Virtualization

Module 3 – Part A

How do Cloud Services Work?

- Assume a cloud service provider has a datacenter with 4 CPUs and 8 GB RAM
  - Alice wants a system with 1 CPU and 2 GB RAM
  - Bob wants a system with 2 CPUs and 2 GB RAM
  - Carol wants a system with 1 CPU and 4 GB RAM
- In a traditional IT setup, this would be impossible!
- ***Solution:*** Create **virtual machines** with the required specifications and provide to the customers
- This can be done by using a technology known as **virtualization**

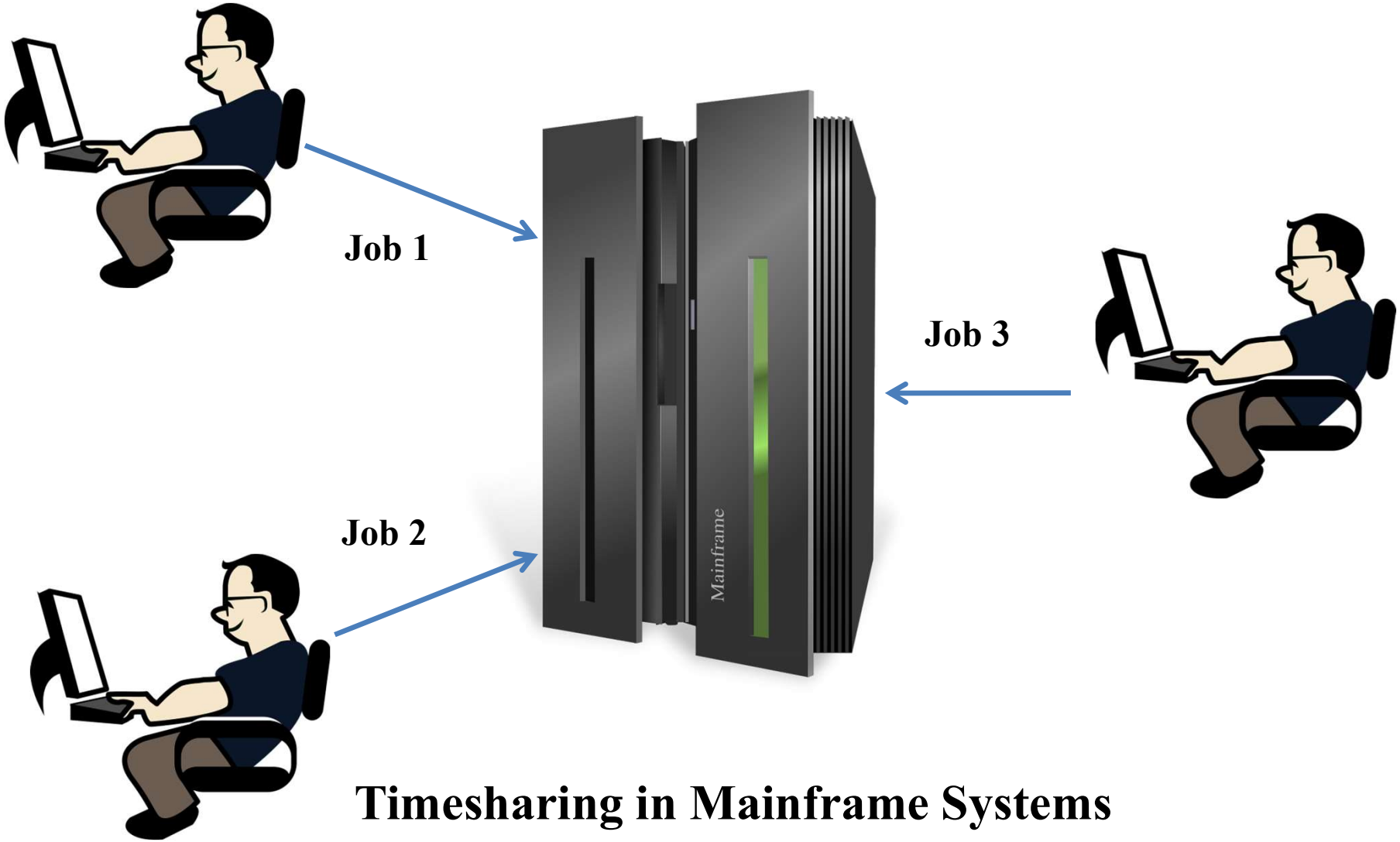
Contd…

- Put in simple terms, virtualization means

  creating an illusion of something which is not actually present

- Virtualization is used very commonly nowadays
  - **Virtual memory** gives us the illusion of a significantly larger memory than we physically have
  - **Virtual Reality** games allow users to perceive a world that doesn't physically exist

A Brief History of Virtualization

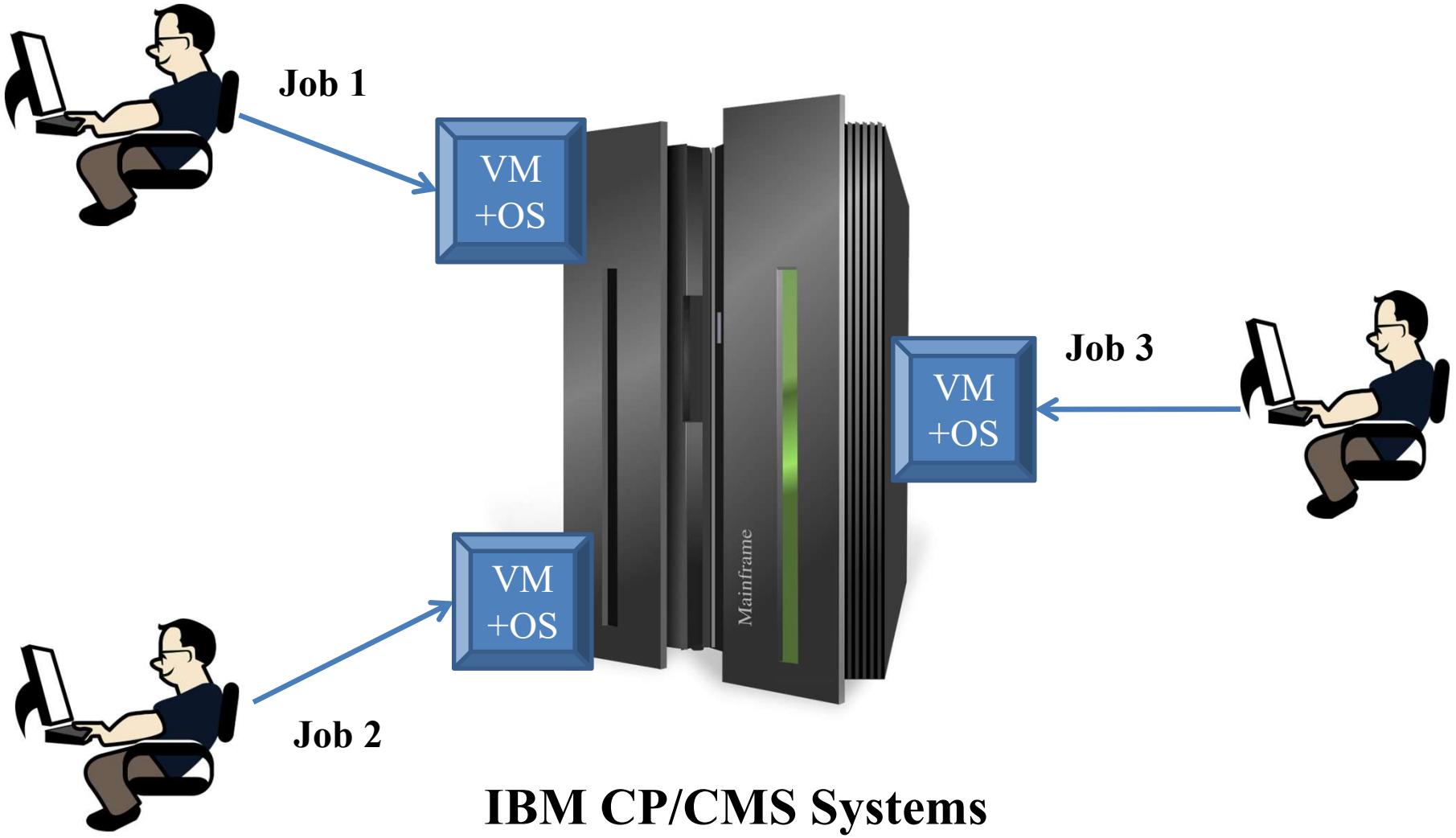


Timesharing in Mainframe Systems

A Brief History of Virtualization

- **Timesharing in Mainframes**
  - Support multiple users through terminals
  - When users block for I/O, system executes jobs from other users
  - System still executes only one job at a time
  - Creates an illusion of multiple jobs being processed at the same time
  - Later, a time quantum was introduced to increase server utilization

