Great Place To Work®
Certified
FEB 2023 – FEB 2024
INDIA

UPES

UNIVERSITY OF TOMORROW

nirf RANKING 2023

#RANKED 52 IN INDIA

#University Category

QS WORLD UNIVERSITY RANKINGS

NO.1 PVT. UNIVERSITY IN ACADEMIC REPUTATION IN INDIA

ACCREDITED WITH GRADE A NAAC

ACCREDITED **GRADE 'A'** BY NAAC

E-LEAD

PERFECT SCORE OF **150/150** AS A TESTAMENT TO EXCEPTIONAL E-LEARNING METHODS

Unit 1 : Introduction to Operating System

Lecture 4

Dr. Hemant Petwal

School of Computer Science

UPES, Dehradun

India

Table of Contents

Learning & Course Outcomes

Learning Outcomes

LO1: Understand the concept of virtualization and Containerization in Operating System

LO2: Understand Operating system structures

Course Outcomes

CO1: Demonstrate a comprehensive understanding of operating systems

Virtualization and Containerization

Virtualization

Virtualization is the process of creating a virtual version of a physical machine or resource, such as a computer, server, storage device, or network. It allows multiple virtual instances to run on a single physical system, enabling better resource utilization, flexibility, and isolation

In other words:

Virtualization in operating systems (OS) is a technology that creates a virtual version of a computing resource, such as hardware platforms, storage devices, or network resources. It enables the creation of multiple simulated environments or dedicated resources from a single physical hardware system.

Virtualization: Key Concepts

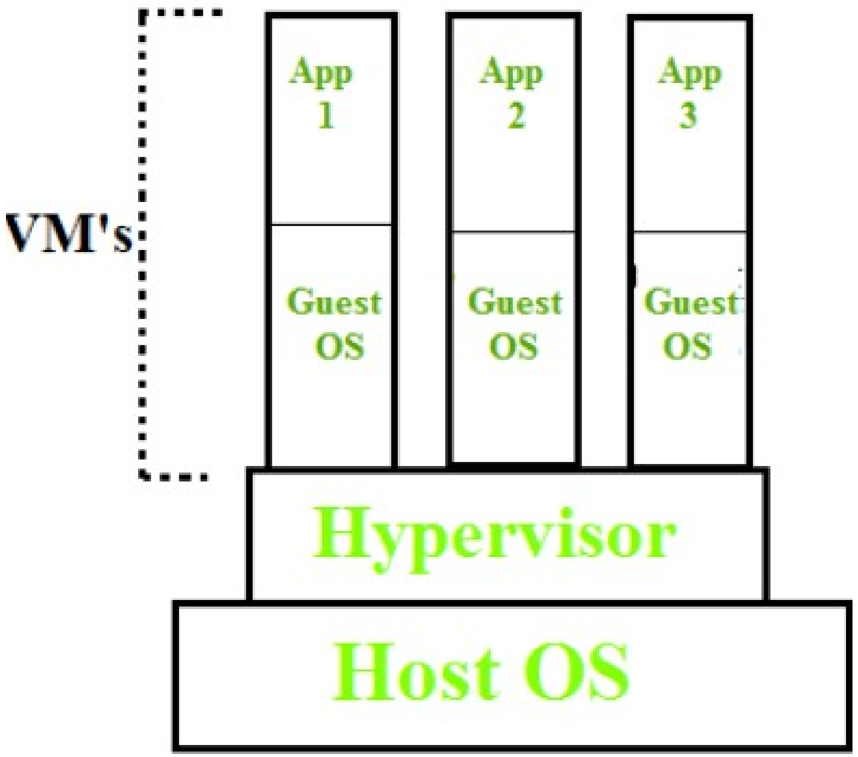

Fig 1. Virtualization

1. Virtual Machines (VMs)
A virtual machine is an emulation of a computer system. Each VM runs its own operating system and applications, functioning as if it were a separate physical machine.

2. Hypervisor
The hypervisor, also known as a virtual machine monitor (VMM), is the software layer that enables virtualization. It manages the hardware resources and ensures that VMs are isolated from each other while sharing the same physical hardware.

There are two types of hypervisors:
Type 1 (Bare-metal): Runs directly on the physical hardware (e.g., **VMware ESXi**, **Microsoft Hyper-V**).
Type 2 (Hosted): Runs on top of a host operating system (e.g., **VMware Workstation**, **Oracle VM VirtualBox**).

UPES
UNIVERSITY OF TOMORROW

Virtualization: Key Concepts

Isolation

Virtualization provides strong isolation between VMs. Each VM operates independently and securely, ensuring that issues in one VM do not affect others.

Resource Management

Virtualization allows for efficient resource allocation and management. Resources such as CPU, memory, and storage can be dynamically allocated to different VMs based on demand.

Virtualization: Types

Server Virtualization It Divides a physical server into multiple virtual servers, each running its own operating system and applications.

