Cloud Computing

Dr.P.Chinnasamy
Department of CS/IS

BITS Pilani
Pilani Campus

innovate achieve lead

BITS Pilani

Pilani Campus

CSI ZG527/ SE ZG527 – Virtualization Continued– L3

Recap

- ✓ Introduction to Virtualization
- ✓ Use & demerits of Virtualization
- ✓ Types of Virtualization
- ✓ Manage the resources for the SaaS, PaaS and IaaS models

Key Terms to Remember

VM : Virtual Machine

VMM: Virtual Machine Monitor **Hypervisor** : VMM

Multiplexed: Many or several

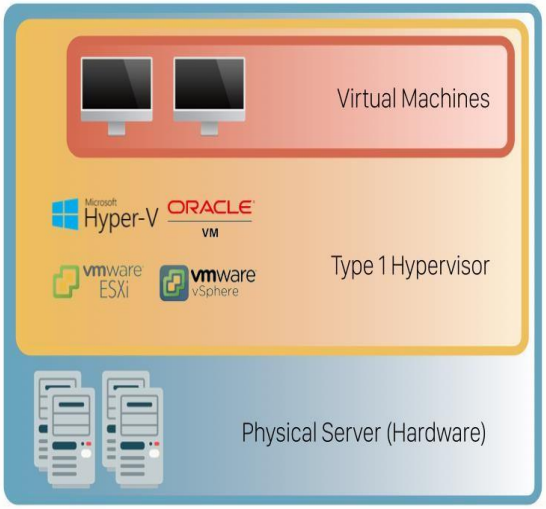
Host: System where the VMM resides

Guest : Virtual Machines created

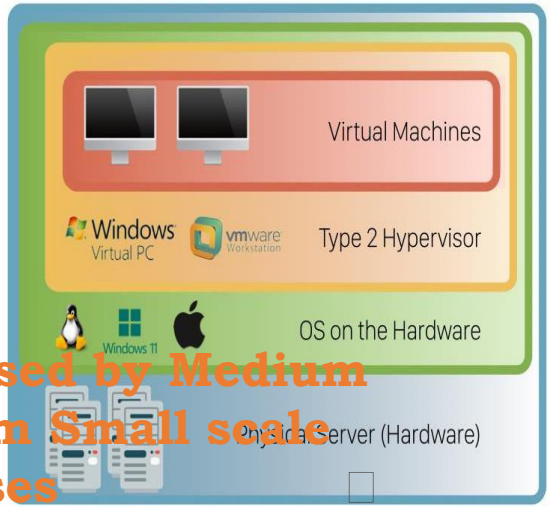
Agenda

- ✓ Comparison of Type I and Type II Hypervisor
- ✓ Resource Sharing VM-CPU
- ✓ Resource Sharing VM-Memory
- ✓ Resource Sharing VM-IO

Type I & II at a Glance

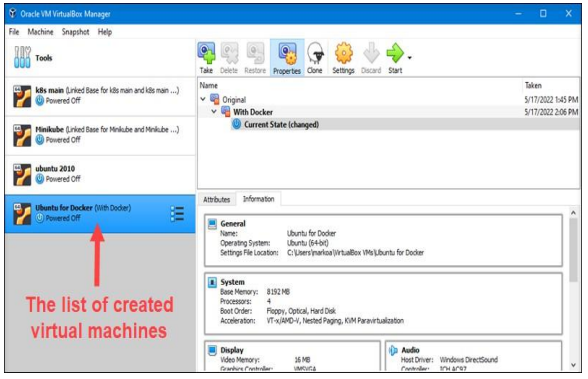


- ✓ **Type 1 or Bare Metal is installed directly on hardware**
- ✓ **Type 2 is installed on top of an existing OS**
- ✓ **Mostly used by Enterprise CSP**



- ✓ **Mostly used by Medium & Medium Small scale enterprises**

- ✓ **Need to have a separate VM Management console to monitor the VM**



- ✓ **Has an in-built console**

Comparison

innovate achieve lead

Aspect	Type 1 Hypervisor (Bare Metal)	Type 2 Hypervisor (Hosted)
Deployment	**Installed directly on the physical hardware**	**Installed on top of a host operating system**
Examples	**VMware vSphere/ESXi, Microsoft Hyper-V, Xen**	**Oracle VirtualBox, VMware Workstation, Parallels Desktop**
Performance	Generally higher performance due to direct access to hardware resources	Lower performance due to reliance on host OS for resource allocation
Resource Overhead	Minimal overhead as it operates directly on hardware	Higher overhead due to running within a host OS
Isolation	**Better isolation between virtual machines (VMs)**	**Weaker isolation, as issues in host OS can affect VMs**

Comparison

innovate achieve lead

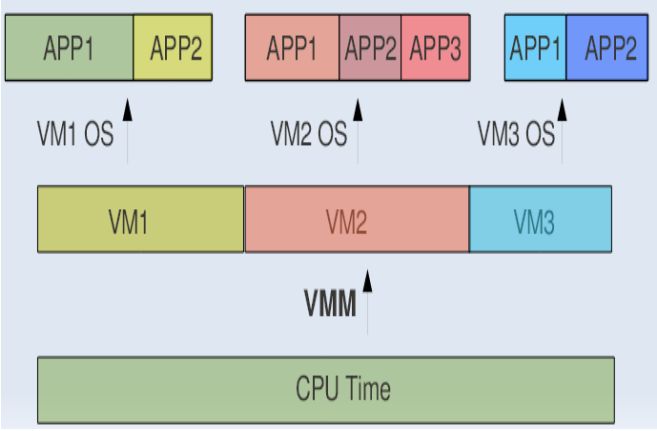
Aspect	Type 1 Hypervisor (Bare Metal)	Type 2 Hypervisor (Hosted)
Security	Generally more secure due to reduced attack surface	Less secure due to reliance on host OS security
Scalability	Typically more scalable for large deployments	Limited scalability compared to Type 1 hypervisors
Management	May require additional management tools	Often easier to manage with GUI interfaces
Disadvantages	• **Requires dedicated hardware** • **Potentially higher upfront costs** • **More complex setup and management**	• **Performance overhead** • **Limited scalability** • **Reduced security compared to Type 1 hypervisors**