A Brief History of Virtualization

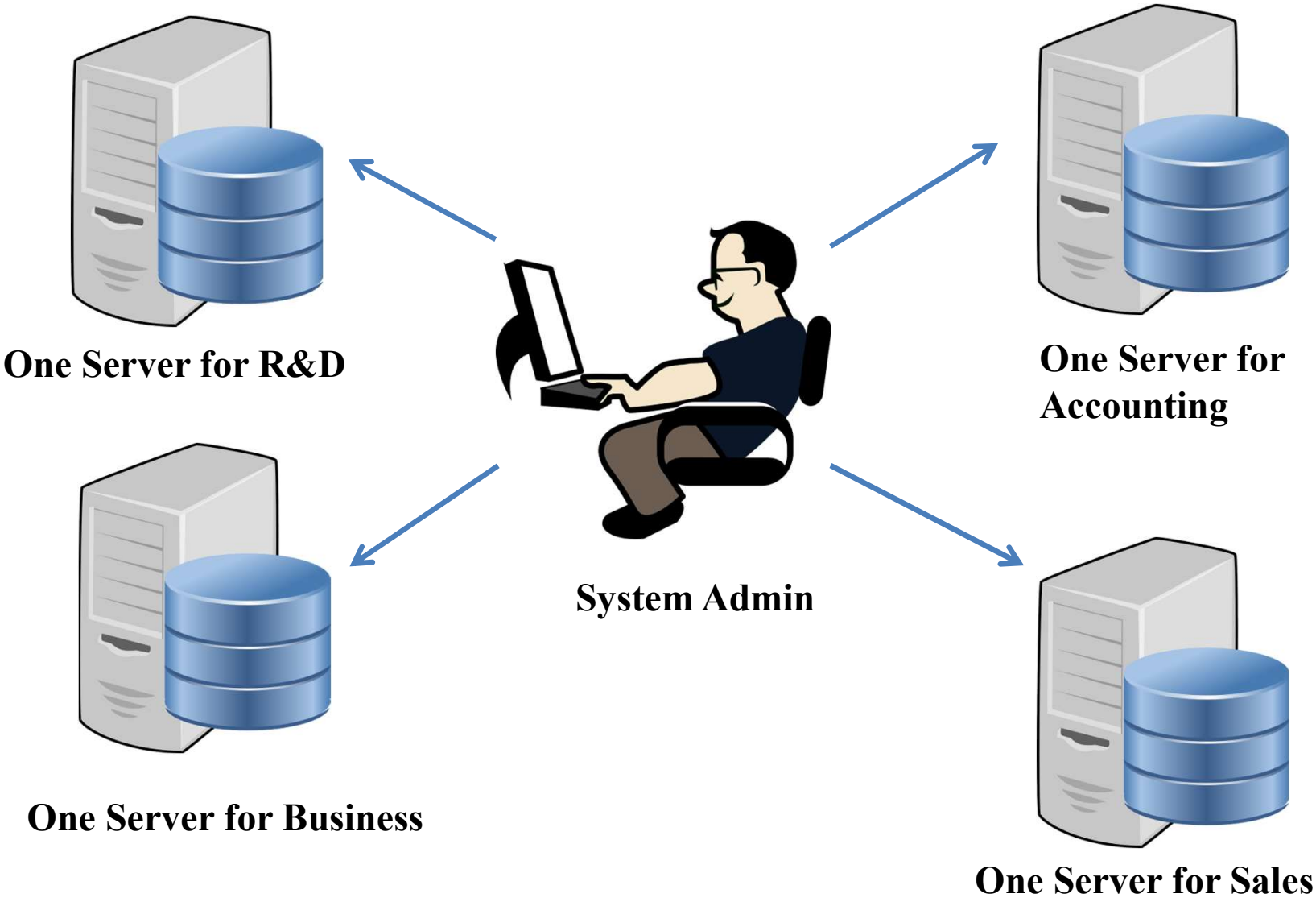


IBM CP/CMS Systems

A Brief History of Virtualization

- **IBM CP/CMS Systems**
  - First virtualized operating system
  - Every user gets a separate “virtual machine” for operating
  - Every user interacts with their own version of OS
  - No concept of time sharing – multiple tasks can be run simultaneously
  - No conflicts between users, so more reliable
  - The rise of personal computers led to a small decline in the importance of virtualization for a period of time

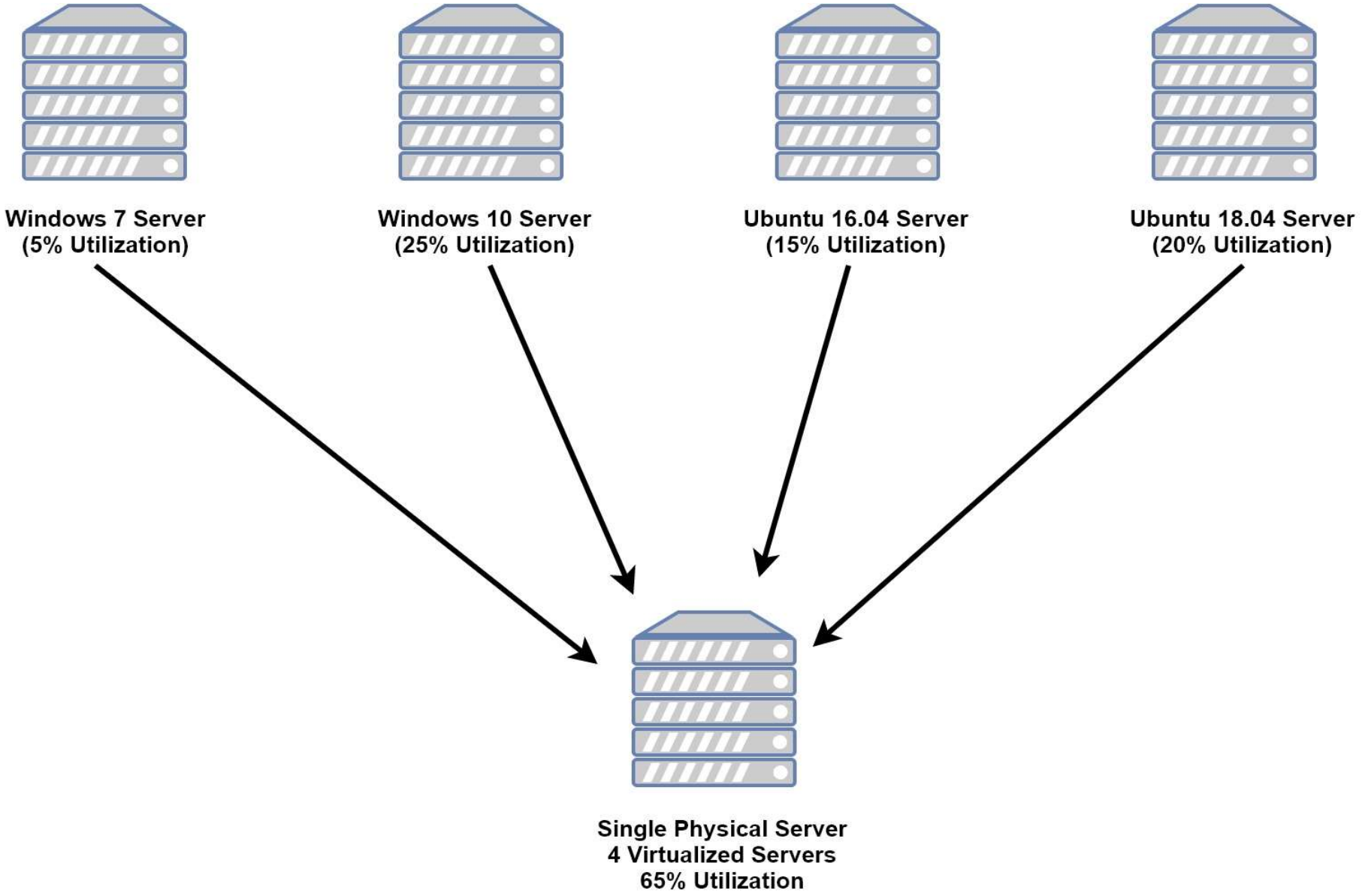
Need for Virtualization in Data Centers



Contd…

- System administrators allocated one machine per application
  - Increased stability – what if one application interfered with the other?
  - Increased security – hiding “sensitive” data
- Issues
  - Increased capital cost
  - Low server utilization