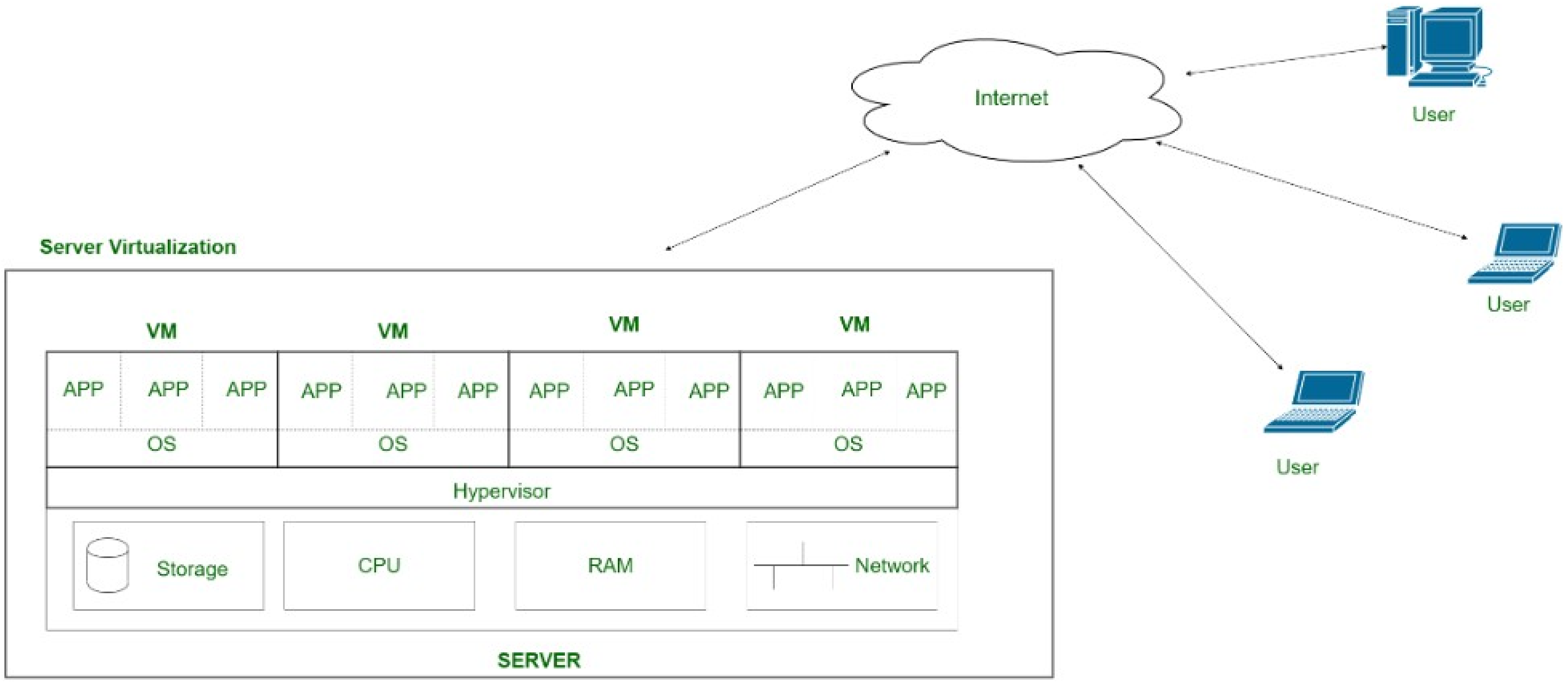

Fig 2. Server Virtualization

Virtualization: Types

Desktop Virtualization

It Provides virtual desktops to end-users, which they can access from various devices. Examples include Virtual Desktop Infrastructure (VDI) solutions.

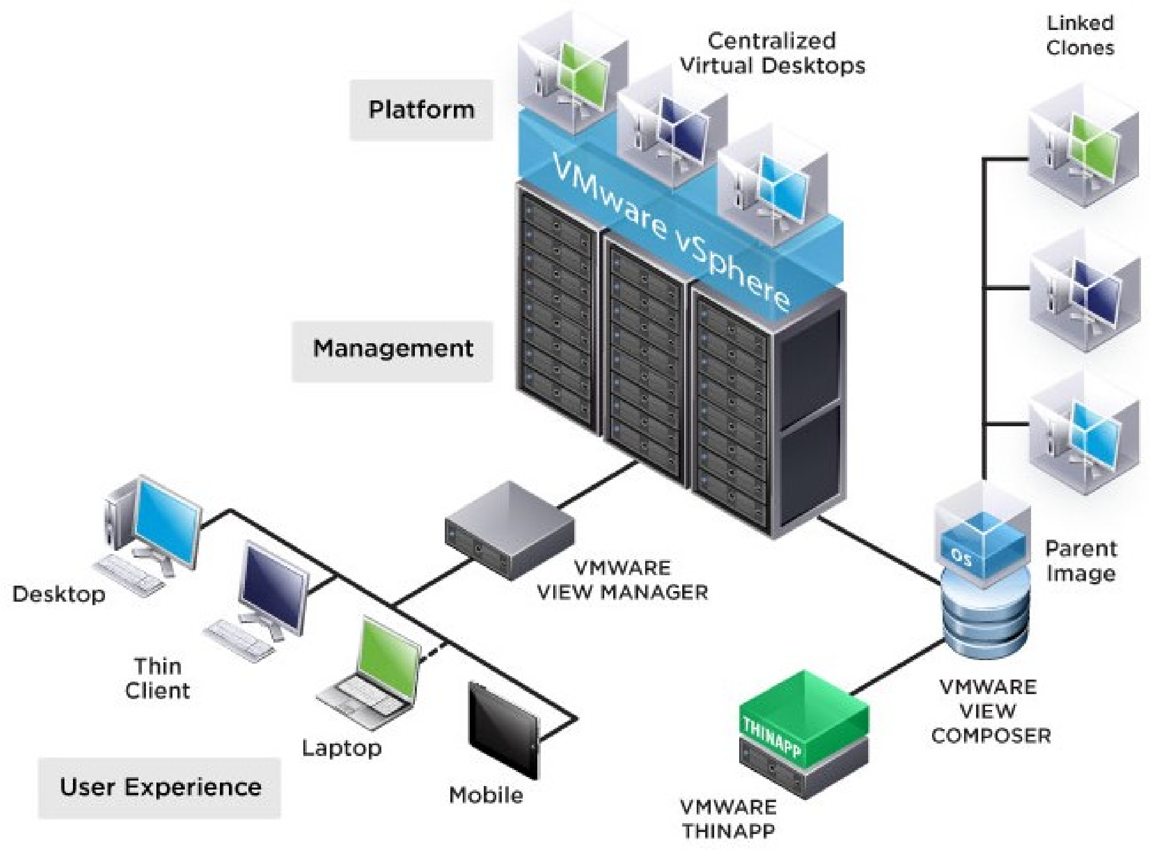

Fig 3. Desktop Virtualization

UPES
UNIVERSITY OF TOMORROW

Virtualization: Types

Storage Virtualization

It Abstracts physical storage resources to create a single storage pool that can be managed and allocated efficiently.

