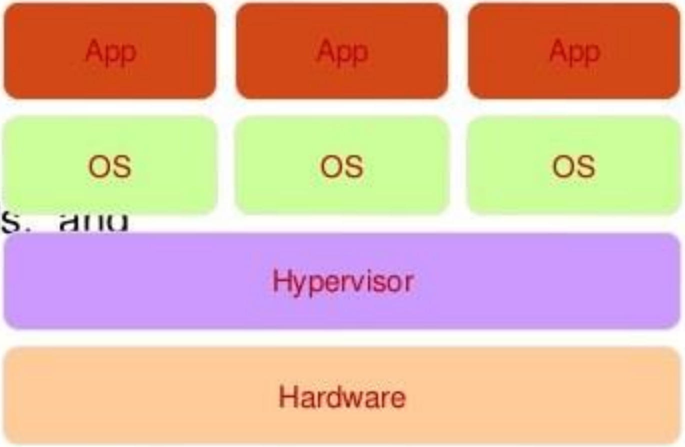
Implementation Levels of Virtualization

Virtualization technology benefits the computer and IT industries by enabling users to share expensive hardware resources by multiplexing VMs on the same set of hardware hosts. Virtual workspaces:

- An abstraction of an execution environment that can be made dynamically available to authorized clients by using well-defined protocols,
- Resource quota (e.g. CPU, memory share),
- Software configuration (e.g. O/S, provided services).

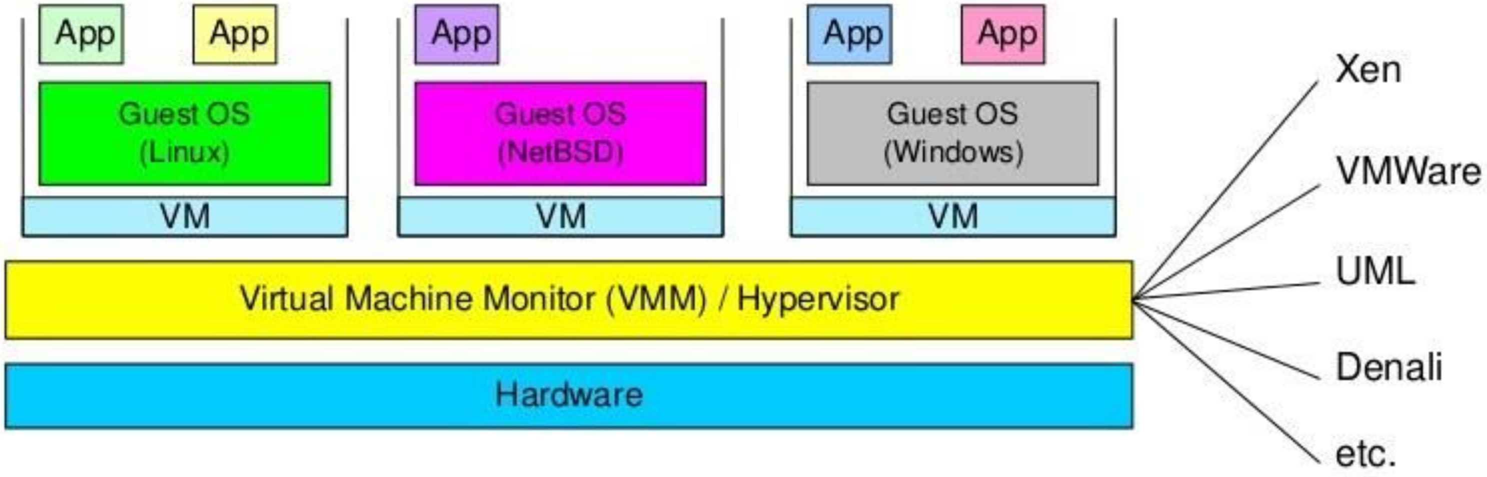
- Implement on Virtual Machines (VMs):
  - Abstraction of a physical host machine,
  - Hypervisor intercepts and emulates instructions from VMs, and allows management of VMs,
  - VMWare, Xen, etc.
- Provide infrastructure API:
  - Plug-ins to hardware/support structures



Virtualized Stack

Virtual Machines

- VM technology allows multiple virtual machines to run on a single physical machine.