Virtualization



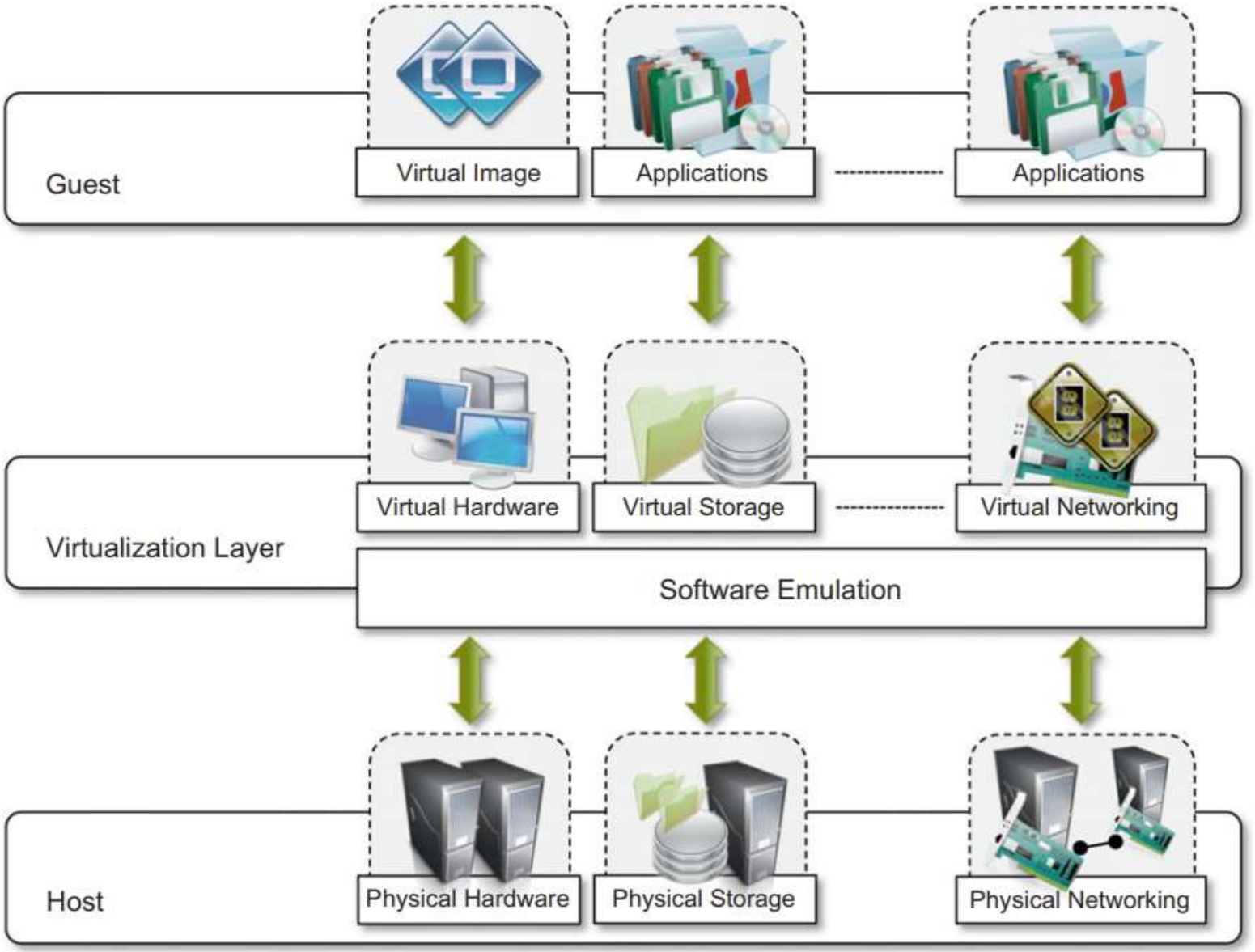
Virtualization

- Virtualization is a broad concept that refers to the creation of a virtual version of something, whether hardware, a software environment, storage, or a network.
- In a virtualized environment there are three major components: guest (virtual machines), host (physical machines), and virtualization layer.
- The guest represents the system component that interacts with the virtualization layer rather than with the host.
- The host represents the original environment where the guest is supposed to be managed.
- The virtualization layer is responsible for recreating the same or a different environment where the guest will operate

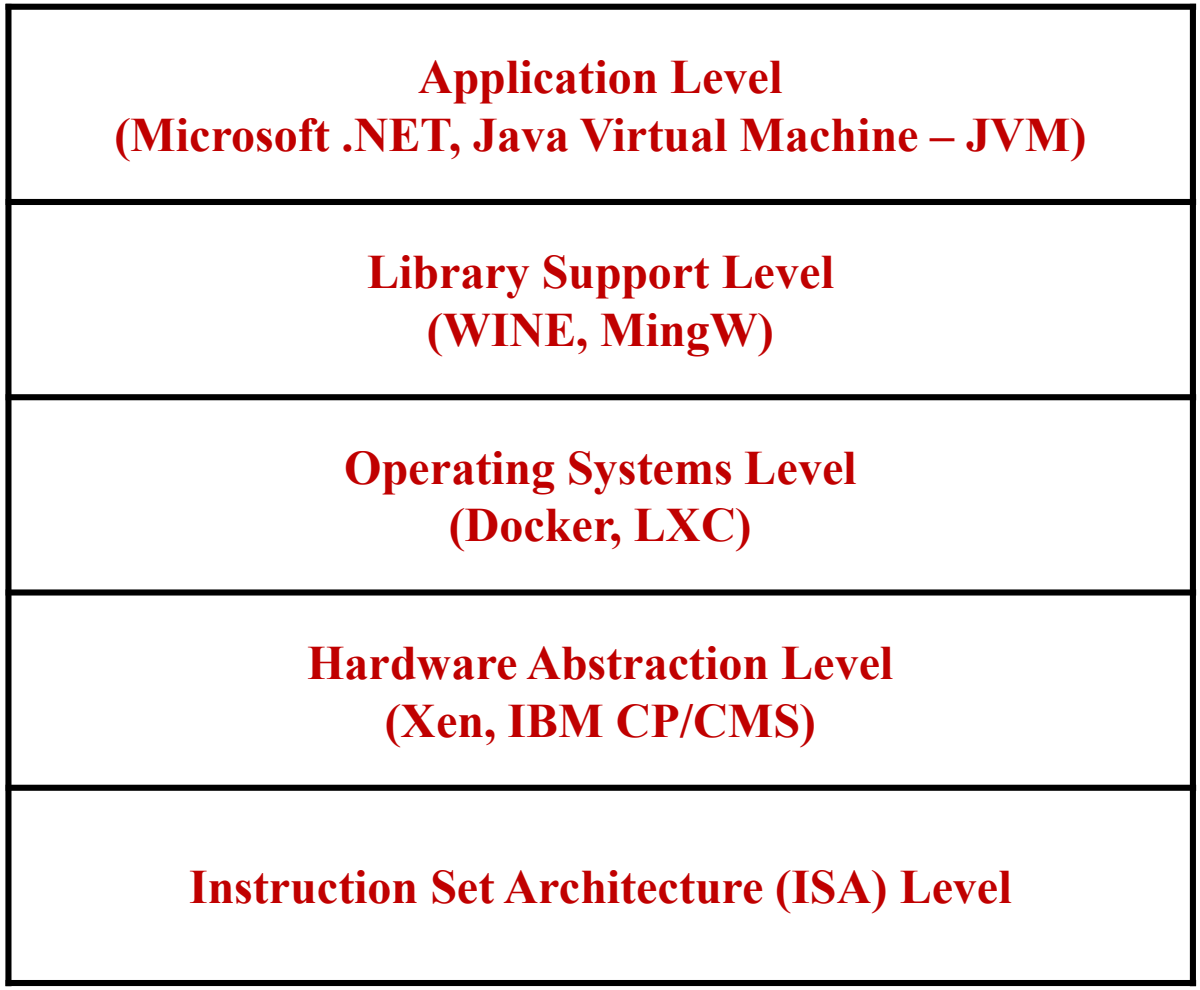
What can be virtualized?

- Desktop
- Application
- Server
- Storage
- Network

The Virtualization Reference Model



Levels of Virtualization

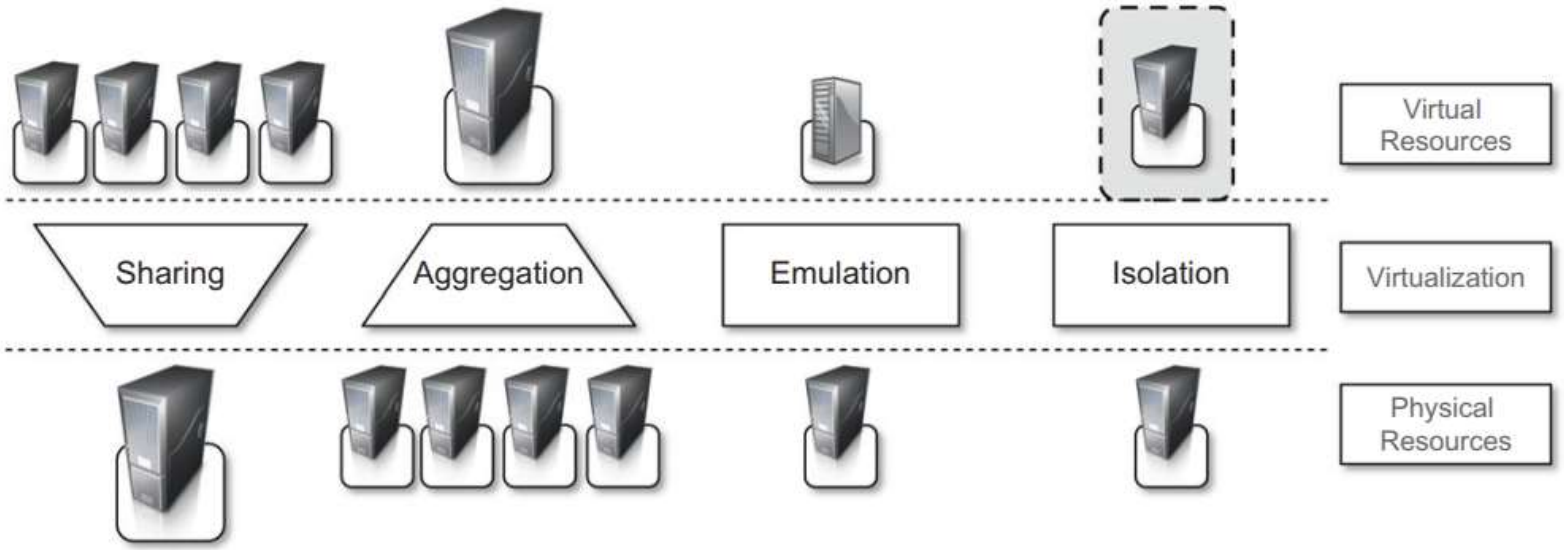


Issues/Challenges

1. ***Increased performance and computing capacity.***
2. ***Underutilized hardware and software resources.***
3. ***Lack of space.***
4. ***Greening initiatives:*** reducing the number of servers through server consolidation will definitely reduce the impact of cooling and power consumption of a data center. Virtualization technologies can provide an efficient way of consolidating servers.
5. ***Rise of administrative costs:*** Virtualization can help reduce the number of required servers for a given workload, thus reducing the cost of the administrative personnel.