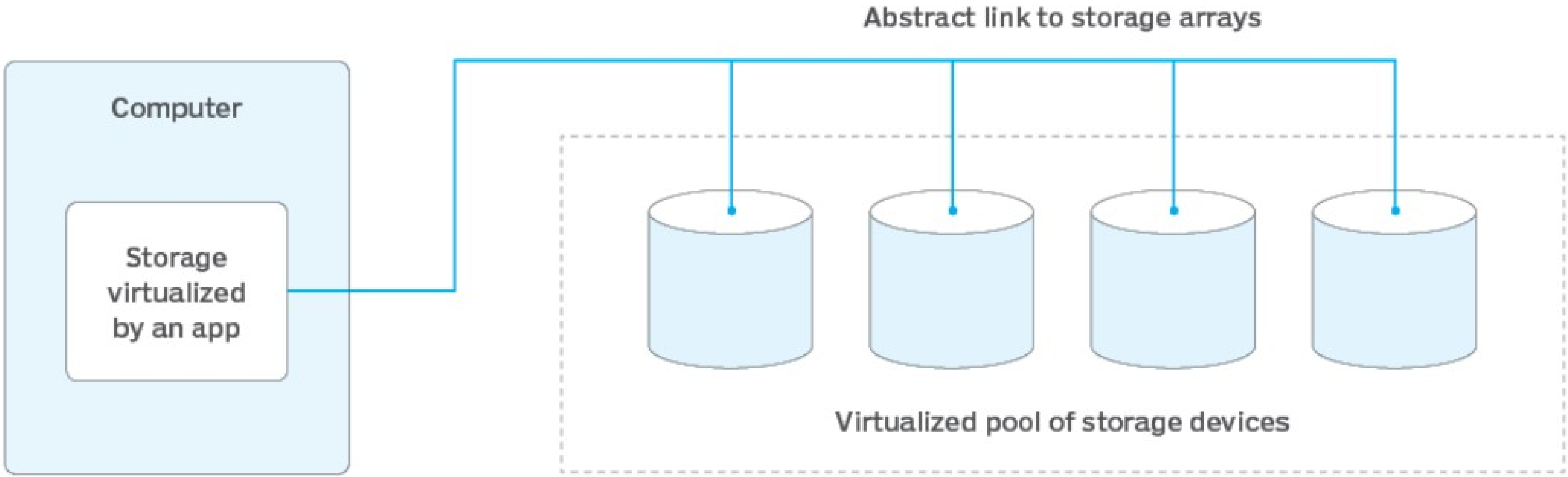

Fig 4. Storage Virtualization

Virtualization: Types

Network Virtualization

It Combines hardware and software network resources into a single, software-based administrative entity.

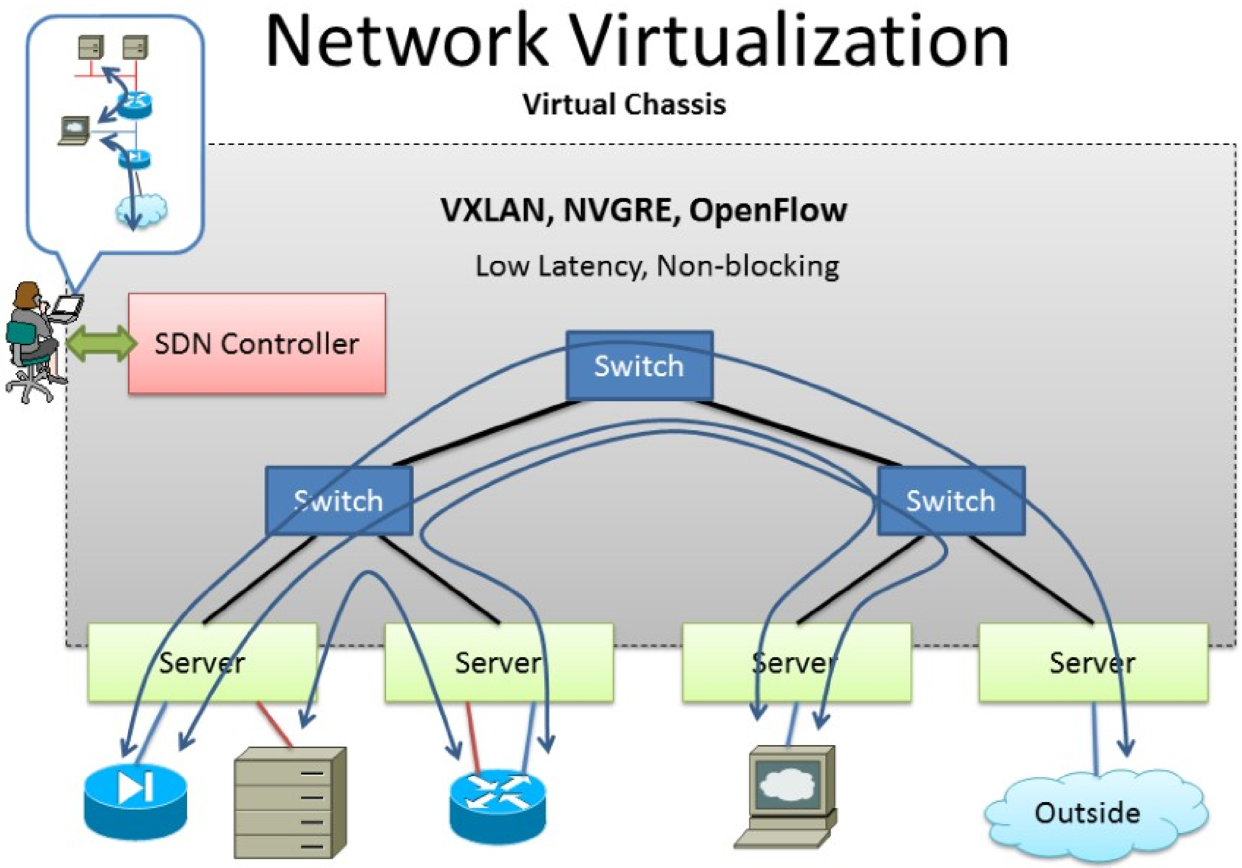

Fig 5. Network Virtualization

Containerization

Containerization

Containerization in operating systems (OS) involves encapsulating an application and its dependencies into a container, ensuring it runs consistently across different environments.

Containerization was introduced to address certain limitations and inefficiencies associated with traditional virtualization. While virtualization provided significant benefits over physical hardware by allowing multiple operating systems to run on a single physical machine, containerization introduced a more efficient and streamlined approach to application deployment and management

