defines the maturity levels as they relate to the four key components.

TABLE 5-3 Comparison of maturity levels for IAM components

Level	Immature	Aware	Capable	Mature	Industry-leading
User Management, New Users	Manual, ad hoc, with no formal process	Manual, ad hoc, following established processes	Automated where appropriate Disparate processes	Automated using more than one process	Automated using a single provisioning process
User Management, User Modifications	Manual, ad hoc, per application	Manual, ad hoc, per application group	Manual or automated per application group	Automated per class of application and resource	Automated across the application space
Authentication Management	Manual, ad hoc No common security policy	Addressed per application No common authorization mechanism	Common authentication mechanism No common authentication module	Common authentication module Minimal credentials Common security policy	Common authentication mechanism as a component service to applications Common security policy
Authorization Management	Manual, ad hoc No rule- or role-based authorization	Addressed per application No common authorization mechanism	Common service No common module	Common module Application-specific attributes disparately maintained	Common mechanism Centrally managed attributes Support role Rule-based

By matching the model's descriptions of various maturity levels with the cloud services delivery model's (SaaS, PaaS, IaaS) current state of IAM, a clear picture emerges of IAM maturity across the four IAM components. If, for example, the service delivery model (SPI) is "immature" in one area but "capable" or "aware" in all others, the IAM maturity model can help focus attention on the area most in need of attention.

Although the principles and purported benefits of established enterprise IAM practices and processes are applicable to cloud services, they need to be adjusted to the cloud environment. Broadly speaking, user management functions in the cloud can be categorized as follows:

- Cloud identity administration

• Federation or SSO

• Authorization management

• Compliance management

We will now discuss each of the aforementioned practices in detail.

Cloud Identity Administration Cloud identity administrative functions should focus on life cycle management of user identities in the cloud—provisioning, deprovisioning, identity federation, SSO, password or credentials management, profile management, and administrative management. Organizations that are not capable of supporting federation should explore cloud-based identity management services. This new breed of services usually synchronizes an organization's internal directories with its directory (usually multitenant) and acts as a proxy IdP for the organization. By federating identities using either an internal Internet-facing IdP or a cloud identity management service provider, organizations can avoid duplicating identities and attributes and storing them with the CSP. Given the inconsistent and sparse support for identity standards among CSPs, customers may have to devise custom methods to address user management functions in the cloud. Provisioning users when federation is not supported can be complex and laborious. It is not unusual for organizations to employ manual processes, web-based administration, outsourced (delegated) administration that involves uploading of spreadsheets, and execution of custom scripts at both the customer and CSP locations. The latter model is not desirable as it is not scalable across multiple CSPs and will be costly to manage in the long run. Federated Identity (SSO) Organizations planning to implement identity federation that enables SSO for users can take one of the following two paths (architectures): • Implement an enterprise IdP within an organization perimeter. • Integrate with a trusted cloud-based identity management service provider. Both architectures have pros and cons.

Enterprise identity provider

In this architecture, cloud services will delegate authentication to an organization's IdP. In this delegated authentication architecture, the organization federates identities within a trusted circle of CSP domains. A circle of trust can be created with all the domains that are authorized to delegate authentication to the IdP. In this deployment architecture, where the organization will provide and support an IdP, greater control can be exercised over user identities, attributes, credentials, and policies for authenticating and authorizing users to a cloud service. Figure 5-7 illustrates the IdP deployment architecture.

