

Chapter Three

Virtualization In Cloud Computing

Prepared By: Dr. Fatma Omara

Hand-Book of Cloud: Chapters 1- 8

CLOUD COMPUTING Principles and Paradigms: Chapter 1

Virtualization

Virtualization

- ❑ It is a technology to **run multiple same or different OSs on a single physical system** which are **completely isolated from each other** to.
 - Share underlying hardware resources*
 - Ex: Run both Windows and Linux on the same machine
- ❑ It is defined as **the abstraction over computing resources, such as**
 - storage, processing power, memory, and network, I/O, etc..
- ❑ It is the **process** by which **one computer behaves as many computers.**
- ❑ Virtualization used to improve ***IT throughput*** and ***costs*** by using physical resources as ***a pool*** from which ***virtual resources*** can be allocated.
 - VMWare white paper, *Virtualization Overview*

Virtualization

A technology to run multiple same or different isolated OSs on a single physical system by abstracting and partitioning its physical resource (storage, processing power, memory, and network or I/O) into multiple Virtual Machines (VMs) with different workloads to improve *IT throughput* and *cost*.

Dual Boot , Emulation and Virtualization

Dual Boot System

- A computer system in which *two operating systems* are *installed* on the same hard drive, allowing *either operating system* to be loaded and given control.

Emulation System

- A system *pretends* to be *another system*

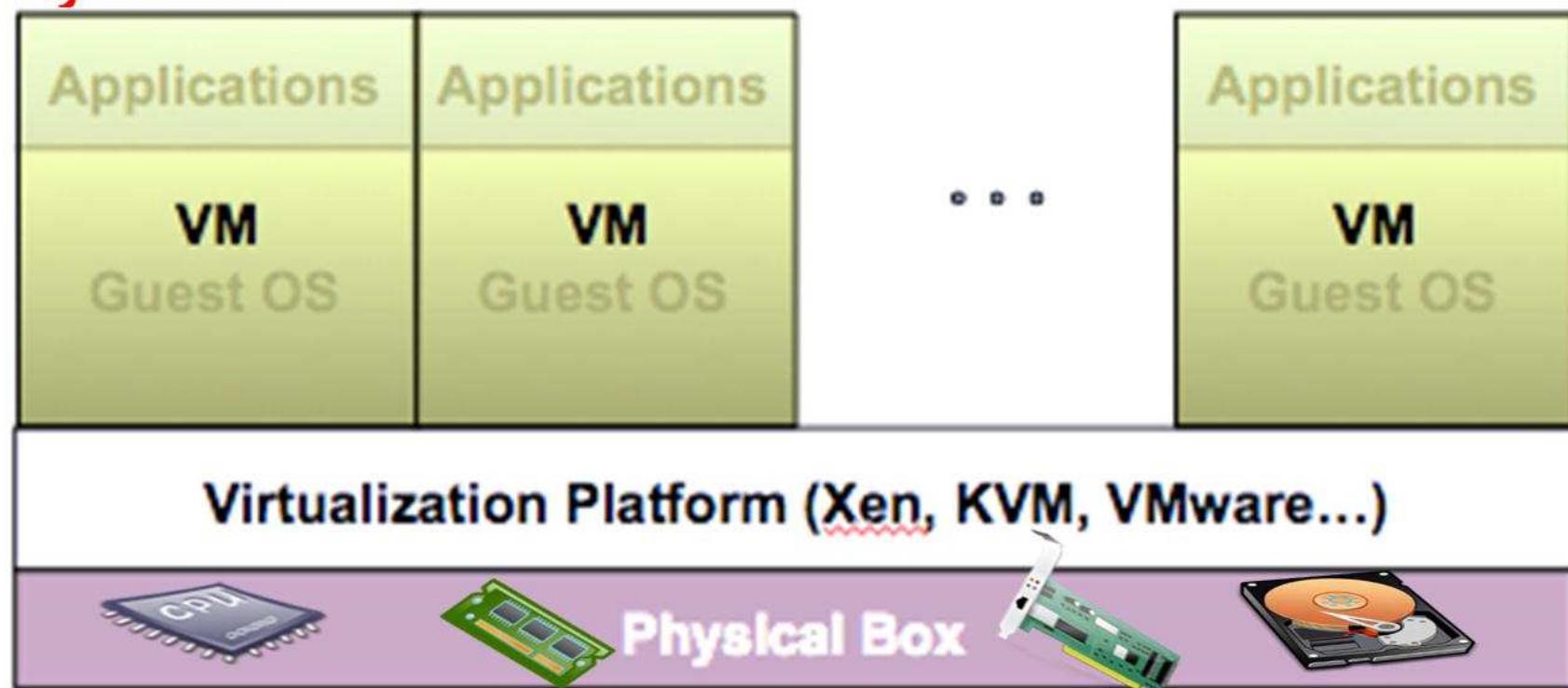
Virtualization System

- A system *pretends* to be *two or more of the same system*.
- *Virtualization layer* partitions *physical resource* of the *underlying physical server* into multiple *Virtual Machines (VMs)* *with different workloads*.

Similarities & differences

Virtualization Architecture

- ❑ A Virtual Machine (VM) is an *isolated runtime environment (guest OS and applications)*
- ❑ *Multiple VMs* can *run* on a *single physical system*



Virtualization Overview

❑ VMs can be scaled *up* and *down* on demand with a high level of resources' abstraction.

❑ Virtualization enables:

- *High reliable*, and *agile deployment mechanisms*, and *management of services*

❑ Virtualization provides on-demand *cloning* and *live migration* services which improve *reliability*.

****Virtualization Overview

Fundamental Idea :

- ***Abstract*** hardware of a single computer into several different execution environments
 - i.e., Virtualization creates VMs , and a VM can run both OS and application
- ***Single physical machine*** can run ***multiple operating systems concurrently***, each in its ***own virtual machine***