Resource Sharing VM-CPU

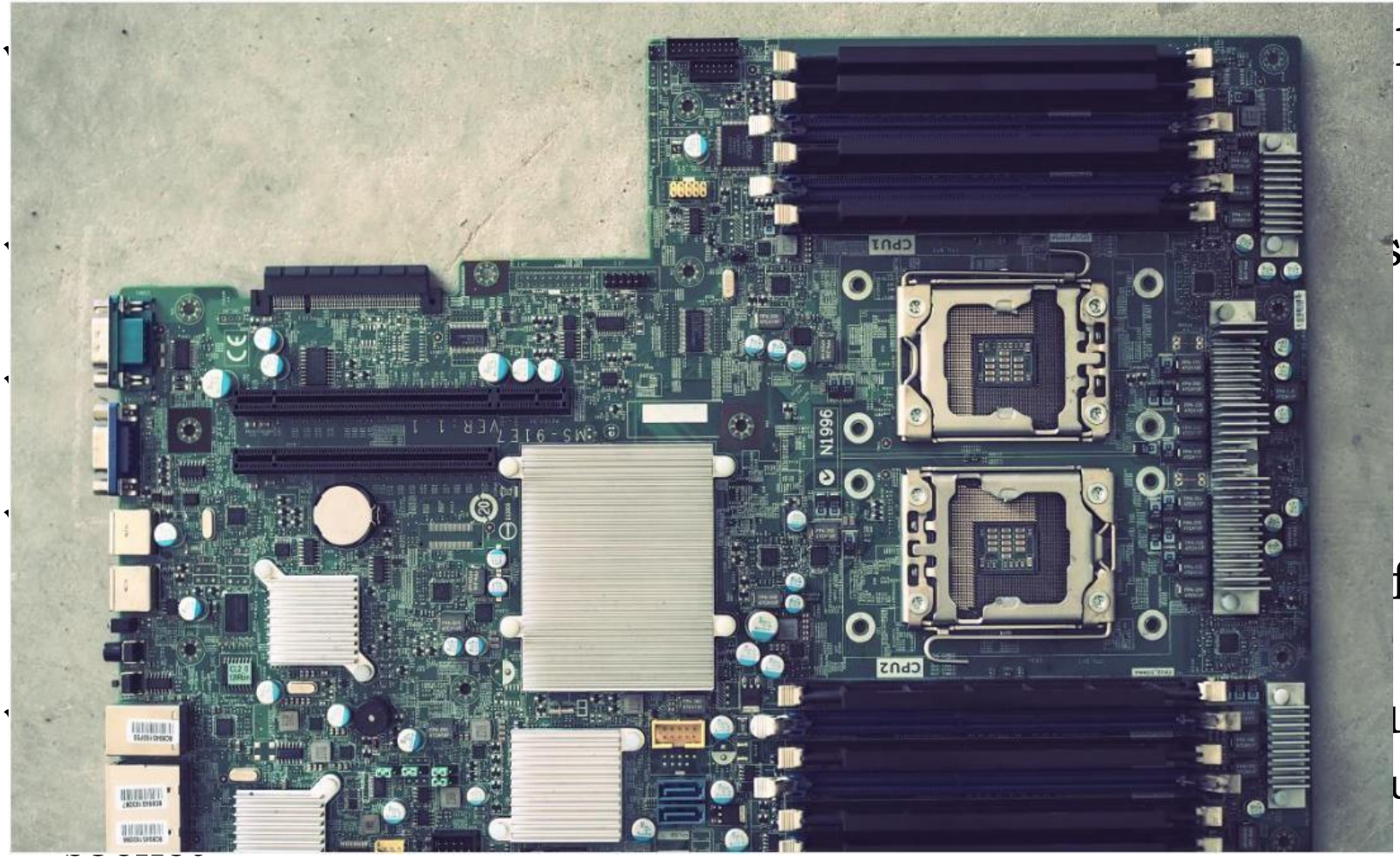
VMM or Hypervisor provides a virtual view of CPU to VMs.

In multi processing, CPU is allotted to the different processes in form of time slices by the OS.

Similarly VMM or Hypervisor allots CPU to different VMs.



Resource Sharing VM-CPU

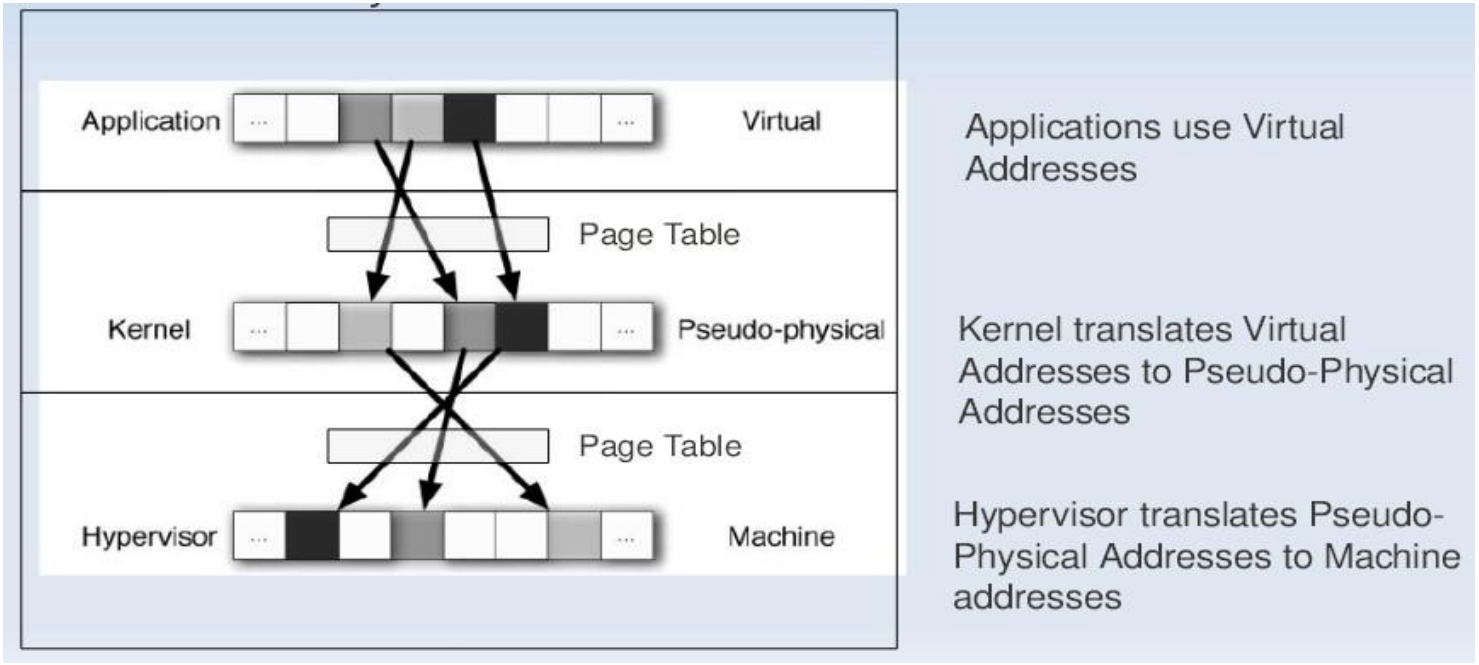


he
is
sor
is
as
f a
lse
ual

Resource Sharing VM- Memory

In Multiprogramming there is a single level of indirection maintained by Kernel.

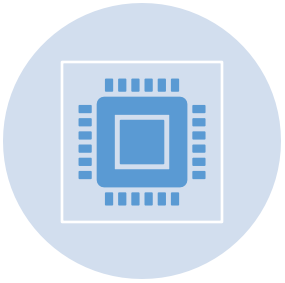
In case of Virtual Machines there is one more level of indirection maintained by VMM



Resource Sharing – VM - Memory

- ✓ **Memory Sharing** relies on the observation that several **virtual machines** might be running instances of the same **guest operating system**.

