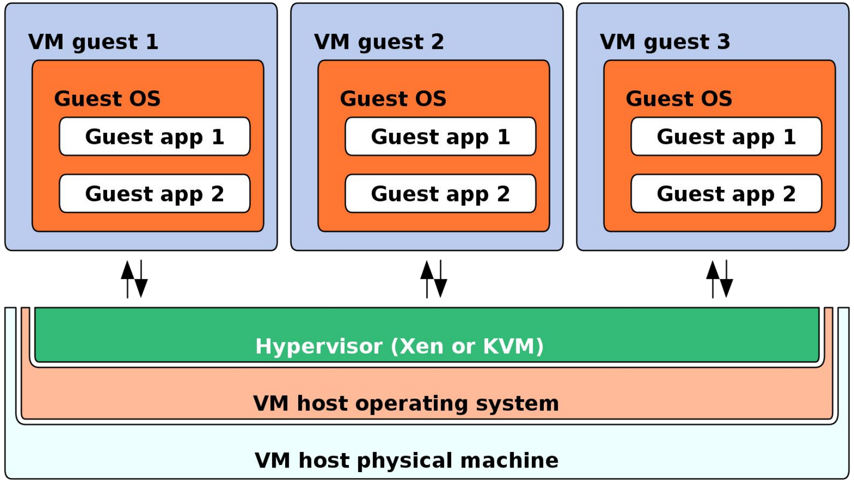
Virtualization and Cloud Computing

Dr. Manish T I
Professor
Dept of CSE
SCMS School of Engineering and Technology, Kerala
manish@scmsgroup.org, manishti2004@gmail.com

- Virtualization is the ability to run multiple operating systems on a single physical system and share the underlying hardware resources
- It is the process by which one computer hosts the appearance of many computers.
- Virtualization is used to improve IT throughput and costs by using physical resources as a pool from which virtual resources can be allocated.

Virtualization Architecture

A Virtual machine (VM) is an isolated runtime environment (guest OS and applications). Multiple virtual systems (VMs) can run on a single physical system



Hypervisor

A hypervisor is a virtual machine manager or monitor (VMM), or virtualization manager, is a program that allows multiple operating systems to share a single hardware host.

Each guest operating system appears to have the host's processor, memory, and other resources all to itself.

Benefits of Virtualization

Sharing of resources helps cost reduction

Isolation: Virtual machines are isolated from each other as if they are physically separated

Encapsulation: Virtual machines encapsulate a complete computing environment

Hardware Independence: Virtual machines run independently of underlying hardware

Portability: Virtual machines can be migrated between different hosts.

Virtualization in Cloud Computing

You don’t need to own the hardware. Resources are rented as needed from a cloud. You get billed only for what you used

creating virtual servers

–Choose the OS and software each instance will have

–The chosen OS will run on a large server farm

–Can instantiate more virtual servers or shut down existing ones within minutes

Virtualization

Virtualization refers to the representation of physical computing resources in simulated form having made through the software.

This special layer of software (installed over active physical machines) is referred as layer of virtualization.

This layer transforms the physical computing resources into virtual form which users use to satisfy their computing needs.

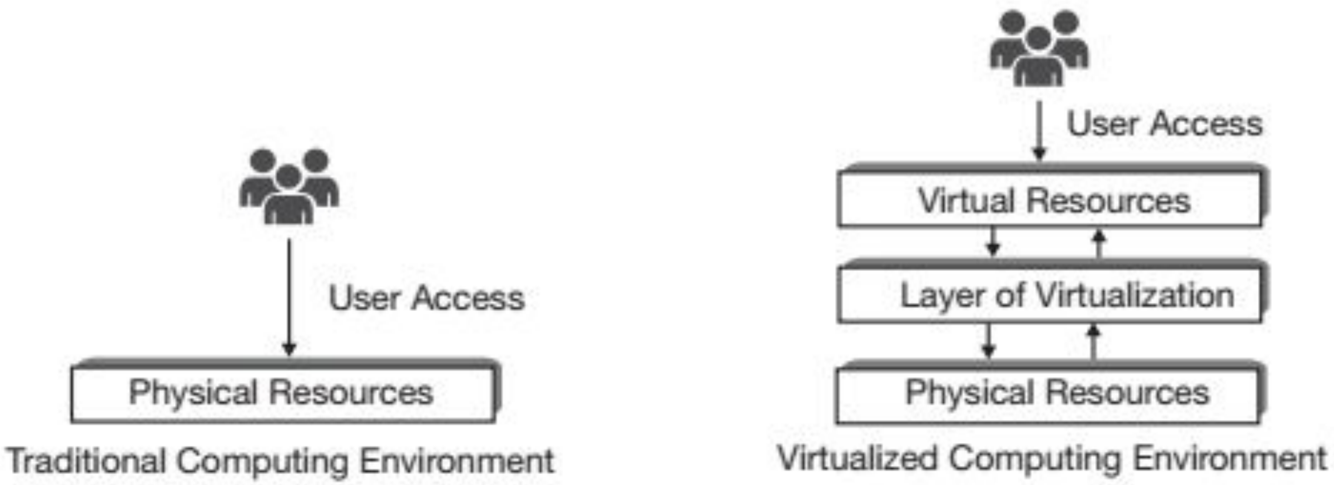


FIG 7.1: Users' interaction with computer in traditional and virtualized computing environment

Virtualization provides a level of logical abstraction that liberates user-installed software (starting from operating system and other systems as well as application software) from being tied to a specific set of hardware.

