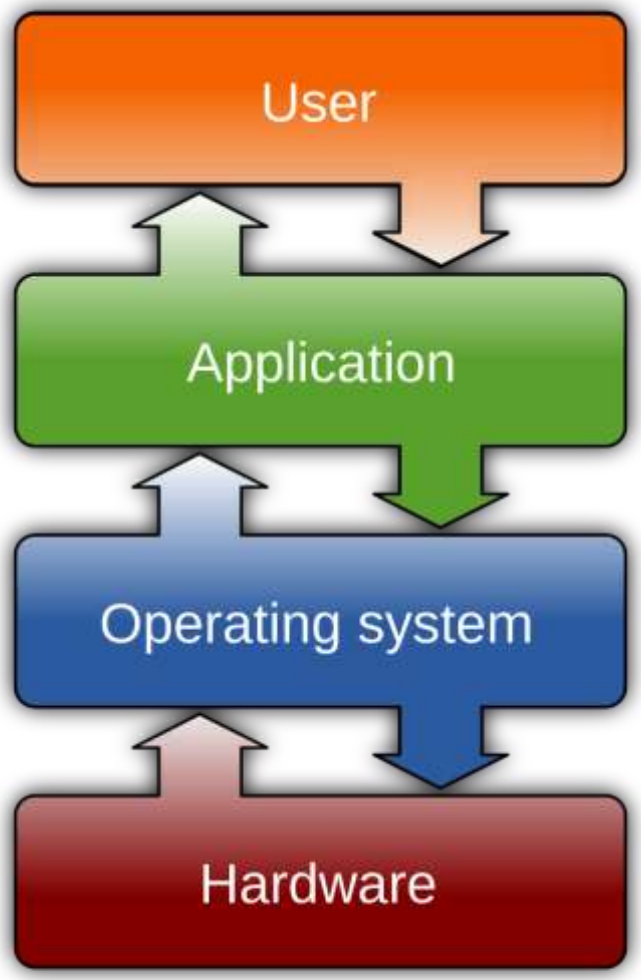
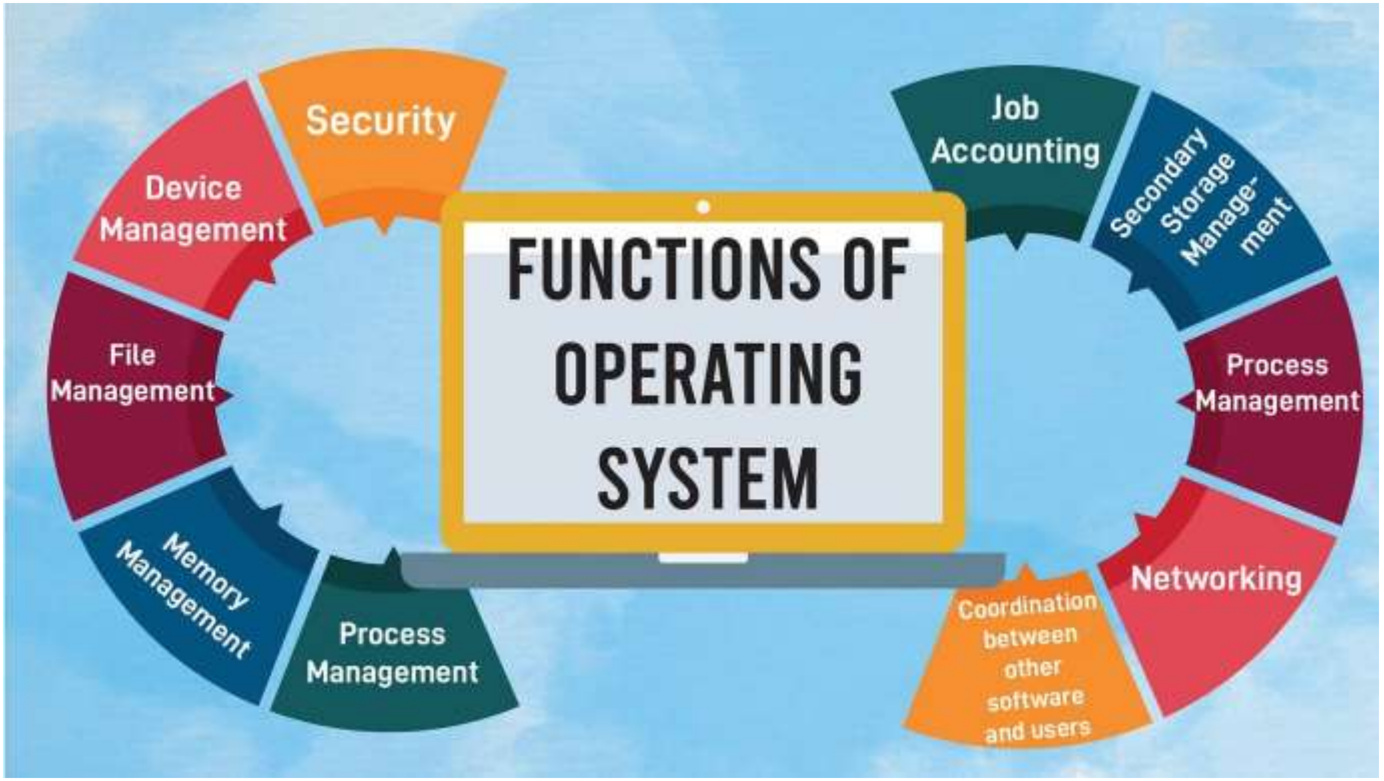
VIRTUALIZATION

Program executing without OS

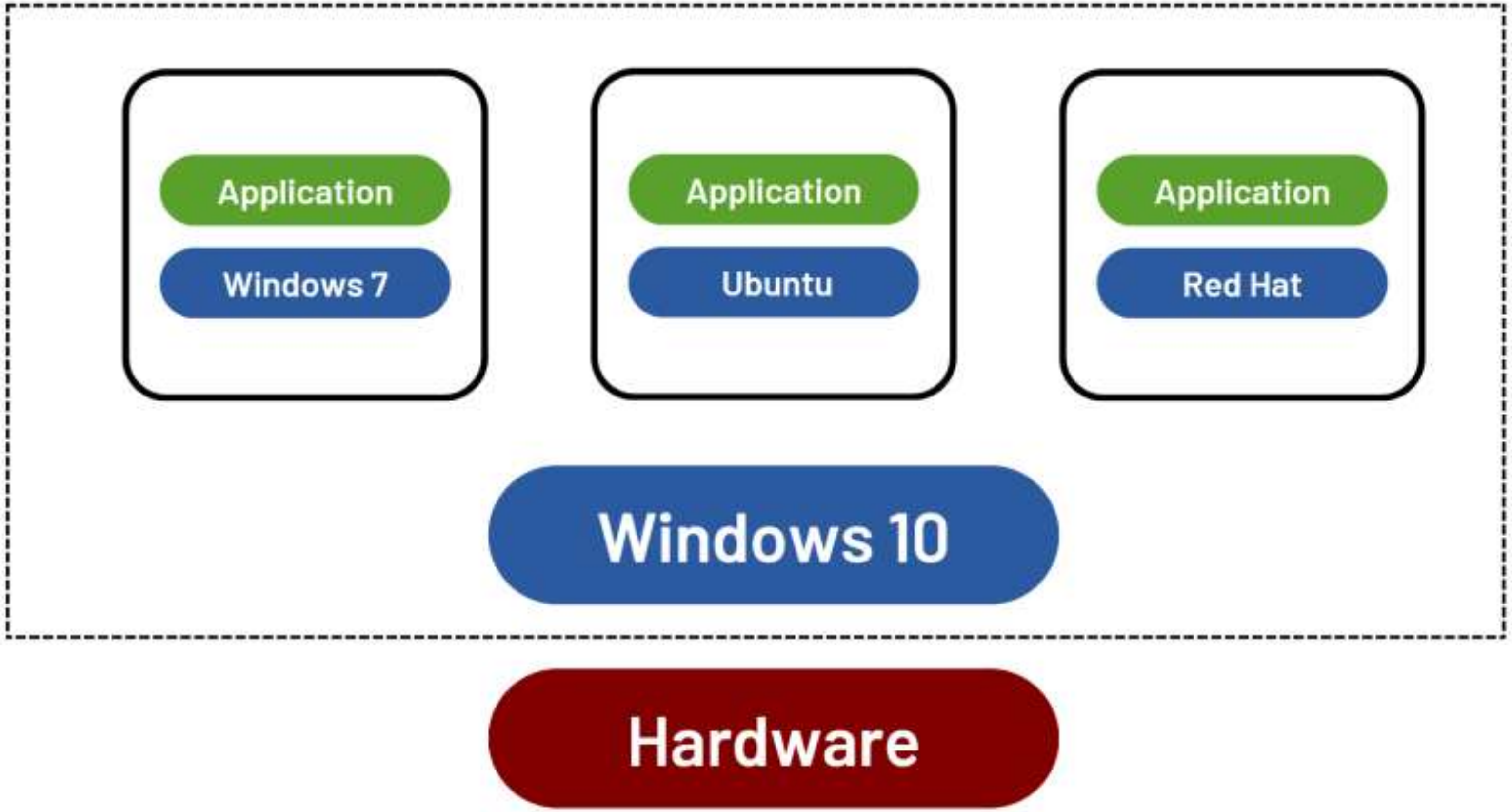
- Can you create assembly programs that can print ‘Hello World’ on screen without loading operating system?

Operating System



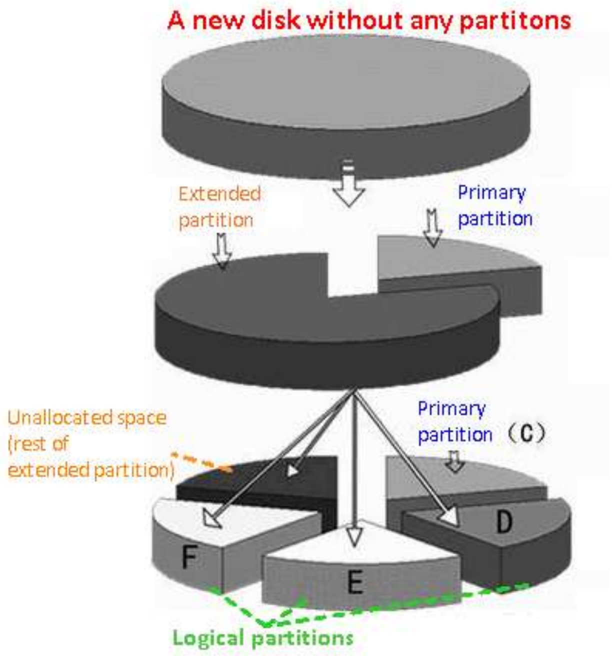


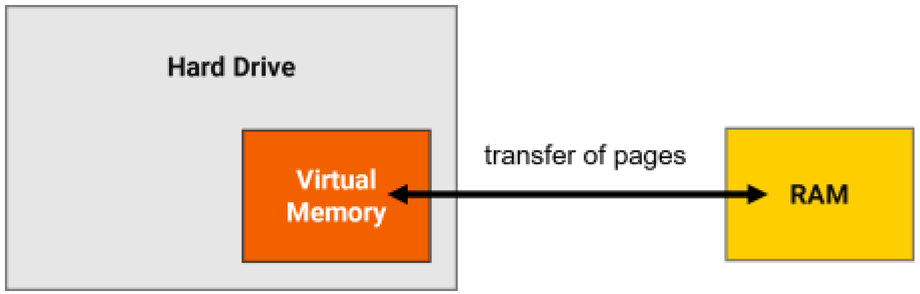
Operating System



ABSTRACTION

How the following are related?


A new disk without any partitons
Extended partition
Primary partition
Unallocated space (rest of extended partition)
Primary partition (C)
D
E
F
Logical partitions



Hard Drive

Virtual Memory

transfer of pages

RAM

Java Virtual Machine (JVM)

Virtual Desktop Infrastructure (VDI)

Keywords

Level of Abstraction

Degree of Freedom

Virtualization

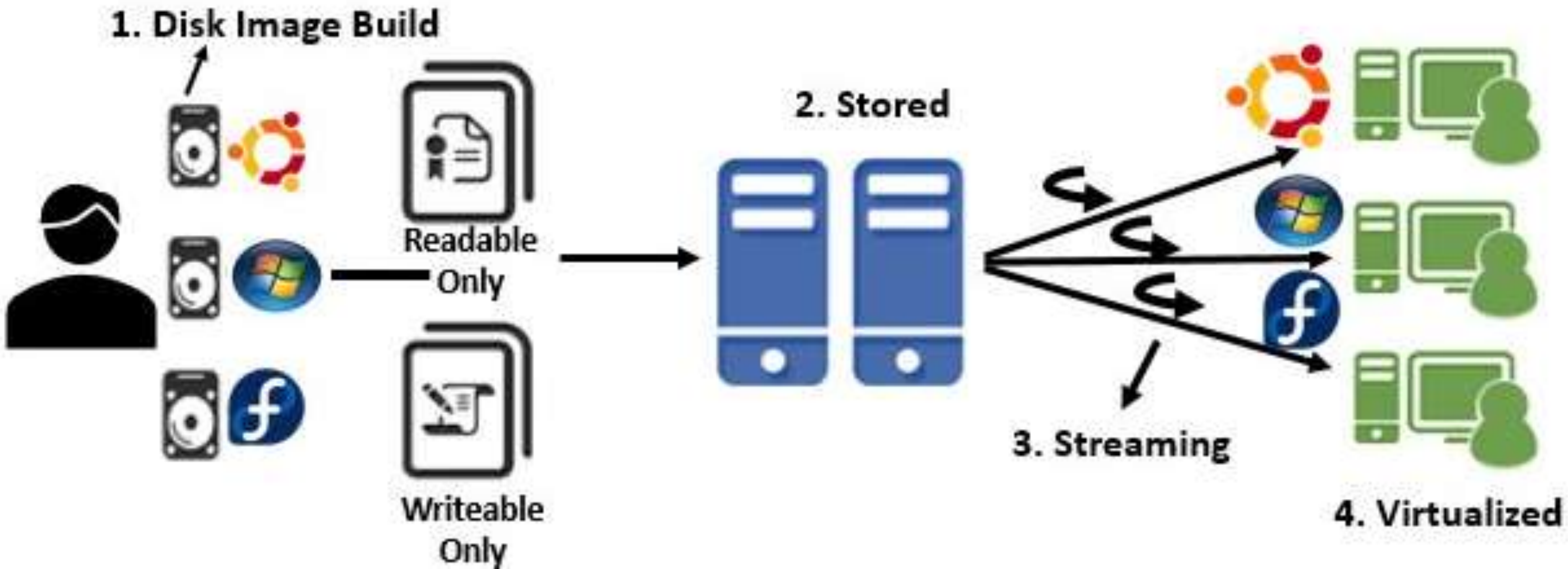
- It is technique which allows to share a single physical instance of an application or resource among multiple organizations or tenants.

Virtualization - Definition

- Virtualization is the **creation of a virtual** (rather than actual) **version** of computing resource, such as **computer hardware, an operating system, a server, a storage device or network resources**.
- This is usually done by introducing a layer of abstraction at appropriate place in computing stack.