Resource Sharing – VM - IO



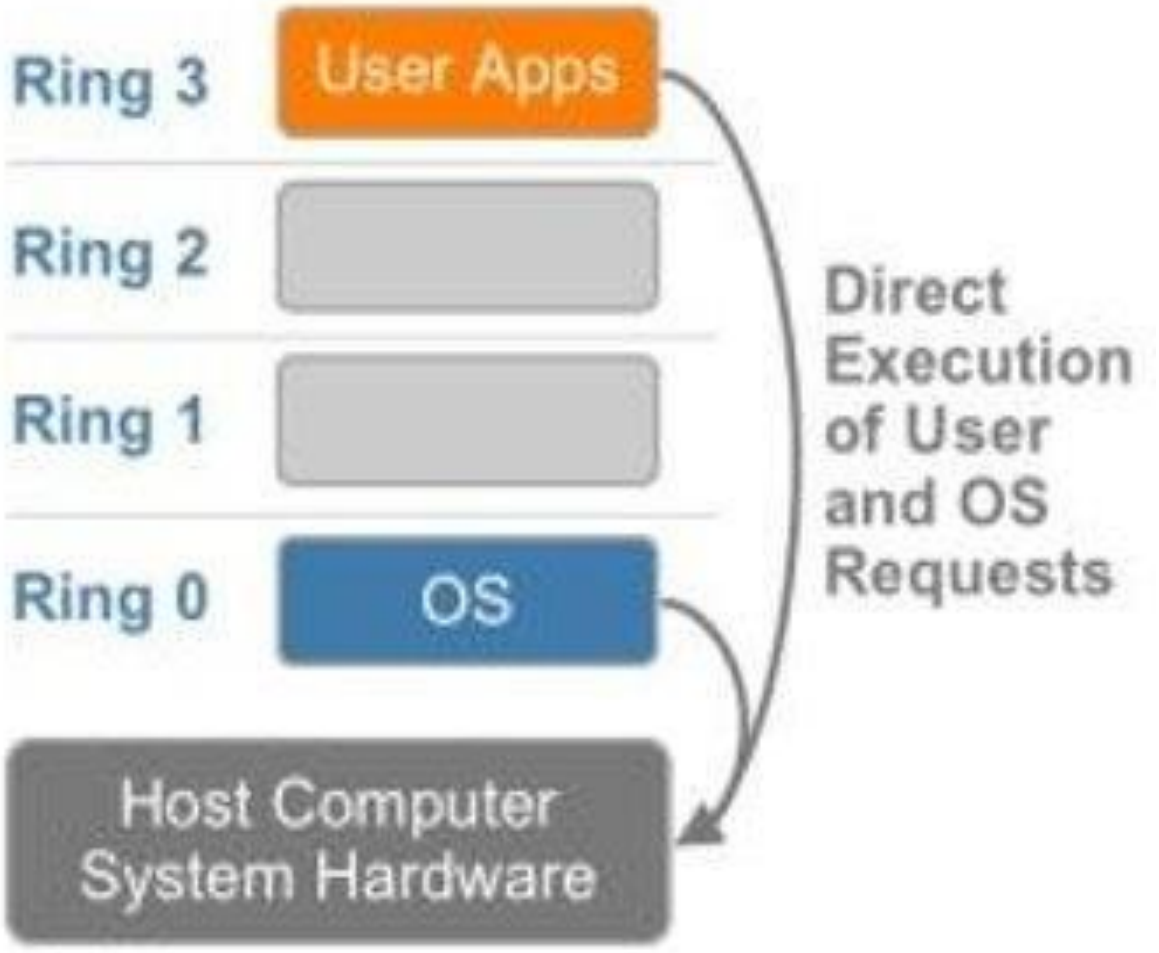
Device needs to use Physical Memory location.

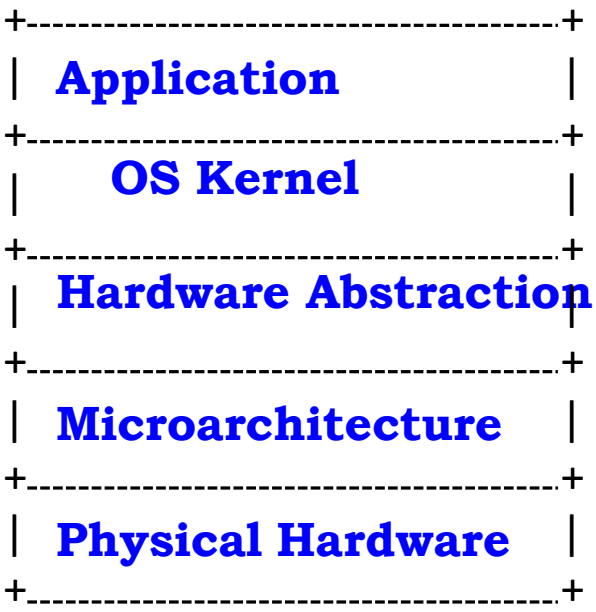
In a virtualized environment, the kernel is running in a hypervisor-provided virtual address space

Allowing the guest kernel to convey an arbitrary location to device for writing is a serious security hole

Each device defines its own protocol for talking to drivers

X86 Architecture





X86 Architecture

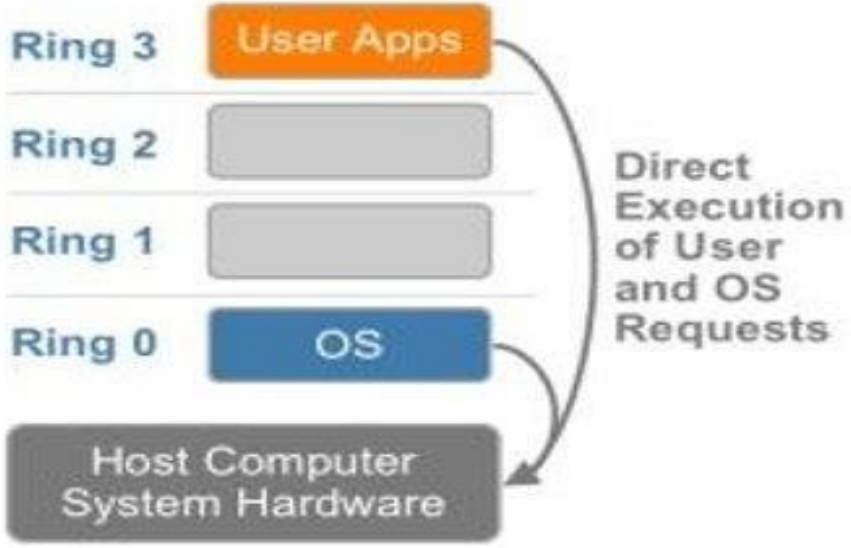
- ✓ The x86 architecture uses a system of privilege rings to **control access to sensitive areas of the computer memory and resources**.
- ✓ This also known as **Ring Architecture.**
- ✓ **Ring 0:** Also known as the **kernel mode**, ring 0 is the **most privileged level**, and it has full access to all of the computer's resources, **including memory and I/O devices**. This is where the **operating system kernel runs.**
- ✓ **Ring 1:** This ring is used for **device drivers** and other **low-level system components**. Ring 1 has access to more resources than ring 2 and ring 3

