UNIT II

Introduction to Virtualization Technology

Virtualization is a technology that allows you to create multiple simulated environments or dedicated resources **from a single physical hardware system**. It works by using software called a **hypervisor** to divide physical resources—such as CPU, memory, storage, or network—into multiple isolated virtual machines (VMs) or virtual components.

Key benefits of virtualization:

- Better utilization of hardware
- Reduced IT costs
- Easier management and deployment
- Improved security through isolation
- Flexibility and scalability

i. Application Virtualization

Definition:
Application virtualization separates an application from the underlying operating system, allowing it to run in a self-contained environment.
Apps are packaged with all required components.
They run in a virtual environment independent of the installed OS.
No need for installation on the host machine.

Benefits:

- Avoids software conflicts
- Easy application deployment and updates
- Improved security (sandboxing)
- Works across different OS versions

ii. Network Virtualization

Definition:
Network virtualization combines hardware and software network resources into a single virtualized network.

How it works:

- Physical networks are divided into multiple virtual networks (VLANs, VPNs).
- Software-defined networking (SDN) allows centralized control.
- Virtual routers, switches, and firewalls can be created.

Benefits:

- Better network efficiency and scalability
- Faster provisioning and reconfiguration
- Improved network security
- Lower operational costs

iii. Desktop Virtualization

Definition:
Desktop virtualization allows users to access a virtual desktop operating system remotely from any device.

How it works:

- The desktop OS (e.g., Windows 10/11) runs on a server.
- Users connect through thin clients, laptops, or mobile devices.
- Virtual Desktop Infrastructure (VDI) delivers centralized desktops.

Benefits:

- Centralized management of desktop OS
- Improved security (data stays on the server)
- Access from anywhere
- Reduced hardware requirements on client devices

iv. Storage Virtualization

Definition:
Storage virtualization pools physical storage from multiple devices into a single storage resource.

How it works:

- A virtualization layer abstracts physical disks.
- Creates logical storage volumes.
- Simplifies backup, recovery, and management.

Benefits:

- Better storage utilization
- Easier data migration
- High availability and reliability
- Simplified management

v. Server Virtualization

Definition:
Server virtualization divides a physical server into multiple isolated **virtual machines (VMs)**, each with its own OS and applications.

How it works:

- A hypervisor (Type 1 or Type 2) allocates resources (CPU, RAM, storage) to VMs.
- Multiple VMs run simultaneously on a single physical server.

Benefits:

- Increases efficiency and reduces hardware costs
- Allows running different OS environments on one machine
- Easier backup, recovery, and scaling
- Isolation improves security and stability

2. Comparison Table:

Virtualization Type	What Is Virtualized?	How It Works	Benefits	Examples
Application	Applications	Runs apps in a sandbox separate from OS	No conflicts, easy updates	VMware ThinApp, App-V
Network	Network resources	Creates virtual switches, routers, VLANs, VPNs	Scalability, security	VMware NSX, Cisco ACI
Desktop	Desktop OS	OS runs on server; accessed remotely	Centralized mgmt, secure	VDI, VMware Horizon
Storage	Storage devices	Pools disks into logical units	Better storage usage	SAN, NAS
Server	Server hardware	Hypervisor creates multiple VMs	Cost-efficient, scalable	Hyper-V, ESXi

Introduction to Load Balancing in Virtualization

Load balancing is the process of distributing incoming network traffic or system workload across multiple servers, virtual machines (VMs), or resources.

In a virtualized environment (like cloud or data centers):

- Many virtual machines run on shared physical hardware.
- Traffic might go to many servers or virtual services.
- Load balancers ensure no single machine gets overloaded.

Why Load Balancing is Important in Virtualization

- Prevents server overload
- Increases application performance
- Improves reliability and uptime
- Enables scalability (more servers/VMs can be added easily)
- Supports failover (if one VM fails, traffic shifts automatically)

i. Software-Based Load Balancers

Definition

Software-based load balancers are applications or programs that run on standard hardware or virtual machines to distribute network traffic.

They are **flexible, cost-effective, and easy to scale**, especially in cloud and virtualized environments.

How They Work

- Installed on a physical or virtual server
- Use algorithms to distribute client requests (e.g., round robin, least connections)
- Can run on cloud platforms like AWS, Azure, GCP or on local servers
- Easily updated and configured through software

Features

- Layer 4 and Layer 7 load balancing

- SSL offloading
- Health checking of backend servers
- Auto-scaling support
- Integration with cloud platforms

Advantages

- Low cost (sometimes free/open-source)
- Highly flexible and configurable
- Runs easily on virtual machines or containers
- Quick updates and patches
- Easy to integrate with automation and DevOps tools

Disadvantages

- Depends on the performance of the underlying hardware
- Might be slower than dedicated hardware appliances under heavy load
- Requires software maintenance

ii. Hardware-Based Load Balancers

Definition

Hardware-based load balancers are **dedicated physical devices** designed specifically for high-performance traffic distribution.
They include specialized processors (ASICs) and networking components optimized for speed and security.