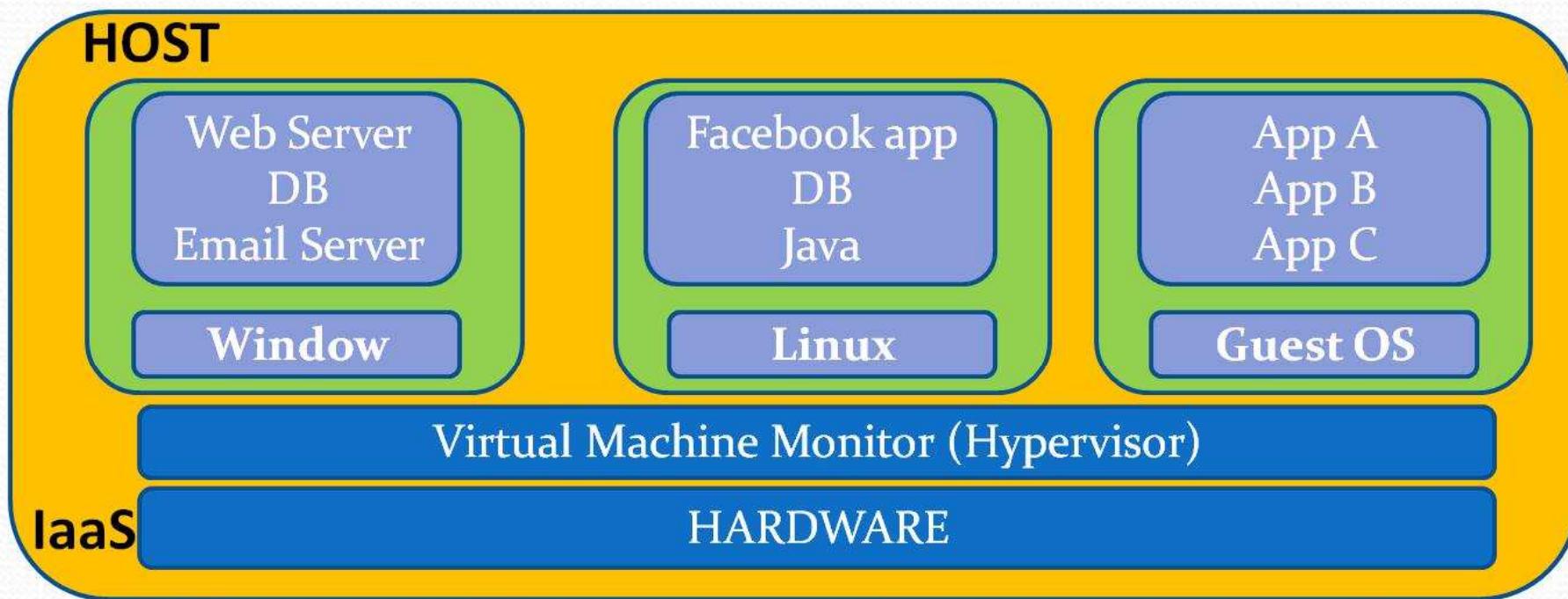
Virtualization Overview

Fundamental Idea :

- Abstract hardware of a single computer into several different execution environments by
 - ❖ Creating virtual system (**virtual machine**, or VM) on which operating systems and/or applications can run
 - (i.e., Virtualization creates VMs , and a VM can run both OS and application)
- Single physical machine can run ***multiple operating systems concurrently***, each in its ***own virtual machine***

Virtualization

VIM: **V**irtualization **I**nfrastructure **M**anagement



□ Several Components

- **Host** – underlying hardware system
- **Virtual Machine Manager (VMM)** or **hypervisor** – creates and runs virtual machines by providing interface that is *identical* to the host
- **Guest** – process provided with virtual copy of the Host
 - Usually an operating system

Virtual Machine, Guest Operating System & VMM (Virtual Machine Monitor)

Virtual Machine (VM)

- A representation of a real machine using software that provides an **operating environment** which can run **Host** or **Guest OS**
- A VM provides interface identical to underlying **bare hardware**
 - ✓ i.e. all devices, interrupts, memory, page tables etc.

Guest Operating System

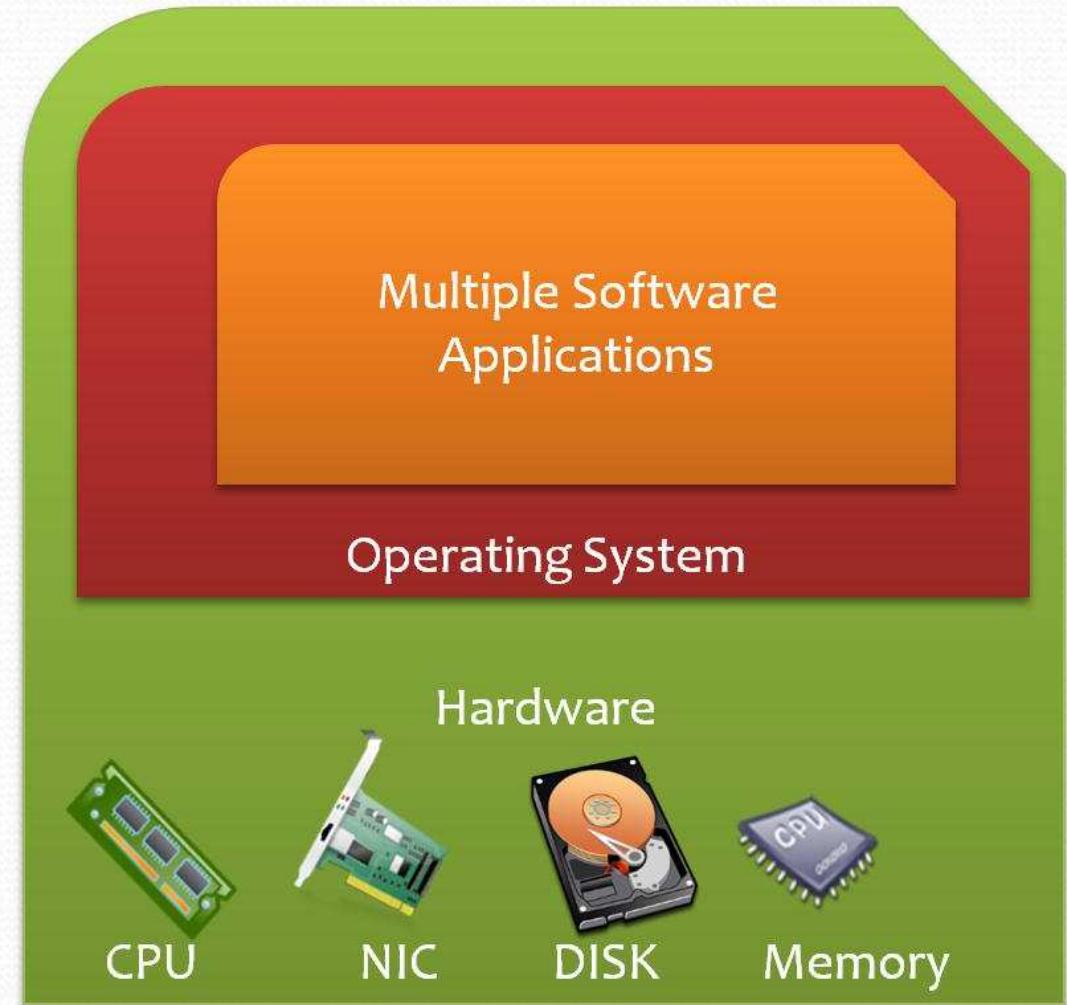
An operating system running in a virtual machine environment that would otherwise run directly on a separate physical system.

Virtualization Layer

Middleware between the underlying hardware and virtual machines represented in the system, also known as **Virtual Machine Monitor (VMM)** or **hypervisor**.