Rather, the users install everything over the logical operating environment (rather than physical ones) having created through virtualization.

Virtualization decouples the physical computing resources from direct access of users.

VIRTUALIZING PHYSICAL COMPUTING RESOURCES

The layers of virtualization transforms these physical computing devices into virtual form and presents them before user.

Apart from basic computing devices like processor, primary memory, other resources like storage, network devices, the communication links and peripheral devices can also be virtualized.

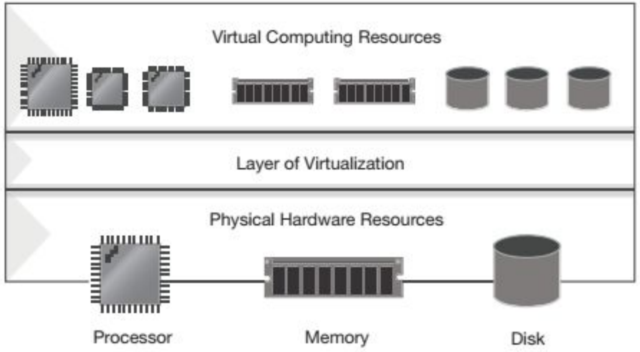


FIG 7.2: Transformation of physical resources into virtual resources through virtualization

The software for virtualization consists of a set of control programs. It offers all of the physical computing resources in custom made simulated (virtual) form which users can utilize to build virtual computing setup or virtual computers or virtual machines (VM).

Abstraction is the process of hiding the complex and non-essential characteristics of a system.

A system can be presented in simplified manner for some particular use after omitting unwanted details from users.

Virtualization creates a layer of abstraction and masks physical resources from external access

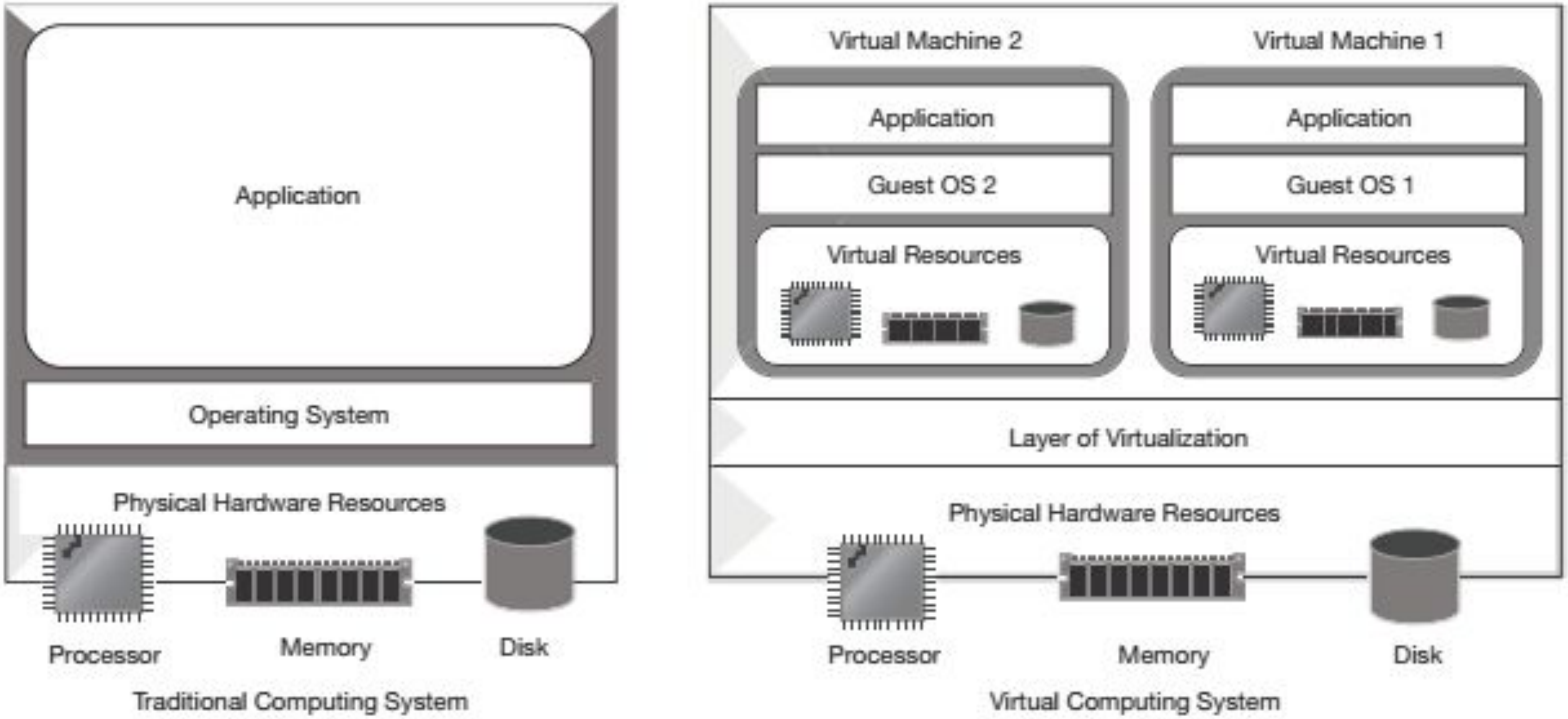

Application
Operating System
Physical Hardware Resources
Processor
Memory
Disk
Traditional Computing System
Virtual Machine 2
Virtual Machine 1
Application
Application
Guest OS 2
Guest OS 1
Virtual Resources
Virtual Resources
Layer of Virtualization
Physical Hardware Resources
Processor
Memory
Disk
Virtual Computing System

Table 7.1 Comparison between non-virtualized and virtualized machine environments

Non-Virtualized Machine Environment	*Virtualized Machine Environment*
At a moment, one single OS can run on a physical machine.	Multiple OS can run simultaneously on one physical machine.
Application and hardware system remain tightly coupled.	Virtual Machines isolates applications from the underlying hardware.
Resource utilization rate is low in most of the times.	Resource utilization improves as multiple VMs share same set of physical resources.
These increase cost of business due to low resource utilization.	They are cost-effective if planned properly.
They have the inflexible approach.	They provide lot of flexibility to system designers.