How They Work

- Installed as physical appliances in the network
- Sit between users and servers
- Distribute traffic using built-in optimized hardware
- Provide very high throughput (Gbps to Tbps)
- Include built-in security (firewalls, DDoS protection)

Features

- Extremely high performance (low latency)
- Dedicated chips for SSL encryption/decryption
- Layer 4 and Layer 7 switching
- Advanced security
- Hardware reliability features

Advantages

- Very fast and efficient
- Can handle large enterprise-level traffic
- High reliability
- Lower latency compared to software balancers
- Better built-in security

Disadvantages

- Very expensive
- Harder to scale (requires buying more hardware)
- Less flexible compared to software balancers
- Requires physical installation and maintenance

Software vs Hardware Load Balancers

Feature	Software Load Balancer	Hardware Load Balancer
Cost	Low / Free	Very expensive
Performance	Moderate to High	Extremely high
Scalability	Easy (just add VMs)	Harder (buy more hardware)
Flexibility	Very flexible, easy to modify	Limited flexibility
Deployment	Virtual machines, cloud	Physical appliance
Updates	Easy software updates	Requires hardware refresh
Use Cases	Small–large businesses, cloud	Large enterprises, ISPs, high-traffic apps

Understanding Hypervisors

A **hypervisor** is software that creates and manages **virtual machines (VMs)**.
It allows multiple operating systems (OS) to run on one physical computer by dividing hardware resources such as CPU, memory, and storage.

Hypervisors are also known as:

- Virtual Machine Monitors (VMM)
- Virtualization Managers

Main job:
Allocate resources to VMs
Isolate VMs from each other
Manage and monitor VM performance

There are **two types** of hypervisors:

I. Type 1 Hypervisor (Bare-Metal Hypervisor)

Definition

A Type 1 hypervisor runs **directly on the physical hardware**.
There is **no host operating system** in between.

It acts as the operating system itself for virtual machines.

Advantages

- High performance
- More secure (no host OS layer)
- Better resource management
- Scales well for large systems

Disadvantages

- Requires dedicated hardware
- More complex to set up
- Usually used by professionals / IT admins

II. Type 2 Hypervisor (Hosted Hypervisor)

Definition

A Type 2 hypervisor runs **on top of a host operating system** (like Windows or Linux).
It depends on the host OS for hardware communication.

Mostly used in personal laptops, testing environments, and learning.

Advantages

- Simple installation
- No special hardware required
- Perfect for small-scale or personal use

Disadvantages

- Less efficient than Type 1
- Performance depends on the host OS
- Not ideal for heavy enterprise workloads

Type 1 vs Type 2 Hypervisor – Comparison Table

Feature	Type 1 Hypervisor	Type 2 Hypervisor
Runs on	Hardware directly	Host OS
Performance	Very high	Moderate
Use case	Enterprise, servers	Students, developers
Security	More secure	Less secure
Setup	Complex	Easy

Virtual Machines: Provisioning and Manageability

Virtualization allows multiple virtual machines (VMs) to run on a single physical server.
To use VMs effectively, organizations need efficient **provisioning**, **management**, and **migration** processes.

Virtual Machine Provisioning

Definition

VM provisioning is the process of **creating, configuring, and deploying virtual machines** from scratch or from templates.

It involves allocating:

- CPU
- RAM
- Storage
- Network settings
- Operating system
- Applications

Provisioning Methods

a) Manual Provisioning

- Administrator manually installs OS, drivers, apps
- Slower and used for small environments

b) Template-Based Provisioning

- Pre-configured VM templates (with OS + apps)
- Very fast and common in enterprises
- Ensures consistency

c) Automated Provisioning

- Uses automation tools
- Ideal for cloud environments
- Auto-creates VMs based on demand

Virtual Machine Manageability

Definition

Manageability refers to all activities involved in **monitoring, controlling, and maintaining virtual machines**.

Key Management Tasks

a) Resource Allocation

- Adjusting CPU, RAM, storage
- Adding/removing virtual hardware

b) Monitoring

- Tracking performance (CPU, RAM usage)
- Identifying VM overload or failures

c) Snapshots and Cloning

- Snapshots → capture VM state at a moment
- Cloning → duplicate an existing VM

d) Updates & Patch Management

- Keeping OS and applications updated

e) Security Management

- Setting firewalls
- Managing user access
- Isolating VMs

f) Backup & Recovery

- Creating VM backups
- Restoring in case of disaster

Virtual Machine Migration Services

Definition

VM Migration is the process of **moving a virtual machine from one host (server) to another**—with or without downtime.