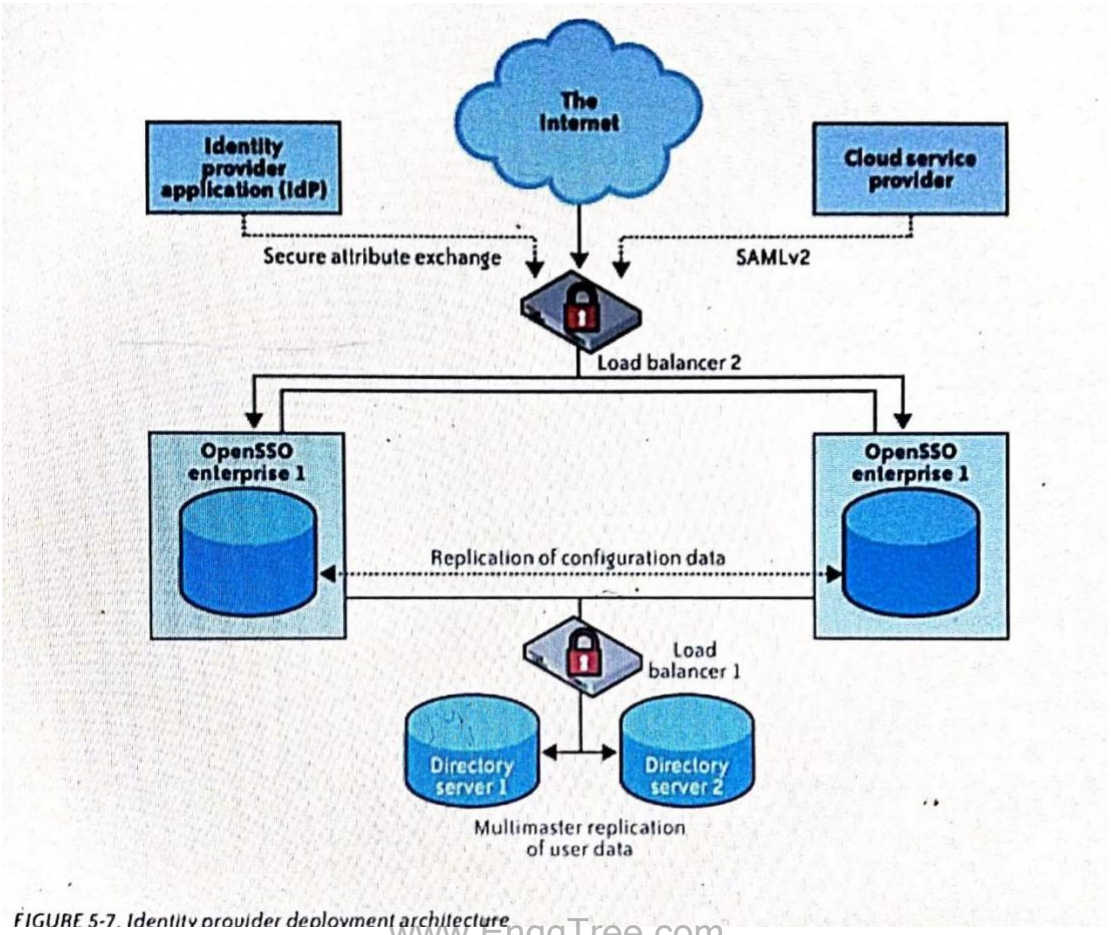


FIGURE 5-7. Identity provider deployment architecture

Here are the specific pros and cons of this approach: Pros Organizations can leverage the existing investment in their IAM infrastructure and extend the practices to the cloud. For example, organizations that have implemented SSO for applications within their data center exhibit the following benefits:

• They are consistent with internal policies, processes, and access management frameworks. • They have direct oversight of the service-level agreement (SLA) and security of the IdP. • They have an incremental investment in enhancing the existing identity architecture to support federation. Cons By not changing the infrastructure to support federation, new inefficiencies can result due to the addition of life cycle management for non-employees such as customers. Most organizations will likely continue to manage employee and long-term contractor identities using organically developed IAM infrastructures and practices. But they seem to prefer to outsource the management of partner and consumer identities to a trusted cloudbased identity provider as a service partner. Identity management-as-a-service In this architecture, cloud services can delegate authentication to an identity management-asa-service (IDaaS) provider. In this model, organizations outsource the federated identity management technology and user management processes to a third-party service provider, such as Ping Identity, TriCipher's Myonelogin.com, or Symplified.com. When federating identities to the cloud, organizations may need to manage the identity life cycle using their IAM system and processes. However, the organization might benefit from an outsourced multiprotocol federation gateway (identity federation service) if it has to interface with many different partners and cloud service federation schemes. For example, as of this writing, Salesforce.com supports SAML

1.1 and Google Apps supports SAML 2.0. Enterprises accessing Google Apps and Salesforce.com may benefit from a multiprotocol federation gateway hosted by an identity management CSP such as Symplified or TriCipher. In cases where credentialing is difficult and costly, an enterprise might also outsource credential issuance (and background investigations) to a service provider, such as the GSA Managed Service Organization (MSO) that issues personal identity verification (PIV) cards and, optionally, the certificates on the cards. The GSA MSO† is offering the USAccess management end-to-end solution as a shared service to federal civilian agencies. In essence, this is a SaaS model for identity management, where the SaaS IdP stores identities in a "trusted identity store" and acts as a proxy for the organization's users accessing cloud services, as illustrated in Figure 5-8.• They are consistent with internal policies, processes, and access management frameworks. • They have direct oversight of the service-level agreement (SLA) and security of the IdP. • They have an incremental investment in enhancing the existing identity architecture to support federation. Cons By not changing the infrastructure to support federation, new inefficiencies can result due to the addition of life cycle management for non-employees such as customers. Most organizations will likely continue to manage employee and long-term contractor identities using organically developed IAM infrastructures and practices. But they seem to prefer to outsource the management of partner and consumer identities to a trusted cloudbased identity provider as a service partner. Identity management-as-a-service In this architecture, cloud services can delegate authentication to an identity management-asa-service (IDaaS) provider. In this model, organizations outsource the federated identity management technology and user management processes to a third-party service provider, such as Ping Identity, TriCipher's Myonelogin.com, or Symplified.com. When federating identities to the cloud, organizations may need to manage the identity life cycle using their IAM system and processes. However, the organization might benefit from an outsourced multiprotocol federation gateway (identity federation service) if it has to interface with many different partners and cloud service federation schemes. For example, as of this writing, Salesforce.com supports SAML

1.1 and Google Apps supports SAML 2.0. Enterprises accessing Google Apps and Salesforce.com may benefit from a multiprotocol federation gateway hosted by an identity management CSP such as Symplified or TriCipher. In cases where credentialing is difficult and costly, an enterprise might also outsource credential issuance (and background investigations) to a service provider, such as the GSA Managed Service Organization (MSO) that issues personal identity verification (PIV) cards and, optionally, the certificates on the cards. The GSA MSO† is offering the USAccess management end-to-end solution as a shared service to federal civilian agencies. In essence, this is a SaaS model for identity management, where the SaaS IdP stores identities in a "trusted identity store" and acts as a proxy for the organization's users accessing cloud services, as illustrated in Figure 5-8.