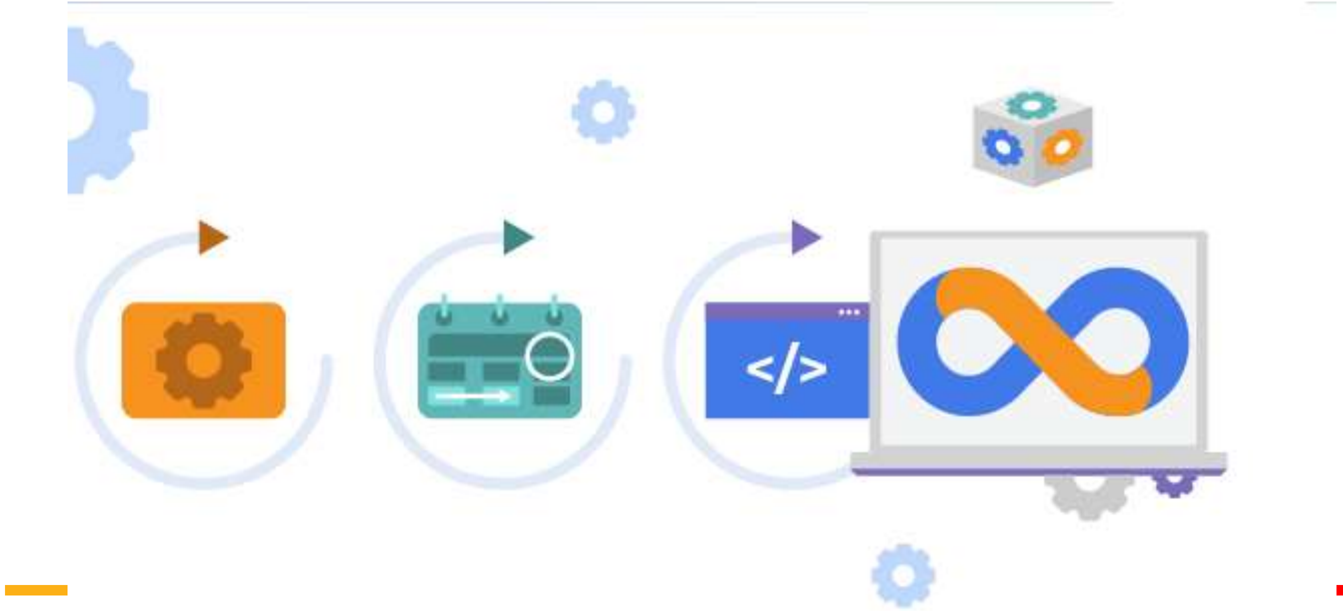
Virtualization

- 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

- Virtualization is used to improve IT throughput and costs by using physical resources as a pool from which virtual resources can be allocated.

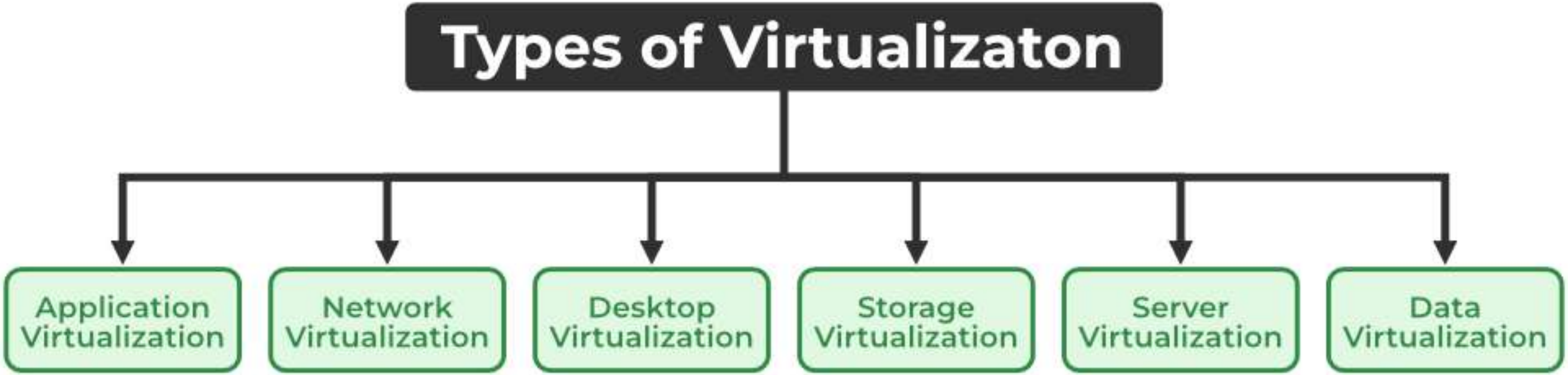


Why called as Virtualization?

- Because physically (or actually) the device or software is not what it is pretending to be. Actual configuration is different, and hence called virtual.
- However, there is always some physical entity down the line.

Some people aren't who they say they are. Be cautious of the company you keep.
TRUTH

Virtualization Types



How Does it Happen??

Learning Virtualization

ACTIVITY-MOM.COM

Simulation or Emulation

- **You want to duplicate the behavior of an old calculator, there are two options**:
  - **You write new program** that draws the calculator's display and keys, and when the user clicks on the keys, your programs does what the old calculator did. This is a *Simulator*
  - **You get a dump of the calculator's firmware**, then write a program that loads the firmware & interprets it the same way the microprocessor in the calculator did. This is an *Emulator.*

Simulation vs. Emulation

- An emulator attempts to duplicate the entire behavior of a device from an external viewpoint. It may or may not need to duplicate the inner workings of a device to accomplish that.

- A simulation attempts to duplicate a specific subset of a device's behavior.

Driving Forces for Virtualization

- Increased performance and computing capacity
- Underutilized hardware and software resources
- Lack of space
- Green initiative
- Rise of administrative cost

Virtualization in General

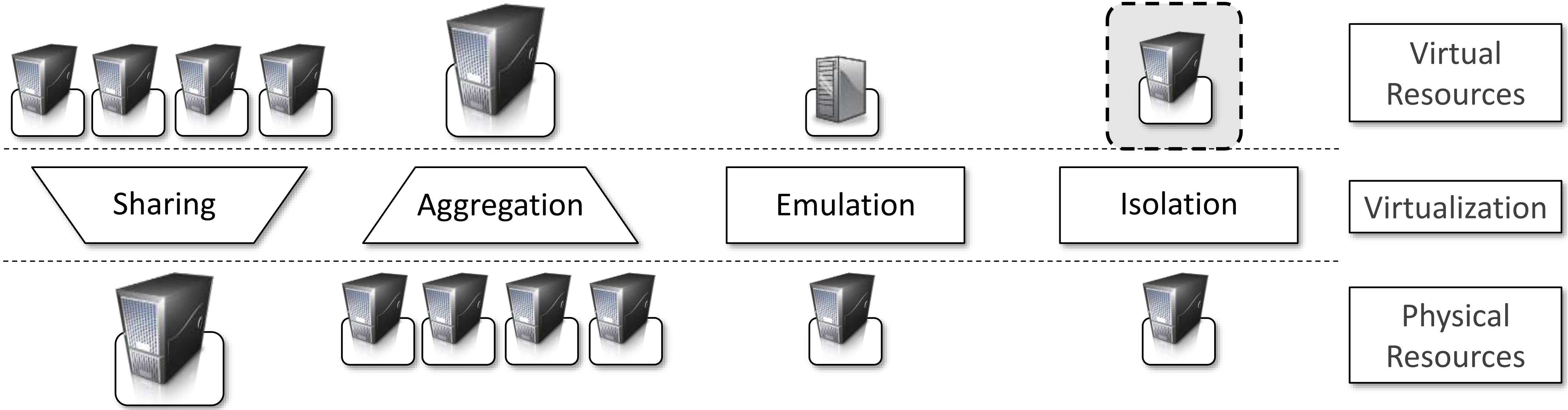
Advantages of virtual machines:

- Run operating systems where the relevant 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,
- Debug problems (suspend and resume the problem machine),
- Easy migration of virtual machines (shutdown needed or not).
- Run legacy systems!

Characteristics of Virtualization

- Increased security
- Managed execution
  - Sharing
  - Aggregation
  - Emulation
  - Isolation
- Portability

Functions Enabled by Managed Execution



Virtual Resources

Sharing

Aggregation

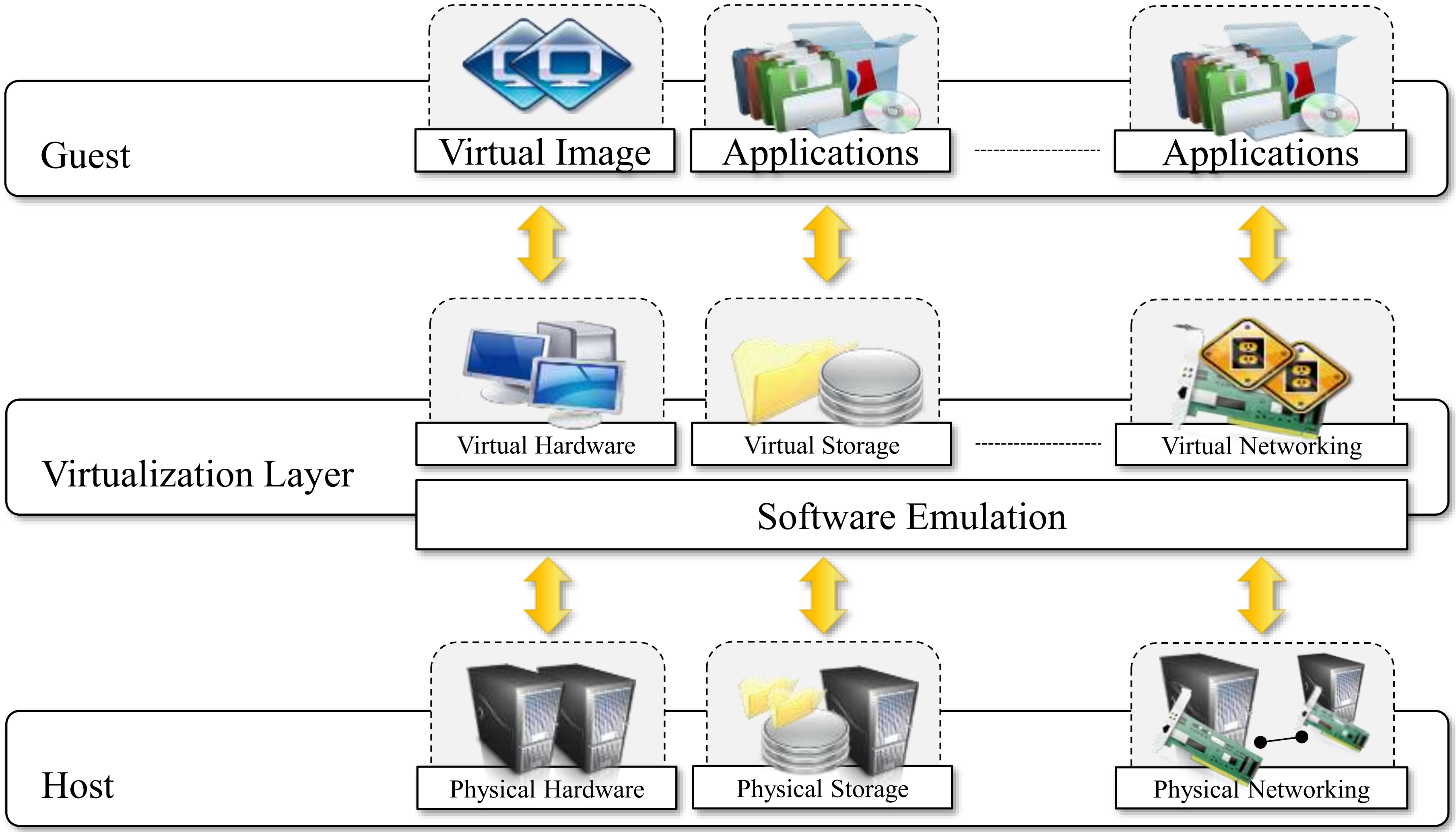
Emulation

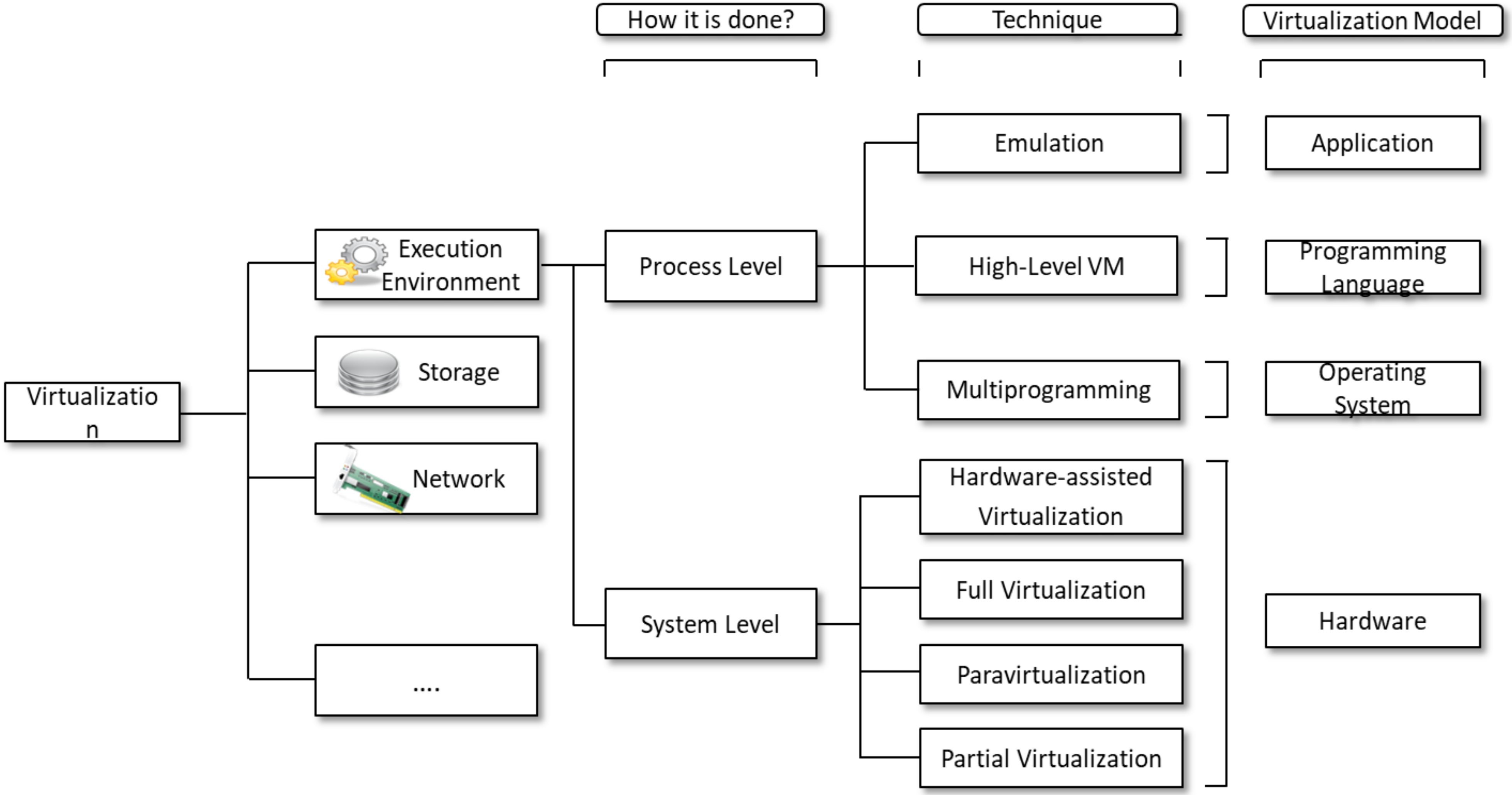
Isolation

Virtualization

Physical Resources

Virtualization Reference Model

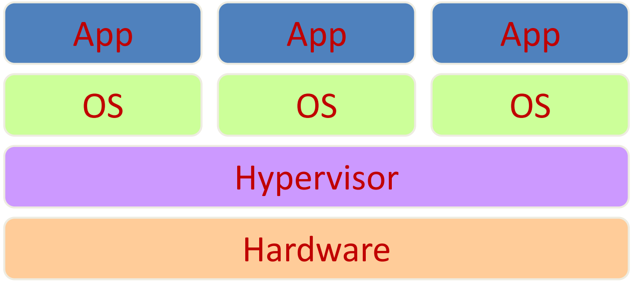




How it is done?
Technique
Virtualization Model
Virtualizatio
n
Execution
Environment
Storage
Network
....
Process Level
System Level
Emulation
High-Level VM
Multiprogramming
Hardware-assisted
Virtualization
Full Virtualization
Paravirtualization
Partial Virtualization
Application
Programming
Language
Operating
System
Hardware