MACHINE OR SERVER LEVEL VIRTUALIZATION

- **Machine virtualization** (also called **server virtualization**) is the concept of creating virtual machine (or **virtual computer**) on actual physical machine.
- The parent system on which the virtual machines run is called the **host system**, and the virtual machines are themselves referred as **guest systems.**

- Virtual machines are created over the virtualization layers.
- A set of control programs that creates the environment for the virtual machines to run on.
- This layer provides the access to the system resources to the virtual machines.
- It also controls and monitors the execution of the virtual machines over it.
- Software layer is referred as the **Hypervisor or Virtual Machine Monitor (VMM).**

- The hypervisor abstracts the underlying software and/or hardware environments and represents virtual system resources to its users.
- Facilitates the existence of multiple VMs those are not bound to share same OS kernel.
- The hypervisor layer provides an **administrative system console** through which the virtual system environment can be managed.

Machine Virtualization Techniques

Two different techniques of server or machine virtualization as

- ➔ Hosted approach
- ➔ Bare metal approach

The techniques differ depending on the **type of hypervisor used**. Although the techniques are different but they have the same end or ultimate goal by creating a platform where multiple virtual machines can share same system resources

Hosted Approach

- An operating system is first installed on the physical machine to activate it.
- OS installed over the host machine is referred as **host operating system**.
- The hypervisor is then installed over this host OS
- **Type 2 hypervisor or Hosted hypervisor**
- Host OS works as the **first layer** of software over the physical resources.
- Hypervisor is the **second layer** of software
- Guest operating systems run as the **third layer** of software.
- **VMWare Workstation and Microsoft Virtual PC** are the most common examples of type 2 hypervisors.

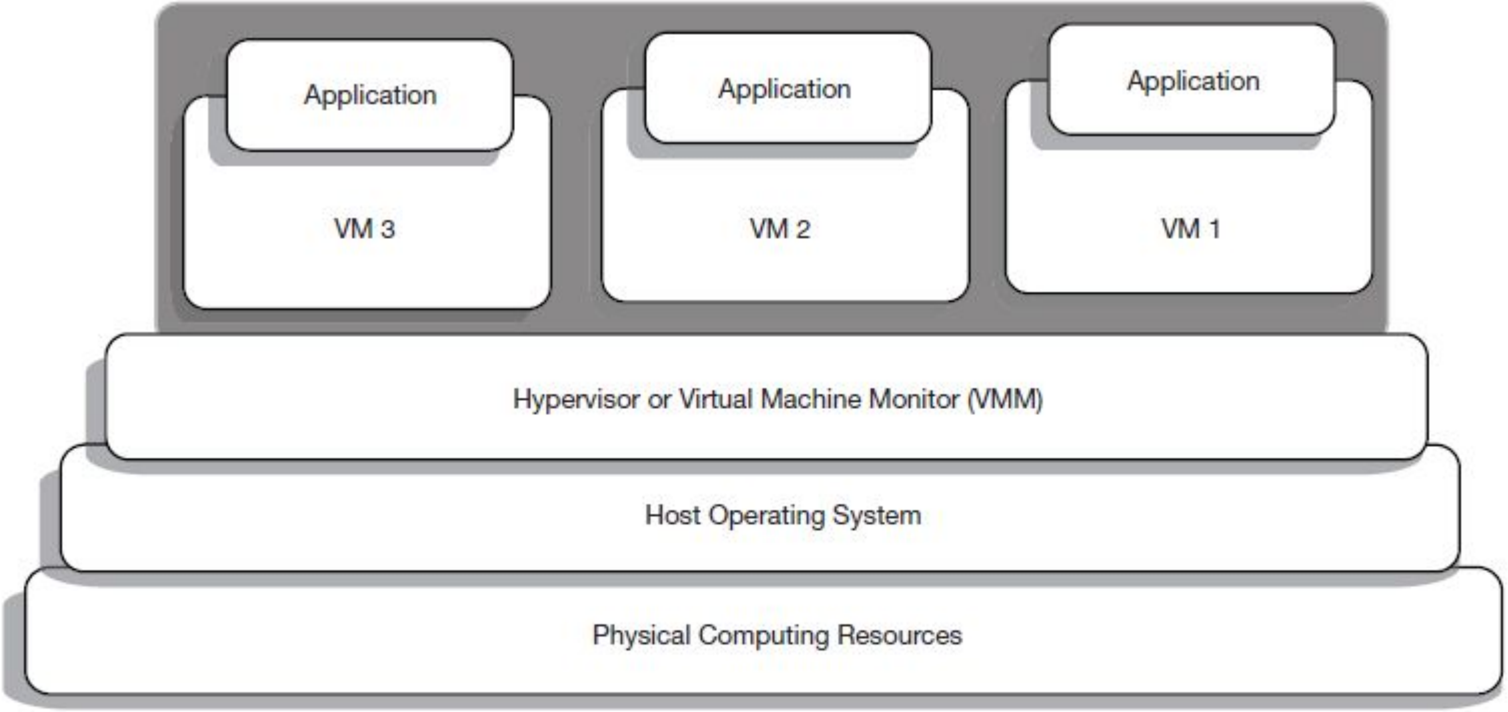


FIG 7.4: A model of hosted machine virtualization approach

Benefits:

- The host OS supplies the hardware drivers for the underlying physical resources.
- This eases the installation and configuration of the hypervisor.
- It makes the type-2 hypervisors compatible for a wide variety of hardware platform.