Performance: Para-virtualization (e.g. Xen) is very close to raw physical performance!

Virtualization in General

Advantages of virtual machines:

- Run operating systems where the physical hardware is unavailable,
- Easier to create new machines, backup machines, etc.,
- Software testing using “clean” installs of operating systems and software,
- Emulate more machines than are physically available,
- Timeshare lightly loaded systems on one host,
- Debug problems (suspend and resume the problem machine),
- Easy migration of virtual machines (shutdown needed or not).
- Run legacy systems!

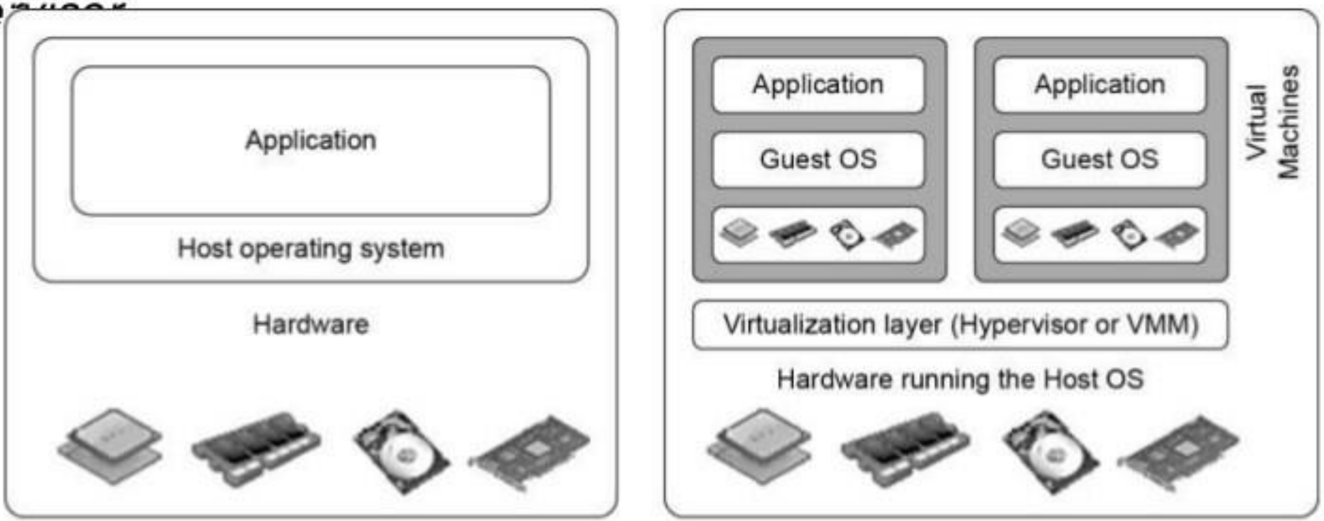
What is the purpose and benefits?

- Cloud computing enables companies and applications, which are system infrastructure dependent, to be infrastructure-less.
- By using the Cloud infrastructure on “pay as used and on demand”, all of us can save in capital and operational investment!
- Clients can:
  - Put their data on the platform instead of on their own desktop PCs and/or on their own servers.
  - They can put their applications on the cloud and use the servers within the cloud to do processing and data manipulations etc.
  - In computing, virtualization means to create a virtual version of a device or resource, such as a server, storage device, network or even an operating system where the framework divides the resource into one or more execution environments.

Difference between Traditional and Virtual machines

- A traditional computer runs with a host operating system specially tailored for its hardware architecture
- After virtualization, different user applications managed by their own operating systems (guest OS) can run on the same hardware, independent of the host OS.
- The Virtualization layer is the middleware between the underlying hardware and virtual machines represented in the system, also known as *virtual machine monitor* (VMM) or *hypervisor*.