Implementation: Levels of Virtualization

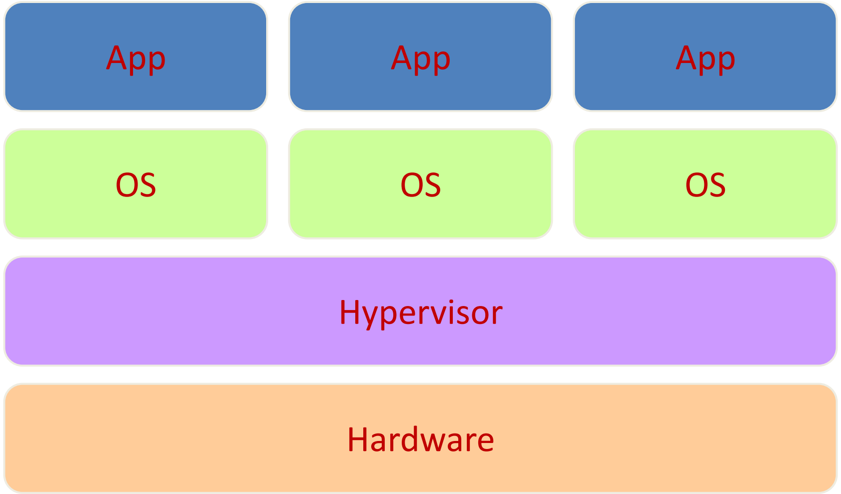
- Virtualization technology benefits the computer and IT industries by enabling users to share expensive hardware resources by multiplexing VMs on the same set of hardware hosts. Virtual workspaces:
  - *An abstraction of an execution environment that can be made dynamically available to authorized clients by using well-defined protocols, resource quota (e.g. CPU, memory share), software configuration (e.g. O/S, provided services).*



Virtualized Stack

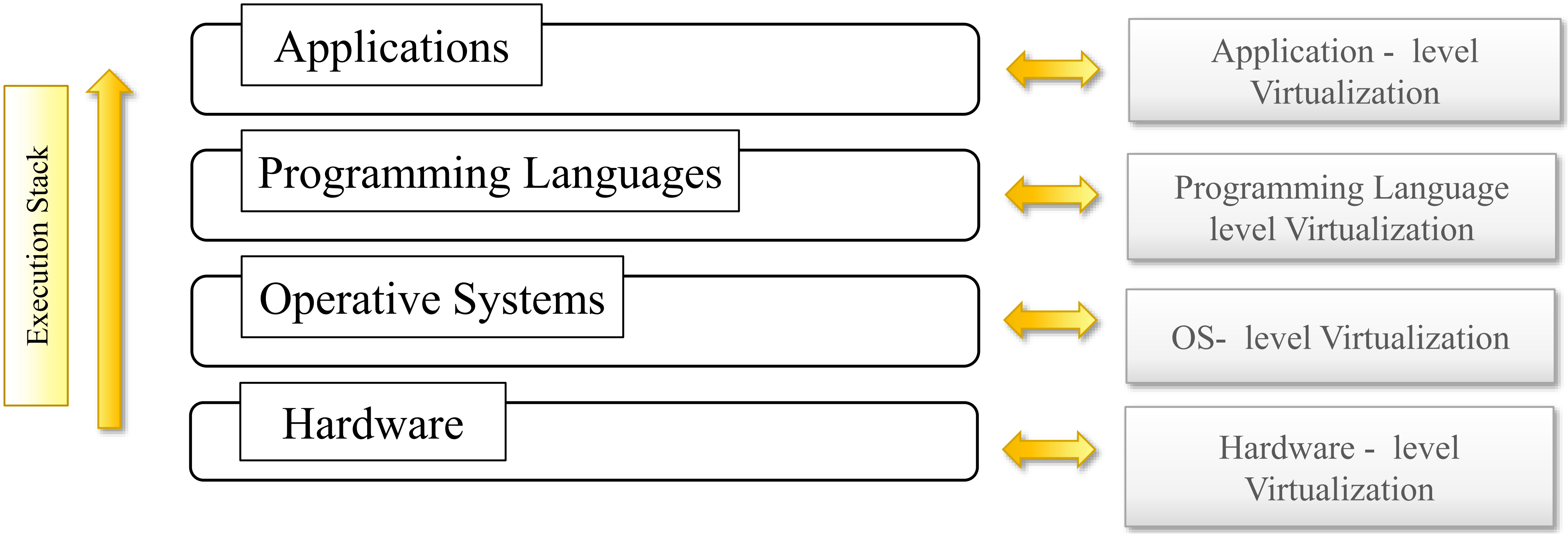
Implementation: Levels of Virtualization

- 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

Virtualization Levels

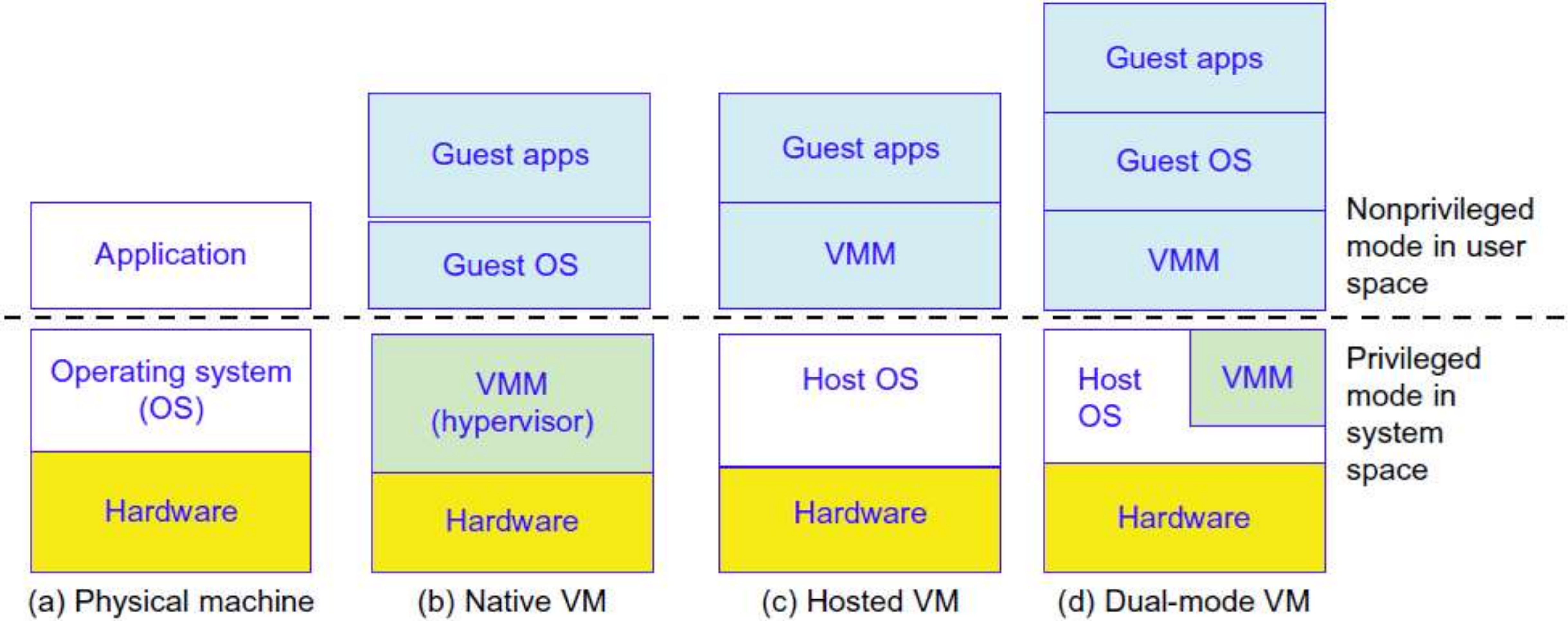

Execution Stack
Applications
Programming Languages
Operative Systems
Hardware
Application - level Virtualization
Programming Language level Virtualization
OS- level Virtualization
Hardware - level Virtualization

Taxonomy of Virtualization Techniques

- Execution Virtualization
  - Hardware level virtualization
    - Hardware assisted virtualization
    - Full virtualization
    - Para-virtualization
    - Partial virtualization
  - Operating system level virtualization
  - Programming language level virtualization
  - Application-level virtualization
    - Interpretation
    - Binary translation

Other Type of Virtualization

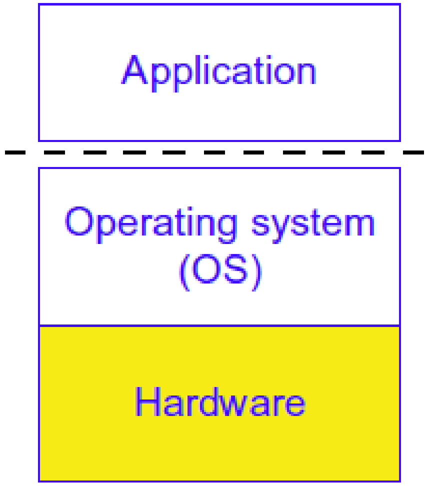
- Storage virtualization
- Network virtualization
- Desktop virtualization
- Application server virtualization


Application
Operating system (OS)
Hardware
(a) Physical machine
Guest apps
Guest OS
VMM (hypervisor)
Hardware
(b) Native VM
Guest apps
VMM
Host OS
Hardware
(c) Hosted VM
Guest apps
Guest OS
VMM
Host OS
VMM
Hardware
(d) Dual-mode VM
Nonprivileged mode in user space
Privileged mode in system space

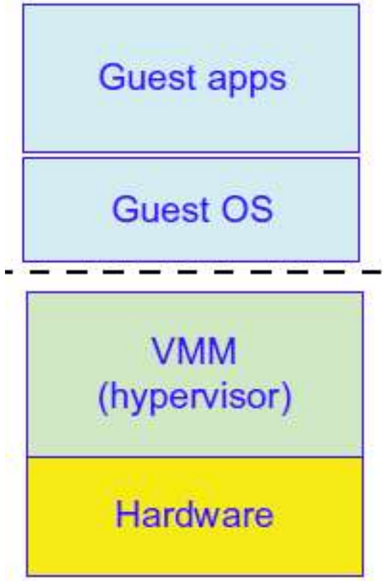
Understanding VM Creation

The host machine is equipped with the physical hardware

- The VM is built with virtual resources managed by a guest OS to run a specific application.
- Between the VMs and the host platform, one needs to deploy a middleware layer called a virtual machine monitor (VMM).
- This hypervisor approach is also called bare-metal VM, because the hypervisor handles the bare hardware (CPU, memory, and I/O) directly.



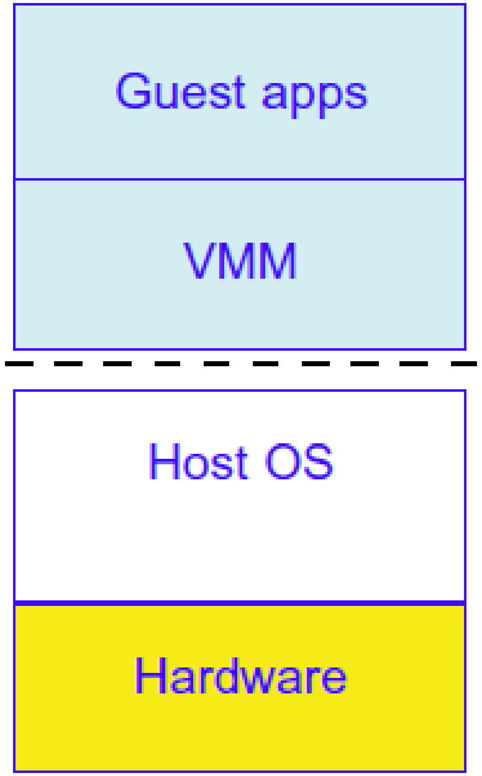
(a) Physical machine



(b) Native VM

Understanding VM Creation

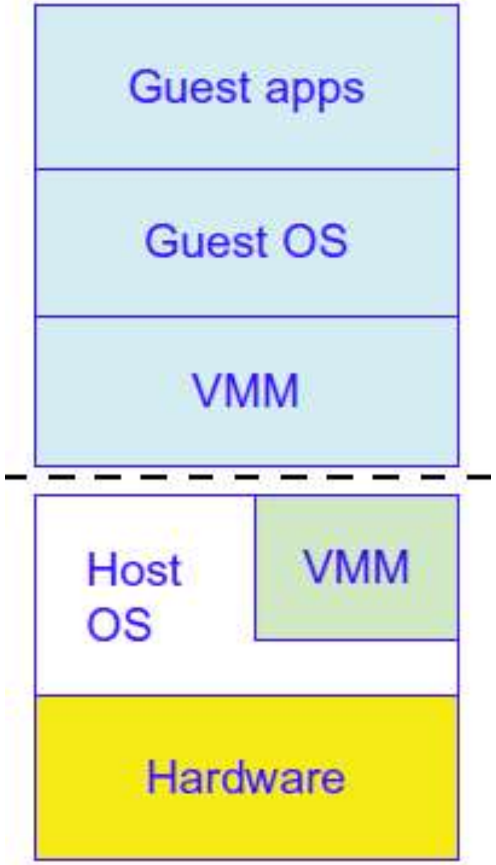
- A native VM is installed with the use of a VMM called a hypervisor in privileged mode
- Example:
  - The hardware is an x-86 architecture running the Windows system
  - The guest OS could be a Linux system and the hypervisor as the XEN system
  - Here the VMM runs in nonprivileged mode.



(c) Hosted VM

Understanding VM Creation

- The VM can also be implemented with a dual mode also
- Part of the VMM runs at the user level and another part runs at the supervisor level.
- In this case, the host OS may have to be modified to some extent.
- Multiple VMs can be ported to a given hardware system to support the virtualization process.



(d) Dual-mode VM

Understanding VM Creation

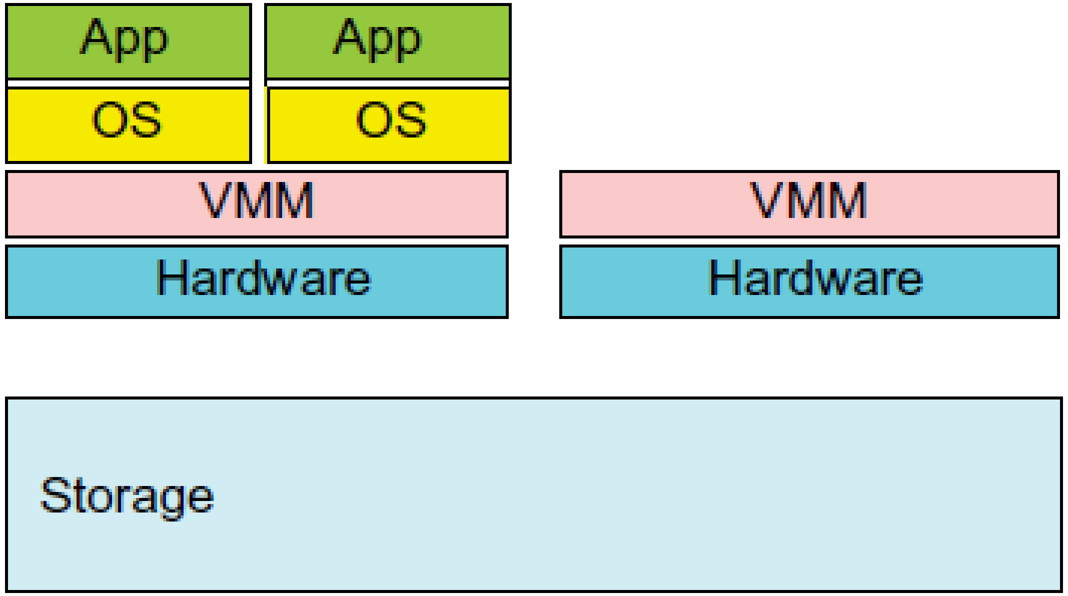
- Finally, VM approach offers hardware independence of the OS and applications.

- The user application running on its dedicated OS could be bundled together as a virtual appliance that can be ported to any hardware platform.

- The VM could run on an OS different from that of the host computer.