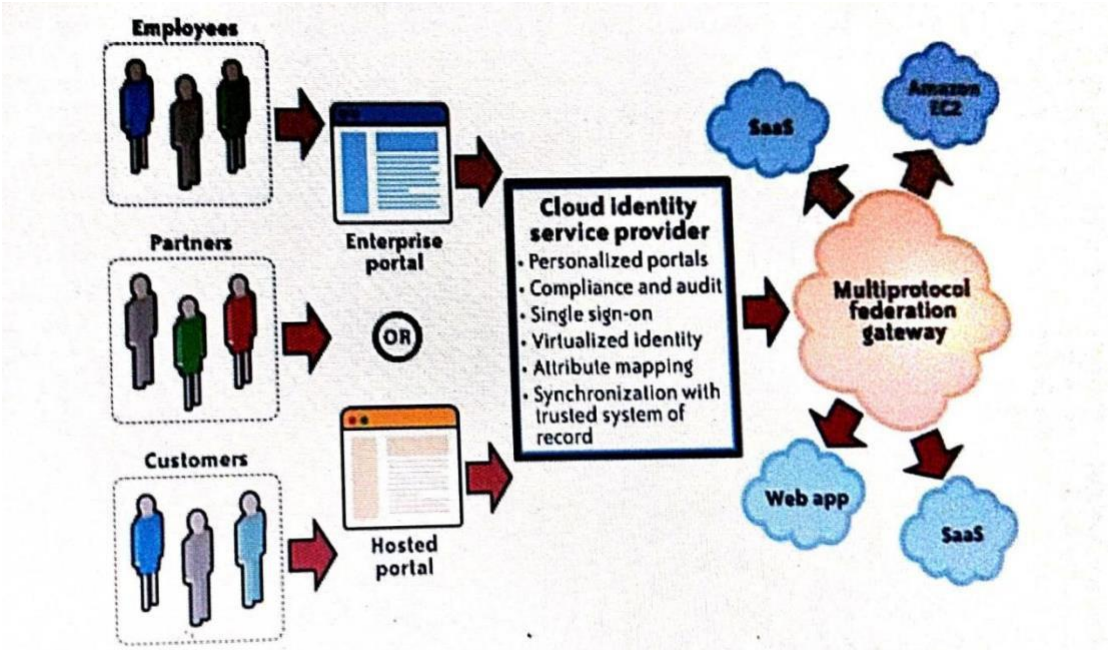


FIGURE 5-8. Identity management-as-a-service (IDaaS)

The identity store in the cloud is kept in sync with the corporate directory through a providerproprietary scheme (e.g., agents running on the customer's premises synchronizing a subset of an organization's identity store to the identity store in the cloud using SSL VPNs). Once the IdP is established in the cloud, the organization should work with the CSP to delegate authentication to the cloud identity service provider. The cloud IdP will authenticate the cloud users prior to them accessing any cloud services (this is done via browser SSO techniques that involve standard HTTP redirection techniques). Here are the specific pros and cons of this approach:

Pros

Delegating certain authentication use cases to the cloud identity management service hides the complexity of integrating with various CSPs supporting different federation standards. Case in point: Salesforce.com and Google support delegated authentication using SAML. However, as of this writing, they support two different versions of SAML: Google Apps supports only SAML 2.0, and Salesforce.com supports only SAML 1.1. Cloudbased identity management services that support both SAML standards (multiprotocol federation gateways) can hide this integration complexity from organizations adopting cloud services. Another benefit is that there is little need for architectural changes to support this model. Once identity synchronization between the organization directory or trusted system of record and the identity service directory in the cloud is set up, users can sign on to cloud services using corporate identity, credentials (both static and dynamic), and authentication policies.

Cons

When you rely on a third party for an identity management service, you may have less visibility into the service, including implementation and architecture details. Hence, the availability and authentication performance of cloud applications hinges on the

identity management service provider's SLA, performance management, and availability. It is important to understand the provider's service level, architecture, service redundancy, and performance guarantees of the identity management service provider. Another drawback to this approach is that it may not be able to generate custom reports to meet internal compliance requirements. In addition, identity attribute management can also become complex when identity attributes are not properly defined and associated with identities (e.g., definitions of attributes, both mandatory and optional). New governance processes may be required to authorize various operations (add/modify/remove attributes) to govern user attributes that move outside the organization's trust boundary. Identity attributes will change through the life cycle of the identity itself and may get out of sync. Although both approaches enable the identification and authentication of users to cloud services, various features and integration nuances are specific to the service delivery model— SaaS, PaaS, and IaaS—as we will discuss in the next section.