With sufficient storage, any computer platform can be installed in another host computer, even if they use processors with different instruction sets and run with distinct operating systems on the same hardware



(a) Traditional computer

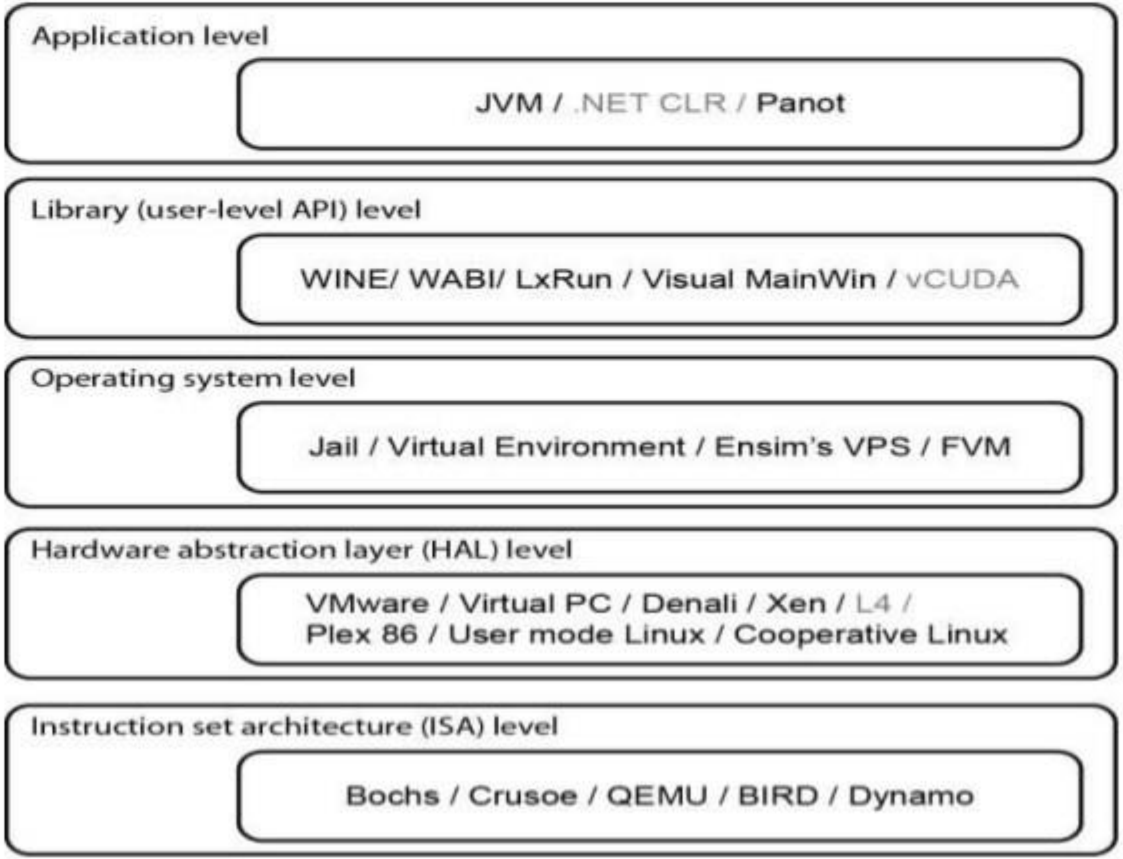
(b) After virtualization

Virtualization Layers

The virtualization software creates the abstraction of VMs by interposing a virtualization layer at various levels of a computer system. Common virtualization layers include

1. the instruction set architecture (ISA) level,
2. hardware level,
3. operating system level,
4. library support level, and
5. application level