Server Without Virtualization

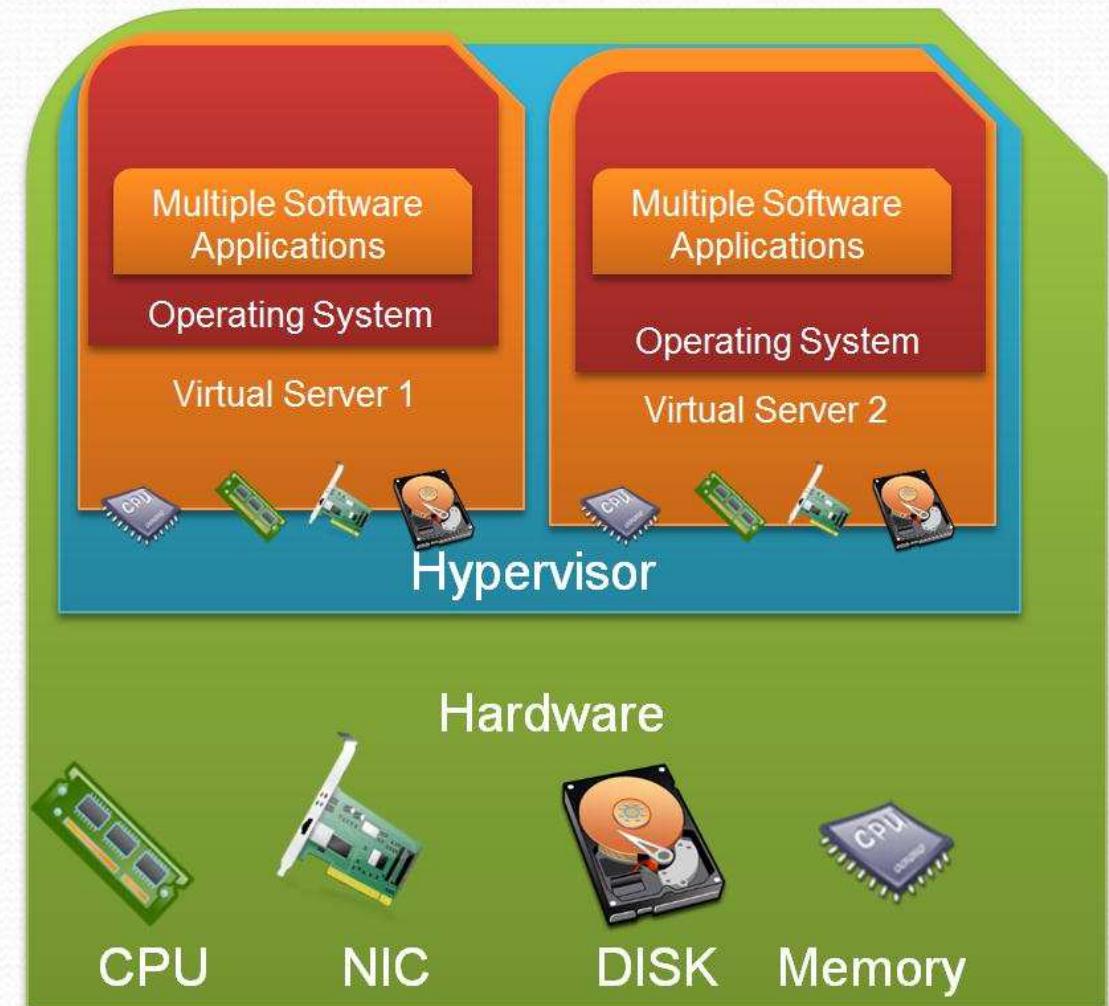
- **Single OS** can run at a time within a server.
- Software and hardware tightly coupled
 - **Running** multiple applications on same machine creates conflict
 - **Under** utilization of resources.
 - **Inflexible** and **costly** infrastructure.
 - Hardware changes require manual effort and access to the physical server.



(Courtesy of VMWare, 2008)

Server With Virtualization

- Hardware-*independence* of OS and applications
- Can run *multiple OSs simultaneously*.
- *Each OS can have different hardware configuration.*
 - *Efficient utilization* of hardware resources.
 - Each virtual machine is independent and can be *provisioned* any time .
 - *Save electricity*, initial *cost* to buy servers, *space* etc.
 - *Easy* to manage and monitor virtual machines centrally.



(Courtesy of VMWare, 2008)

Hypervisor

A **hypervisor**

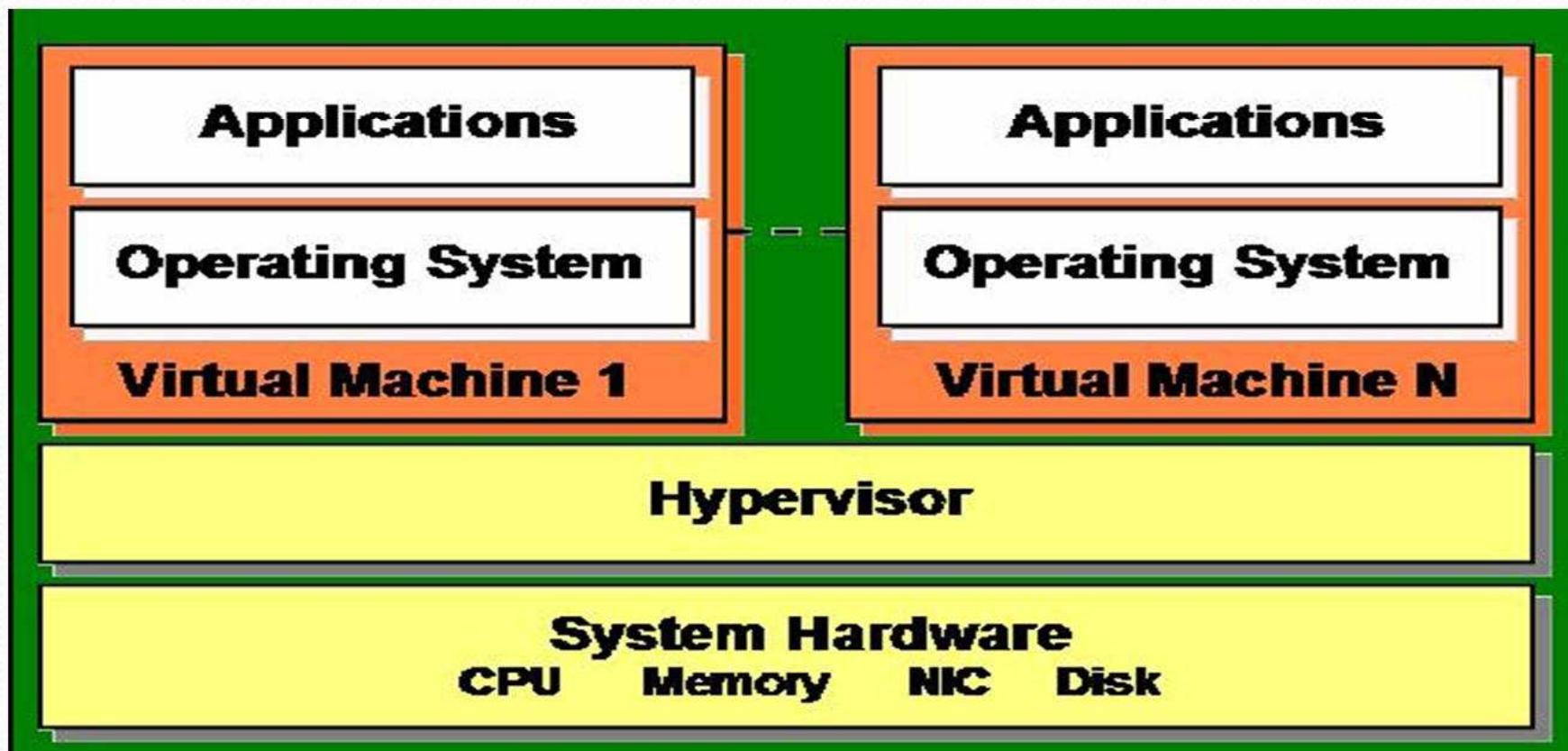
- Virtual Machine Manager/Monitor (VMM), or virtualization manager,
- A software that allows multiple **OSs** (guest) to share **a single hardware host**
- Each guest OS appears to have the host's processor, memory, and other resources all to itself.
- However, the hypervisor is actually
 - ✓ **Controlling** the host processor and resources,
 - ✓ **Allocating** what is needed to each operating system in turn, and
 - ✓ **Making sure** that the guest operating systems (called **Virtual Machines(VMs)**) cannot disrupt each other.