Drawbacks:

- A hosted hypervisor does not have direct access to the hardware resources and all of the requests from virtual machines must go through the host OS.
- Degrade the performance of the virtual machines.
- Lack of support for real-time operating systems.
- Host OS controls the scheduling of jobs it becomes unrealistic to run a real-time OS inside a VM using hosted virtualization.

Bare Metal Approach: Removal of the Host OS

- The hypervisor is directly installed over the physical machine.
- The hypervisor is the first layer over hardware resources, hence, the technique is referred as bare metal approach.
- The VMM or the hypervisor communicates directly with system hardware.
- The hypervisor acts as low-level virtual machine monitor and also called as **Type 1 hypervisor or Native Hypervisor.**
- **VMware's ESX and ESXi Servers, Microsoft's Hyper-V, solution Xen** are some of the examples of bare-metal hypervisors.

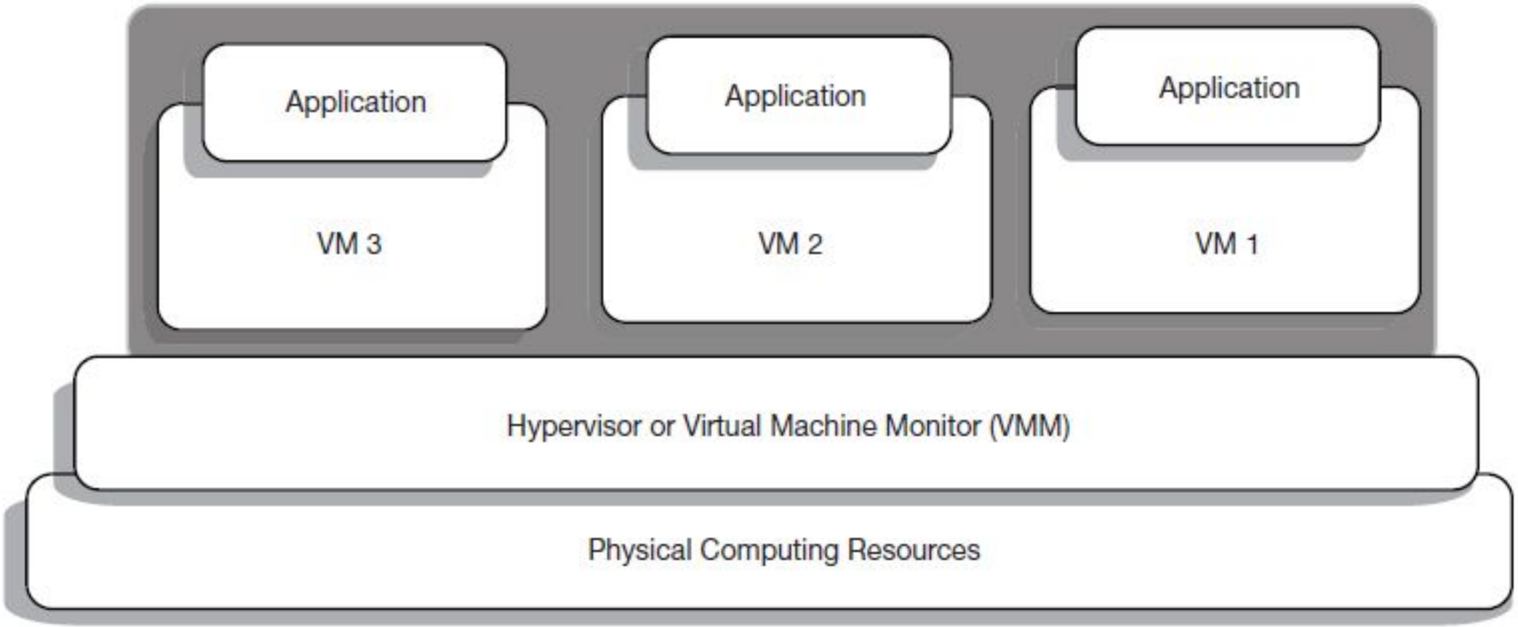


FIG 7.5: A model for the bare metal approach of machine virtualization

Benefits:

- It provides better performance in comparison to the hosted hypervisor.
- For bigger application like enterprise data centers, bare-metal virtualization is more suitable because usually it provides advanced features for resource and security management.
- Administrators get more control over the host environment.

Drawbacks:

- As any hypervisor usually have limited set of device drivers built into it, so the bare metal hypervisors have limited hardware support and cannot run on a wide variety of hardware platform.

Summary

Type 2 or hosted hypervisors run within an operating-system environment.

Type 1 or bare metal hypervisor does not use any host operating system.

Exploring Hypervisor or Virtual Machine Monitor

The hypervisor or virtual machine monitor (VMM) presents a virtual operating platform before the guest systems.

It also monitors and manages the execution of guest systems and the virtual machines.