X86 Architecture

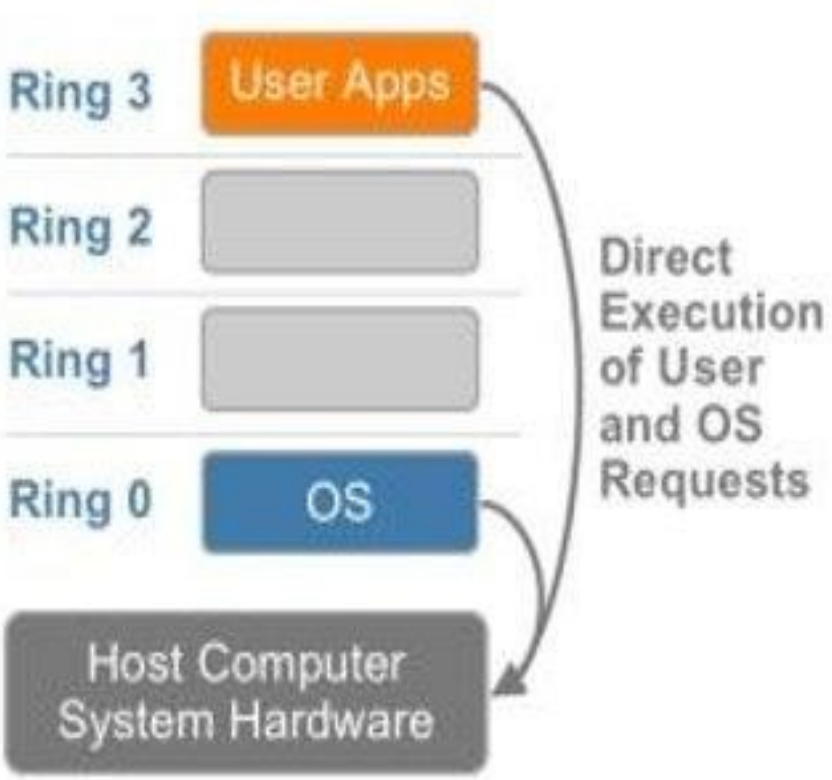
- ✓ **Ring 2**: Ring 2 is not used in **modern x86 systems**, but it was used in the past for system extensions, such as **device and drivers.**
- ✓ **Ring 3:** Also known as **user mode**, ring 3 is the **least privileged level**, and it is where **user applications run.**

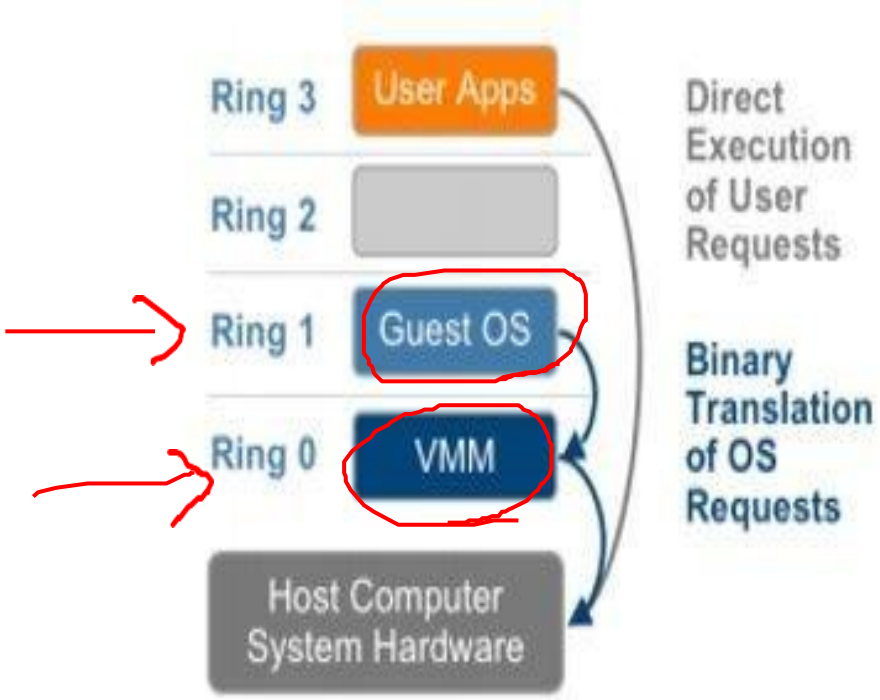
Challenges of x86

- ✓ **X86 operating systems** are designed to run directly on the **bare-metal hardware.**
- ✓ **Virtualizing** the x86 architecture requires placing a **virtualization layer at ring 0** to deliver the shared resources efficiently



VM Solution


Ring 3
User Apps
Ring 2
Ring 1
Ring 0
OS
Host Computer
System Hardware
Direct
Execution
of User
and OS
Requests


Ring 3
User Apps
Ring 2
Ring 1
Guest OS
Ring 0
VMM
Host Computer
System Hardware
Direct
Execution
of User
Requests
Binary
Translation
of OS
Requests

Evolution of Virtualization

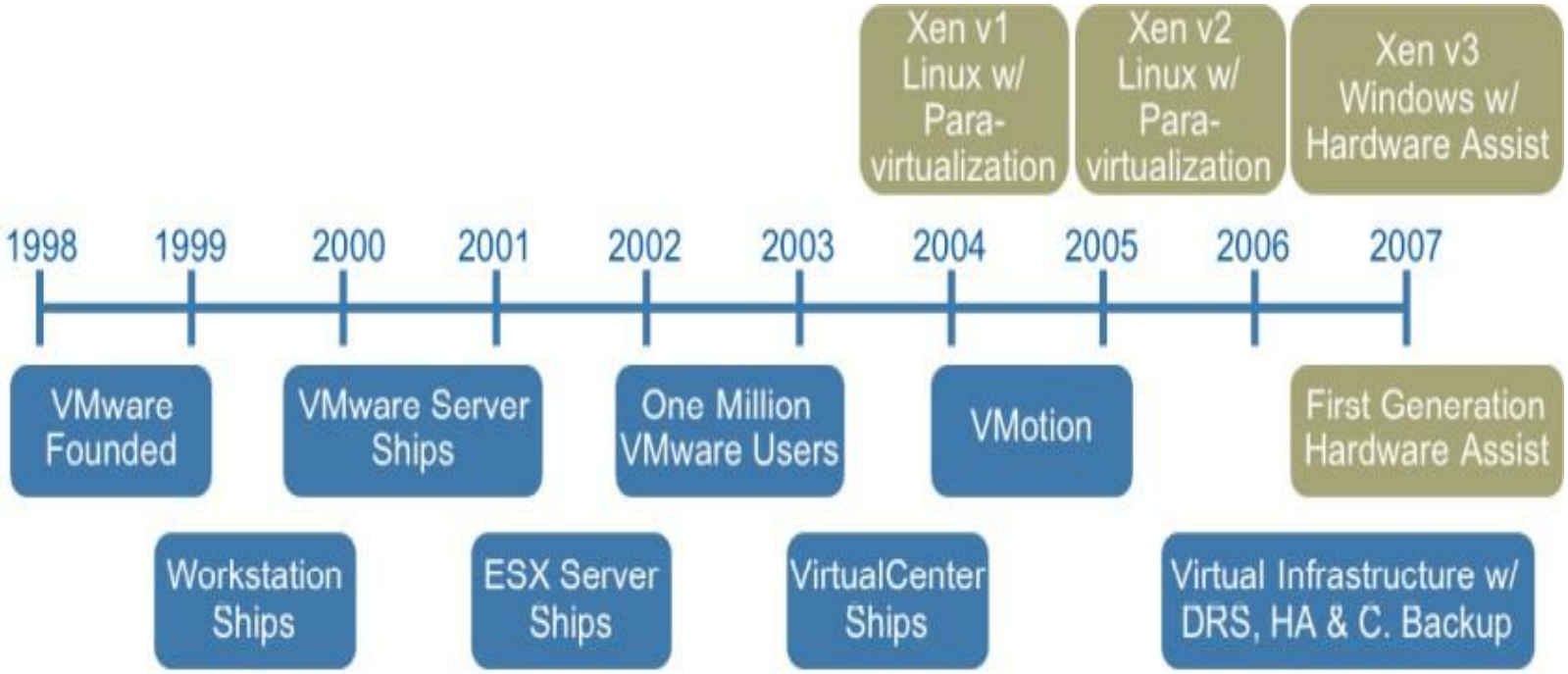
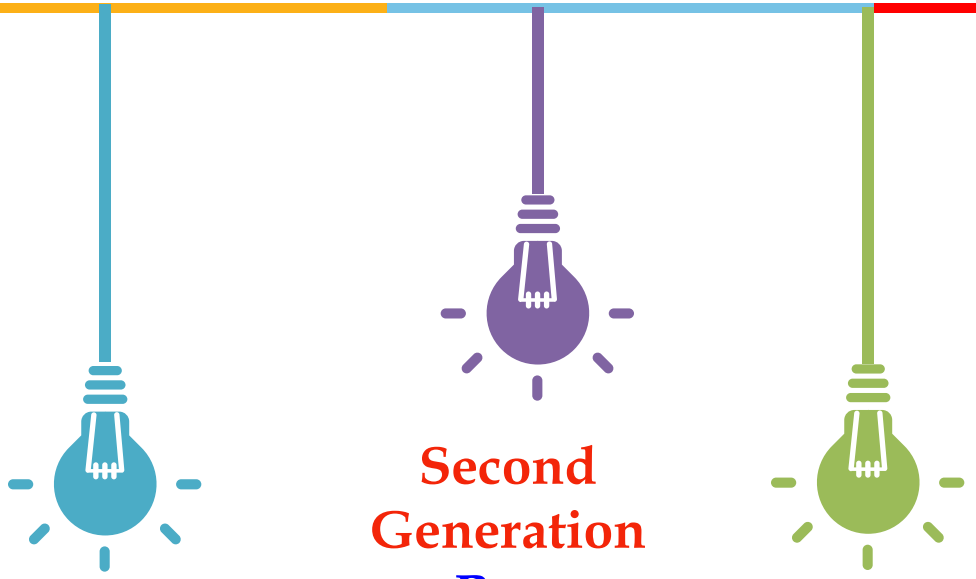