Characteristics

1. ***Increased Security:*** The ability to control the execution of a guest in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment.
2. ***Managed execution:*** Virtualization of the execution environment not only allows increased security, but a wider range of features also can be implemented as given below.



Contd…

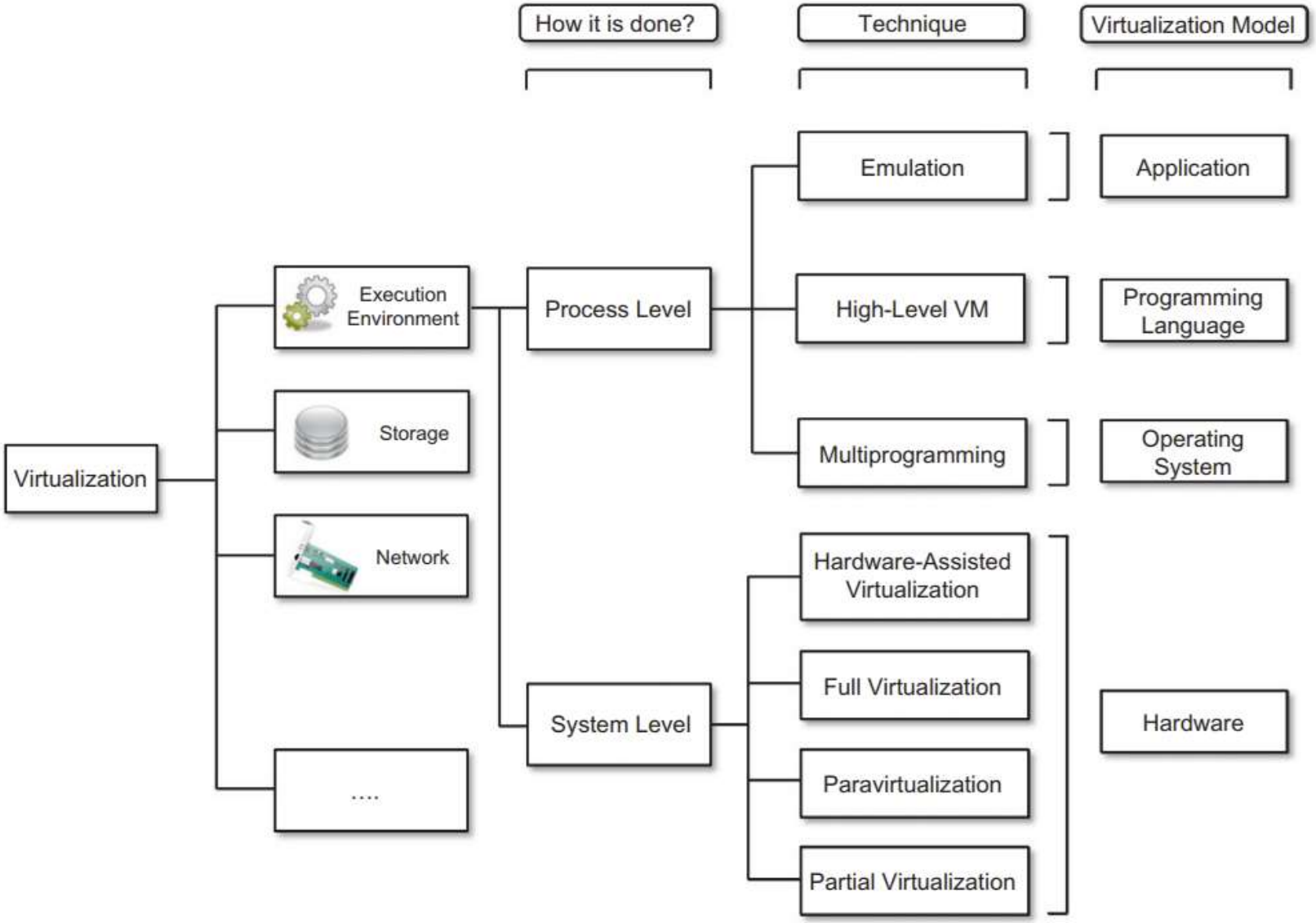
i. **Sharing:** Virtualization allows the creation of a separate computing environments within the same host. In this way it is possible to fully exploit the capabilities of a powerful guest, which would otherwise be underutilized.

ii. **Aggregation:** A group of separate hosts can be tied together and represented to guests as a single virtual host.

iii. **Emulation:** a completely different environment with respect to the host can be emulated, thus allowing the execution of guest programs requiring specific characteristics that are not present in the physical host.

iv. **Isolation:** 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. The virtual machine can filter the activity of the guest and prevent harmful operations against the host.

Taxonomy of Virtualization Techniques



Execution Virtualization

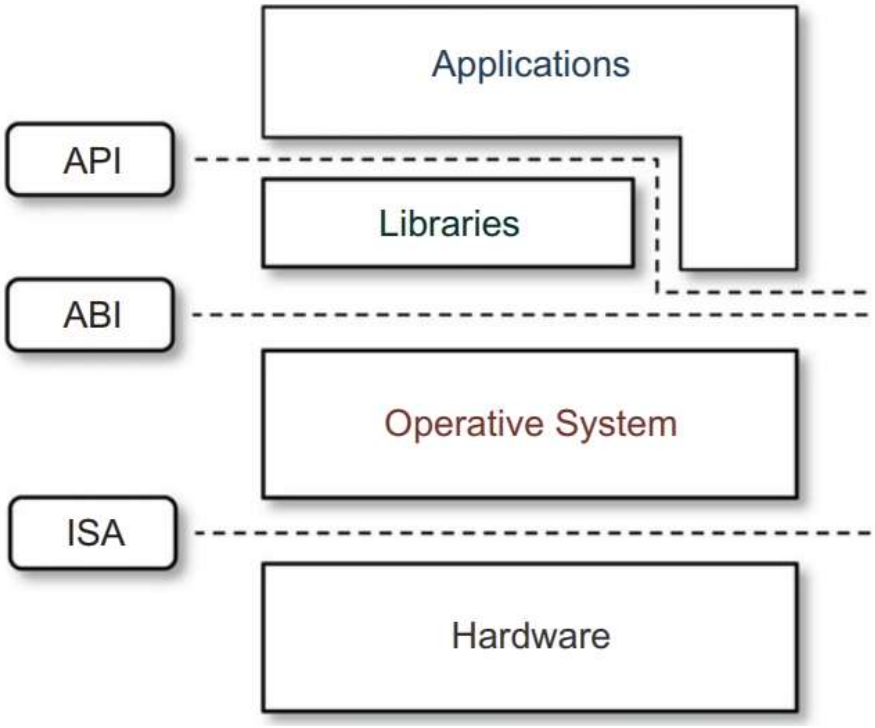
- Execution virtualization includes all techniques that aim to emulate an execution environment that is separate from the one hosting the virtualization layer.

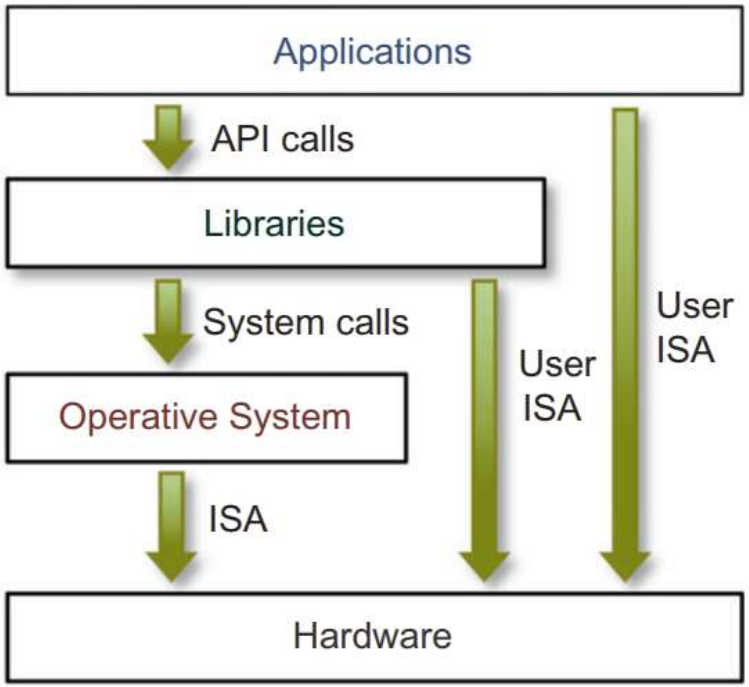
1. Machine reference model

✓ Virtualizing an execution environment at different levels of the computing stack requires a reference model that defines the interfaces between the levels of abstractions, which hide implementation details.