Virtualization Ranging from Hardware to Applications in Five Abstraction Levels



Application level

JVM / .NET CLR / Panot

Library (user-level API) level

WINE/ WABI/ LxRun / Visual MainWin / vCUDA

Operating system level

Jail / Virtual Environment / Ensim's VPS / FVM

Hardware abstraction layer (HAL) level

VMware / Virtual PC / Denali / Xen / L4 /
Plex 86 / User mode Linux / Cooperative Linux

Instruction set architecture (ISA) level

Bochs / Crusoe / QEMU / BIRD / Dynamo

1.Virtualization at Instruction Set Architecture (ISA) level:

- At the ISA level, virtualization is performed by emulating a given ISA by the ISA of the host machine. Instruction set emulation leads to virtual ISAs created on any hardware machine. e.g, MIPS binary code can run on an x-86-based host machine with the help of ISA emulation.
- With this approach, it is possible to run a large amount of legacy binary code written for various processors on any given new hardware host machine.
- code interpretation – dynamic binary translation - virtual instruction set architecture (V-ISA)
- Advantage:
  - It can run a large amount of legacy binary codes written for various processors on any given new hardware host machines
  - best application flexibility
- Shortcoming & limitation:
  - One source instruction may require tens or hundreds of native target instructions to perform its function, which is relatively slow.
  - V-ISA requires adding a processor specific software translation layer in the

2.Virtualization at Hardware Abstraction level:

- Hardware-level virtualization is performed right on top of the bare hardware.
- On the one hand, this approach generates a virtual hardware environment for a VM.
- On the other hand, the process manages the underlying hardware through virtualization.
- The idea is to virtualize a computer's resources, such as its processors, memor and I/O devices. The intention is to upgrade the hardware utilization rate by multiple users concurrently.

Advantage:

- Has higher performance and good application isolation

Shortcoming & limitation:

3.Virtualization at Operating System (OS) level:

- OS-level virtualization creates isolated containers on a single physical server and the OS instances to utilize the hardware and software in data centers. The containers behave like real servers.
- OS-level virtualization is commonly used in creating virtual hosting environments to allocate hardware resources among a large number of mutually distrusting users.

Advantage:

- Has minimal startup/shutdown cost, low resource requirement, and high scalability; synchronize VM and host state changes.

Shortcoming & limitation:

- All VMs at the operating system level must have the same kind of guest OS
- Poor application flexibility and isolation.

Virtualization at OS Level

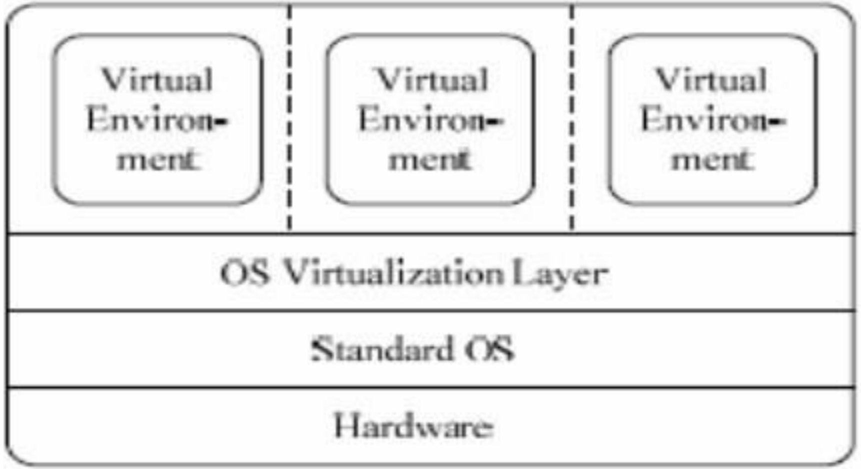


Figure 6.3 The virtualization layer is inserted inside an OS to partition the hardware resources for multiple VMs to run their applications in virtual environments

Advantages of OS Extension for Virtualization

1. VMs at OS level has minimum startup/shutdown costs
2. OS-level VM can easily synchronize with its environment

Disadvantage of OS Extension for Virtualization

- All VMs in the same OS container must have the same or similar guest OS, which restrict application flexibility of different VMs on the same physical machine.