By

Hypervisor

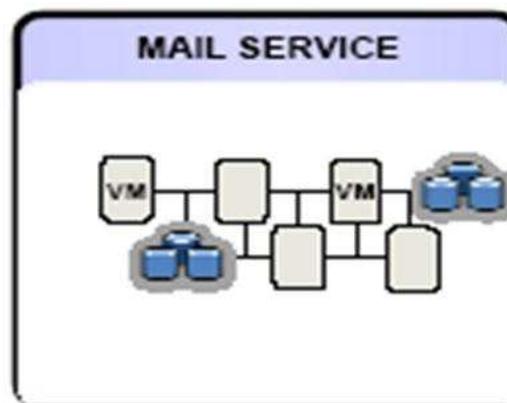
- Virtualization Software
 - ✓ VMWare, KVM, Xen, QEMU

The term **hypervisor** is a variant of **supervisor**, a traditional term for the kernel of an operating system: the **hypervisor** is the **supervisor** of the **supervisor**, with hyper- used as a stronger variant of super



User's view of virtualization

LOGICAL VIEW



Virtualization Layer - Optimize HW utilization, power, etc.

PHYSICAL VIEW



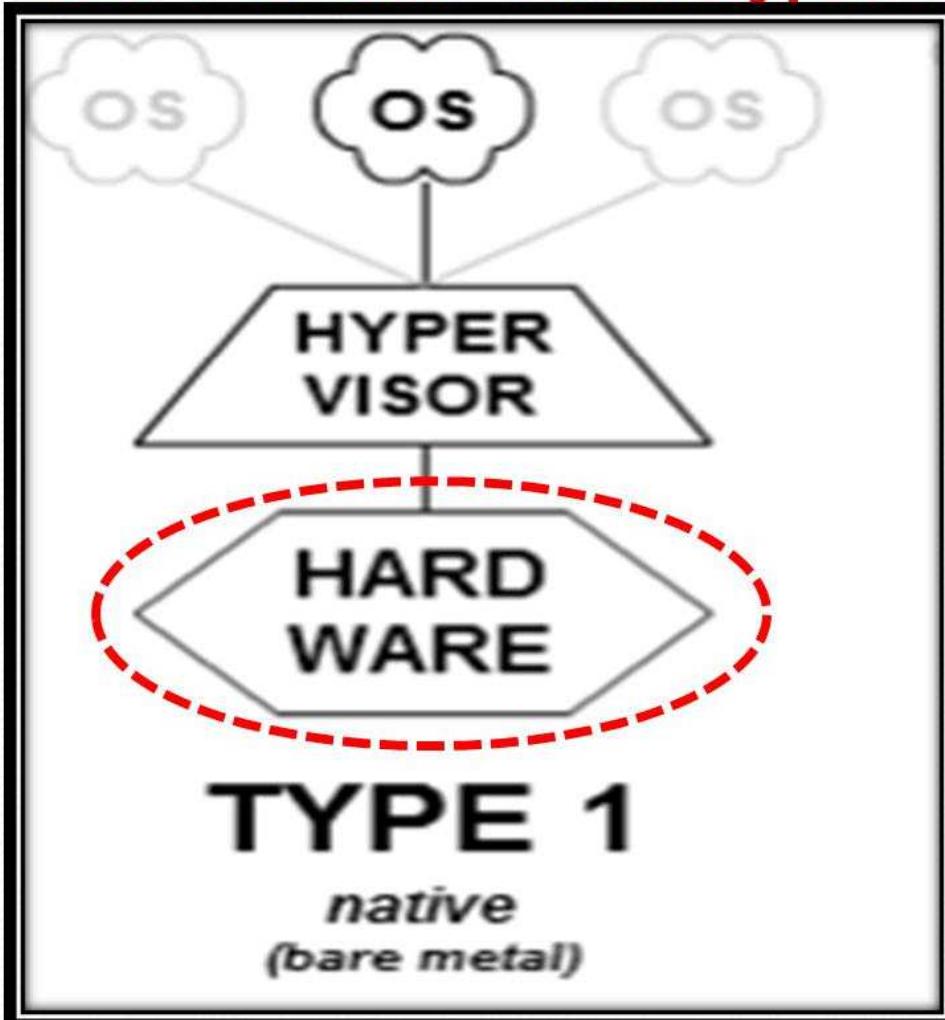
(Courtesy of VMWare, 2008)

Hypervisor Types

Hypervisor Types

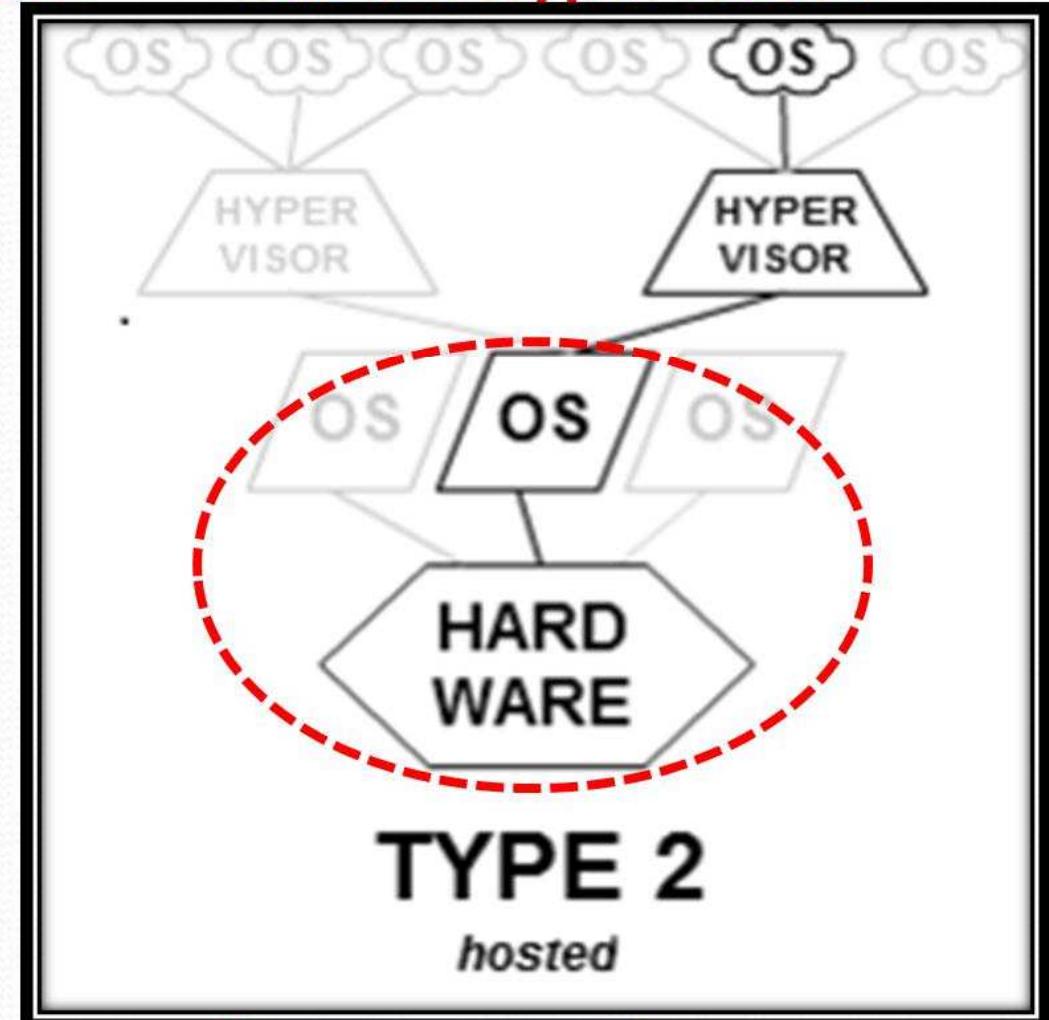
Type-1 (Full Virtualization)