A hypervisor or VMM facilitates and monitors the execution of virtual machines and allows the sharing of the underlying physical resources among them.

Hypervisor-Based Virtualization Approaches

Hypervisor-based virtualization techniques can be divided into three categories as

- Full virtualization
- Para-virtualization
- Hardware-assisted virtualization

Full Virtualization

- In full virtualization (also called as native virtualization), the hypervisor fully simulates or emulates the underlying hardware.
- Virtual machines run over these virtual set of hardware.
- The guest operating systems assume that they are running on actual physical resources and thus remain unaware that they have been virtualized.
- This enables the unmodified versions of available operating systems (like **Windows, Linux** and else) to run as guest OS over hypervisor.

- In this model, it is the responsibility of the hypervisor to handle all OS-to-hardware (i.e.guest OS to physical hardware) requests during running of guest machines.
- The guest OS remains **completely isolated from physical resource** layers by the hypervisor.
- This provides flexibility as almost all of the available operating systems can work as guest OS.
- VMware's virtualization product **VMWare ESXi Server and Microsoft Virtual Server** are few examples of full virtualization solution.

In full virtualization technique, the guest operating systems can directly run over hypervisor.

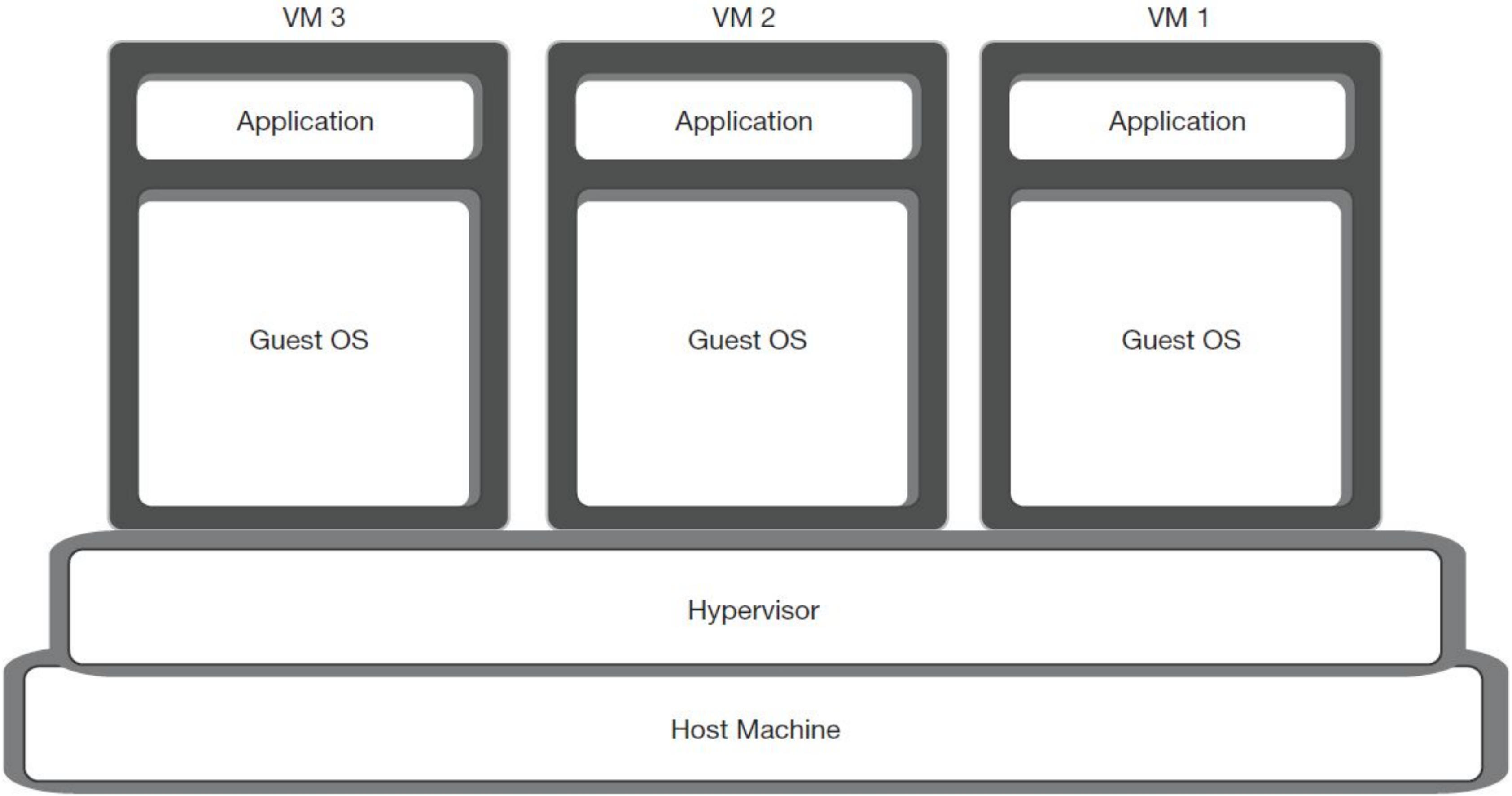


FIG 7.6: A model of full virtualization

Para-Virtualization or OS-Assisted Virtualization

- ‘Para’ is an English affix of Greek origin that means **‘beside’ or ‘alongside’.**
- In para-virtualization, a portion of the virtualization management task is transferred (from the hypervisor) towards the guest operating systems.
- Normal versions of available operating systems are not capable of doing this. They need special modification for this capability inclusion. This modification is called **porting.**

- Each guest OS is explicitly ported for the **para-application program interface (**API).
- Each guest OS needs to have prior knowledge that it will run over the virtualized platform. It also has to know on which particular hypervisor they will have to run. Depending on the hypervisor, the guest OS is modified as required to participate in the virtualization management task.
- **OS-Assisted Virtualization**
- Best known example of paravirtualization hypervisor is the open-source Xen project which uses a customized Linux kernel.