VM Primitive Operations

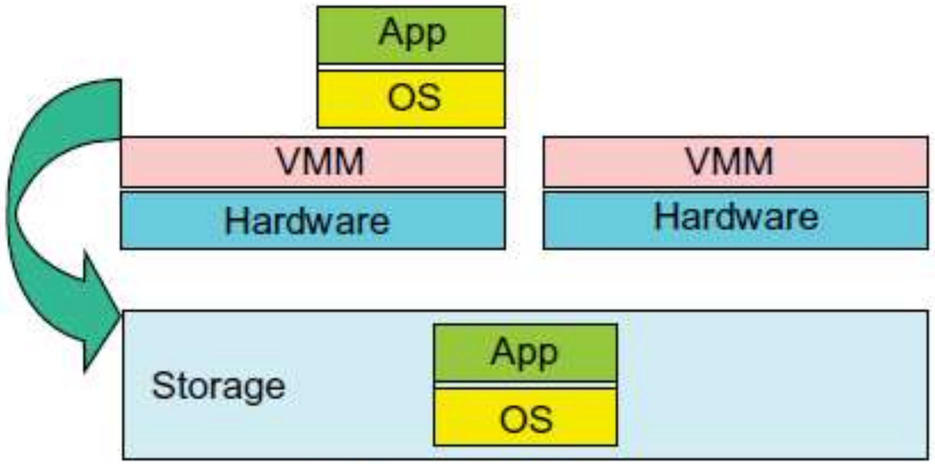
- The VMs can be multiplexed between hardware machines



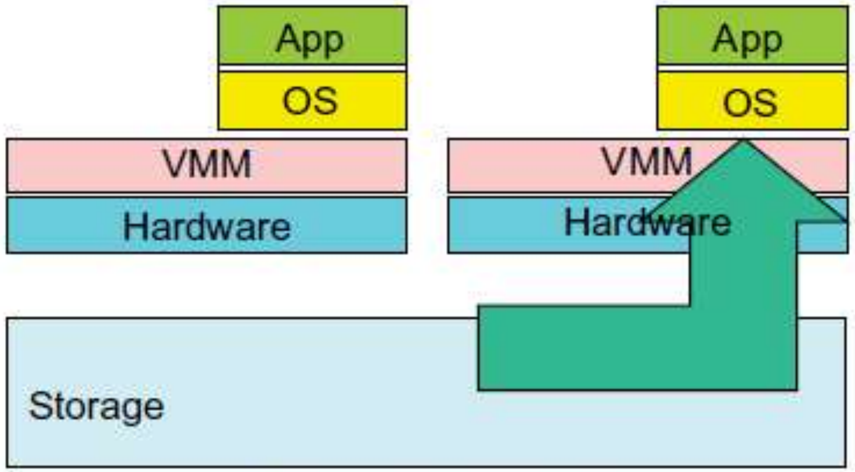
(a) Multiplexing

VM Primitive Operations

- The VM can be suspended and stored in stable storage
- A suspended VM can be resumed or provisioned to a new hardware platform



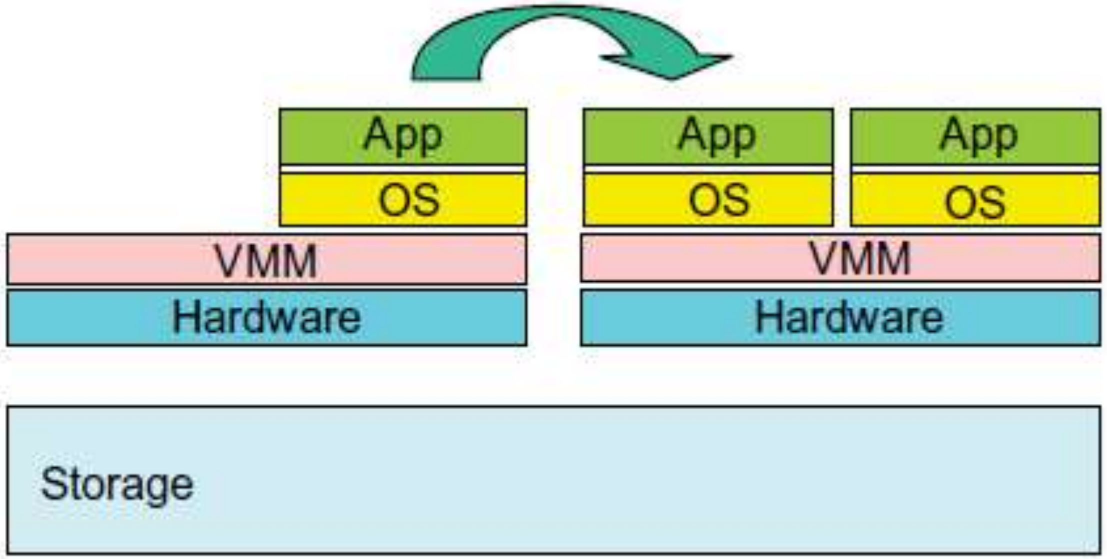
(b) Suspension (storage)



(c) Provision (resume)

VM Primitive Operations

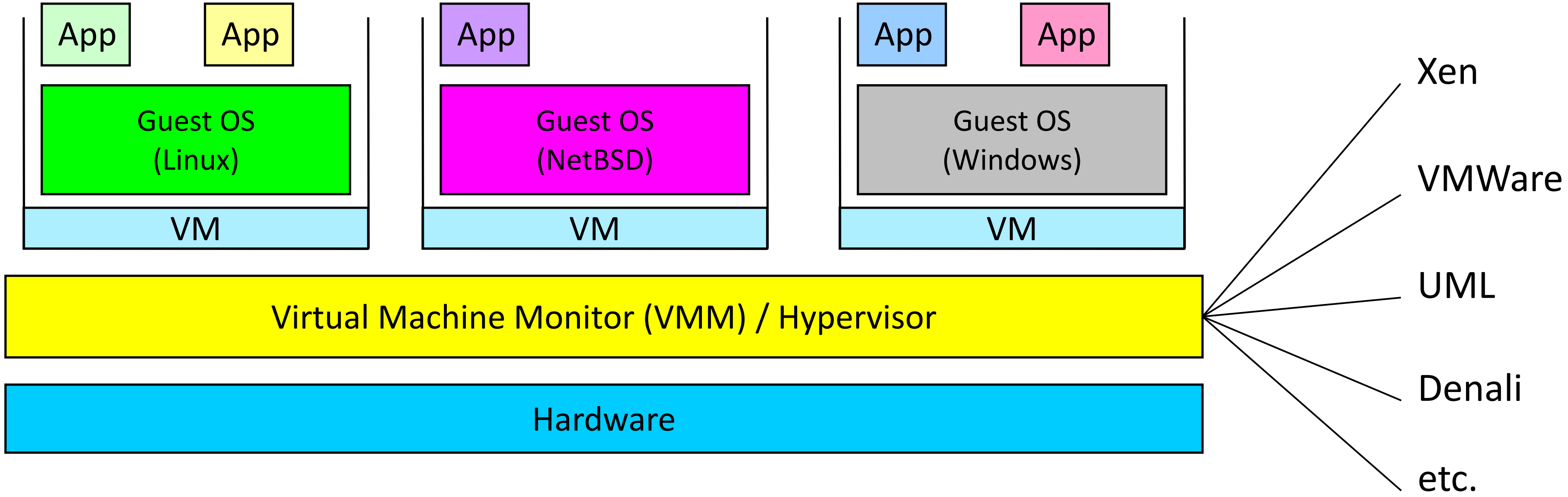
- The VM can be migrated from one hardware platform to another
- They also enable flexibility in porting distributed application executions.



(d) Life migration

Virtual Machines

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



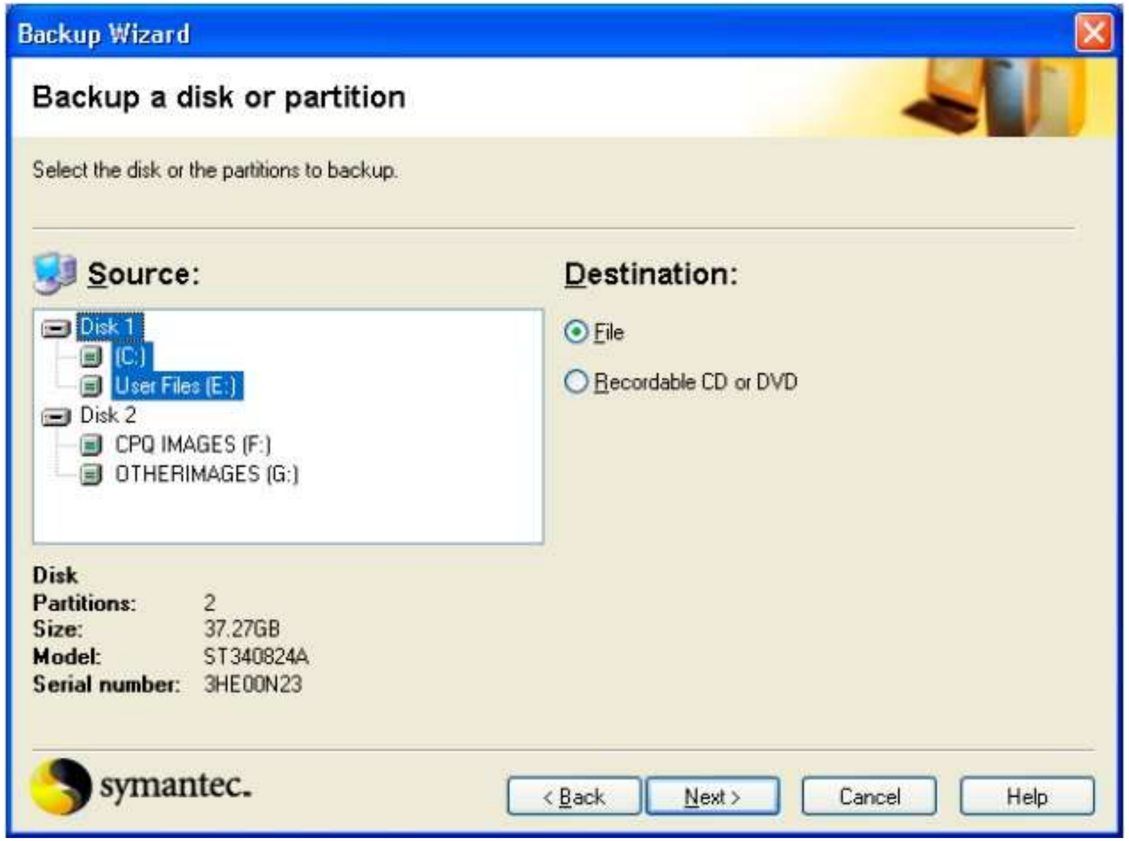
Performance: Para-virtualization (e.g. Xen) is very close to raw physical performance!

Need for Virtual Machine

How easy or hard?

Clone complete installation to new computer

Virtual Machines - Analogy



Backup a disk or partition

Select the disk or the partitions to backup.

Source:

- Disk 1
  - (C:)
  - User Files (E:)
- Disk 2
  - CPQ IMAGES (F:)
  - OTHERIMAGES (G:)

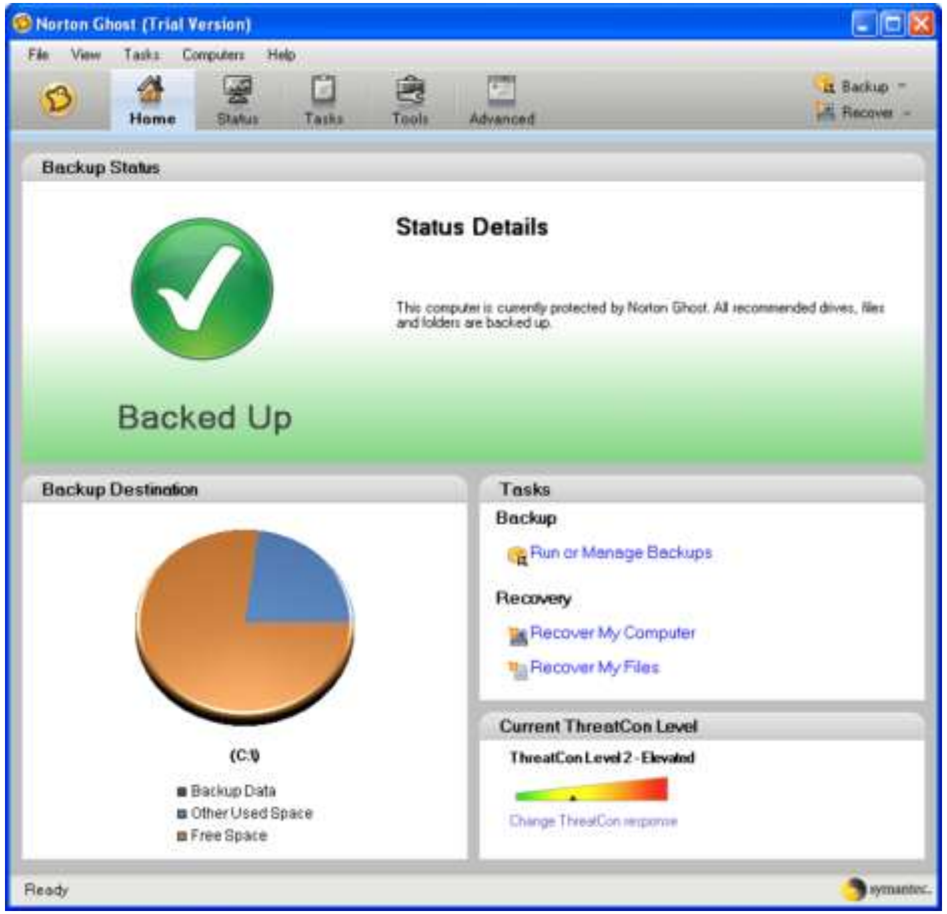
Destination:

● File

○ Recordable CD or DVD

Disk

Partitions:	2
Size:	37.27GB
Model:	ST340824A
Serial number:	3HE00N23



Backup Status

Backed Up

Status Details

This computer is currently protected by Norton Ghost. All recommended drives, files and folders are backed up.

ThreatCon Level 2 - Elevated

Norton Ghost

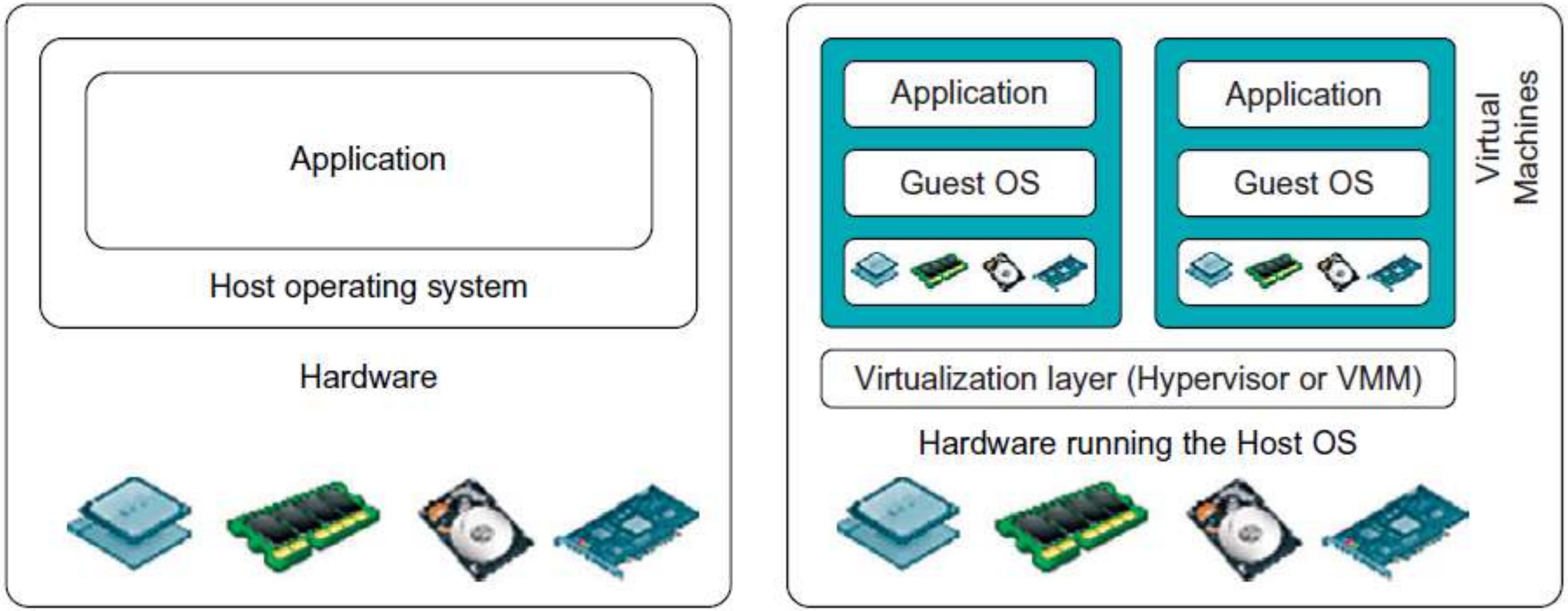
Virtual Machines - Analogy

Image

Virtual Machine

Virtualization Levels

The virtualization software creates the abstraction of VMs by interposing a **virtualization layer (Hypervisor or VMM)** at various levels of a computer system.