native or bare-metal hypervisors



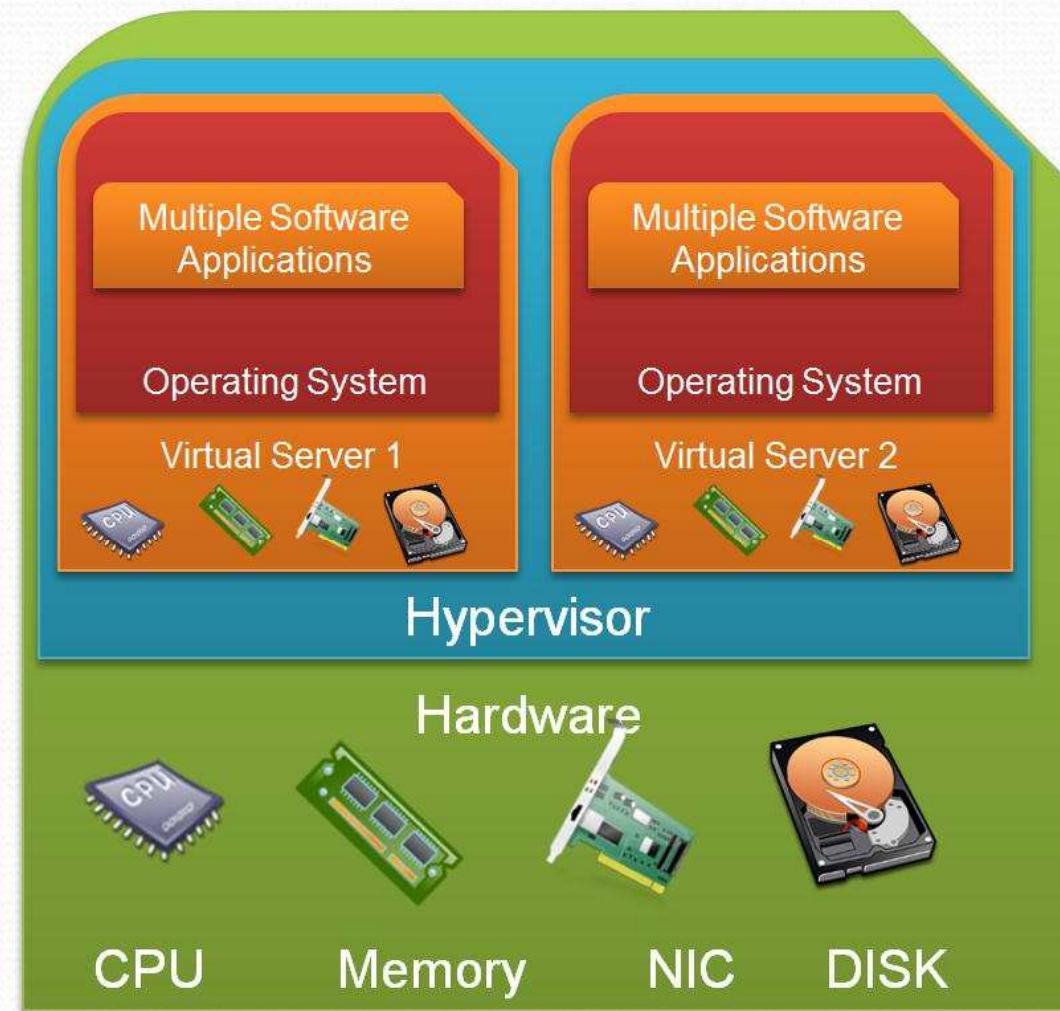
Type-2 (Para Virtualization)

hosted hypervisors



Full Virtualization

- It is called “**native or bare-metal hypervisors**”
- It directly sitting on top of the **bare hardware** devices
- Hypervisors Enable to run multi- unmodified **guest operating system**
- Guest OS is not aware that it is being virtualized.
- **Note:** No **host OS** is used here



e.g.: **VMware** uses a combination of direct execution and binary translation techniques to achieve full virtualization of server systems.