✓ From this perspective, virtualization techniques actually replace one of the layers and intercept the calls that are directed toward it.

A Machine Reference Model





Contd…

- Modern computing systems can be expressed in terms of the 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 interrupt management.
- ISA is the interface between hardware and software, and it is important to the operating system (OS) developer (System ISA) and developers of applications that directly manage the underlying hardware (User ISA).

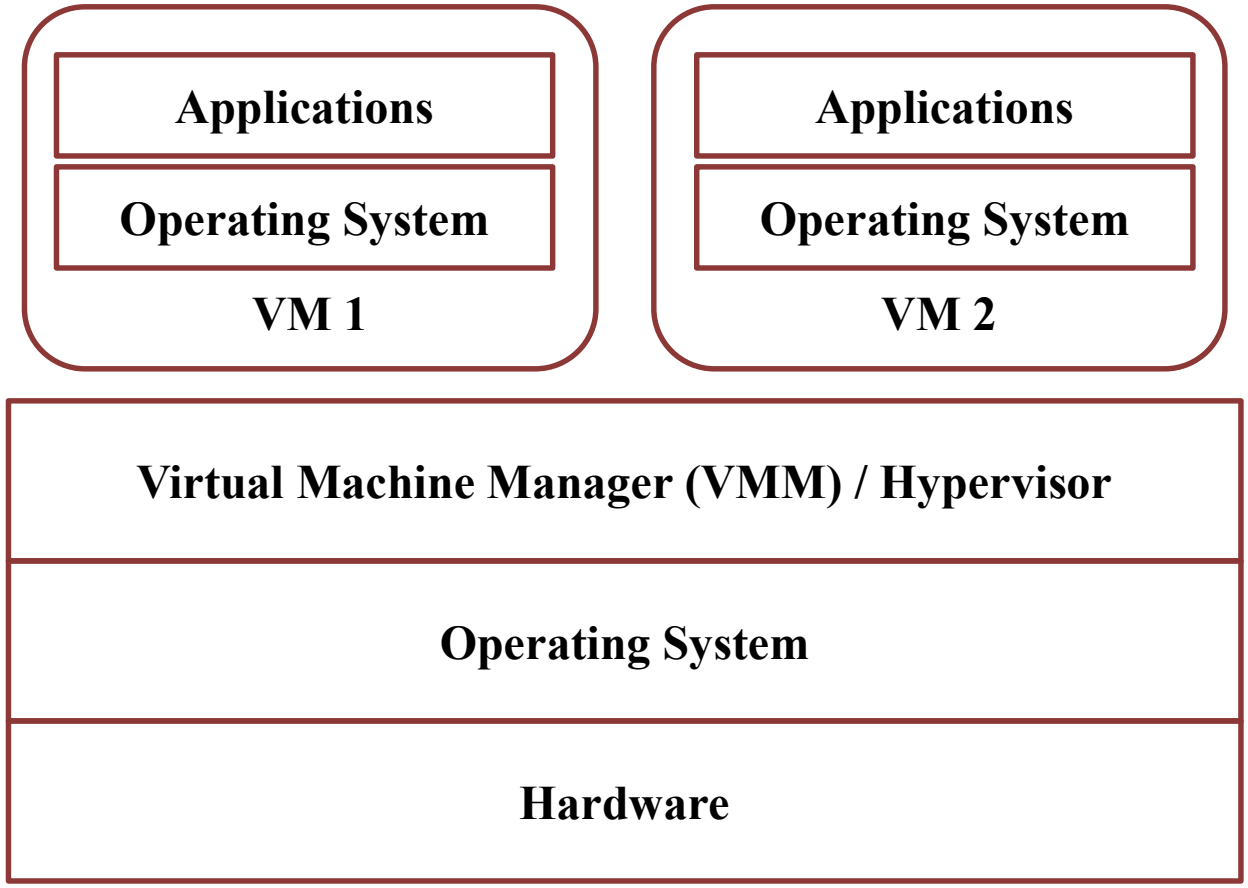
2. Hardware Level Virtualization

- 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 physical hardware.
- 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.

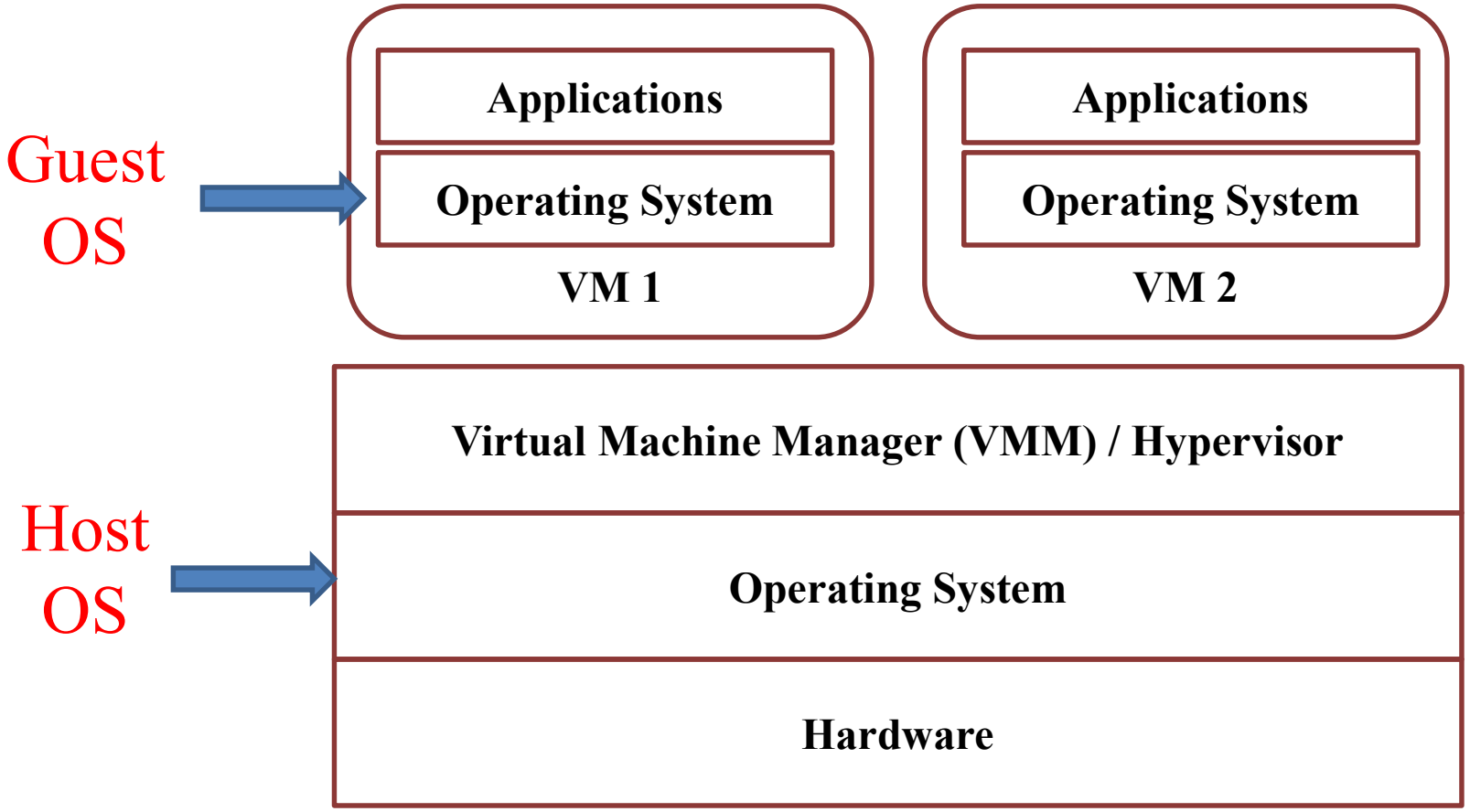
Hypervisor

- It is usually a layer of software that mediates between the VMs and the underlying hardware.
- Also called as Virtual Machine Manager (VMM).
- Provide an environment for programs which is essentially identical to the original machine
- Ensure complete control of the system resources.
  - Allocation
  - Separation
  - Preemption