4.Library Support level:

- Since most systems provide well-documented APIs, such an interface becomes another candidate for virtualization.
- Virtualization with library interfaces is possible by controlling the communication link between applications and the rest of a system through API hooks.
- The software tool WINE has implemented this approach to support Windows applications on top of UNIX hosts.
- Another example is the vCUDA which allows applications executing within VMs leverage GPU hardware acceleration.

Advantage:

- It has very low implementation effort

Shortcoming & limitation:

- poor application flexibility and isolation

5.User-Application Level

- Virtualization at the application level virtualizes an application as a VM. On a traditional OS, an application often runs as a process.
- Therefore, application-level virtualization is also known as process-level virtualization.
- The most popular approach is to deploy high level language (HLL) VMs. In this scenario, the virtualization layer sits as an application program on top of the operating system, and the layer exports an abstraction of a VM that can run programs written and compiled to a particular abstract machine definition.
- Other forms of application-level virtualization are known as
- application isolation,
- application sandboxing, or application streaming.

Advantage:

- has the best application isolation

Shortcoming & limitation:

- low performance, low application flexibility and high implementation complexity.

Virtualization Structures/Tools and Mechanisms

- In general, there are three typical classes of VM architecture. Figure showed th architectures of a machine before and after virtualization.
- Before virtualization, the operating system manages the hardware.
- After virtualization, a virtualization layer is inserted between the hardware and operating system. In such a case, the virtualization layer is responsible for converting portions of the real hardware into virtual hardware.
- Therefore, different operating systems such as Linux and Windows can run on same physical machine, simultaneously.
- Depending on the position of the virtualization layer, there are several classes VM architectures, namely the hypervisor architecture, para-virtualization, and h based virtualization.
- The hypervisor is also known as the VMM (Virtual Machine Monitor). They both perform the same virtualization operations.

Hypervisor

- A hypervisor is a hardware virtualization technique allowing multiple operating systems, called guests to run on a host machine. This is also called the Virtual Machine Monitor (VMM).

Type 1: bare metal hypervisor

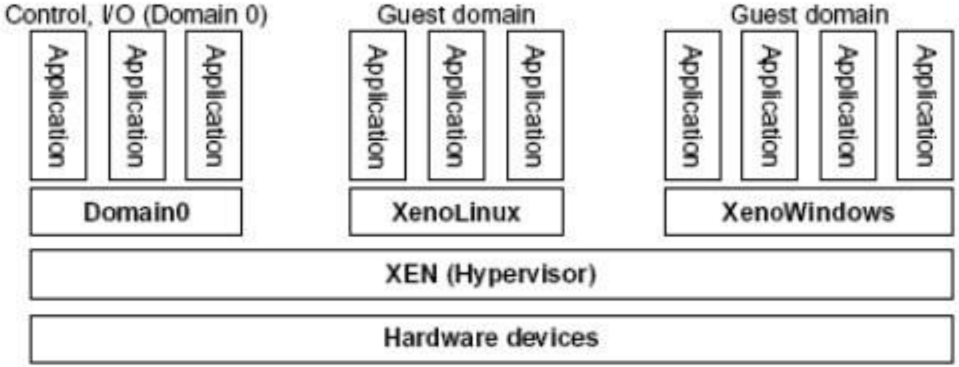
- sits on the bare metal computer hardware like the CPU, memory, etc.
- All guest operating systems are a layer above the hypervisor.
- The original CP/CMS hypervisor developed by IBM was of this kind.