Advantages

- Para-virtualization allows calls from guest OS to directly communicate with hypervisor (without any binary translation of instructions). The use of modified OS **reduces the virtualization overhead** of the hypervisor as compared to the full virtualization.

- In para-virtualization, the system is **not restricted by the device drivers provided by the virtualization software layer.** In fact, in para-virtualization, the virtualization layer (hypervisor) does not contain any device drivers at all. Instead, the **guest operating systems contain the required device drivers.**

Limitations

■ Unmodified versions of available operating systems (like Windows or Linux) are not compatible with para-virtualization hypervisors. Modifications are possible in **Open source operating systems** (like Linux) by the user. But for proprietary operating systems (like Windows), it depends upon the owner. If owner agrees to supply the required modified version of the OS for a hypervisor, then only that OS becomes available for the paravirtualization system.

■Security is compromised in this approach as the guest **OS has a comparatively more control of the underlying hardware.** Hence, the users of some VM with wrong intentions have more chances of causing harm to the physical machine.

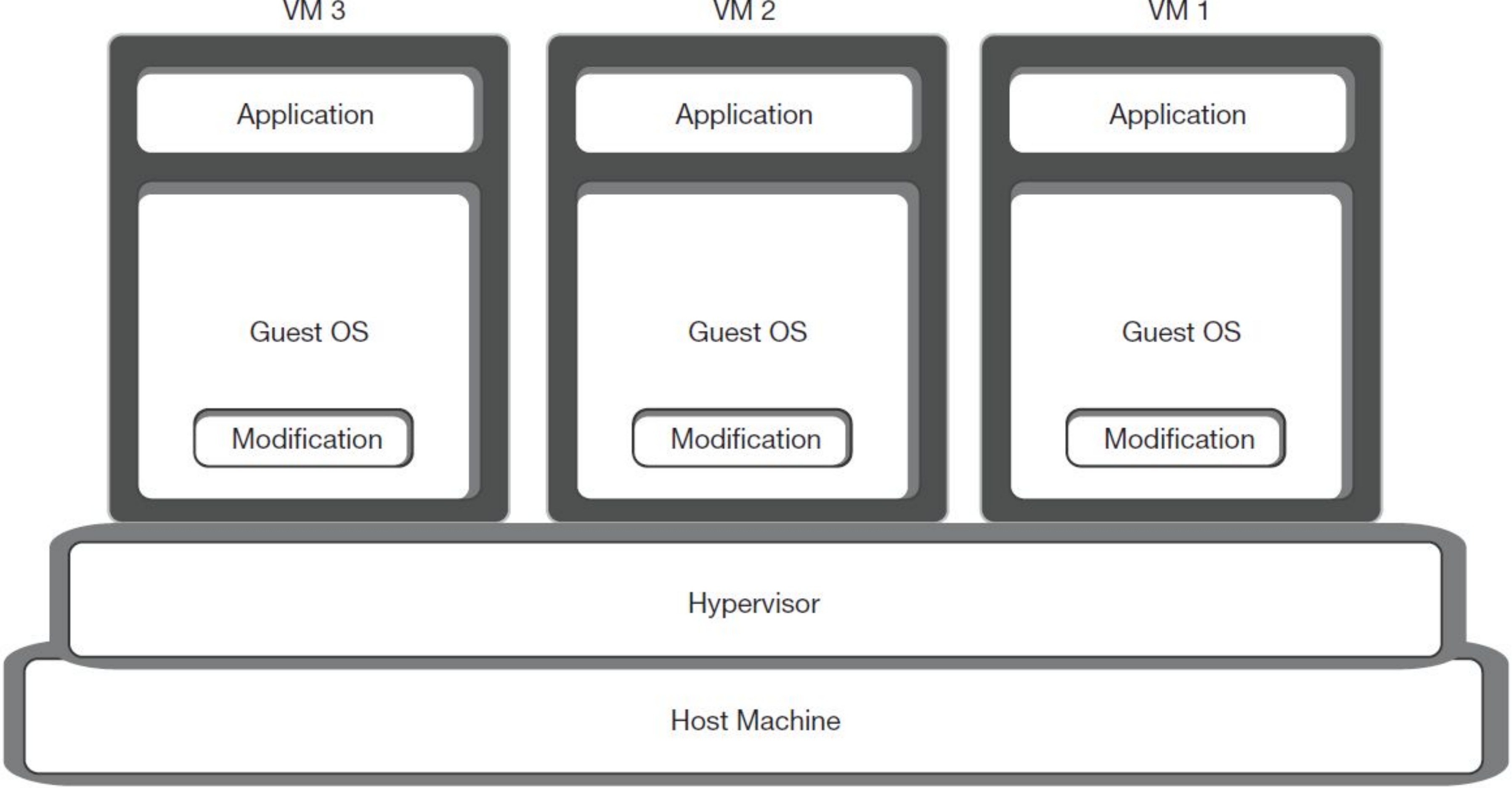


FIG 7.7: A model of para-virtualization

Summary

Para-virtualization requires hypervisor-specific modifications of guest operating systems.