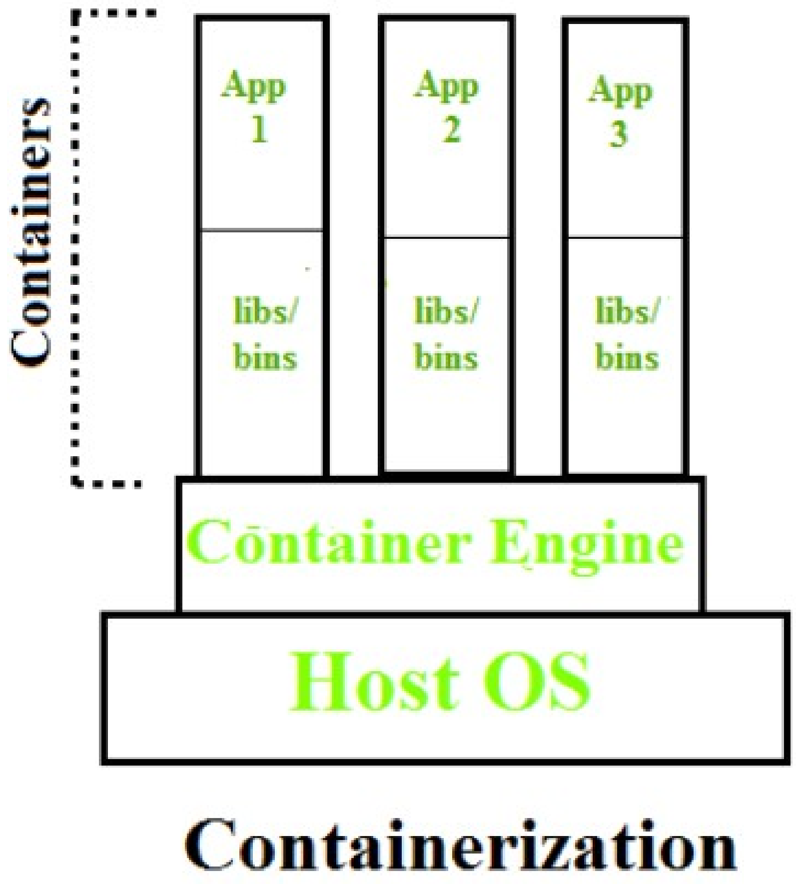

Fig 6. Containerization

Virtualization vs Containerization

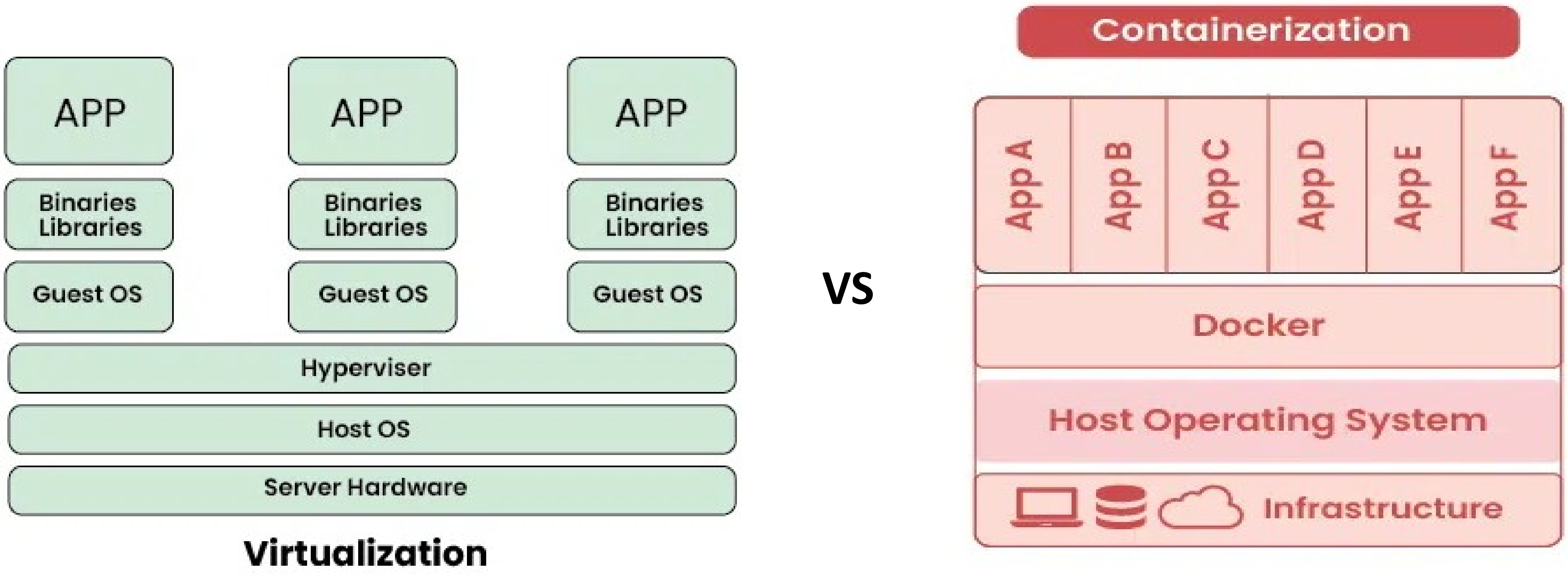

Fig 7. Architectural Difference between Virtualization and Containerization

Virtualization vs Containerization

Feature	Virtualization	Containerization
Isolation	Each VM runs its own OS, providing strong isolation from the host and other VMs.	Containers share the host OS kernel, providing lightweight isolation, but not as strong as VMs.
Resource Usage	VMs require a separate OS instance and consume more resources (CPU, memory, storage).	Containers share the host OS kernel, resulting in more efficient resource utilization.
Performance	Slightly slower performance due to overhead of hypervisor and separate OS instances.	Generally faster performance due to less overhead, as containers leverage the host OS directly.
Portability	VMs can be migrated between different hypervisors and platforms, but with some effort.	Containers are highly portable and can run consistently across different environments (development, testing, production).
Scalability	Scalability is limited by the available resources on the host machine.	Highly scalable as containers are lightweight and can be spun up or down quickly to match demand.
Overhead	Higher overhead due to running multiple OS instances.	Lower overhead as containers share the host OS kernel.
Examples	VMware, VirtualBox, Hyper-V.	Docker, Podman.

UPES
UNIVERSITY OF TOMORROW

Virtualization vs Containerization

Best Scenarios for Virtualization and Containerization

Scenario	Containerization	Virtualization
Running legacy applications	✗	✓
Isolating applications from one another	✗	✓
Building microservices architecture	✓	✗
Rapid development and deployment (CI/CD)	✓	✗
Maximizing resource utilization	✓	✗
Lifting and shifting existing applications to the cloud	✗	✓

Fig 8. Scenarios for Containerization and Virtualization

UPES
UNIVERSITY OF TOMORROW

Operating System Structures

- The operating system can be implemented with the help of various structures.
- The structure of the OS depends mainly on how the various standard components of the operating system are interconnected and melded into the kernel.
- A design known as an operating system enables user application programs to communicate with the machine’s hardware.