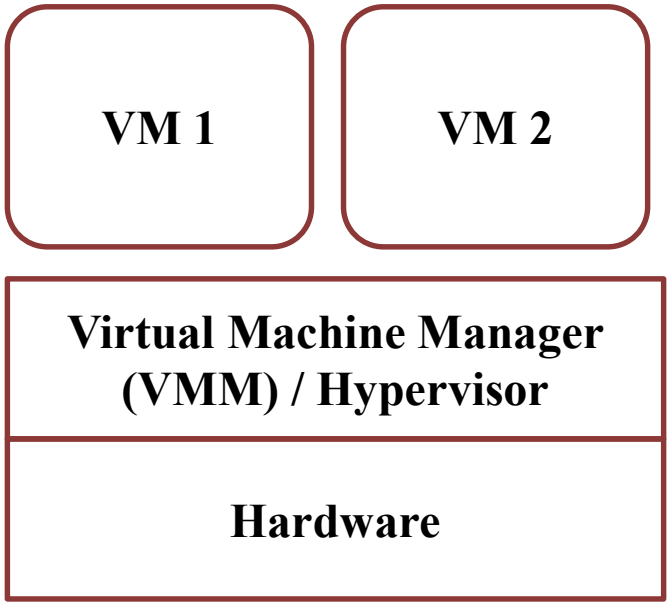
A General Architecture



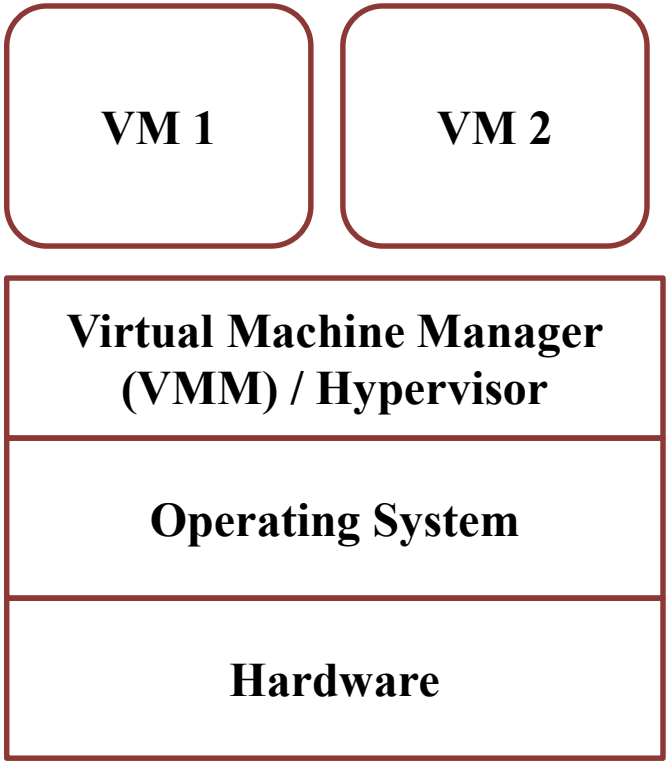
A General Architecture



Types of Hypervisor



Type 1 (Bare-Metal) Hypervisor



Type 2 (Hosted) Hypervisor

Contd…

Type 1 : hypervisors run directly on top of the hardware

- ✓ take the place of the operating systems and interact directly with the ISA interface.
- ✓ emulate this interface in order to allow the management of guest operating systems.
- ✓ also called as native virtual machine since it runs natively on hardware.

Type 2 : require the support of an operating system

- ✓ 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.

Comparison of Hypervisors

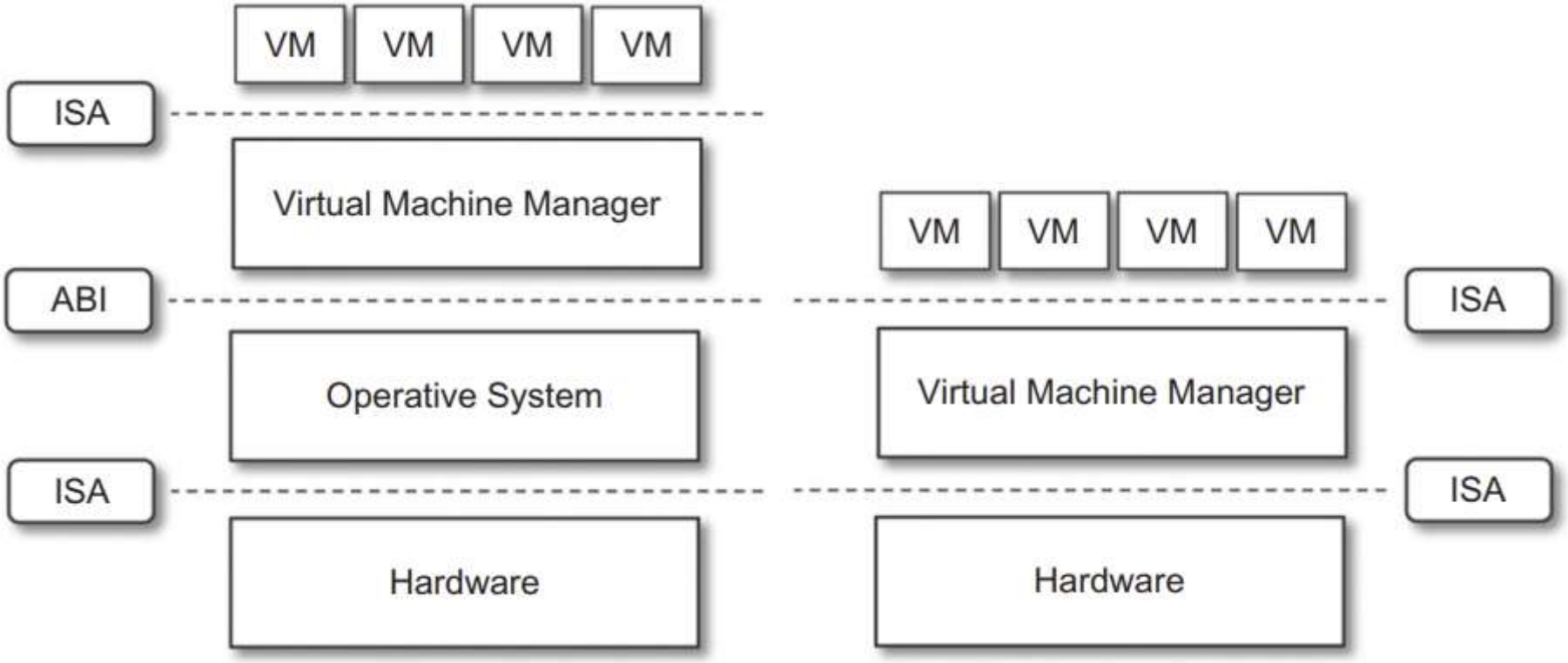
Type - 1 Hypervisor

- Resides directly on the hardware (“bare metal”)
- Communicates directly with the hardware resources
- More efficient
- More secure
- Eg: Citrix/Xen Server, VMware ESXi and Microsoft Hyper-V

Type – 2 Hypervisor

- Resides on top of the operating system (“hosted”)
- Communicates with hardware through the OS
- Less efficient
- Less secure
- Eg: Oracle Virtual Box, VMware Workstation etc.