Para-virtualization can provide enhanced virtualization performance at the cost of security.

Performance Comparison Through an Example:

Para-virtualization reduces the load of host machine and can run more number of VMs over a host machine in comparison to full virtualization.

In a fully virtualized environment, each guest machine consumes 10 percent of the processing power of host machine whereas in para-virtualization, they consume 4 percent only. In addition to this, the hypervisor (on which the virtual machines are running) consumes a certain amount of processing power to support each virtual machines.

Table 7.2 A comparison between processing power utilization in full- and para-virtualization

	VM Instances	*Resource engaged to process Virtualization Overhead*	*Resource engaged to process VMs*	*Total Processing power engaged*
Full virtualization	5	10% per VM (50% of total)	10% per VM (50% of total)	100%
Para-virtualization	5	4% per VM (20% of total)	10% per VM (50% of total)	70%
Para-virtualization	7	4% per VM (28% of total)	10% per VM (70% of total)	98%

The table suggests that in full virtualization, six guest system as running, it can make a host system starve. While in para-virtualization, more number of guest machines can be accommodated which creates scope for better scaling. Data presented here are indicative in nature and much better para-virtualization performance is achieved in reality.

Hardware-Assisted Virtualization

- Inspired by software-enabled virtualization, hardware vendors later started manufacturing devices tailored to support virtualization.
- Intel and AMD started this by including new virtualization features in their processors.
- The **AMD-Virtualization (AMD-V) and Intel Virtualization Technology (Intel-VT)** allows some privileged CPU calls from the guest OS to be directly handled by the CPU. These calls do not require to be translated by the hypervisors.
- This kind of virtualization is only possible when specific combinations of hardware components are used, and that did not happen until **2006** when both Intel and AMD started to include new virtualization features in their processors.
- **Hypervisors like Xen, Microsoft's Hyper-V or VMWare ESXi Server** can take the advantages of the hardware-assisted virtualization.

Hardware-assisted virtualization requires explicit features in the host machine's CPU.

Table 7.3 Comparison among Full Virtualization, Para-Virtualization and Hardware-Assisted Virtualization

Full Virtualization	*Para-Virtualization or OS-Assisted Virtualization*	*Hardware-Assisted Virtualization*
Guest OS has no role in virtualization.	Guest OS plays role in virtualization.	Guest OS has no role in virtualization.
Guest OS remains unaware about the virtualization.	Guest OS has to be aware about the virtualization.	Guest OS remains unaware about the virtualization.
Normal version of available OS can be used as guest OS.	Modified version of available OS is required.	Normal version of available OS can be used as guest OS.
It provides good options for guest OS.	It provides lesser options for guest OS.	It provides good options for guest OS.
Guest OS is not hypervisor-specific.	Guest OS is tailored to be hypervisor-specific.	Guest OS is not hypervisor-specific.
Here it requires no special feature in the host CPU.	Here it requires no special feature in the host CPU.	Here it requires explicit features in the host CPU.
Hardware does not play role in virtualization.	Hardware does not play role in virtualization.	Hardware plays role in virtualization.
Hypervisor takes care of all of the virtualization tasks.	Guest OS along with hypervisor take care of the virtualization tasks.	Specialized hardware device along with hypervisor take care of virtualization tasks.
Virtualization overhead of hypervisor is more.	Virtualization overhead of hypervisor is less.	Virtualization overhead of hypervisor is less.
Virtualization performance is little slow.	Virtualization performance is better.	Virtualization performance is better.
It provides high level of security as all of the virtualization controls remain with the hypervisor.	Here the security is compromised as guest OS has some control in virtualization.	Here the security is compromised as calls from guest OS can directly access the hardware.

OPERATING SYSTEM LEVEL VIRTUALIZATION: REMOVAL OF THE HYPERVISOR

No hypervisor is used and the virtual servers are enabled by the kernel of the operating system of physical machine.

The kernel of the operating system installed over physical system is shared among all of the virtual servers running over it.

All of the virtual servers share a single kernel, it is evident that all of them will have same OS as the parent system.

The goal of this approach is to create multiple logically-distinct user-space instances (virtual servers) over a single instance of an OS kernel.