Type 2: hosted hypervisor

- Run over a host operating system.
- Hypervisor is the second layer over the hardware.
- Guest operating systems run a layer over the hypervisor.
- The OS is usually unaware of the virtualization

The XEN Architecture

- Xen is an open source hypervisor program developed by Cambridge University. Xen is a micro-kernel hypervisor, which separates the policy from the mechanism.
- Xen does not include any device drivers natively . I t just provides a mechanism by which a guest OS can have direct access to the physical devices.
- As a result, the size of the Xen hypervisor is kept rather small. Xen provides a virtual environment located



Control, I/O (Domain 0)

Guest domain

Guest domain

Application Application Application

Application Application Application

Application Application Application Application

Domain0

XenoLinux

XenoWindows

XEN (Hypervisor)

Hardware devices

FIGURE 3.5

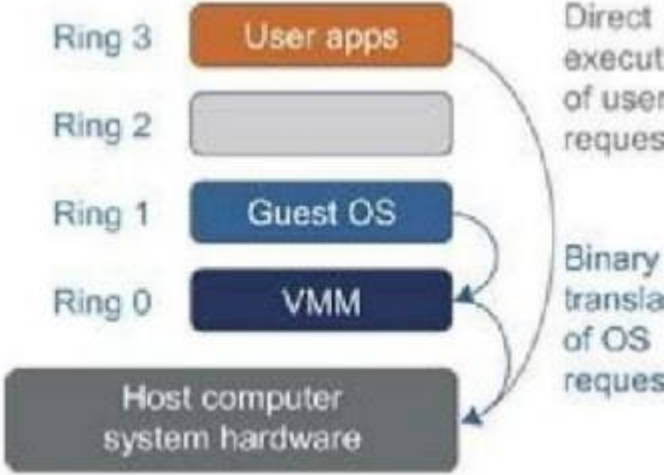
The Xen architecture's special domain 0 for control and I/O, and several guest domains for user applic

Binary Translation with Full Virtualization

- Depending on implementation technologies, hardware virtualization can be classified into two categories: full virtualization and host-based virtualization.
- Full virtualization does not need to modify the host OS. It relies on binary translation to trap and to virtualizes the execution of certain sensitive, non virtualizable instructions. The guest OSes and their applications consist of noncritical and critical instructions.
- In a host-based system, both a host OS and a guest OS are used. A virtualization software layer is built between the host OS and guest OS.
- These two classes of VM architecture are introduced next.

Binary Translation of Guest OS Requests Using a VMM

- This approach was implemented by VMware and many other software companies.
- VMware puts the VMM at Ring 0 and the guest OS at Ring 1. The VMM scans the instruction stream and identified the privileged, control- and behavior sensitive instructions.
- When these instructions are identified, they are trapped into the VMM, which emulates the behavior of these instructions.
- The method used in this emulation is called binary translation. Therefore, full virtualization combines binary translation

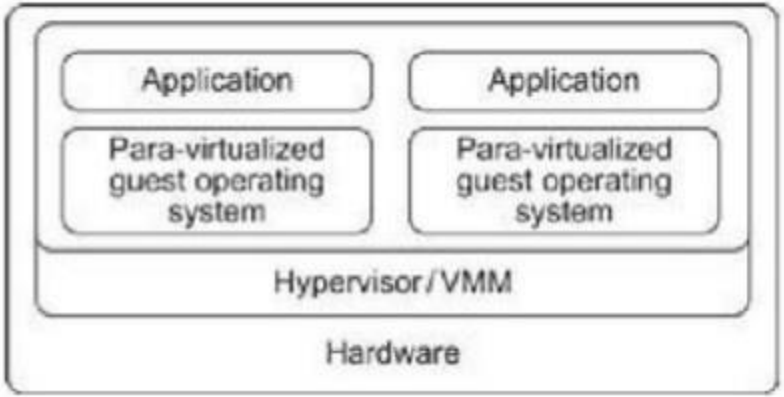

Ring 3
User apps
Ring 2
Ring 1
Guest OS
Ring 0
VMM
Host computer system hardware
Direct execut of user reques
Binary transla of OS reques

Host-Based Virtualization

- An alternative VM architecture is to install a virtualization layer on top of the host OS. This host OS is still responsible for managing the hardware.
- This host-based architecture has some distinct advantages. First, the user can install this VM architecture without modifying the host OS. The virtualizing software can rely on the host OS to provide device drivers and other low-level services. This will simplify the VM design and ease its deployment.
- Second, the host-based approach appeals to many host machine configurations. Compared to the hypervisor/VMM architecture, the performance of the host-based architecture may also be low

Para -virtualization

- Para -virtualization needs to modify the guest operating systems. A para-virtualized VM
- provides special API s requiring substantial OS modifications in user applications.
- Performance degradation is a critical issue of a virtualized system.