Migration helps with:

- Load balancing
- Hardware maintenance
- Fault tolerance
- Power saving
- Performance improvement

Types of VM Migration

a) Live Migration (Zero/Minimal Downtime)

Definition

The VM continues running while it is being moved from one host to another.

Process

- RAM contents are copied
- CPU state transferred
- VM continues running on target host
- Almost no interruption for users

Benefits

- No downtime
- Flexible maintenance
- Better load distribution

b) Cold Migration (VM Powered Off)

Definition

The VM is **shut down**, then moved to another host.

Use Cases

- Hardware upgrades
- Moving large VMs without speed requirement
- Low-risk environment

Benefits

- Simple
- Safe (no running processes)

Disadvantage

- Requires downtime

c) Hot Migration (Running but Paused Briefly)

Definition

The VM is **suspended**, transferred, then resumed.

Characteristics

- Very short downtime
- Faster than cold migration
- Slower than live migration

1. Provisioning in the Cloud Context

Provisioning in cloud computing refers to the process of preparing and equipping a virtual environment so users can run applications, store data, or use services.
It is essentially about **allocating necessary resources** (compute, storage, network) to customers or virtual machines.

Types of Cloud Provisioning

1. **Manual Provisioning**
    - Administrator manually creates instances, storage, network.
    - Used in small environments.
2. **Automated Provisioning**
    - Uses scripts, templates, or orchestration tools.
    - Common in public clouds (AWS, Azure, GCP).
    - Examples: AWS CloudFormation, Terraform.
3. **Self-Service Provisioning**
    - Users can provision VMs, storage, and networks using a portal.
    - Key feature of cloud computing.
    - Enables rapid scaling.
4. **Dynamic/Auto Provisioning**
    - Cloud automatically increases or decreases resources.
    - Supports elasticity.
    - Uses auto-scaling groups, load monitoring.

Virtualization of CPU

CPU Virtualization allows multiple virtual machines to share a physical CPU as if each VM had its own processor.

A **hypervisor** sits between hardware and VMs.
It **allocates CPU cycles** to each VM using scheduling algorithms.
Creates **virtual CPUs (vCPUs)** that map to physical CPU cores.

Key Techniques

1. **Time Sharing**

    - CPU time is divided among VMs.
    - Each VM gets a time slice.
2. **CPU Scheduling**
    - Hypervisor uses algorithms (round robin, fair share) to allocate CPU time.
    - Ensures no VM starves.
3. **Hardware-Assisted CPU Virtualization**
    - Modern CPUs support virtualization features:
        - Intel VT-x
        - AMD-V
    - Improves performance and security.

Benefits

- Efficient usage of CPU.
- Multiple OS can run on same hardware.
- Supports workload isolation.

Virtualization of Memory

Memory virtualization allows each VM to believe it has access to its own continuous memory space, independent of the physical RAM.

Hypervisor creates **virtual memory** for each VM.
Maps **guest virtual memory → guest physical memory → host physical memory**.
Uses page tables to manage mappings.

Techniques Used

1. **Paging**
    - Splits memory into fixed-size pages.
    - Enables isolation between VMs.
2. **Shadow Page Tables**
    - Hypervisor keeps mapping tables for VM memory translation.
3. **Hardware-Assisted Memory Virtualization**
    - Intel EPT (Extended Page Tables)
    - AMD RVI (Rapid Virtualization Indexing)
    - Improves memory translation speeds.
4. **Memory Overcommitment**
    - More virtual memory is allocated than physical RAM.
    - Works using:
        - **Ballooning**
        - **Swapping**
        - **Compression**

Benefits

- VMs use memory efficiently.
- Supports multiple VMs on limited RAM.
- Provides isolation & security.

Virtualization of I/O Devices

I/O Virtualization allows VMs to share underlying hardware devices like:

- Network Interface Cards (NICs)
- Storage controllers
- USB devices
- Disk I/O

How It Works

Hypervisor creates **virtual devices** for each VM:

- Virtual NIC
- Virtual disk
- Virtual GPU (in some cases)

When a VM performs an I/O operation:

1. The virtual device receives the request.
2. Hypervisor translates it to physical device instructions.
3. Sends data to/from the actual hardware.

Techniques

1. **Emulation**
    - Hardware device is fully emulated in software.
    - Slower but flexible.
2. **Paravirtualization**
    - VM uses drivers optimized for virtualized I/O.
    - Faster than emulation.
    - Example: VirtIO drivers in KVM.
3. **Direct Device Assignment (Passthrough)**
    - VM gets direct access to physical device.
    - Uses IOMMU (Intel VT-d, AMD-Vi).
    - Used for GPU, NICs requiring high performance.

Benefits

- Reliable sharing of hardware devices.
- High-performance I/O for demanding applications.

- Strong isolation and security.