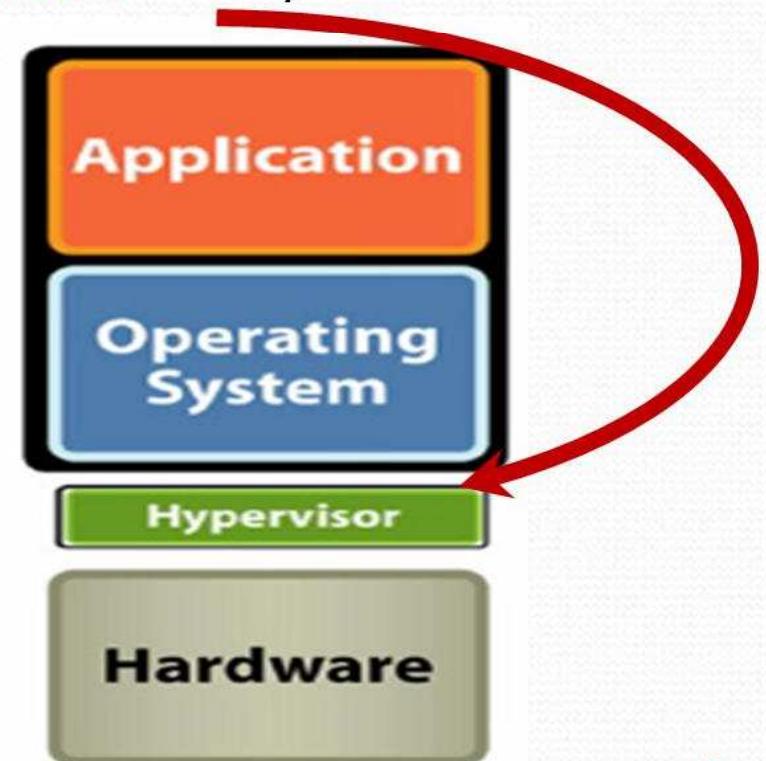
Full Virtualization Concepts

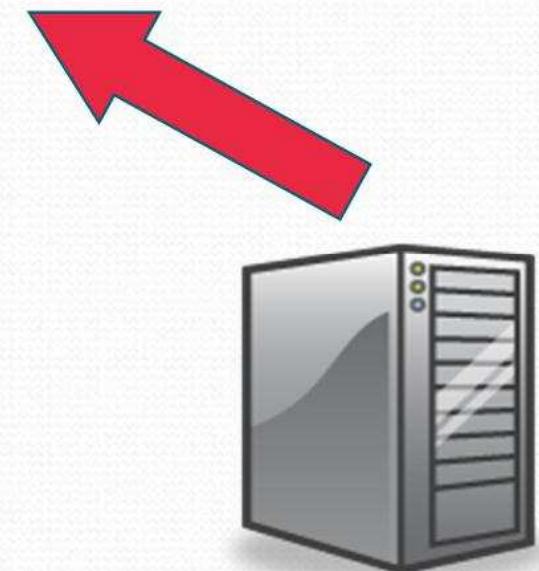
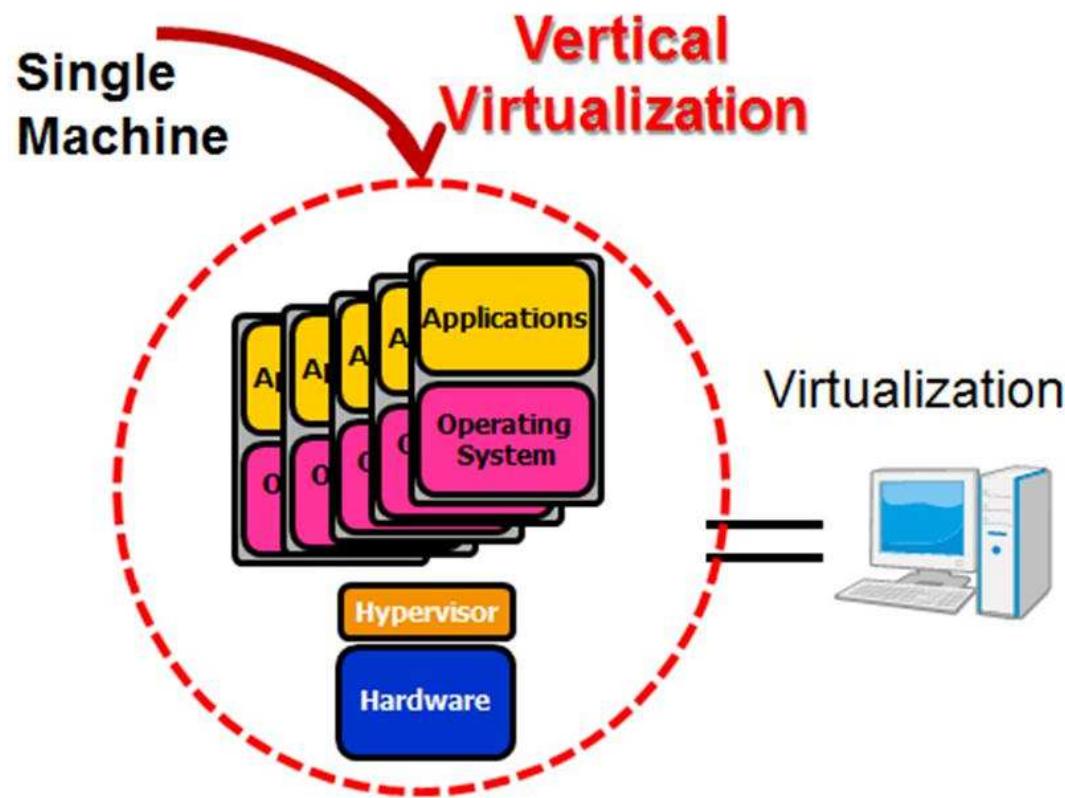
- A **hypervisor** is **a software virtualization technique** allowing multiple operating systems, called **guests** to run on a host machine.
- It is called the **Virtual Machine Monitor (VMM)**.

*The Existing Role of
the Operating System*



*Virtualization is Based on Insertion a
Hypervisor on Top of Hardware*





Full Virtualization

- A certain kind of virtual machine environment that *provides a complete simulation of the underlying hardware.*
- The result is a system in which *all software (including all OS's) capable of executing on the raw (bare) hardware*

Full virtualization has proven highly successful

- Sharing a computer system among multiple users
- Isolating users from each other (and from the control program) and
- Emulating new hardware to achieve:
 - ✓ Improve reliability,
 - ✓ Security, and
 - ✓ Productivity.

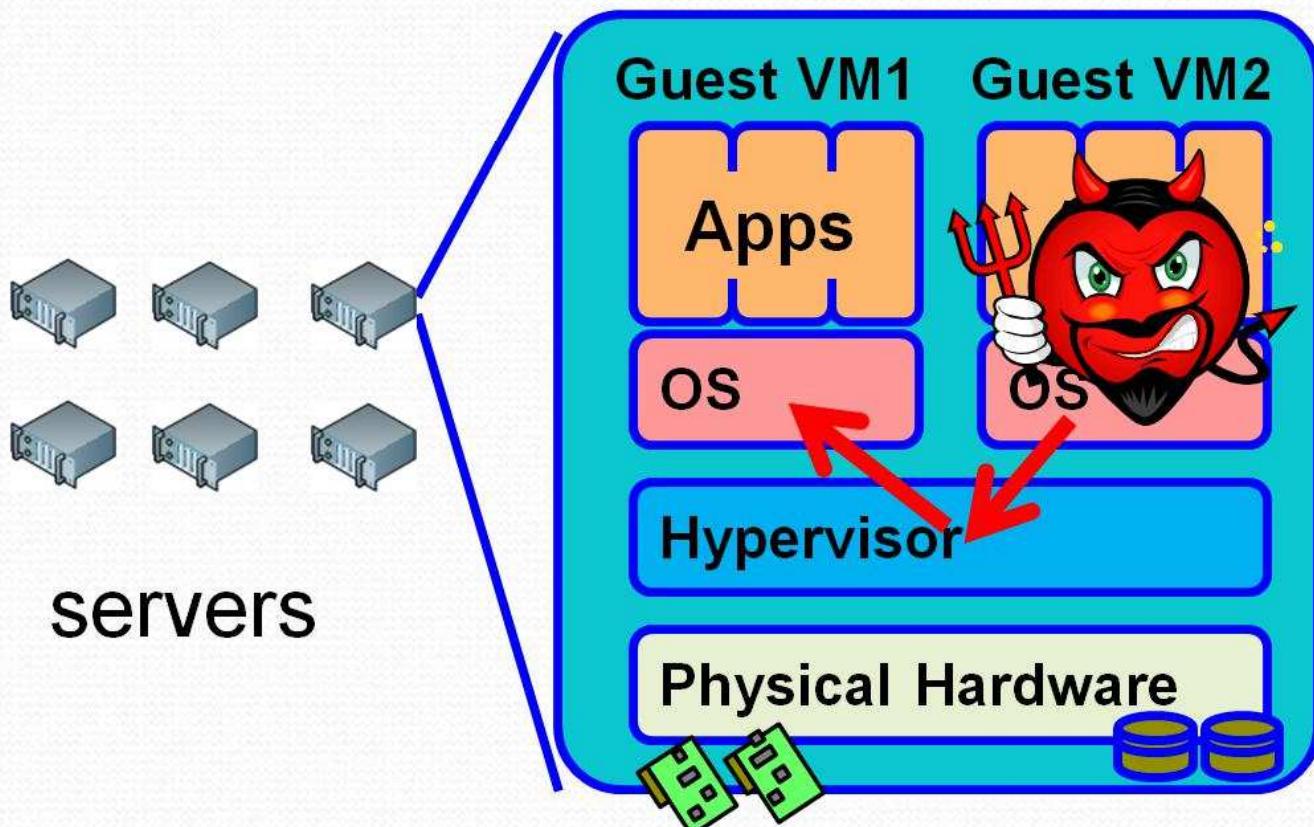
Full Virtualization -- Challenges

- 1) Security issues -- Interception
- 2) Simulation of privileged operations -- I/O instructions
- 3) The effects of every operation performed within a given virtual machine must be kept within that virtual machine – virtual operations **cannot be allowed to alter the state of any other virtual machine, the control program, or the hardware (Encapsulation).**
- 4) Some machine instructions of guest virtual machine can be executed directly by the hardware,
 - E.g., memory locations and arithmetic registers.
- 5) Some instructions of guest virtual machine cannot be allowed to execute directly; instead they must be trapped and simulated.
 - Such instructions **either access or affect state information that is outside** the virtual machine.
- 6) Some hardware is not easy to be used for full virtualization, **e.g.**, x86