Application
Application
Para-virtualized guest operating system
Para-virtualized guest operating system
Hypervisor/VMM
Hardware

Full Virtualization vs. Para-Virtualization

Full virtualization

- Does not need to modify guest OS, and critical instructions are emulated by software through the use of binary translation.
- VMware Workstation applies full virtualization, which uses binary translation to automatically modify x86 software on-the-fly to replace critical instructions.

Advantage: no need to modify OS.

Disadvantage: binary translation slows down the performance.

Para virtualization

- Reduces the overhead, but cost of maintaining a paravirtualized OS is high.
- The improvement depends on the workload.
- Para virtualization must modify guest OS, non-virtualizable instructions are replaced by hyper calls that communicate directly with the hypervisor or VMM.
- Para virtualization is supported by Xen, Denali and VMware ESX.

Virtualization of CPU, Memory, and I/O Devices

CPU Virtualization

- A VM is a duplicate of an existing computer system in which a majority of the VM instructions are executed on the host processor in native mode. Thus, unprivileged instructions of VMs run directly on the host machine for higher efficiency. Other critical instructions should be handled carefully for correctness and stability.
- The critical instructions are divided into three categories: privileged instructions, control –sensitive instructions, and behavior-sensitive instructions.
- Privileged instructions execute in a privileged mode and will be trapped if executed outside this mode.
- Control-sensitive instructions attempt to change the configuration of resources used. Behavior-sensitive instructions have different

- A CPU architecture is virtualizable if it supports the ability to run the VM’s privileged
- and unprivileged instructions in the CPU’s user mode while the VMM runs in supervisor mode.
- When the privileged instructions including control- and behavior-sensitive instructions of a VM are executed, they are trapped in the VMM. I n this case, the VMM acts as a unified mediator for hardware access from different VMs to guarantee the correctness and stability of the whole system. However, not all CPU architectures are virtualizable.
- RI SC CPU architectures can be naturally virtualized because all control and behavior-sensitive instructions are privileged instructions.
- On the contrary, x86 CPU architectures are not primarily designed to support virtualization.

Memory Virtualization

- Virtual memory virtualization is similar to the virtual memory support provided by modern operating systems. I n a traditional execution environment, the operating system maintains mappings of virtual memory to ma chine memory using page tables, which is a one-stage mapping from virtual memory to machine memory.
- However, in a virtual execution environment, virtual memory virtualizatior involves sharing the physical system memory in RAM and dynamically allocating it to the physical memory of the VMs.
- That means a two-stage mapping process should be maintained by the guest OS and the VMM, respectively: virtual memory to physical memory and physical memory to machine memory.

I/O Virtualization

- there are three ways to implement I /O virtualization: full device emulation, para-virtualization, and direct I /O.
- I /O virtualization. Generally, this approach emulates well-known, real-world devices. All the functions of a device or bus infrastructure, such as device enumeration, identification, interrupts, and DMA, are replicated in software. This software is located in the VMM and acts as a virtual device.
- The para-virtualization method of I /O virtualization is typically used in Xen. I t is also known as the split driver model consisting of a frontend driver and a backend driver. It achieves beer device performance than full device emulation, it comes with a higher CPU overhead
- Direct I /O virtualization lets the VM access devices directly. I t can achieve close-to native performance without high CPU costs.