Detailed Representation

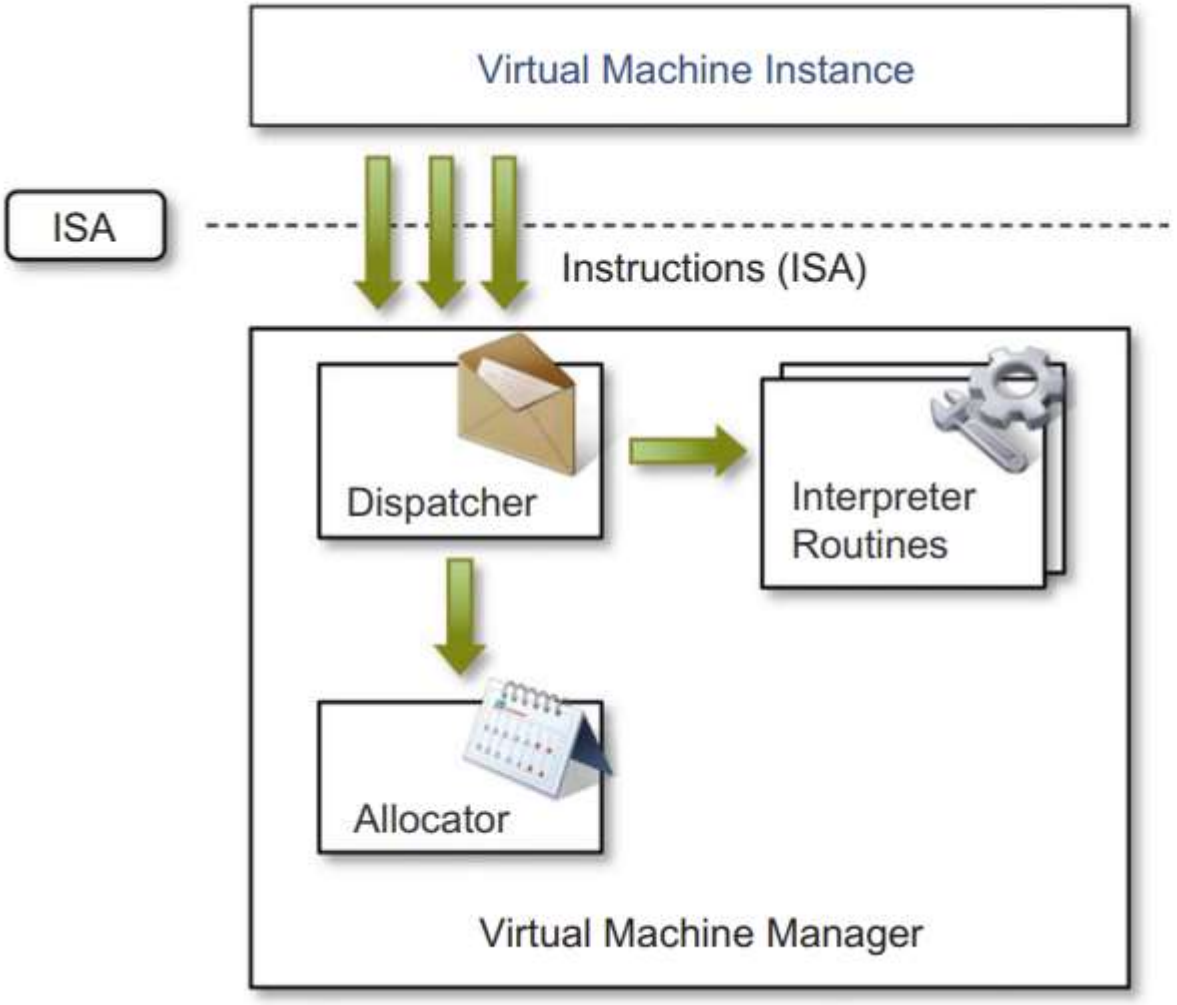


Type 2

Type 1

ABI: Application Binary Interface
ISA: Instruction Set Architecture

Hypervisor Reference Architecture



Contd…

- VMM mainly consists of 3 modules:

1. **Dispatcher:** entry point of the monitor and reroutes the instructions issued by the virtual machine instance to one of the two other modules.
2. **Allocator:** responsible for deciding the system resources to be provided to the VM. The allocator is invoked by the dispatcher.
3. **Interpreter:** consists of interpreter routines, executed whenever a virtual machine executes a privileged instruction: a trap is triggered and the corresponding routine is executed.

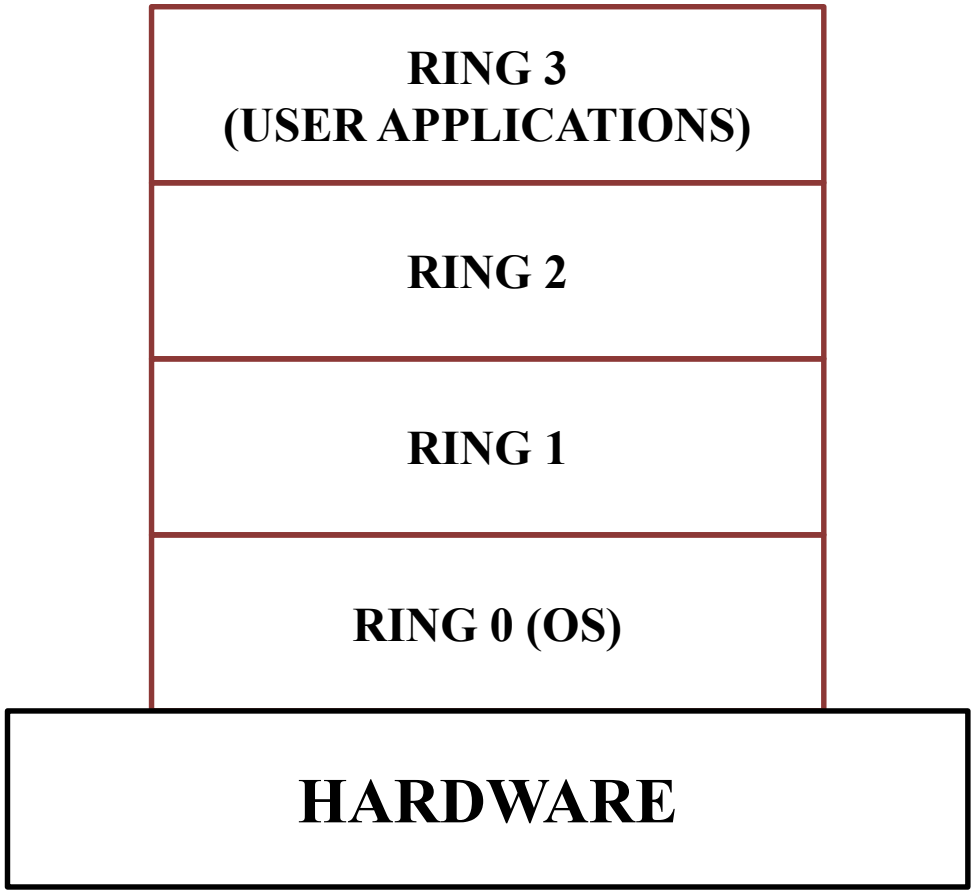
Conditions for Hypervisor

The criteria that need to be met by a VMM to efficiently support virtualization are as follows.

1. **Equivalence.** A guest running under the control of a VMM should exhibit the same behavior as when it is executed directly on the physical host.
2. **Resource control.** The VMM should be in complete control of virtualized resources.
3. **Efficiency.** A statistically dominant fraction of the machine instructions should be executed without intervention from the VMM.

Popek and Goldberg 1974: "Formal Requirements for Virtualizable Third Generation Architectures" Communications of the ACM

“Protection Ring” Concept



The Difficulty with Virtualization

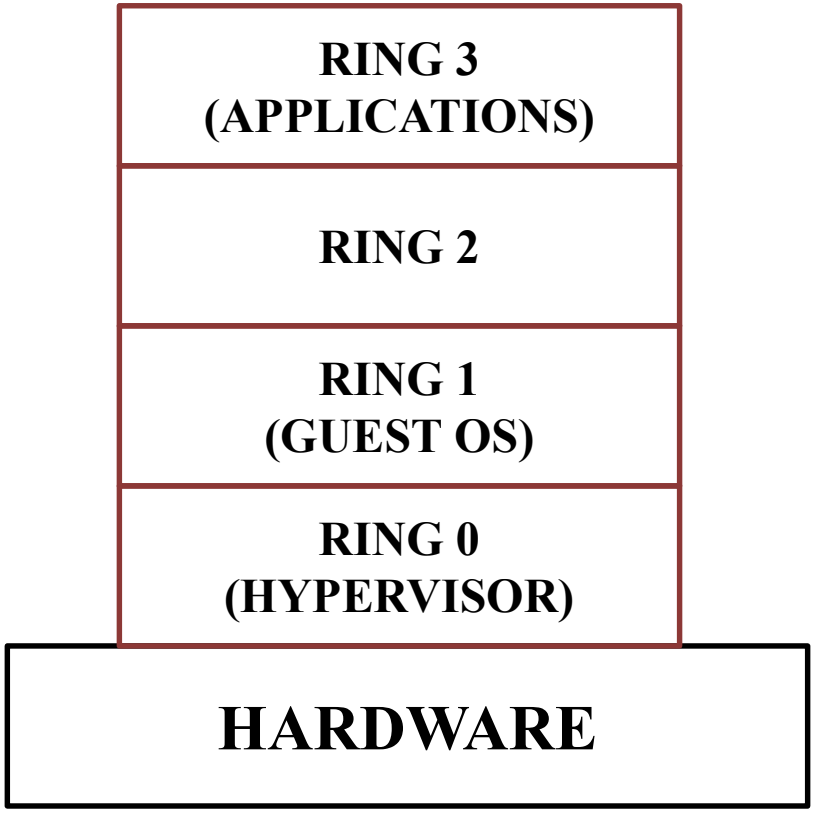
- All modern operating systems are built to run at Ring 0
  - They are designed to issue privileged instructions to modify memory and hardware directly
- For virtualization, guest OS resides on top of a hypervisor
  - Guest OS can only operate at a Ring > 0
  - This causes problems when the guest OS issues privileged instructions
  - The hypervisor must intercept and translate privileged instructions before passing it over to the hardware

Hardware Virtualization Techniques

1. 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.