Types of structures in Operating systems

- Simple/Monolithic Structure
- Micro-Kernel Structure
- Hybrid-Kernel Structure
- Exo-Kernel Structure
- Layered Structure
- Client Server Model
- Virtual Machines

Operating System Structures: Types

Simple/Monolithic Structure

A unified operating system architecture where the kernel provides all essential services directly to user applications

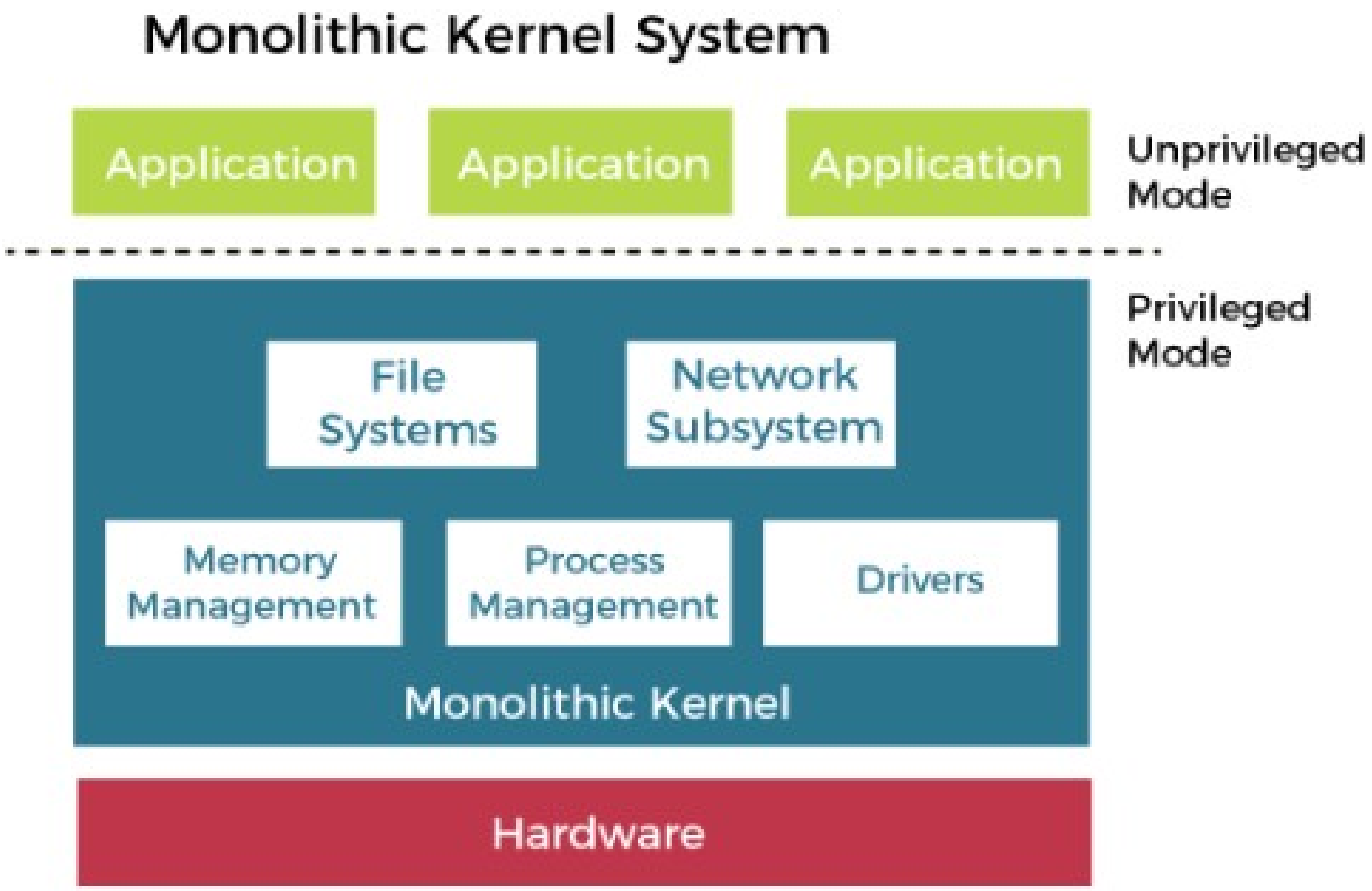

Fig 9. Simple/Monolithic OS Structure

Operating System Structures: Types

Micro-Kernel Structure

A minimalistic operating system architecture where the kernel provides only essential services, and additional functionalities are implemented as user-space processes.

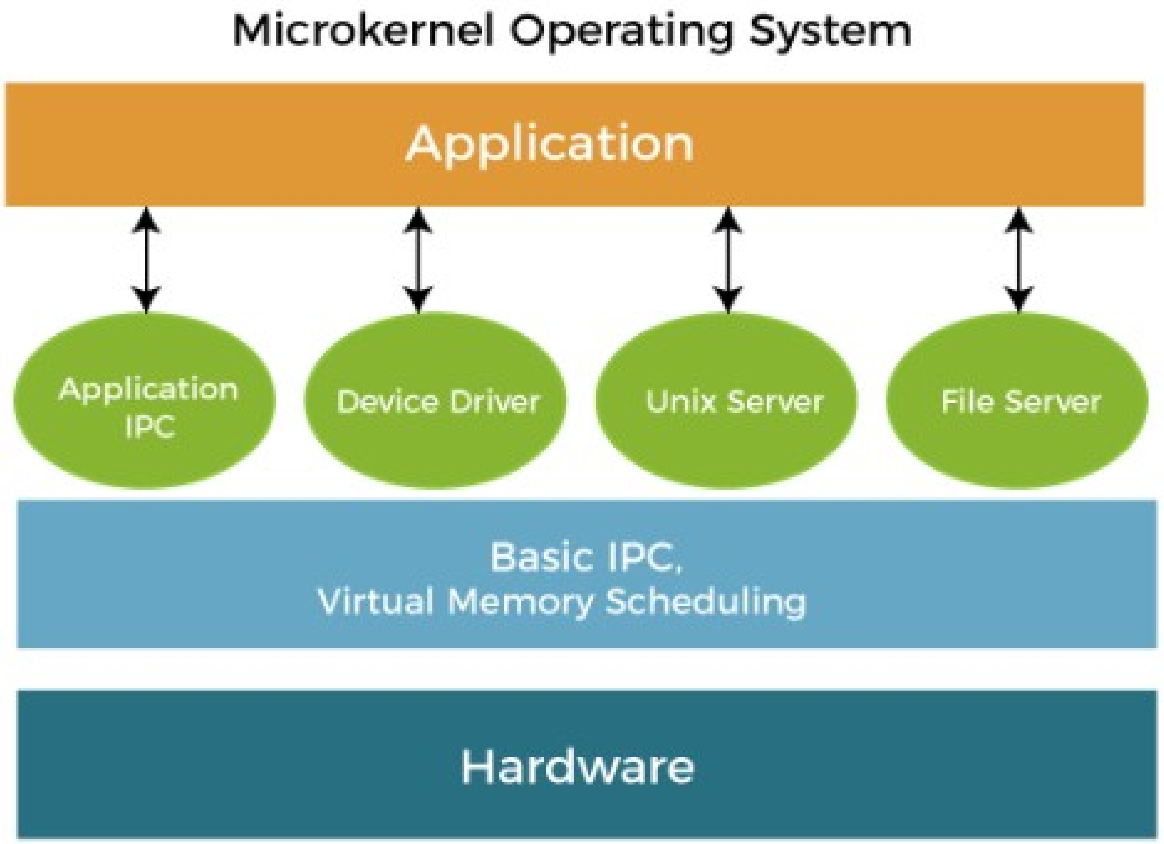

Fig 10. Micro-Kernal Structure

Operating System Structures: Types

Hybrid-Kernel Structure

An operating system architecture combining aspects of both monolithic and microkernel designs, offering a balance between performance and flexibility.