Virtual Clusters and Resource Management

- A physical cluster is a collection of servers (physical machines) interconnected by a physical network such as a LAN
- Virtual clusters are built with VMs installed at distributed servers from on or more physical clusters. The VMs in a virtual cluster are interconnecte logically by a virtual network across several physical networks. Figure illustrates the concepts of virtual clusters and physical clusters. Each virtual cluster is formed with physical machines or a VM hosted by multiple physical clusters. The virtual cluster boundaries are shown as distinct boundaries.

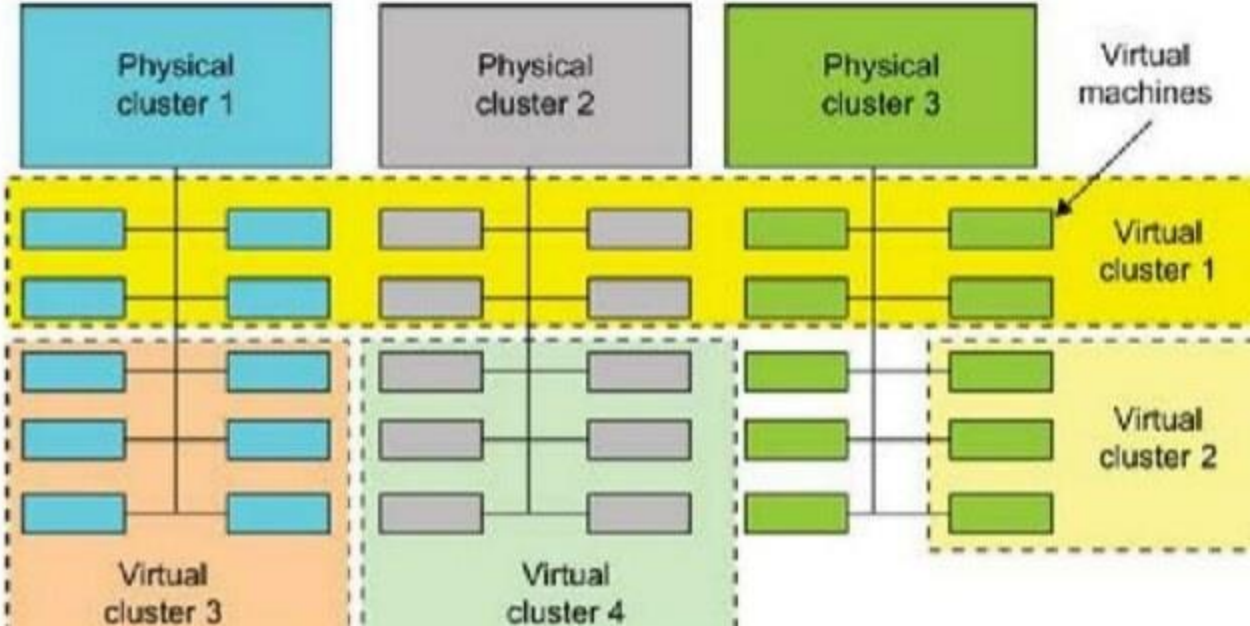
The provisioning of VMs to a virtual cluster is done dynamically to have the following interesting properties
: • The virtual cluster nodes can be either physical or virtual machines. Multiple VMs running with different OSes
can be deployed on the same physical node.

• A VM runs with a guest OS, which is often different from the host OS, that manages the resources in the phys
machine, where the VM is implemented

. • The purpose of using VMs is to consolidate multiple functionalities on the same server. This will greatly enhance server utilization and application flexibility

. • VMs can be colonized (replicated) in multiple servers for the purpose of promoting distributed parallelism, fau
tolerance, and disaster recovery.

- • The size (number of nodes) of a virtual cluster can grow or shrink dynamically, similar
to the way an overlay network varies in size a peer-to-peer (P2P) network.
- • The failure of any physical nodes may disable some VMs installed on the failing


Physical cluster 1
Physical cluster 2
Physical cluster 3
Virtual machines
Virtual cluster 1
Virtual cluster 2
Virtual cluster 3
Virtual cluster 4