Full Virtualization

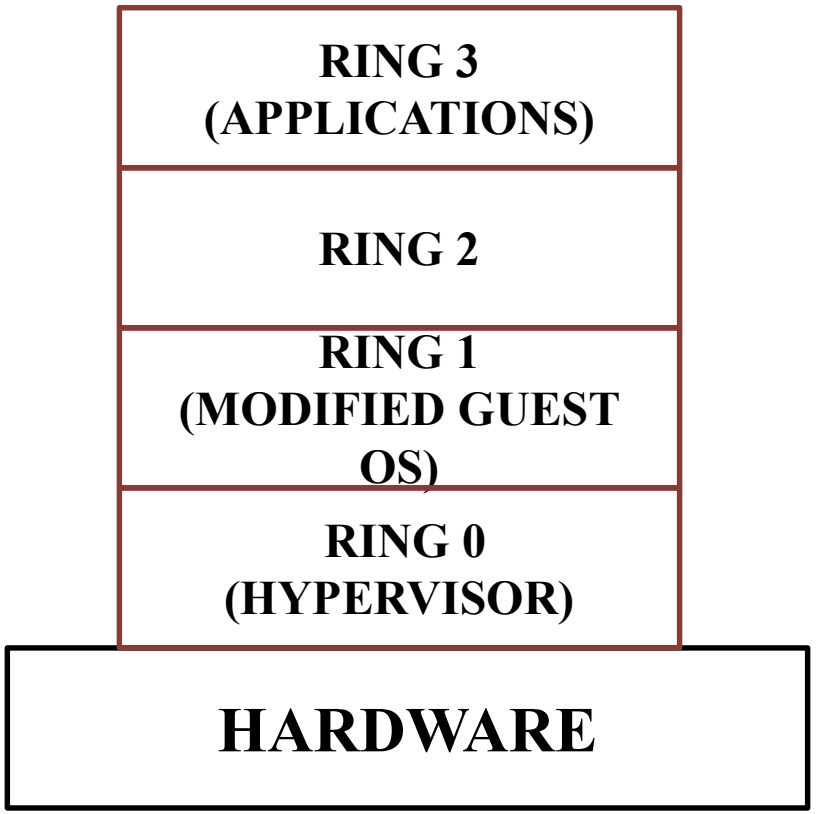


- Hypervisor operates at Ring 0
- Hypervisor scans the request stream
  - Captures and translates privileged instructions
  - Guest OS thinks it is directly working with hardware

2. Para Virtualization

- ✓ Exposes 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.
- ✓ 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.

Para Virtualization

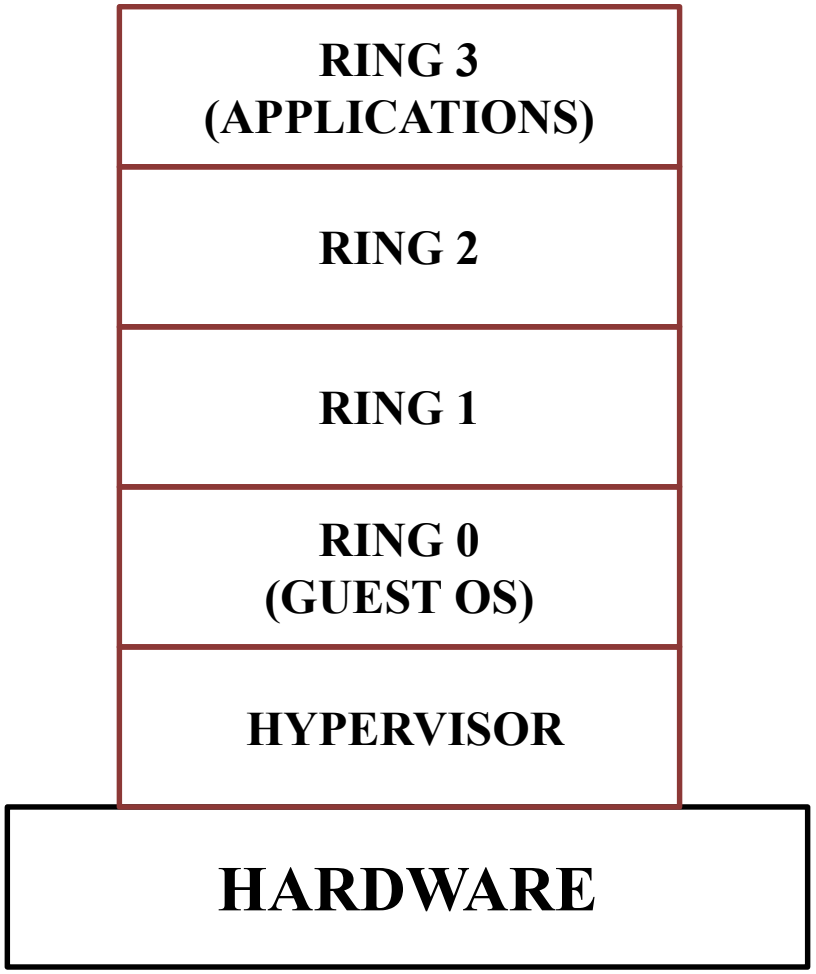


- Hypervisor resides in a privileged layer beneath the guest OS
- Guest OS is modified, so it doesn't execute privileged instructions
  - It executes hypercalls to the hypervisor
- Better performance
  - Limited use due to the need to modify OS

3. 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).

Hardware Assisted Virtualization



- Hardware provides architectural support for building a virtual machine manager able to run a guest operating system in complete isolation.
- Hardware allows hypervisor to reside in a privileged ring
- Privileged and sensitive calls are set to automatically trap to the hypervisor
- Can use unmodified OS + better performance
- Requires hardware support

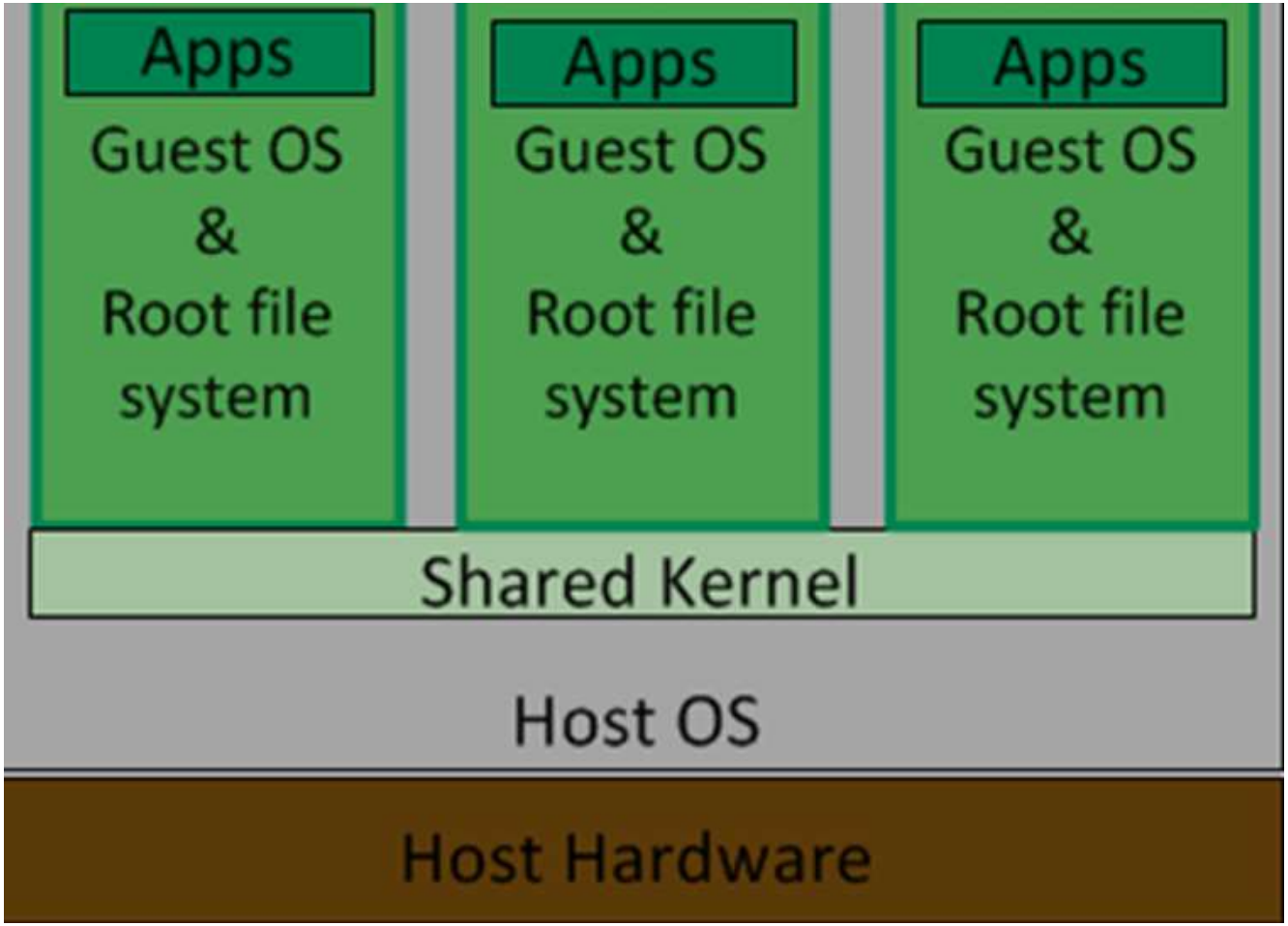
Types of Virtualization

Type of Virtualization	Requires Hardware Support?	Requires Guest OS Modification?
Full Virtualization	No	No
Para virtualization	No	Yes
Hardware Assisted Virtualization	Yes	No

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.

Contd…


Apps
Guest OS
&
Root file
system
Apps
Guest OS
&
Root file
system
Apps
Guest OS
&
Root file
system
Shared Kernel
Host OS
Host Hardware