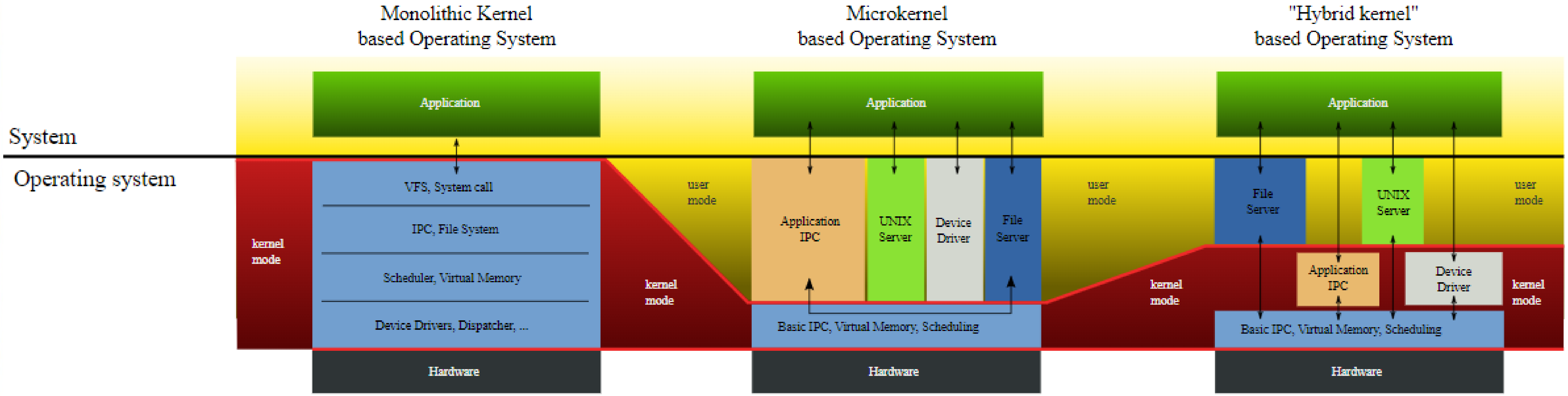

Fig 11. Hybrid-Kernal Structure

Operating System Structures: Types

Exo-Kernel Structure

An operating system architecture that exposes hardware resources directly to applications, allowing greater flexibility but requiring applications to manage resources explicitly.

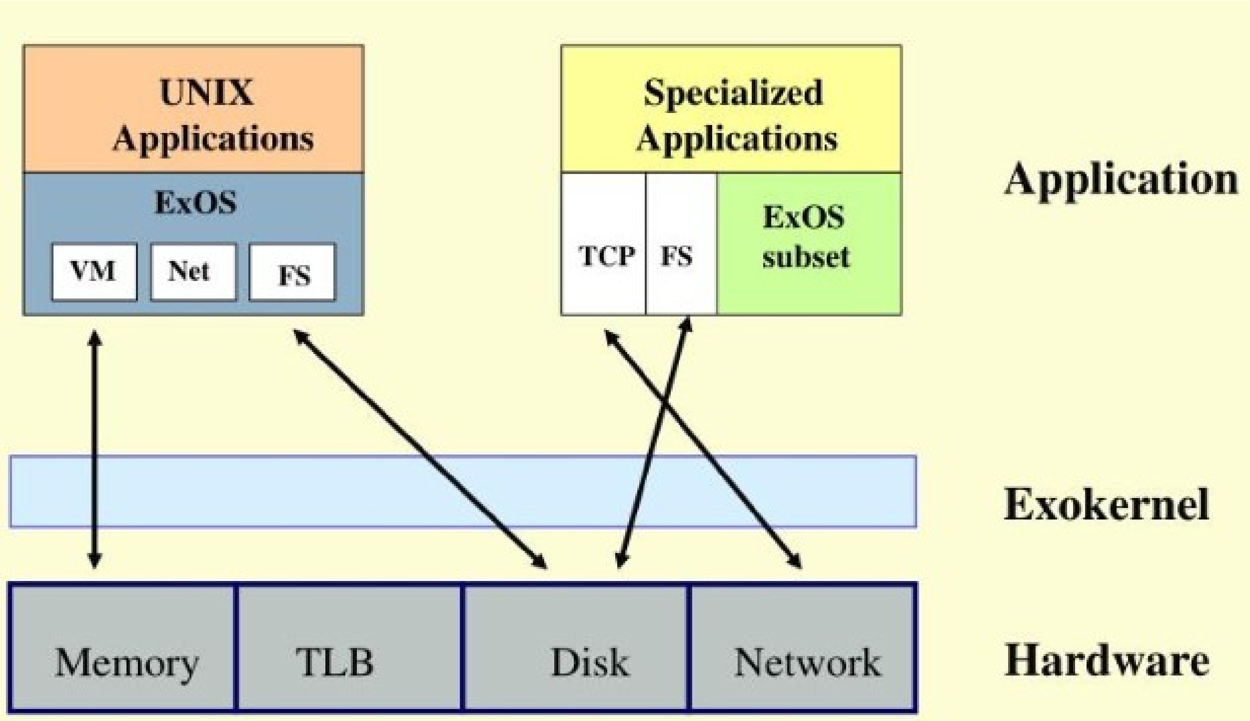

Fig 12. Exo-Kernal Structure

Operating System Structures: Types

Layered Structure

An operating system structure organized into layers, with each layer providing a set of services to the layer above, facilitating modular design and ease of maintenance.