Figure 1 – Summary timeline of x86 virtualization technologies

Evolution of Virtualization



First Generation
Full Virtualization
Binary Rewriting
Software based

Second Generation
Para Virtualization
Modified Guest
VMWare, Xen

Third Generation
Un Modified Guest
Silicon Based
Hardware Assisted

Emulation

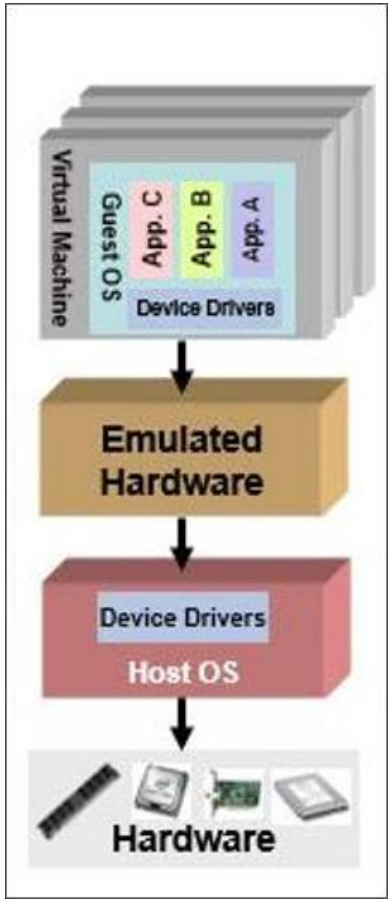
- ✓ Emulation is the process where the **virtualizing software mimics that portion of hardware, which is provided to the guest operating system** in the virtual machine.
- ✓ Emulation provides **VM portability and wide range of hardware compatibility**, which means the possibility of executing any virtual machine on any hardware, as the guest operating system interacts only with the emulated hardware.
- ✓ In an emulated environment, both the application and guest operating system in virtual machines run in the **user mode of base operating system**. In simple terms, the behavior of the hardware is produced by a software program

Emulation

- ✓ Emulation process involves only **those hardware components** so that user or virtual machines does not understand the underlying environment.

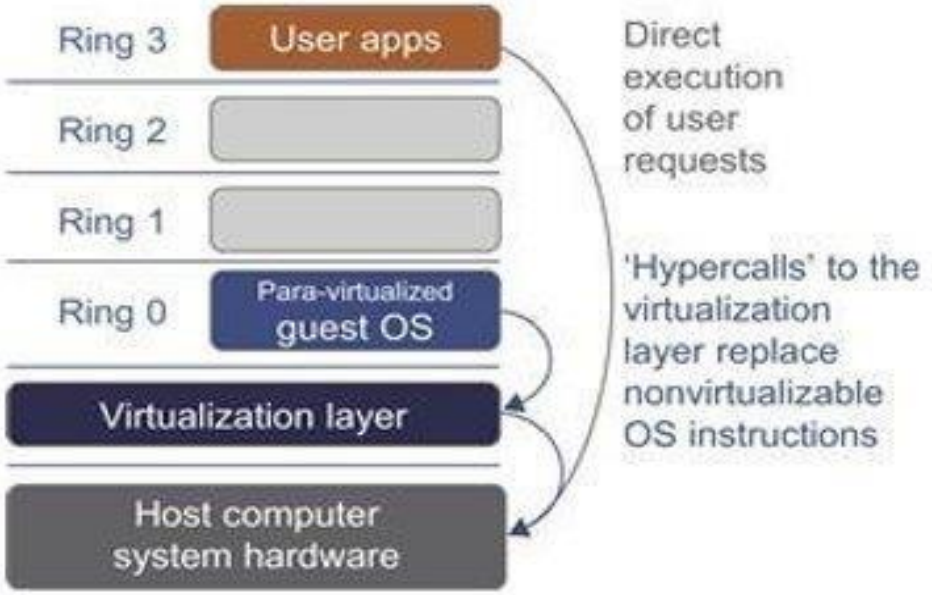
Binary Translation/ Full Virtualization

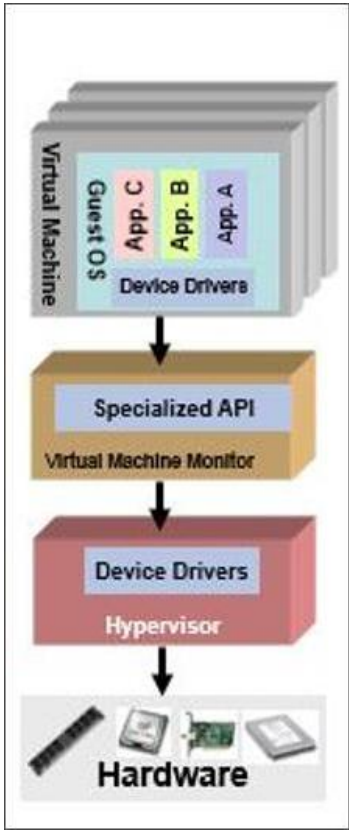
- ✓ In its basic form known as "full virtualization" the hypervisor provides a fully emulated machine in which an operating system can run.
- ✓ **VMWare is a good example.**
- ✓ The biggest advantage to this approach is its flexibility: one could run a **RISC-based OS** as a guest on an Intel-based host.
- ✓ 1st Generation offering of x86/x64 server virtualization .
- ✓ Dynamic binary translation.