(a) Traditional computer

(b) After virtualization

Virtualization Layer

- The main function of the software layer for virtualization is to virtualize the physical hardware of a host machine into virtual resources to be used by the VMs, exclusively.

Virtualization Levels

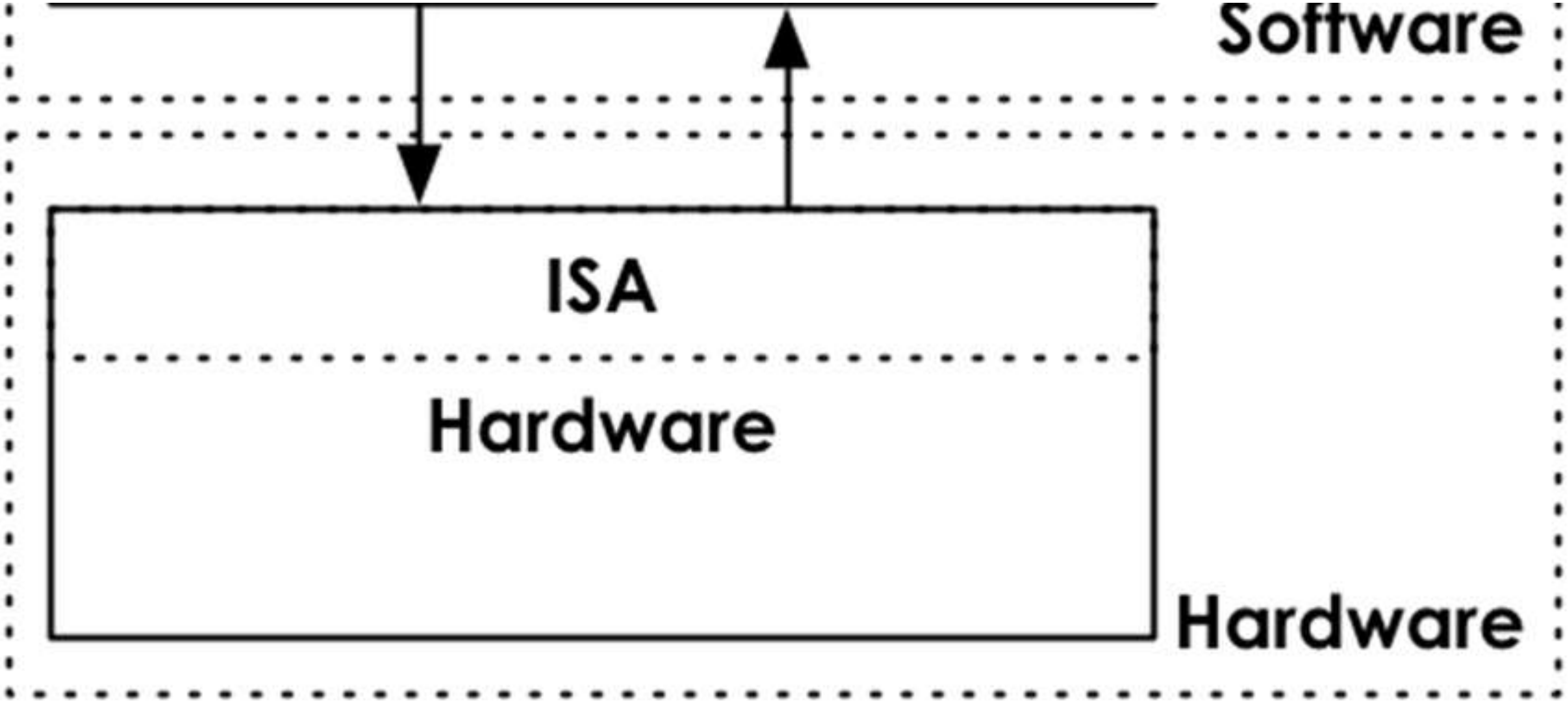
Common virtualization layers include

1. Instruction Set Architecture (ISA) level
2. Hardware level
3. Operating system level
4. Library support level
5. Application level

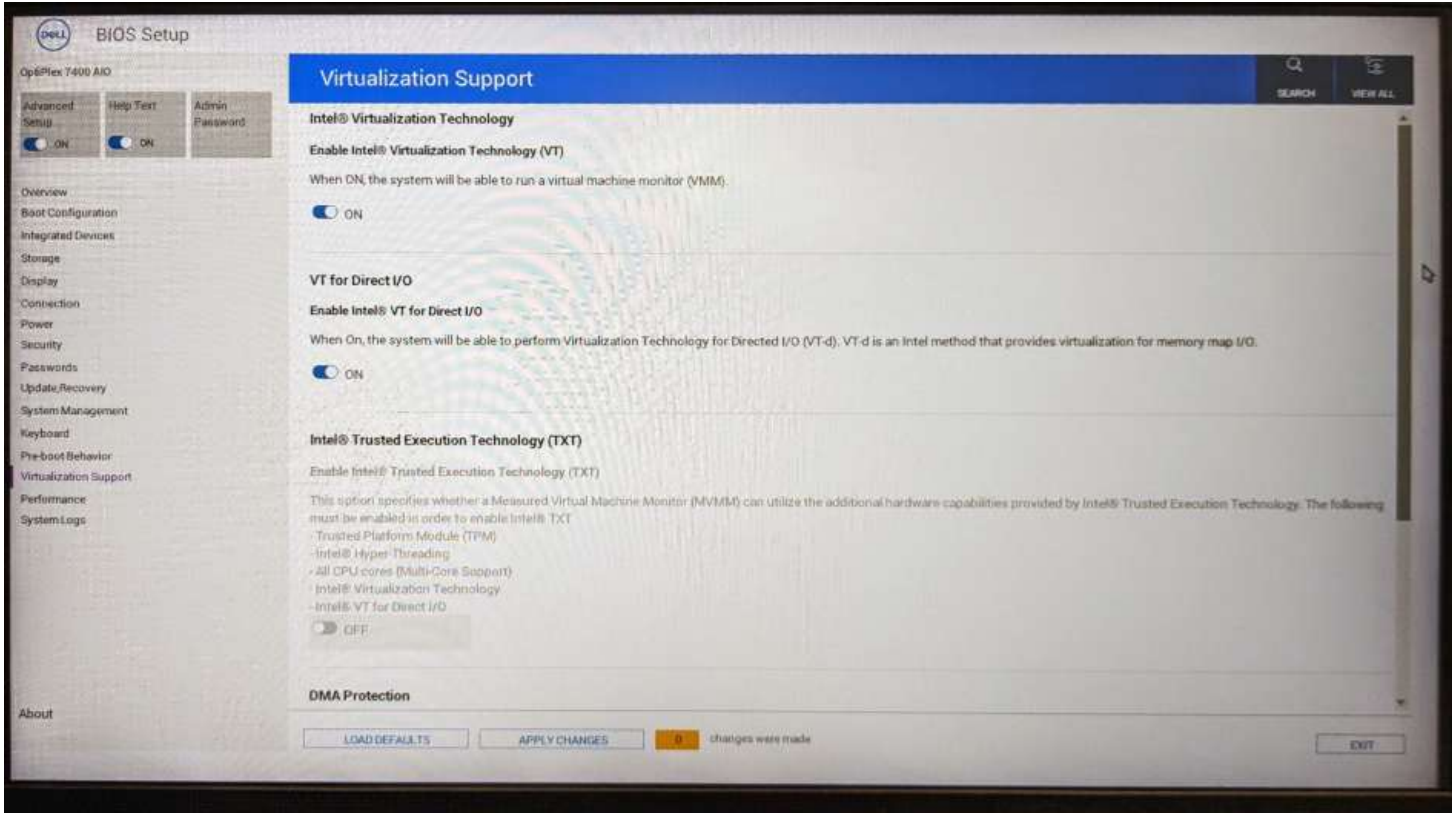
Can you run x86 vm on x64 architecture

??

Instruction Set Architecture Levels



ISA level & Hardware Assisted



Virtualization Support

Intel® Virtualization Technology

Enable Intel® Virtualization Technology (VT)

When ON, the system will be able to run a virtual machine monitor (VMM).

ON

VT for Direct I/O

Enable Intel® VT for Direct I/O

When On, the system will be able to perform Virtualization Technology for Directed I/O (VT-d). VT-d is an Intel method that provides virtualization for memory map I/O.

ON

Intel® Trusted Execution Technology (TXT)

Enable Intel® Trusted Execution Technology (TXT)

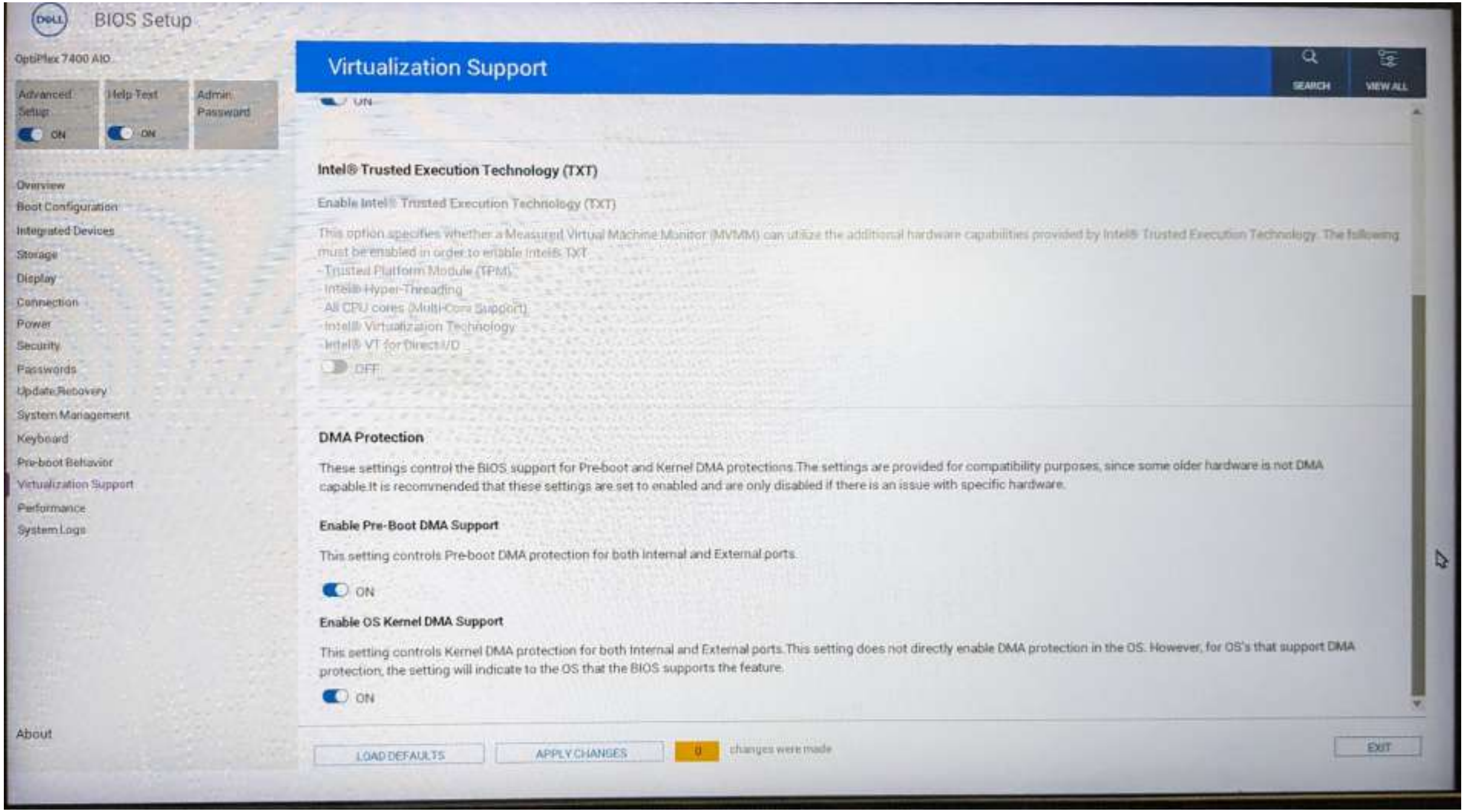
This option specifies whether a Measured Virtual Machine Monitor (MVMM) can utilize the additional hardware capabilities provided by Intel® Trusted Execution Technology. The following must be enabled in order to enable Intel® TXT

- Trusted Platform Module (TPM)
- Intel® Hyper-Threading
- All CPU cores (Multi-Core Support)
- Intel® Virtualization Technology
- Intel® VT for Direct I/O

OFF

DMA Protection

ISA level



Virtualization Support

ON

Intel® Trusted Execution Technology (TXT)

Enable Intel® Trusted Execution Technology (TXT)

This option specifies whether a Measured Virtual Machine Monitor (MVMM) can utilize the additional hardware capabilities provided by Intel® Trusted Execution Technology. The following must be enabled in order to enable Intel® TXT
- Trusted Platform Module (TPM)
- Intel® Hyper-Threading
- All CPU cores (Multi-Core Support)
- Intel® Virtualization Technology
- Intel® VT for Direct I/O

OFF

DMA Protection

These settings control the BIOS support for Pre-boot and Kernel DMA protections. The settings are provided for compatibility purposes, since some older hardware is not DMA capable. It is recommended that these settings are set to enabled and are only disabled if there is an issue with specific hardware.

Enable Pre-Boot DMA Support

This setting controls Pre-boot DMA protection for both Internal and External ports.

ON

Enable OS Kernel DMA Support

This setting controls Kernel DMA protection for both Internal and External ports. This setting does not directly enable DMA protection in the OS. However, for OS's that support DMA protection, the setting will indicate to the OS that the BIOS supports the feature.

ON

0 changes were made

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.

Instruction Set Architecture (ISA) level

- Code interpretation – Dynamic Binary Translation - **Virtual Instruction Set Architecture (V-ISA)**
- The basic emulation method is through code interpretation.
  - An interpreter program interprets the source instructions to target instructions one by one.
  - One source instruction may require tens or hundreds of native target instructions to perform its function.
  - Obviously, this process is relatively slow.

Instruction Set Architecture (ISA) level

- For better performance, dynamic binary translation is desired.
- This approach translates basic blocks of dynamic source instructions to target instructions.
- Instruction set emulation requires binary translation and optimization.
- A virtual instruction set architecture (V-ISA) thus requires adding a processor-specific software translation layer to the compiler

Instruction Set Architecture (ISA) level

- **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 complier.

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.

Hardware Abstraction level

- The idea is to virtualize a computer’s resources, such as its processors, memory, 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

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;

Operating System (OS) Level

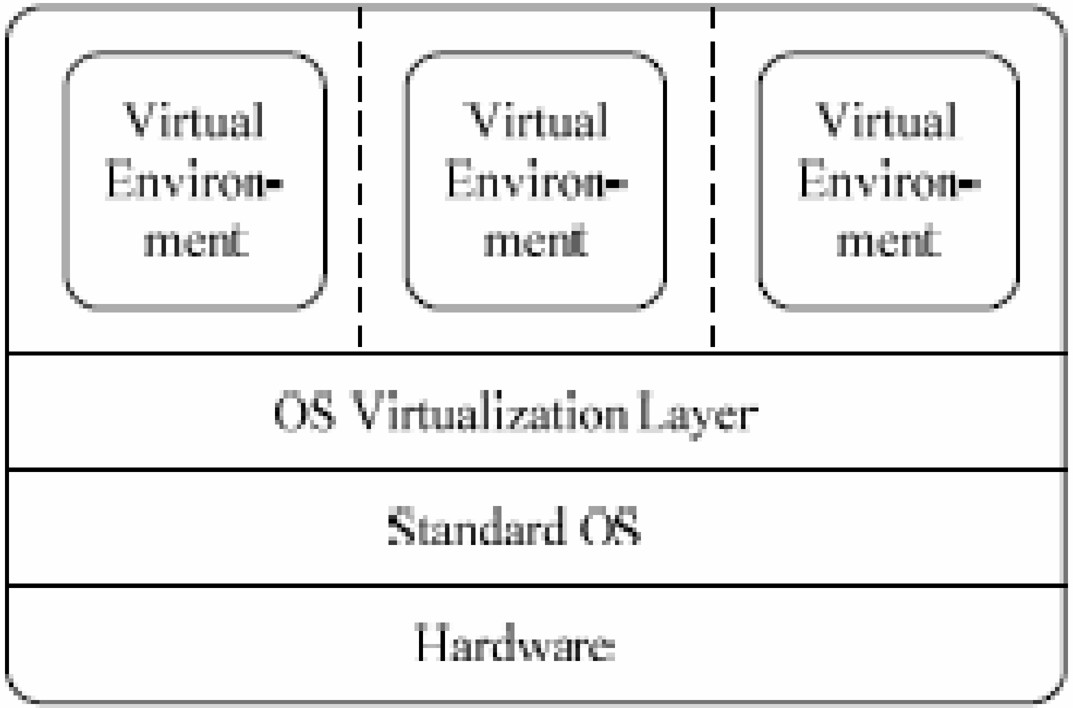


Figure 6.3 The virtualization layer is inserted inside an OS to partition the hardware resources for multiple VMs to run their applications in virtual environments

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.