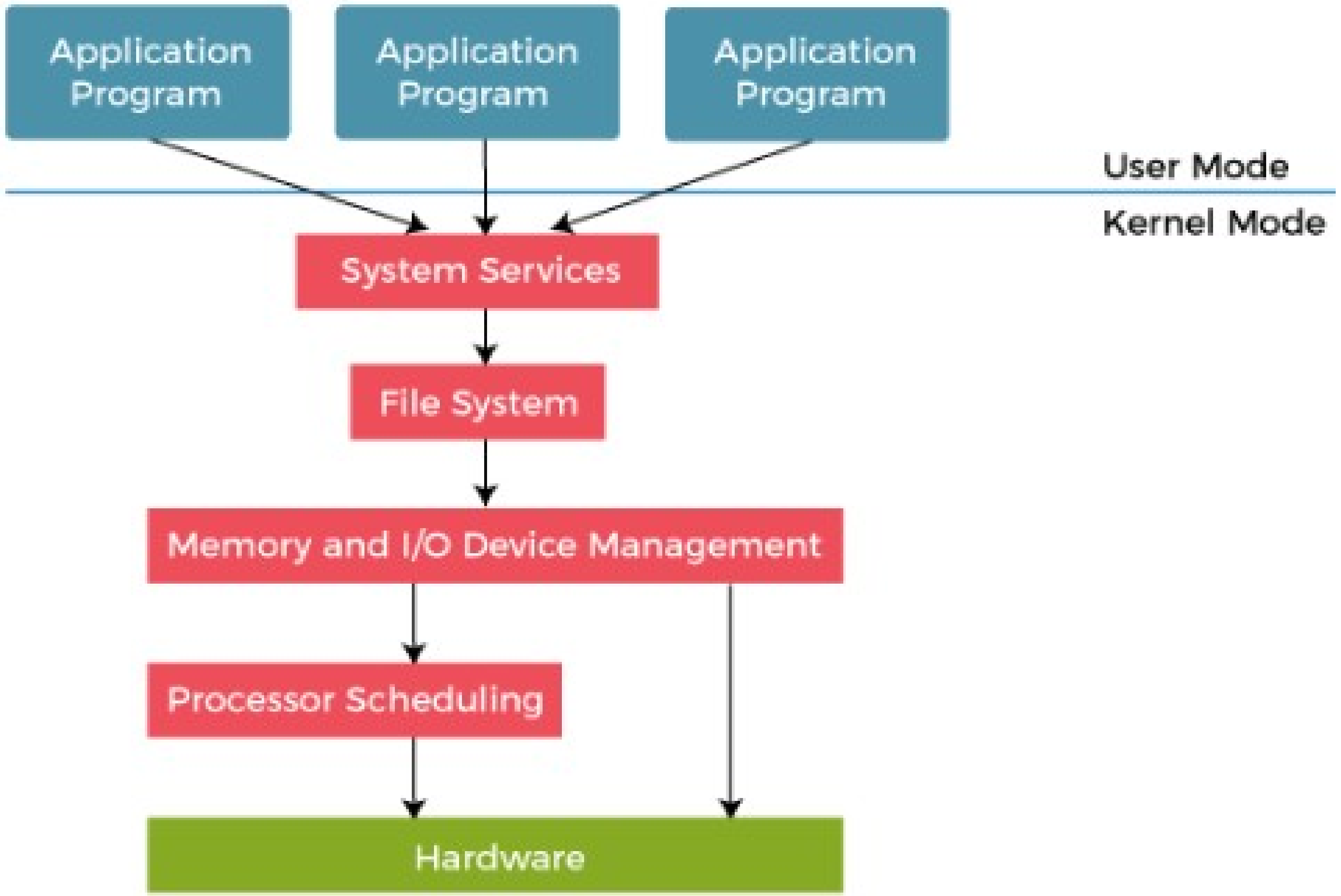

Fig 13. Layered Structure

Operating System Structures: Types

Client-Server Model

An operating system structure where services are divided into client and server components, with servers providing specific functionalities to client applications upon request.

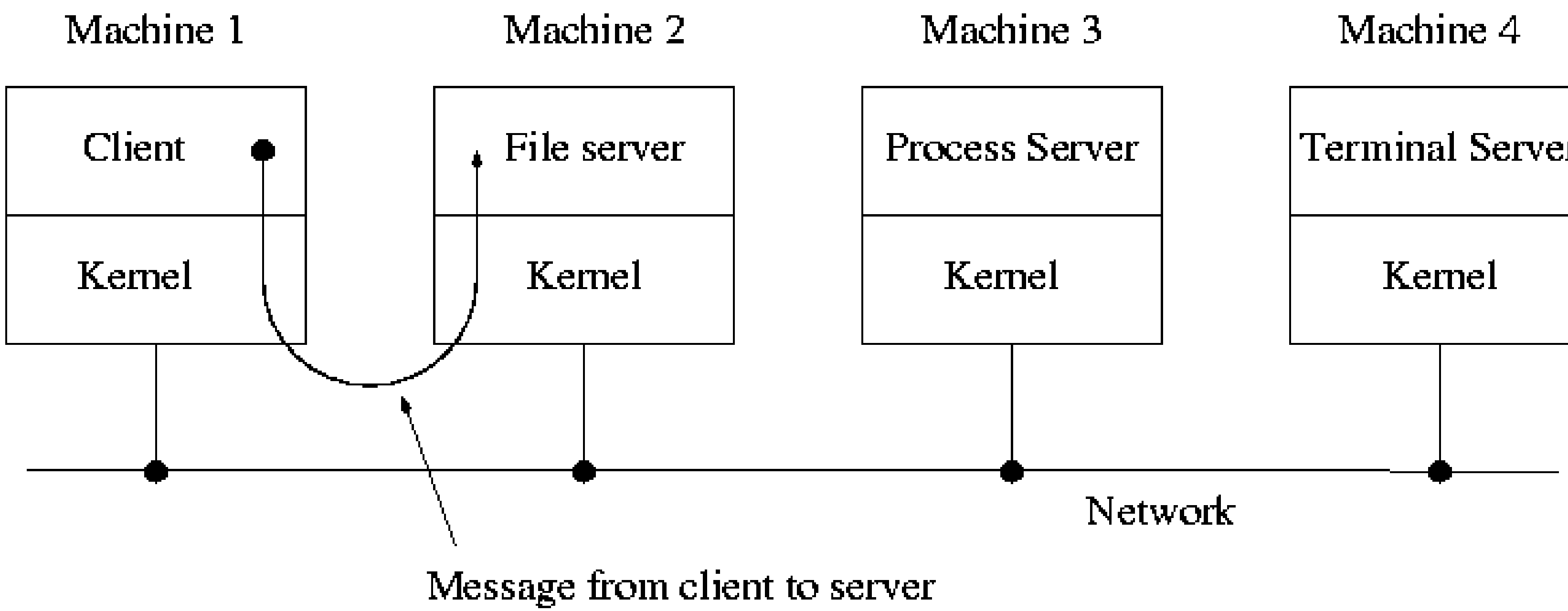

Fig 14. Client-Server Structure

Operating System Structures: Types

Virtual Machines

A layer of software that enables multiple operating systems to run concurrently on the same hardware by virtualizing resources and providing isolation between guest operating systems.