Para Virtualization

- ✓ Paravirtualization," found in the **XenSource, open source Xen** product, wo





un

3)

a

ıd,

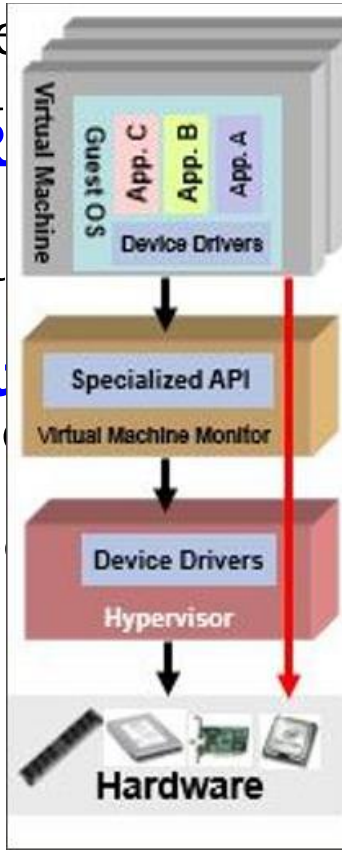
ged

he

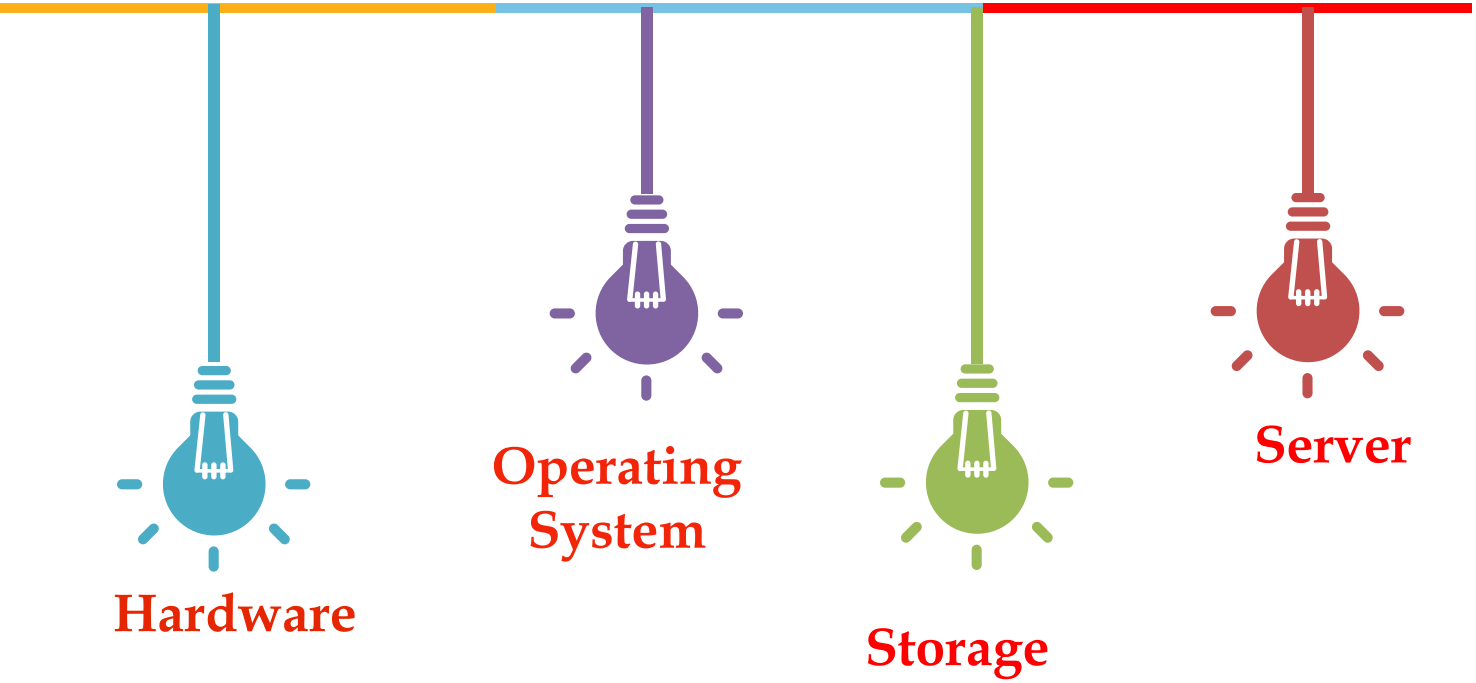
virtualization requests and putting them to the hardware

Hardware Assisted

- ✓ This technique attempts to **simplify virtualization** because full or paravirtualization is complicate
- ✓ **Intel and AMD add** an additional mod **privilege mode level (some people call it R x86 processors.**
- ✓ Therefore, operating systems can still run a **and the hypervisor can run at Ring -1.**
- ✓ This technique removes the **difficu implementing binary translation** virtualization.
- ✓ It also lets the operating run without mo unlike para virtualization



Types of Virtualization



Hardware Virtualization

- ✓ When the virtual machine **software or virtual machine manager *(VMM)*** *is directly installed on the hardware system* is known as hardware virtualization.
- ✓ The main job of hypervisor is to control and monitoring the **processor, memory and other hardware resources.**
- ✓ After virtualization of hardware system we can **install different operating system on it and run different applications on those OS.**
- ✓ Hardware virtualization is mainly done for the **server platforms**, because controlling virtual machines is much easier than controlling a physical server.