Advantage:

- It has very low implementation effort

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.

User-Application Level

- 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 Tools and Mechanisms

- In general, there are three typical classes of VM architecture
- Before virtualization, the operating system manages the hardware.
- After virtualization, a virtualization layer is inserted between the hardware and the operating system. In such a case, the virtualization layer is responsible for converting portions of the real hardware into virtual hardware.

Virtualization Tools and Mechanisms

- Different operating systems such as Linux and Windows can run on the same physical machine, simultaneously.
- Depending on the position of the virtualization layer, there are several classes of VM architectures, namely the hypervisor architecture, para-virtualization, and host-based virtualization.
- The hypervisor is also known as the VMM (Virtual Machine Monitor).

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.

Hypervisor

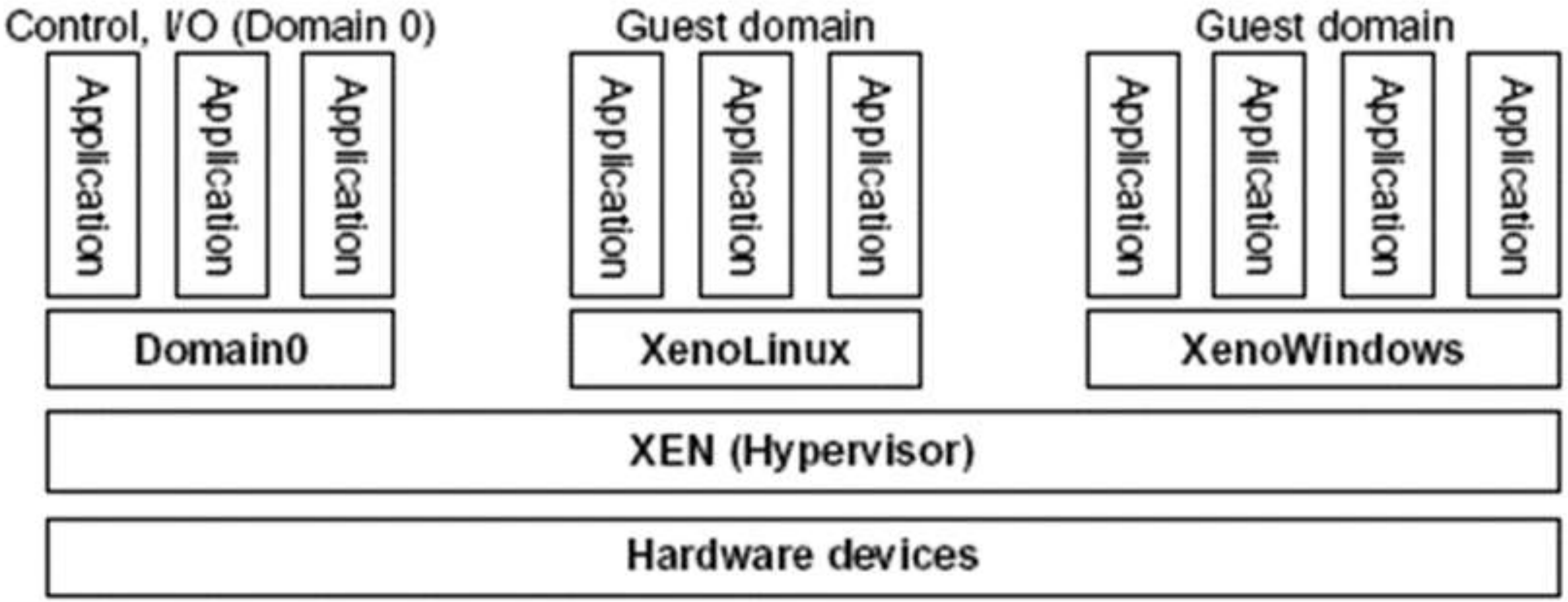
Type 2: Hosted hypervisor

- 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 between the hardware and the OS.

The XEN Architecture



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

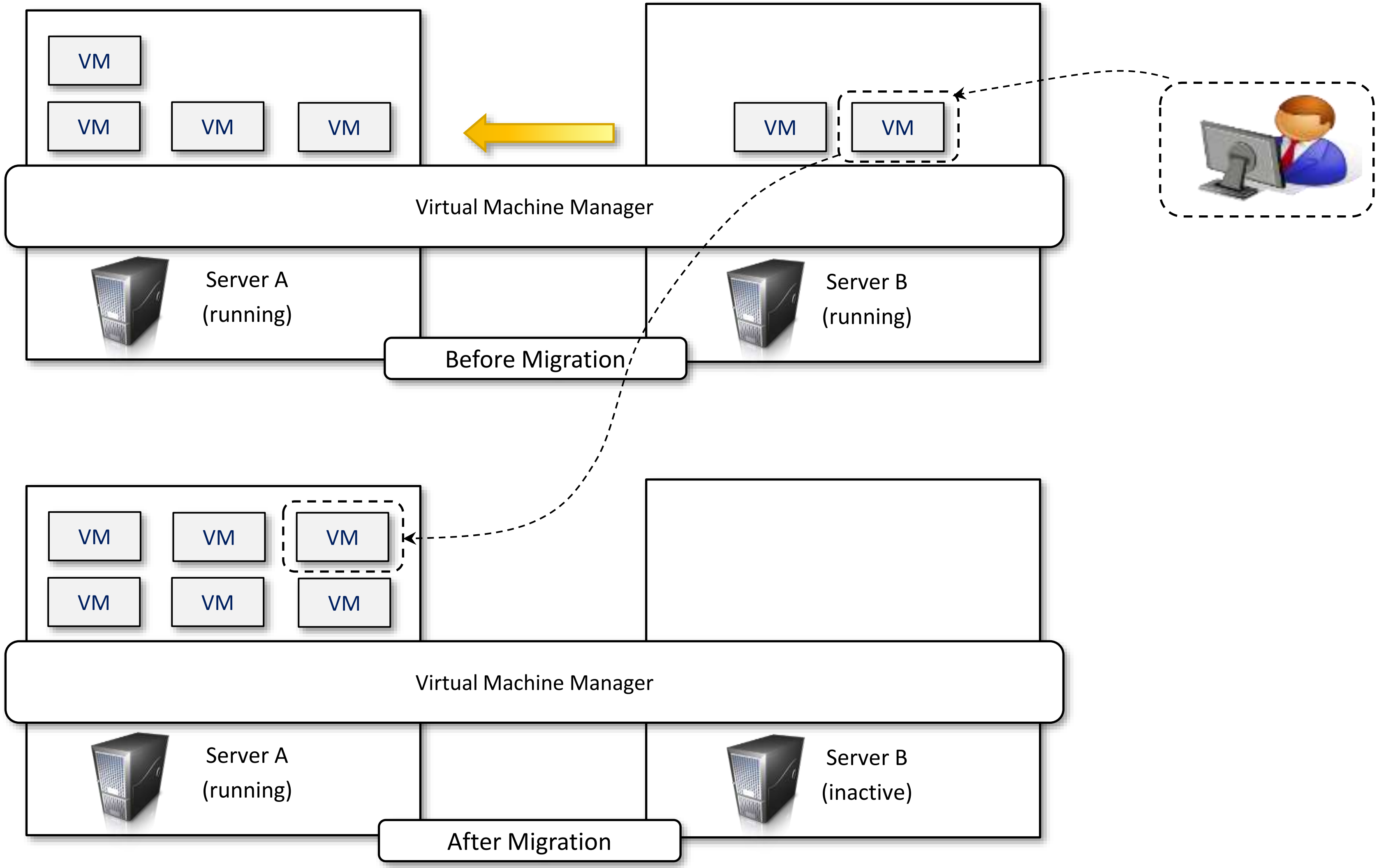
Binary Translation with Full Virtualization

- 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 non-critical and critical instructions.

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 and direct execution.

Live Migration and Server Consolidation



Host-Based Virtualization

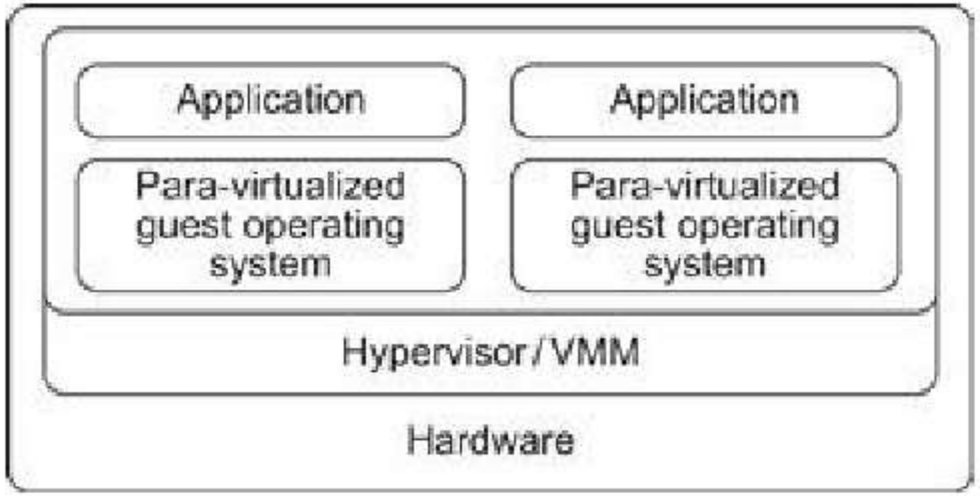
- An alternative VM architecture is to install a virtualization layer on top of the host OS. This host OS is still responsible for managing the hardware.
- This host-based architecture has some distinct advantages. First, the user can install this VM architecture without modifying the host OS. The virtualizing 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.

Host-Based Virtualization

- 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

Host-Based Virtualization: Para-virtualization

- Para-virtualization needs to modify the guest operating systems.
- Provides special API 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.

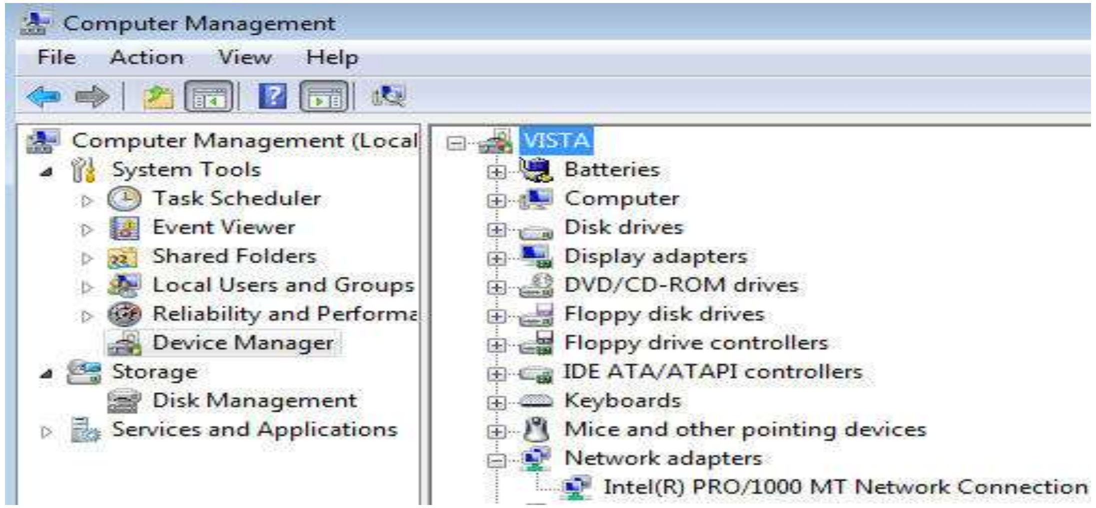
Full Virtualization vs. Para-Virtualization

Para virtualization

- Reduces the overhead, but cost of maintaining a para-virtualized 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.

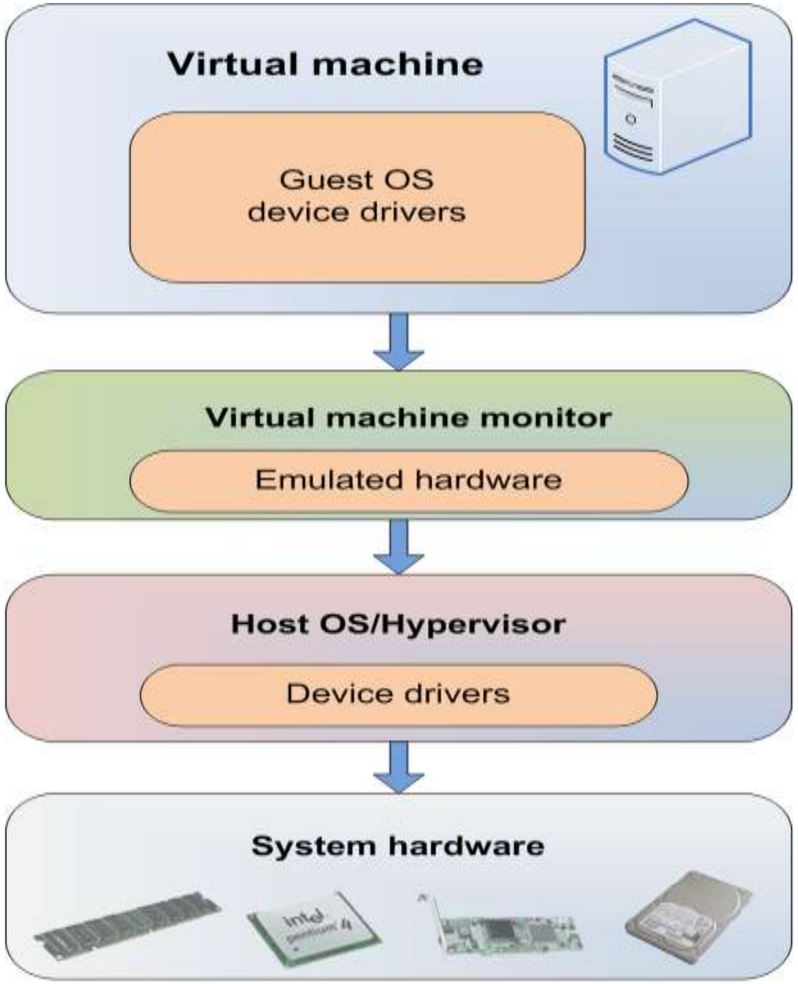
Full Virtualization

- Everything is virtualized
- Full hardware emulation
- Emulation = latency



Computer Management (Local
- System Tools
  - Task Scheduler
  - Event Viewer
  - Shared Folders
  - Local Users and Groups
  - Reliability and Performa
  - Device Manager
- Storage
  - Disk Management
- Services and Applications

VISTA
- Batteries
- Computer
- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Floppy disk drives
- Floppy drive controllers
- IDE ATA/ATAPI controllers
- Keyboards
- Mice and other pointing devices
- Network adapters
  - Intel(R) PRO/1000 MT Network Connection