Hypervisor Vulnerabilities

Malicious software can run on the same server:

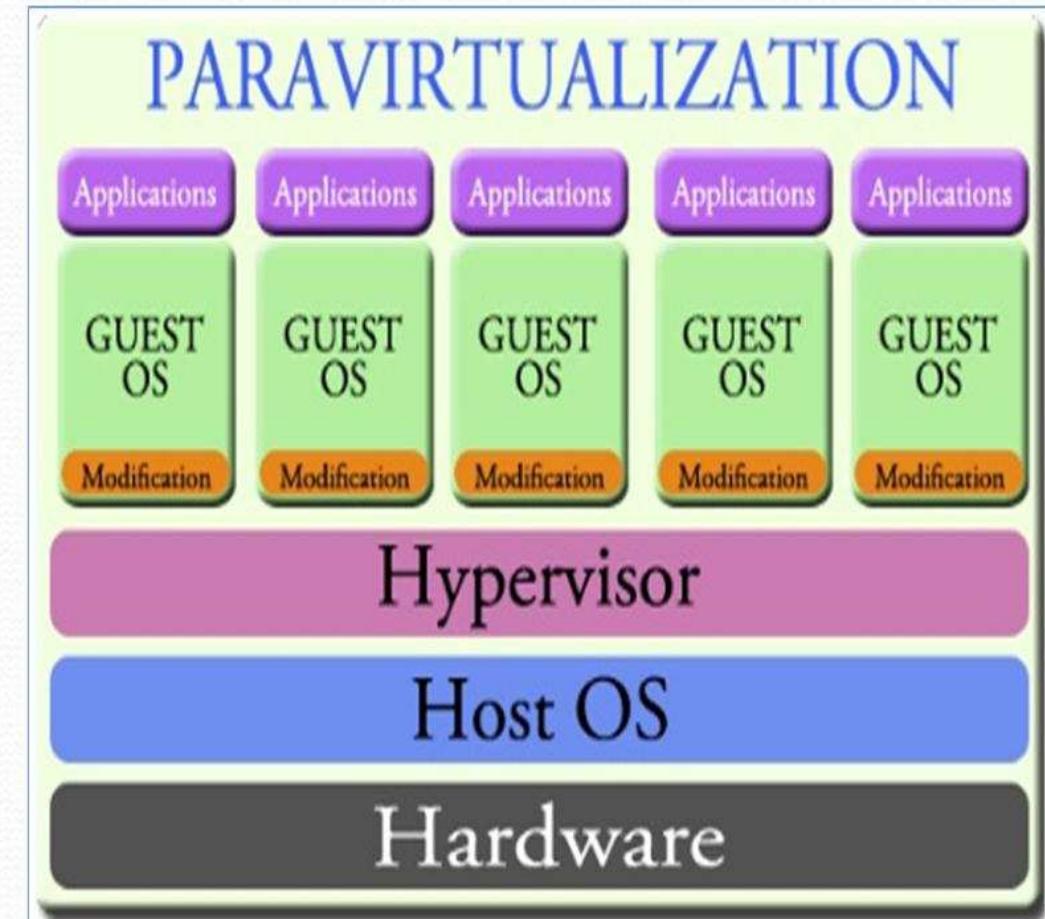
- Attack hypervisor
- Access/Obstruct other VMs



Para Virtualization

These hypervisors run on a conventional operating system (**Host OS**) just as other computer programs do.

- A **Guest OS** runs as a process on the **Host OS**.
- Para hypervisors abstract **Guest OSs** from the **Host OS**.
 - By explicitly *modifying* Guest OS. So that it is aware of being virtualized to allow near native performance.
- Improves performance.
- Lower overhead.



Ex: **Xen** -- modified Linux kernel and a version of Windows XP

Benefits of Virtualization

- **Consolidation:** Operate different OS's and applications on one single server
- **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 Ranging from *Hardware* to *Applications* in Five Abstraction Levels

[5]

Application level

JVM / .NET CLR / Panot

[4]

Library (user-level API) level

WINE/ WABI/ LxRun / Visual MainWin / vCUDA

[3]

Operating system level

Jail / Virtual Environment / Ensim's VPS / FVM

[2]

Hardware abstraction layer (HAL) level

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

[1]

Instruction set architecture (ISA) level

Bochs / Crusoe / QEMU / BIRD / Dynamo

[1] **Virtualization at Instruction Set Architecture (ISA) Level:**

Emulating a given ISA by the ISA of the host machine.

- e.g., MIPS binary code can run on an x-86-based host machine with the help of ISA emulation.
 - Typical systems: Bochs, Crusoe, Qemu, BIRD, Dynamo

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 ISA instruction may require **tens** or **hundreds** of native target ISA instructions to perform its function, which is relatively **slow**.
- **V-ISA** requires adding a processor-specific software translation layer in the compiler.

[2] **Virtualization at Hardware Abstraction Level:**

Virtualization is performed on top of the hardware (full Virtualization).

- It generates virtual hardware environments for VMs, and manages the underlying hardware through virtualization.
- Typical systems: VMware, Virtual PC, Denali

Advantage:

- Has higher performance and good application isolation

Shortcoming & limitation:

- Very expensive to implement (complexity)