Concept	Key Idea	Why It Matters
Cloud Provisioning	Allocating cloud resources	Enables elasticity & automation
CPU Virtualization	Virtual CPUs mapped to physical cores	Runs many VMs efficiently
Memory Virtualization	Virtual memory mapping & isolation	Enables overcommitment & security
I/O Virtualization	Virtual devices + hypervisor mediation	Efficient hardware sharing

1. Virtual Clusters

Definition

A **virtual cluster** is a group of virtual machines (VMs) that work together as a cluster, similar to a physical cluster, but built using **virtualization technology**.

In a virtual cluster:

- VMs act as cluster nodes.
- All VMs are connected through a virtual network.
- VMs can run on one or multiple physical servers.
- Resources (CPU, memory, storage, network) are allocated dynamically.

Why Virtual Clusters Are Used

1. **Cost-Efficiency** – No need for dedicated physical servers for each node.
2. **Scalability** – Add or remove VMs easily.
3. **Flexibility** – Create clusters for different workloads (web servers, databases, Hadoop, etc.).

4. **Isolation** – Each VM/node is isolated; failure in one does not affect others.
5. **Rapid Deployment** – New cluster members can be created using templates.

Architecture of Virtual Clusters

A typical virtual cluster includes:

1. **Physical Host Layer**
   - Servers running a hypervisor (VMware ESXi, KVM, Hyper-V).
2. **Virtualization Layer**
   - Hypervisor creates VMs.
   - Virtual network connects VMs internally.
3. **Cluster Layer**
   - VMs run clustering software such as:
     - Kubernetes clusters
     - Hadoop clusters
     - High-availability clusters
4. **Applications**
   - Distributed applications running on cluster nodes.

Types of Virtual Clusters

1. Dedicated Virtual Clusters

Each VM in the cluster is used only for that particular cluster.

2. Non-Dedicated Virtual Clusters

VMs share hardware with other VMs not part of the cluster.

3. Multi-Cluster Virtual Environments

Multiple virtual clusters run on shared hardware resources.

Advantages of Virtual Clusters

- Easier management
- Fault isolation
- Resource pooling
- Quick recovery (snapshot/restore)
- Supports distributed computing systems

2. Resource Management in Virtualized Environments

Resource management ensures that VMs and virtual clusters receive the **right amount of compute, memory, storage, and network** resources.

Key Resource Management Functions

1. Resource Allocation

- Assigning CPU, memory, and storage to VMs.
- Hypervisor allocates resources using:
  - CPU scheduling
  - Memory ballooning
  - Storage provisioning
  - I/O prioritization

2. Resource Monitoring

- Tracking usage: CPU load, RAM usage, disk I/O, network traffic.
- Tools: VMware vCenter, OpenStack, CloudWatch, Prometheus.

3. Resource Optimization

Includes:

- **Load balancing**
- **Dynamic allocation**
- **Auto-scaling**
- **Live migration** (moving VM to another host without shutdown)

4. Resource Isolation

Ensures one VM cannot affect another. Achieved by:

- CPU scheduling limits
- Memory reservation
- Network isolation (VLANs)
- Storage I/O throttling

Resource Management Techniques

1. Overprovisioning

Allocating more virtual resources than physically available (e.g., 20 vCPUs on a 12-core server).
Benefit: Higher utilization
Risk: Performance issues if all VMs become busy.

2. Underprovisioning

Allocating fewer resources than VMs need to reduce hardware load.
Used for low-priority workloads.

3. Dynamic Resource Scheduling

Resources are adjusted automatically based on workload.

Examples:

- VMware DRS
- Kubernetes auto-scaler

4. VM Migration

Moving a VM from one host to another.

- **Live Migration** – No downtime.
- **Cold Migration** – VM is powered off.

Used for:

- Load balancing
- Maintenance
- Energy saving

5. Admission Control

Determines whether new VMs can be created based on available resources.
Prevents overloading the host.

6. Resource Quotas and Limits

- **Quota** → Minimum resources guaranteed
- **Limit** → Maximum resources allowed
  Used to ensure fairness among VMs.

How Virtual Clusters Work With Resource Management

Virtual Cluster Task	Resource Management Role
Adding more nodes	Automatically allocates more CPU/RAM to VMs
Workload balancing	VMs migrated to less-loaded hosts
High availability	Hypervisor restarts VMs on another host
Scaling applications	Auto-provisioning new VMs
Ensuring performance	CPU/memory limits and reservations

Question Set:-

1.Define Virtualization?

2. Explain types of Virtualization?

3. Define Load Balancer?

4. Explain types of Load Balancer?

5.Define types of Hyperviser?

6.Define Virtual Cluster?

7.Define Virtual Machine Provisioning?

8. Virtual Machine Manageability?

9.Define types of Virtual Migration ?

10.Define Virtualization of Memory?

11.Define C.P.U Virtualization?

12. Explain types of Virtualization?