Operating System Virtualization

- ✓ When the virtual machine **software or virtual**
- ✓

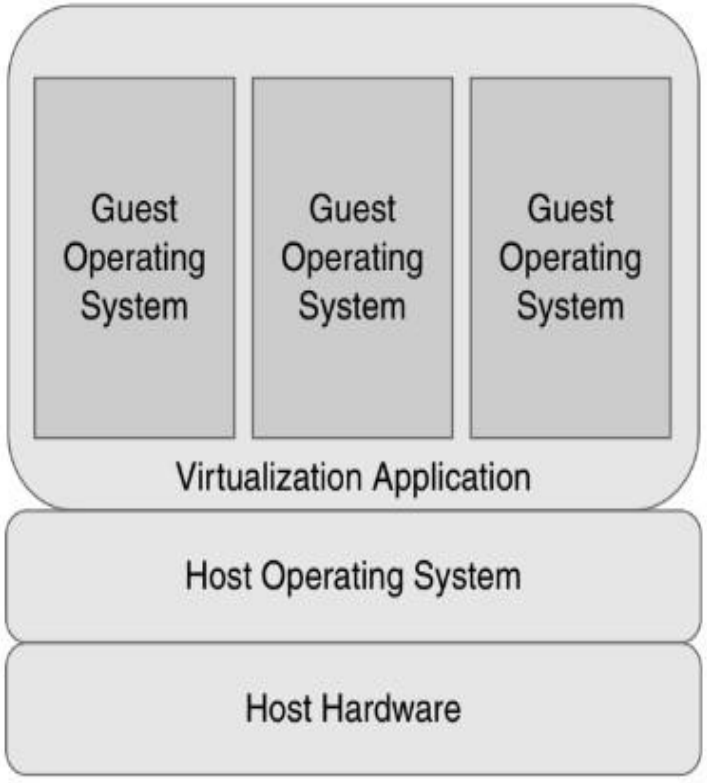


Figure 8.6 OS Virtualization

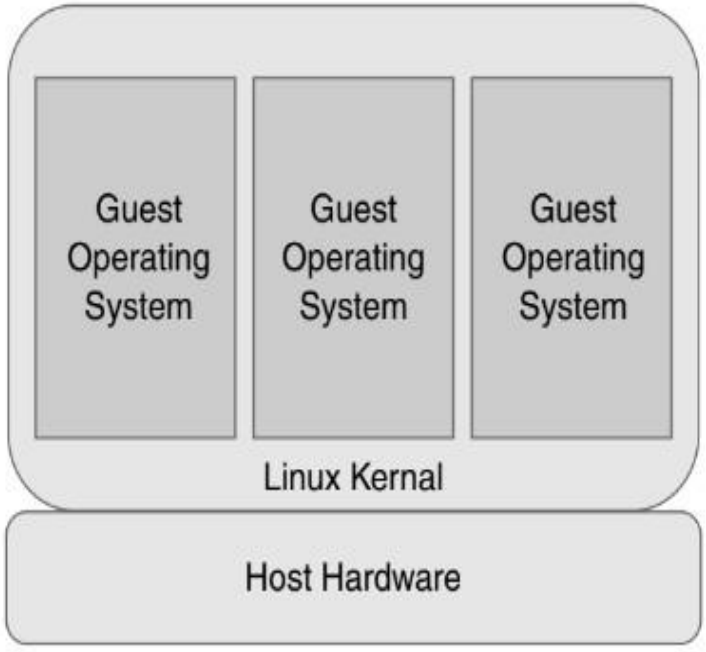
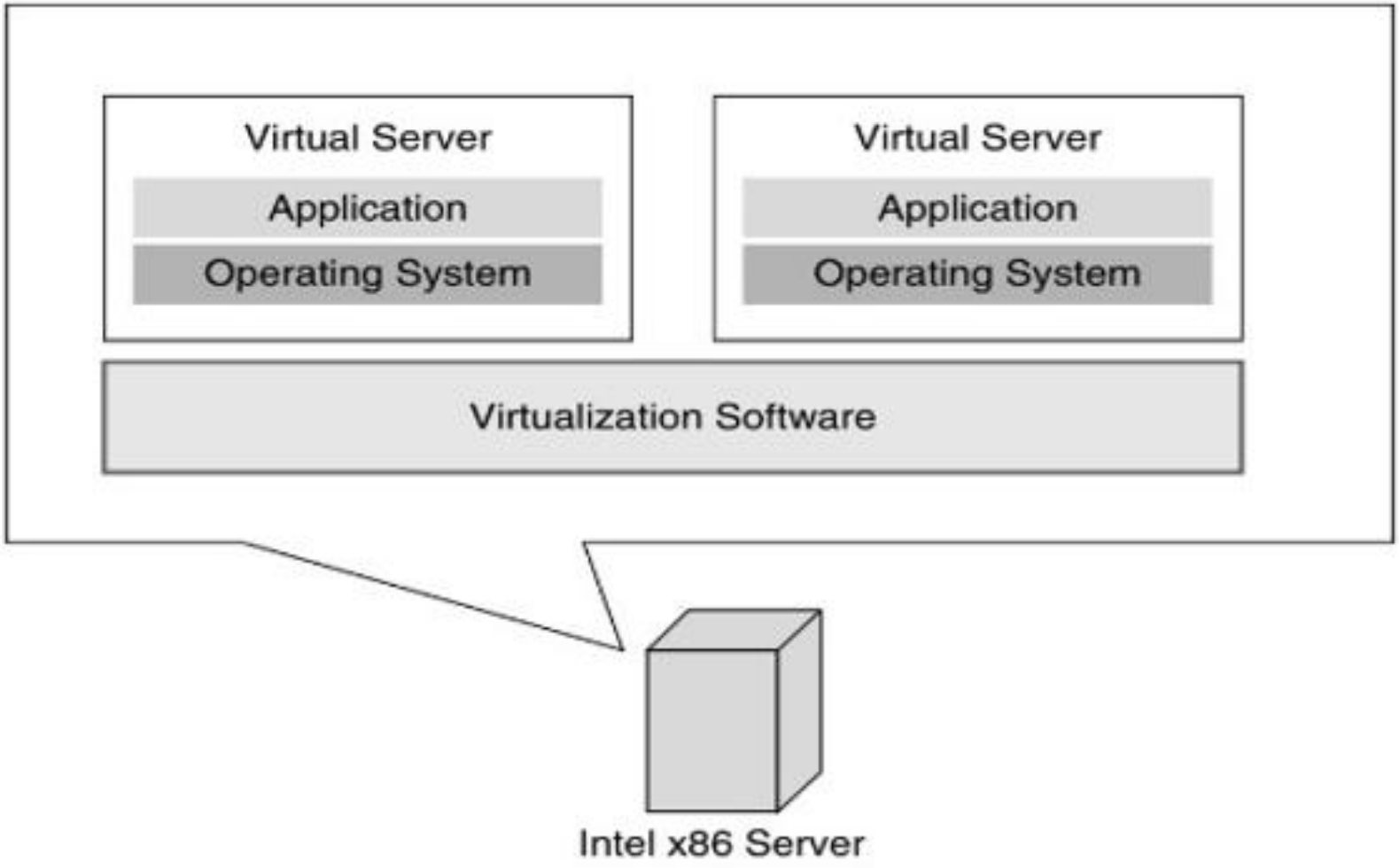
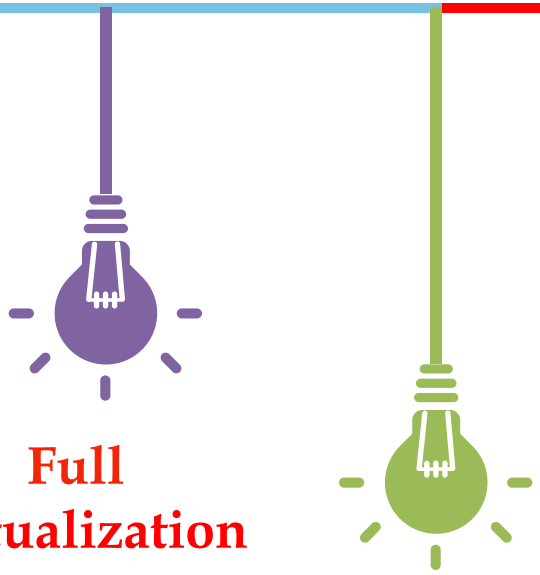


Figure 8.7 Kernel Level Virtualization

Server Virtualization



Types of Server Virtualization



Full Virtualization

Para Virtualization

Full Virtualization

- The physical server's resources are monitored by the hypervisor.
- Virtual servers operate independently and are not aware of other virtual servers using hypervisor.

Para Virtualization

- Hypervisor uses less processing power to manage virtual OS.
- It work based on network administrator

Uses of Virtualization

- Cost reduction in infrastructure such as hardware and its maintenance
- Utilization of resource to the fullest
- Increased efficiency of server
- Increased security

Storage Virtualization

- ✓ Storage virtualization is the *process of* **grouping the physical storage from multiple network storage devices** *so that it looks like a single storage device*.
- ✓ Storage virtualization is also implemented by using **software applications**.
- ✓ Storage virtualization is mainly done for **back-up and recovery purposes.**