Virtual machine

Guest OS
device drivers

Virtual machine monitor

Emulated hardware

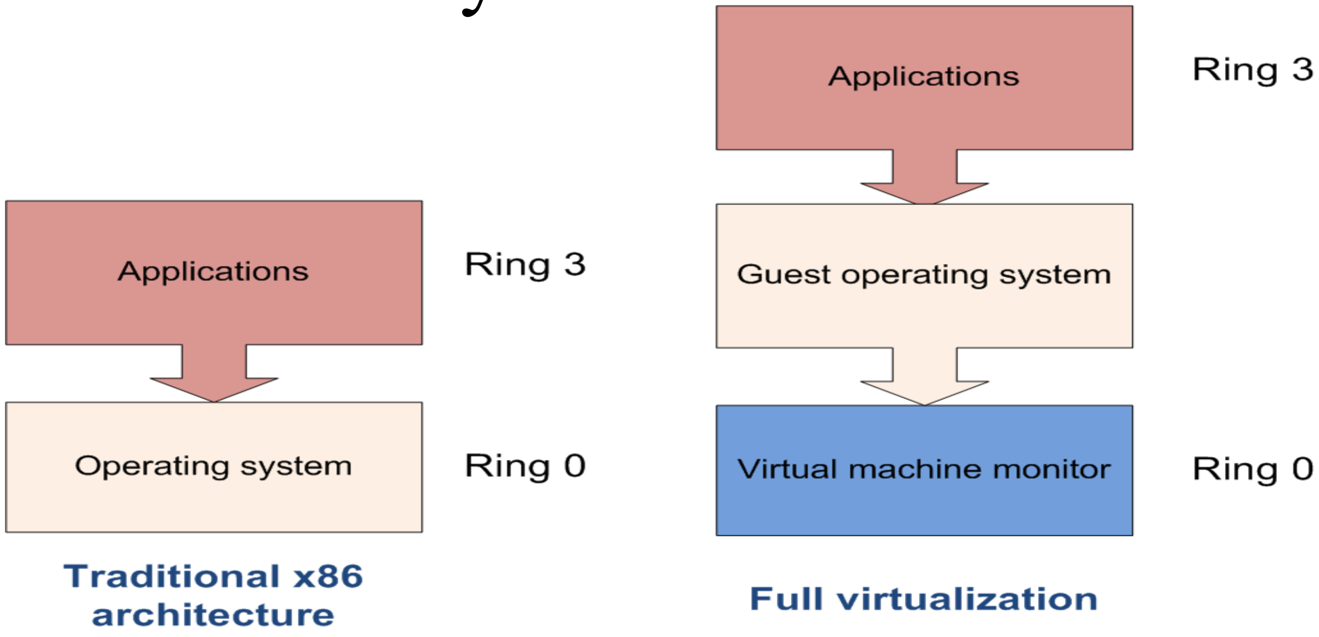
Host OS/Hypervisor

Device drivers

System hardware

Privileged Instructions

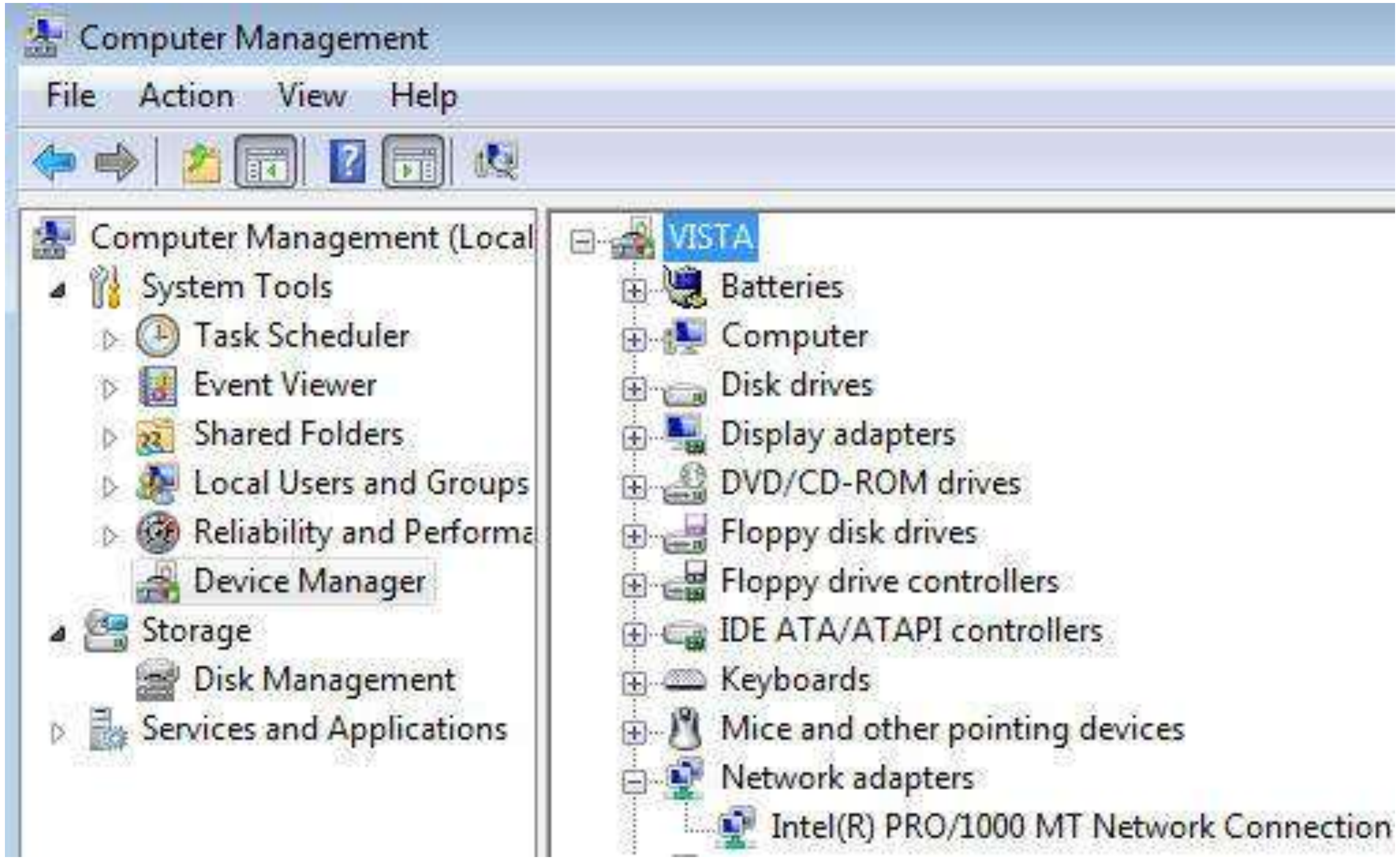
- Privileged instructions: OS kernel and device driver access to system hardware
- Trapped and emulated by VMM



Pros and Cons – Full Virtualization

- **Pros**
  - – Disaster recovery, failover
  - – Virtual appliance deployment
  - – Legacy code on non-legacy hardware
- **Cons** – LATENCY of core four resources
  - – RAM performance reduced 25% to 75%
  - – Disk I/O degraded from 5% to 20%
  - – Network performance decreased up to 10%
  - – CPU privileged instruction dings nearing 1% to 7%

Full Virtualization



Computer Management

File Action View Help

Computer Management (Local
- System Tools
  - Task Scheduler
  - Event Viewer
  - Shared Folders
  - Local Users and Groups
  - Reliability and Performa
  - Device Manager
- Storage
  - Disk Management
- Services and Applications

VISTA
- Batteries
- Computer
- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Floppy disk drives
- Floppy drive controllers
- IDE ATA/ATAPI controllers
- Keyboards
- Mice and other pointing devices
- Network adapters
  - Intel(R) PRO/1000 MT Network Connection

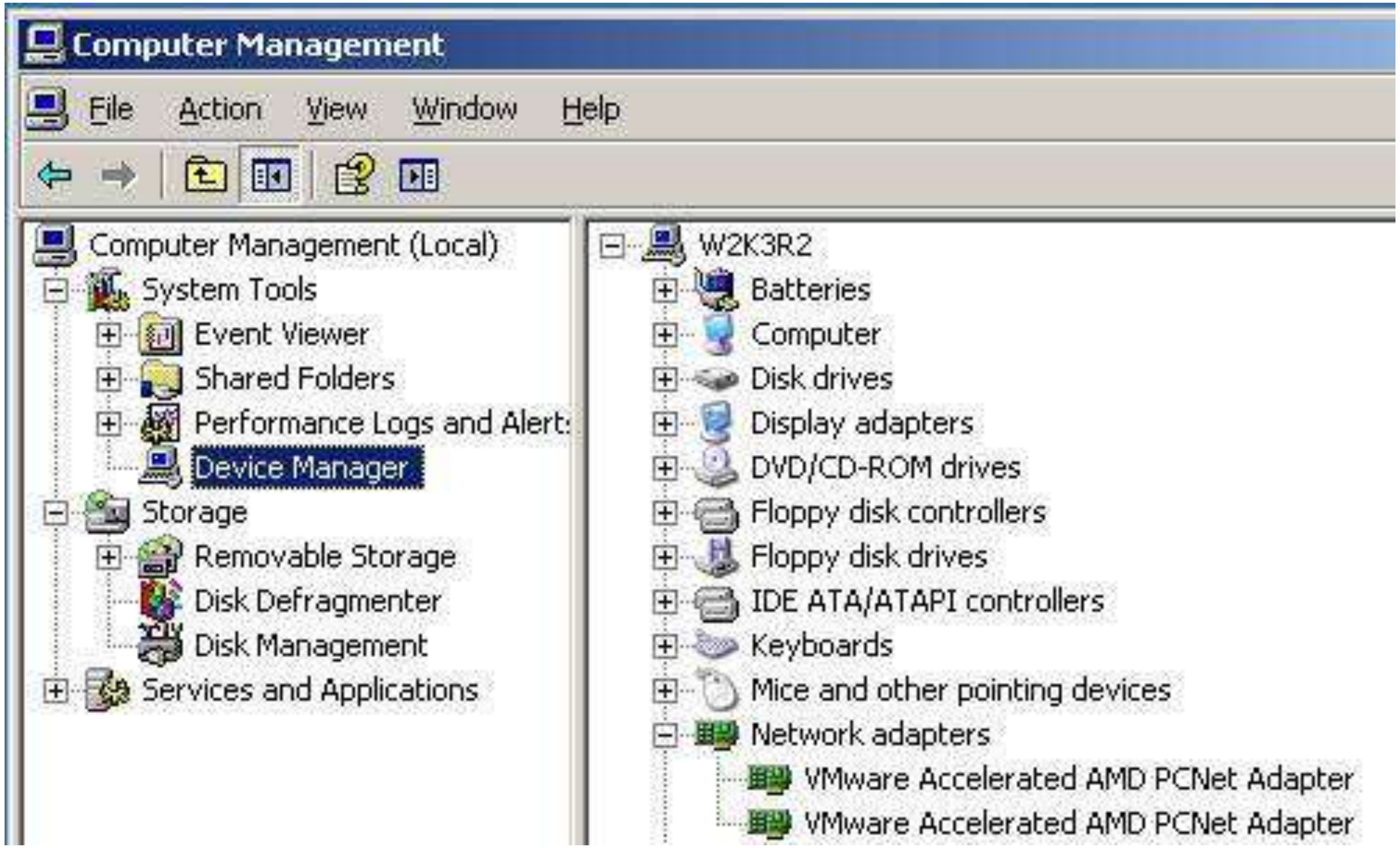
Network adapters are physical adapters. It shows no virtual adapters/drivers here. This is Full Virtualization.

Para-virtualization

- OS or system devices are virtualization aware

- Requirements:
- OS level – recompiled kernel
- Device level – para-virtualized or "enlightened" device drivers

Para-virtualization



Virtualized network adapters are automatically added to the machine after installation is done.

Which means OS kernel is modified and these drivers are pushed into the OS so that it can contact the physical ethernet adapter.

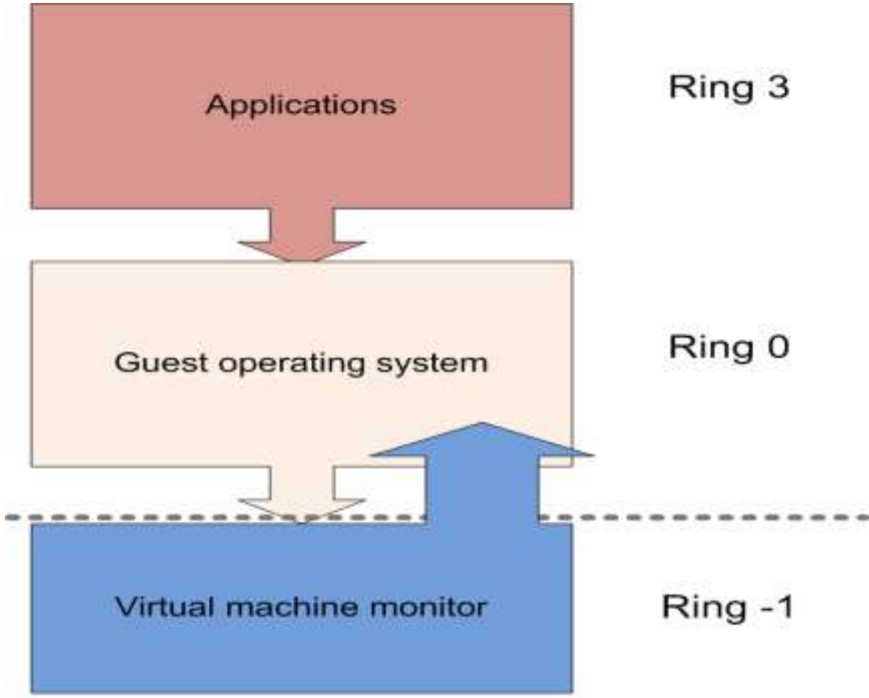
Para-virtualization

- **Pro:** Fast
- **Con:** requires a specially modified guest OS, thus precludes the ability to run off-the-shelf and legacy OS in paravirtual environments

Virtualization - Other Levels

Hardware-assisted Virtualization

- Server hardware is virtualization aware
- Hypervisor and VMM load at privilege Ring -1 (firmware)
- Removes CPU emulation bottleneck
- Memory virtualization coming in quad core AMD and Intel CPUs



Hardware-assisted virtualization

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 behaviors depending on the configuration of resources, including the load and store operations over the virtual memory.

CPU Virtualization

- 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.
- RISC 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. In a traditional execution environment, the operating system maintains mappings of virtual memory to machine memory using page tables, which is a one-stage mapping from virtual memory to machine memory.
- However, in a virtual execution environment, virtual memory virtualization 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. It can achieve close-to native performance without high CPU costs.

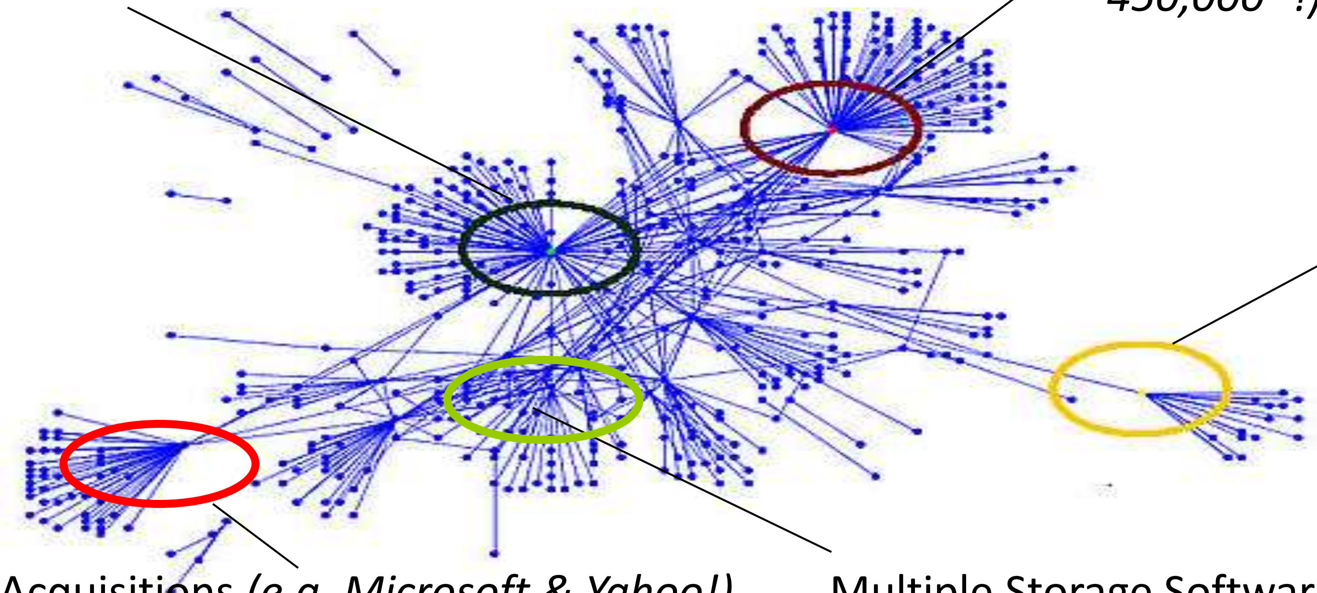
Storage Virtualization

- ***Storage Virtualization*** *is the next frontier in Storage Advances that* ***aims to provide a layer of*** abstraction to reduce complexity.
- Storage Networking Industry Association (SNIA) defines Storage Virtualization as:
  - The act of abstracting, hiding, or isolating the internal functions of a storage (sub) system or service from applications, host computers, or general network resources, for the purpose of enabling application and network-independent management of storage or data.
  - The application of virtualization to storage services or devices for the purpose of aggregating functions or devices, hiding complexity, or adding new capabilities to lower level storage resources.