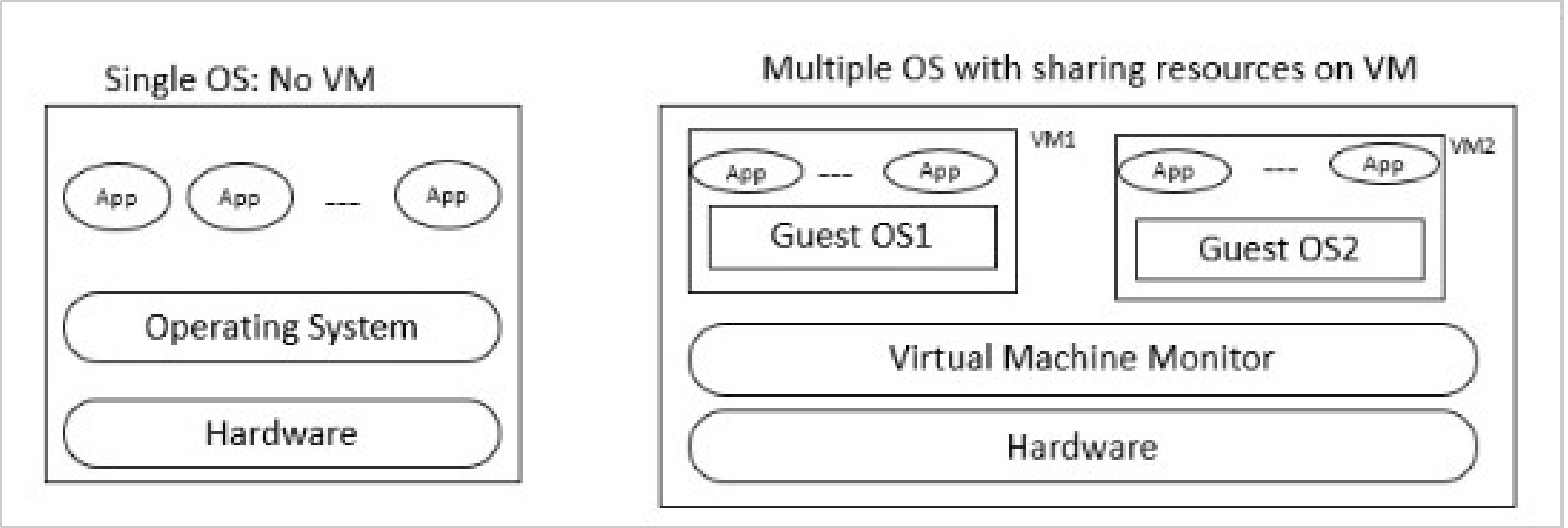

Fig 15. Virtual Machine Structure

UPES
UNIVERSITY OF TOMORROW

MCQ

Q1. What role does the hypervisor play in virtualization?

A. Managing network connections
B. Controlling user access
C. Enabling and managing virtual machines
D. Monitoring internet traffic

Q2. Which type of hypervisor runs directly on the physical hardware?

A. Type 1 (Bare-metal)
B. Type 2 (Hosted)
C. Type 3 (Network-based)
D. Type 4 (Cloud-based)

Q3. What is the primary purpose of server virtualization?

A. Divide a physical server into multiple virtual servers
B. To provide virtual desktops to end-users
C. To abstract physical storage resources
D. To combine hardware software network resources

Q4. Which offers better performance due to less overhead?

A. Virtualization
B. Containerization
C. Both have equal performance
D. Depends on the hardware

MCQ

Q5. What is containerization in operating systems?

A. Creating virtual networks

B. Encapsulating an application and dependencies into a container

C. Dividing physical storage into virtual pools

D. Combining multiple virtual machines into one

Q6. Which has higher overhead due to running multiple OS instances?

A. Virtualization

B. Containerization

C. Both have equal overhead

D. Neither has overhead

Q7. What does the structure of an operating system depend on?

A. The type of applications it runs

B. How its components are interconnected and integrated into the kernel

C. The speed of the underlying hardware

D. The number of users it supports

UPES

UNIVERSITY OF TOMORROW

MCQ

Question No-Answer	Option	Description
Q1- Answer	C	Enabling and managing virtual machines
Q2- Answer	A	Type 1 (Bare-metal)
Q3- Answer	A	Divide a physical server into multiple virtual servers
Q4- Answer	B	Containerization
Q5- Answer	B	Encapsulating an application and dependencies into a container
Q6- Answer	A	Virtualization
Q7- Answer	B	How its components are interconnected and integrated into the kernel

UPES
UNIVERSITY OF TOMORROW

Summary/Key Points

- Virtualization is the process of creating a virtual version of a physical machine or resource, such as a computer, server, storage device, or network.
- The major components associated with virtualization are virtual machines and hypervisor
- Hypervisor is of two types: Type 1 (Bare-Metal hypervisor), Type 2 (Hosted hypervisor)
- Virtualization is categorized into following categories (types): Sever, Desktop, Network, Storage virtualization
- Containerization was introduced to address certain limitations and inefficiencies associated with traditional virtualization
- The operating system can be implemented with the help of various structures e.g. Simple/Monolithic Structure, Micro-Kernel Structure, Hybrid-Kernel Structure, Exo-Kernel Structure, Layered Structure, Client-Server Model, Virtual Machines

UPES
UNIVERSITY OF TOMORROW

Reference Material

- Operating Systems Concepts (10th Ed.) Silberschatz A, Peterson J and Galvin P, John Wiley & Sons, Inc. 2018.

  Topics:
  - Virtualization and Containerization, Page No: 34
  - Operating System Structures, Page No: 81-91

- Modern Operating Systems (4th Ed.) by Andrew S. Tanenbaum and Herbert Bos, 2007
- Operating Systems: Principles and Practice by Thomas Anderson and Michael Dahlin, 2014

UPES
UNIVERSITY OF TOMORROW

Coming Up-Next Lecture

- Various operations of Operating System

Thank You