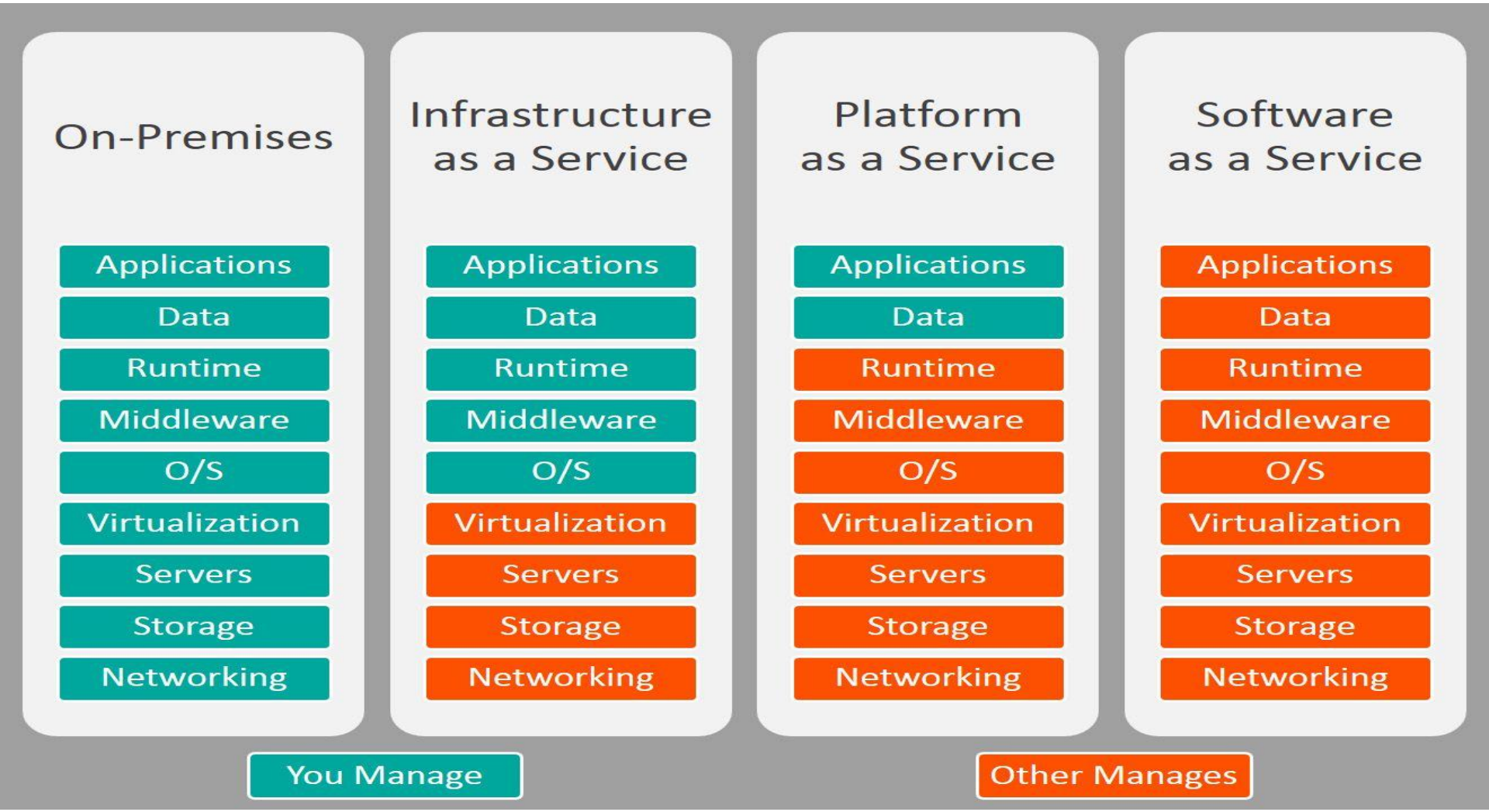
Virtualization Characteristics

- ✓ Increased Security
- ✓ Managed Execution
- ✓ Sharing
- ✓ Aggregation
- ✓ Emulation
- ✓ Isolation
- ✓ Portability

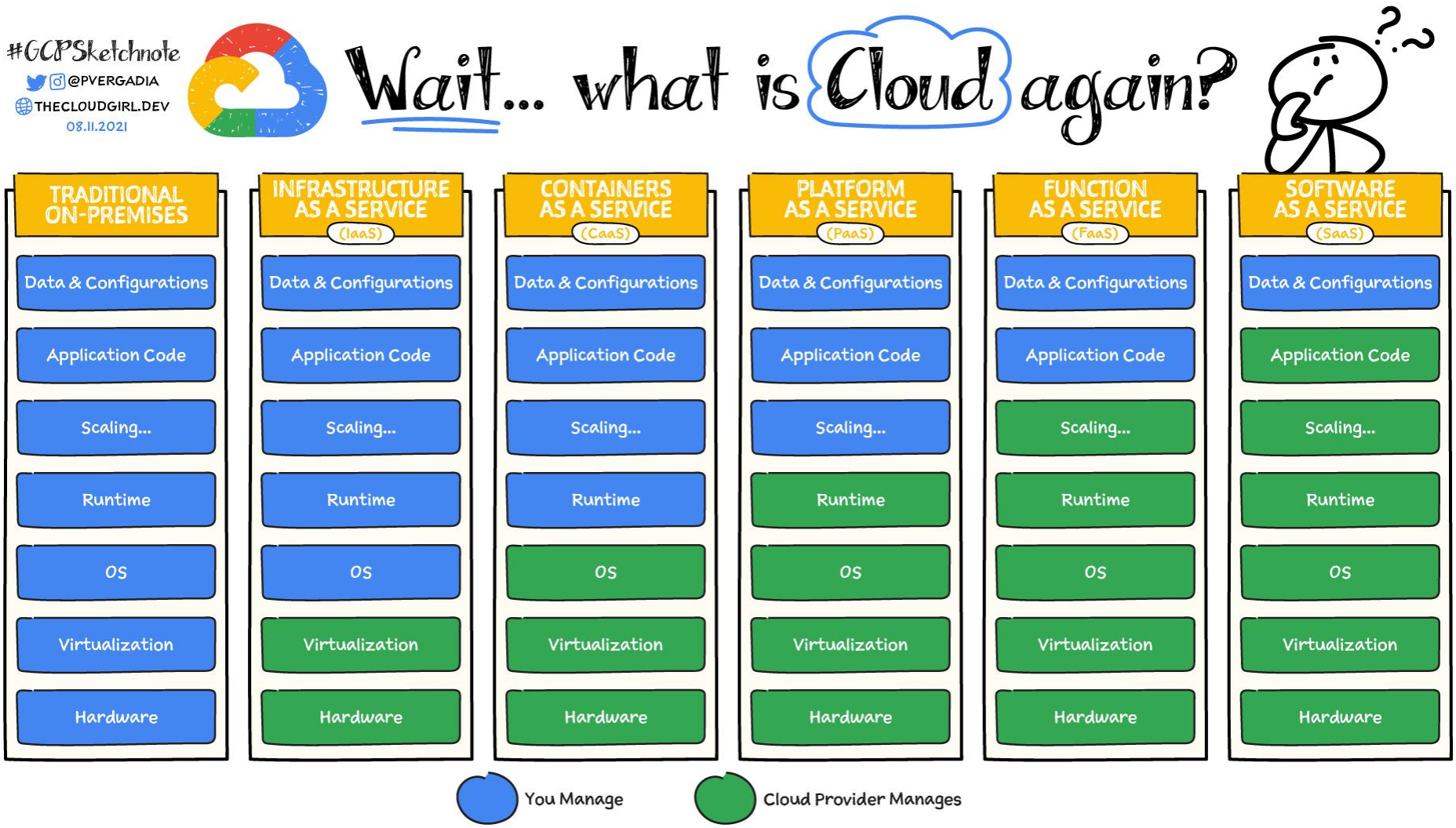
Benefits of Virtualization

- ✓ 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.

Manage Resources in Cloud Service Models



Manage Resources in Cloud Service Models



Thank You