Why Storage Virtualization?

Provided continuous availability despite exponential growth (e.g. *FaceBook- Over 55 billion page views a month, 41 million active users*[1])

Effectively group and manage heterogeneous storage devices & servers (*e.g. Estimated number of Google Servers 450,000* [2]*!)*



Allocate and manage storage in accordance to the Quality of Service (QoS) associated with the data *(e. g. Gartner estimates average data center doubling its storage every 18 to 24 months)*

Mergers and Acquisitions *(e.g. Microsoft & Yahoo!)*

Multiple Storage Software Platforms (*e.g. IBM, EMC, HP,..)*

Storage Virtualization ***aims to provide a layer of*** abstraction to manage storage and reduce complexity !!!

Innovations associated with storage?

Client side storage innovations… variety of *storage device innovations* that are smaller, higher capacity and cheaper have helped end users cope with increasing storage requirements!

Innovations associated with storage?

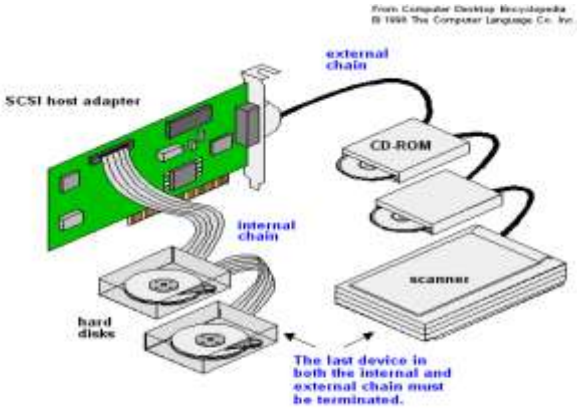
Server side storage innovations… *a combination of* ***<u>storage devices</u>****,* ***<u>storage interfaces</u>*** *and* ***<u>storage software</u>*** *innovations* have helped enterprises cope with exponential growth of data storage requirement!

<u>Storage devices</u> have evolved from tapes to hard drives to RAID hard drives increasing capacity and resiliency.

Innovations associated with storage?

Storage interface innovations *have evolved from* SCSI to ISCI, Fiber Channel (FCP) and InfiniBand to inter ***connect devices and transport the data faster.***

SCSI


external chain
SCSI host adapter
CD-ROM
internal chain
scanner
hard disks
The last device in both the internal and external chain must be terminated.

From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.

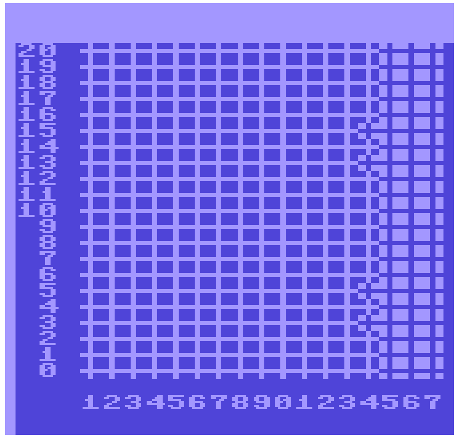
ISCSI

FCP

Infiniband

Innovations associated with storage?

- *Storage Access File level access takes center stage along with conventional Block level access.*
- Block level access: Block addresses are used to Read/Write data [Read/Write, Block #] to the storage media.
- File level access: Files are accessed by "semantics" instructions [example: Open, Close]. Data inside files is accessed by byte-ranges within the file (example: the first 10 bytes of a file). GFS (Google File System) is an example of a large scale distributed file system.



Sample conventional Block Allocation Map

Innovations associated with storage?

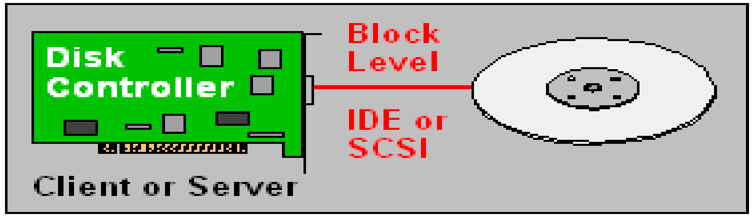
- **Metadata** is Data about data; in the context of storage metadata may describe an individual datum, or content item, or a collection of data including multiple content items.

- Examples include: file size, who created file, attributes such as read only, free block bitmaps, control data.

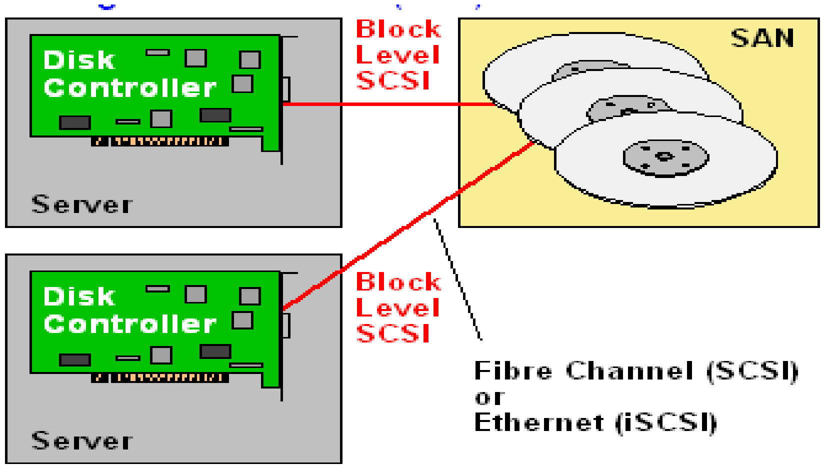
Innovations associated with storage?

Storage Software *from simple back-up and restore to advanced storage networks and storage management software functions.*

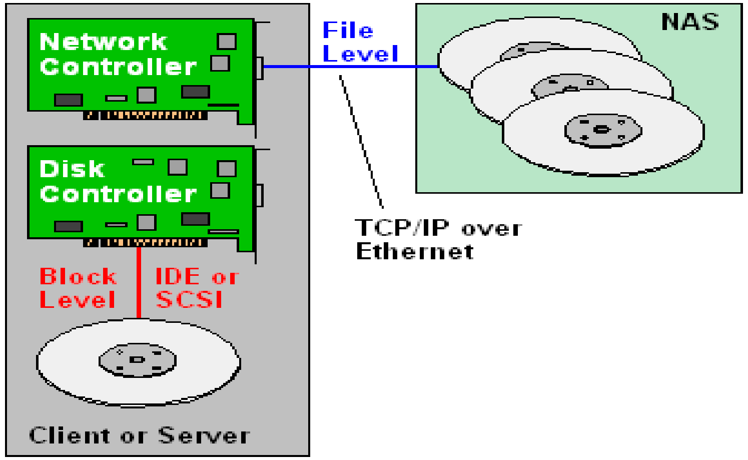
(A) Simple Direct Attached Storage (DAS)



(B) Storage Area Network (SAN)



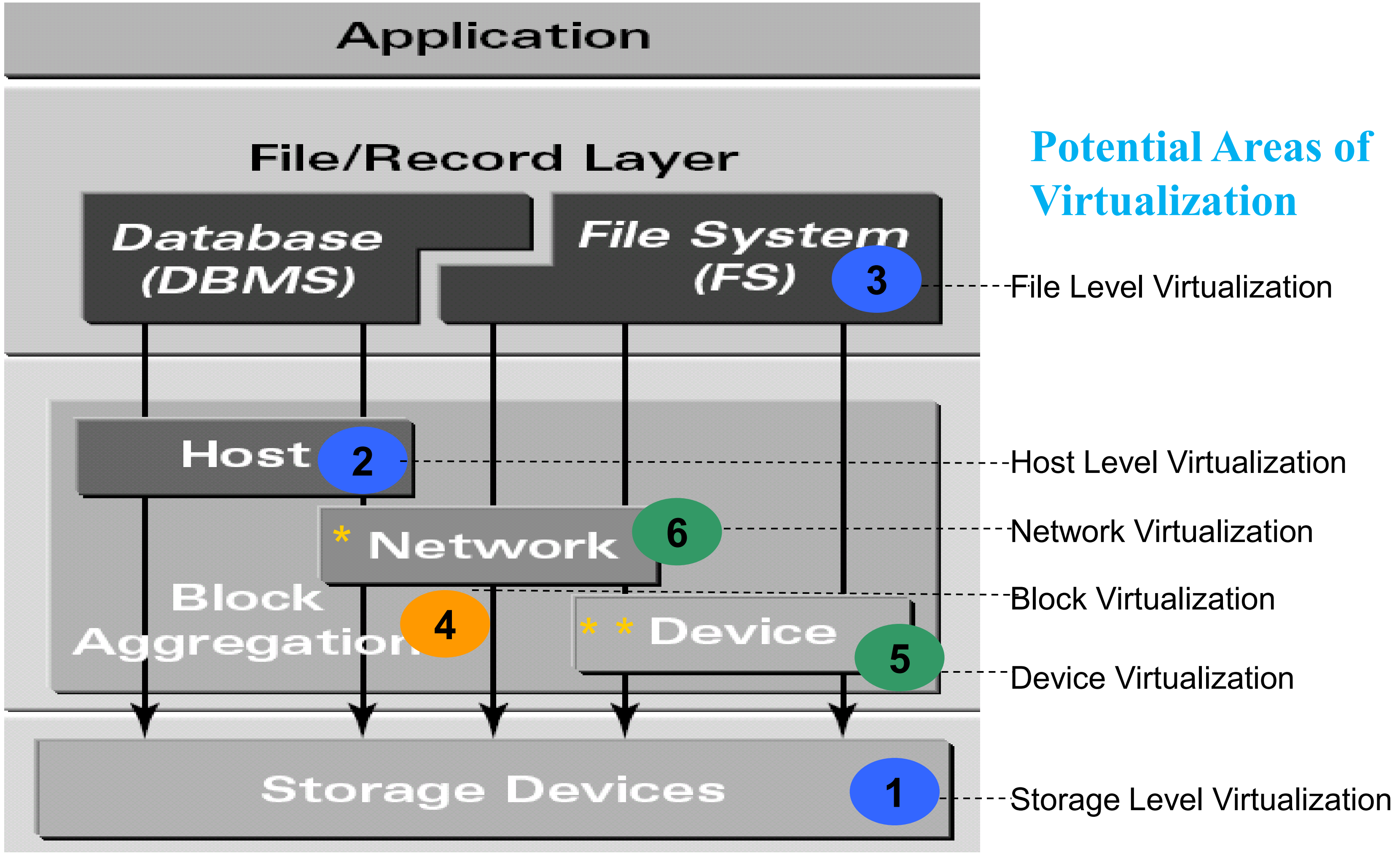
(C) Network Attached Storage (NAS)



NAS vs SAN

	NAS	SAN
Access Methods	**File access**	**Disk block access**
Access Medium	Ethernet	Fiber Channel
Architecture	Decentralized	Centralized
Transport Protocol	Layer over TCP/IP	SCSI/FC and SCSI/IP
Efficiency	Less	More
Sharing and Access Control	Good	Poor
Typical Applications	Web	Database
Typical Clients	Workstations	Database servers

Where can Storage be virtualized?



Potential Areas of Virtualization

File Level Virtualization

Host Level Virtualization

Network Virtualization

Block Virtualization

Device Virtualization

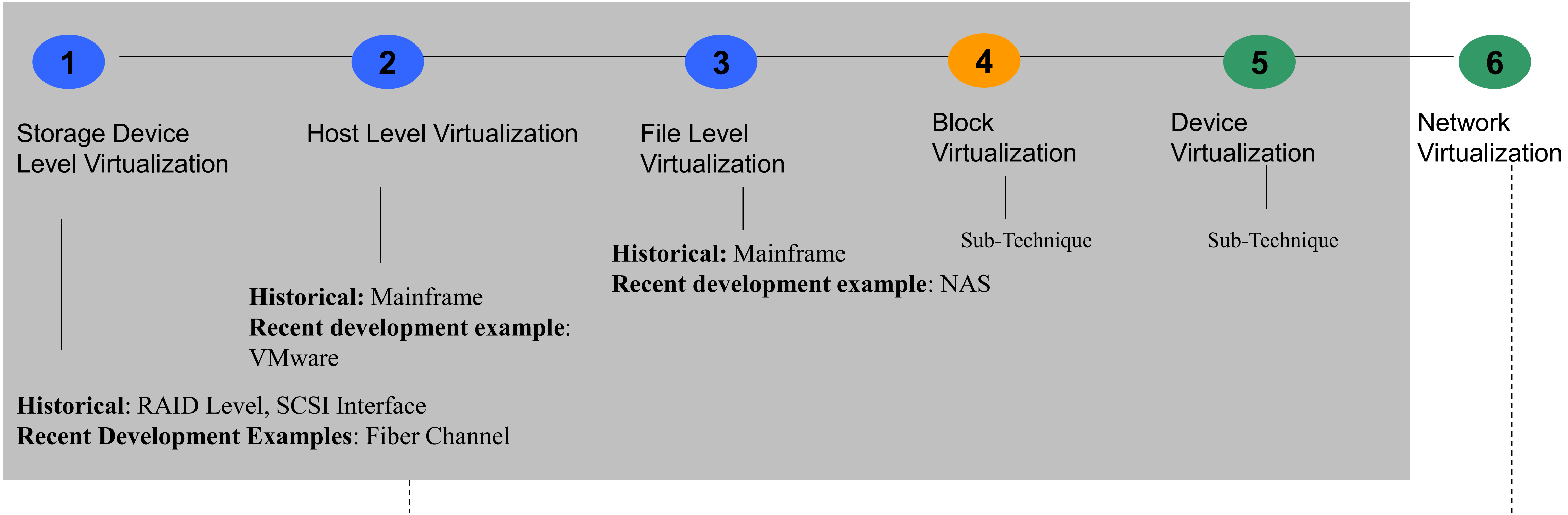
Storage Level Virtualization

Source: The Storage Networking Tutorials, SNIAVIRT- Page 20

http://www.snia.org/education/tutorials/

Where can Storage be virtualized?

Storage Virtualization: Innovations and Trends



1 Storage Device Level Virtualization

Historical: RAID Level, SCSI Interface
Recent Development Examples: Fiber Channel

2 Host Level Virtualization

Historical: Mainframe
Recent development example: VMware

3 File Level Virtualization

Historical: Mainframe
Recent development example: NAS

4 Block Virtualization

Sub-Technique

5 Device Virtualization

Sub-Technique

6 Network Virtualization

Major innovations continue to emerge even in historical areas of storage virtualization

Symmetrical (aka in-band) and Asymmetrical (aka Out-of-Band) are emerging as key areas of abstraction and virtualization.

Potential topics of research in storage virtualization

❶ Bayesian analysis for resource management

❷ Bayesian analysis for diagnostics

❸ Trusted domains for security

❹ Storage Virtualization and Metadata Standards

❺ Algorithm advances for block, device and other component virtualization techniques