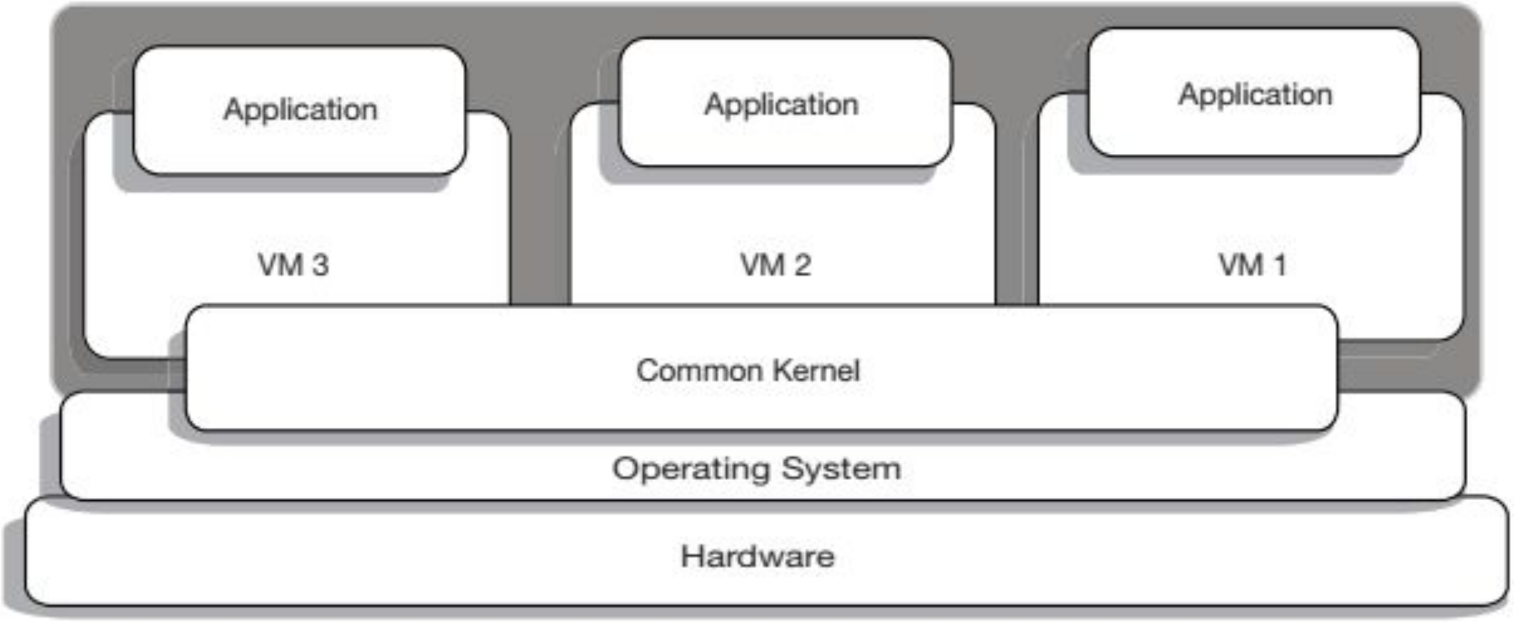


FIG 7.8. A model of operating system-level virtualization approach

Virtualization solutions such as FreeBSD’s jail, Linux VServer, OpenVZ are few examples of OS-level virtualization. All of them can run logically-distinct user-spaces on top of a single kernel.

Advantages: The advantages of OS level virtualization is that it is lighter in weight since all of the virtual servers share a single instance of an OS kernel. This enables a single physical system

to support many others virtual servers than the number of complete virtual machines it could support.

Limitations: All virtual machines have to use the same operating system . Although different distributions (like Linux distribution) of the same system kernel are allowed.

Summary

OS level virtualization facilitates the creation of multiple logically-distinct user-space instances rather than creating complete VMs.

Network Virtualization

Especially cloud-based service development requires each and every computing infrastructure in virtualized mode

Network virtualization is the process of combining network resources and network functionality into a single, software-based administrative entity called as a virtual network.

There are two common forms of network virtualization

- Virtual device-based virtual network
- Protocol based virtual network

Virtual device-based virtual network

Here, virtualized devices form the network.

All virtual networking devices (including virtual computers, virtual switches, virtual routers etc.) communicate using actual (non-virtual) network protocols such as Ethernet as well as virtualization protocols such as the VLAN.

This is actual network virtualization where the network is formed with all virtual components.

Protocol based virtual network

Rather than virtualizing devices, it creates virtual area network.

Virtual LAN (VLAN) and virtual private network (VPN) are examples of such virtualizations.

These are logical local area networks (logical LANs) where the underlying physical LAN’s structure is something else.

Here, several physical LANs which are actually part of public network (such as the Internet) can function as a single logical LAN.

This enables network devices (such as computers and switches) to send and receive data across shared or public networks as if they are part of a private network.

The devices can communicate using LAN protocols which make faster and secure network communication

Storage Virtualization

In traditional computing system, the storages have always been directly linked with the physical servers.

Virtualized storage systems are linked with servers and actual (physical) storage systems remain hidden.

Like other computing resources, virtualization of storage also happens through layer of software which creates logical abstraction of the pooling of physical storage devices having linked together by network.

Data stored in logical (virtualized) storage devices ultimately get stored in some physical storage disks.

The advent of Storage Area Networks (SAN) has made the pooling (and hence the virtualization as well) of physical storage systems easier.

There are many commercial virtualized cloud storage systems available in the market. Google Cloud Storage, Microsoft's Azure Storage, Simple Storage System (S3) and Elastic Block Store (EBS) of Amazon are few to name among them.

Desktop Virtualization

Desktop virtualization does not fall under the core category of computing infrastructure virtualization concept.

But it is the key to business as it can lower the total cost of ownership and enhances security of system, application and data.

Desktop virtualization is different from remote desktop access. Through desktop virtualization technology, any computer's applications can be separated from its desktop and user can get the look and feel of some other environment while using those applications.

For instance, VMware Fusion 7 solution provides Mac-like experience while running Windows applications on a Mac system.

In an enterprise environment, individual virtualized desktops can be maintained in a central server and users can access those desktops by connecting to the central server.

Pros & Cons of Virtualization

Pros

- Better Utilization of Existing Resources
- Reduction in Hardware Cost
- Reduction in Computing Infrastructure Costs
- Improved Fault Tolerance or Zero Downtime Maintenance
- Simplified System Administration
- Simplified Capacity Expansion
- Simplified System Installation
- Support for Legacy Systems and Applications
- Simplified System-Level Development
- Simplified System and Application Testing
- Security

Summary

The benefits of virtualization directly propagate into cloud computing and have empowered it as well

Cons

Single Point of Failure Problem

Lower Performance Issue

Difficulty in Root Cause Analysis

Summary

The positive impulse of virtualization prevails over the negatives by far