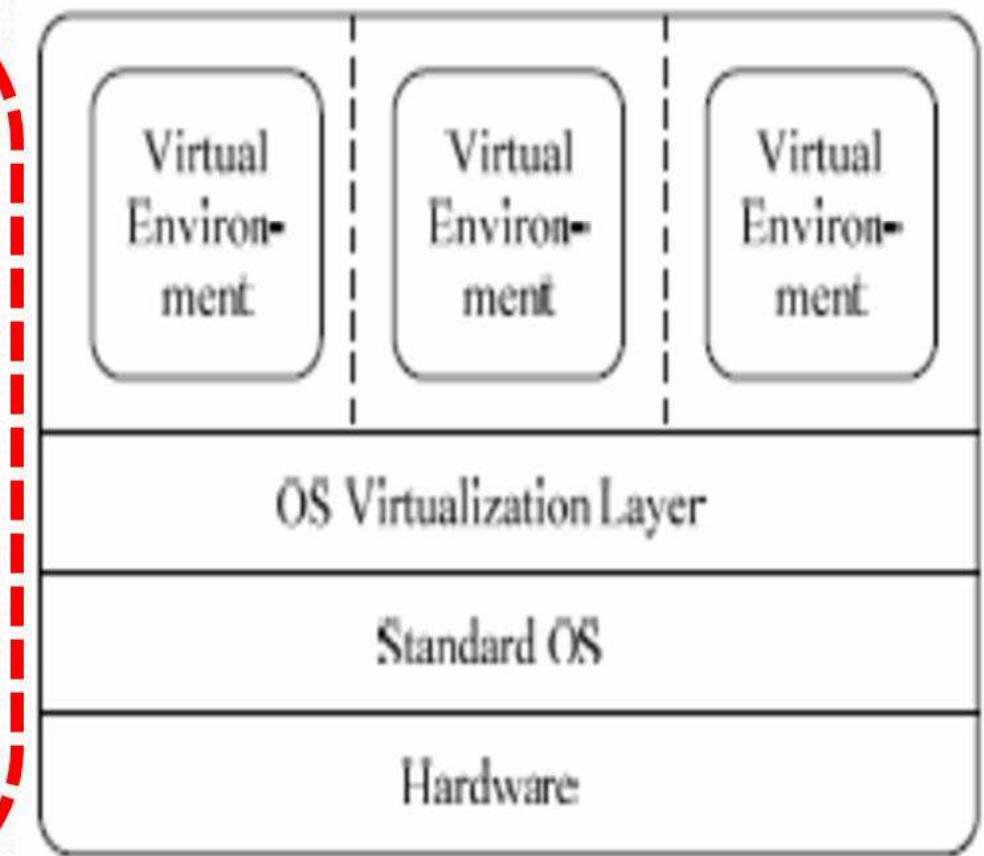
[3] Virtualization at OS Level:

Operating system level

Jail / Virtual Environment / Ensim's VPS / FVM

It is an abstraction layer between traditional OS and user applications

This virtualization layer creates ***isolated containers*** from a ***single physical server*** and the OS-instance to utilize the hardware and software in datacenters.



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

Shortcoming & limitation:

1. All VMs at the **OS-level** must have *the same kind of Host OS*
 - restrict *application flexibility* of different VMs on the same physical machine.
2. Poor application *flexibility* and *isolation*.

[5] User-Application Level: (SaaS)



It virtualizes an application

- This layer sits as an application program *on top of an operating system* and exports an abstraction of a VM that can run programs written and compiled to a particular abstract machine definition.
- Typical systems: **JVM**, **.NET CLI**, **Panot**

Advantage:

- It has the best *application isolation*
- *Support code portability*

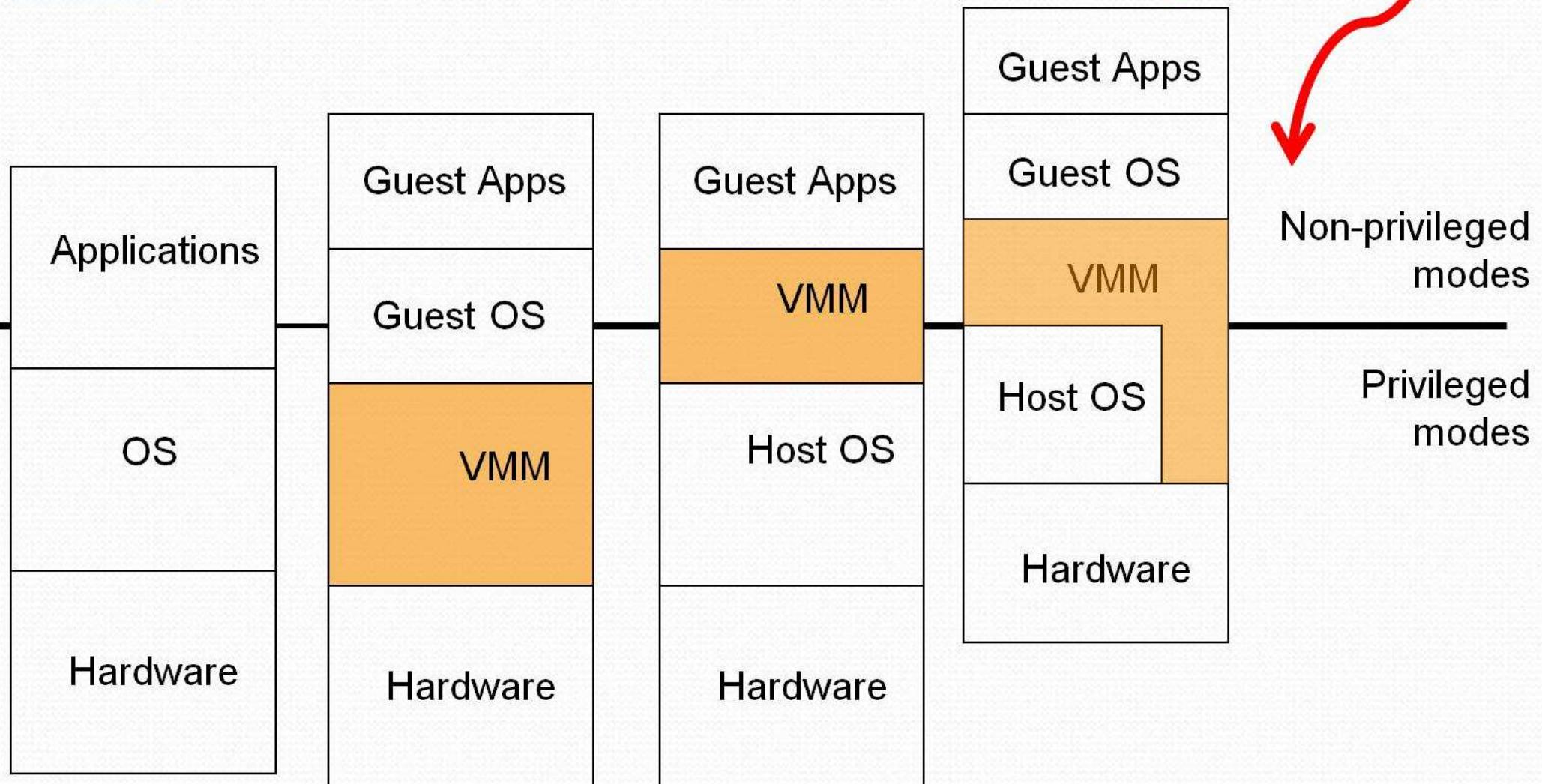
Shortcoming & limitation:

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

Generally; Virtualization

- It is the enabling technology which creates virtual machines that allows a single machine to act as if it were many machines (*OS-Level Virtualization*).
- Sharing the resources of a single hardware across multiple environments (*Full Virtualization*)
- **Host OS** provides an abstraction layer for running virtual **guest OSs** (*Para Virtualization*)
 - Enable portability (migration) of virtual servers between physical servers
 - Increase utilization of physical servers

Native and Hosted VM Systems



Confusion...

Full, Para, and
OS-Level
virtualization

OS-Level Virtualization

- A type of server virtualization technology which works at the OS layer. The *physical server and single instance of the operating system is virtualized into multiple isolated partitions, where each partition replicates a real server.*
- The OS kernel will run a single operating system and provide that operating system functionality to each of the partitions.

Para Virtualization

- Refers to the use of *software to allow system hardware to run multiple instances of different operating systems concurrently*, allowing you to run different applications requiring different operating systems on one computer system. The *operating systems do not interfere with each other or the various applications.*

Terminology

Small Scale Consolidation:

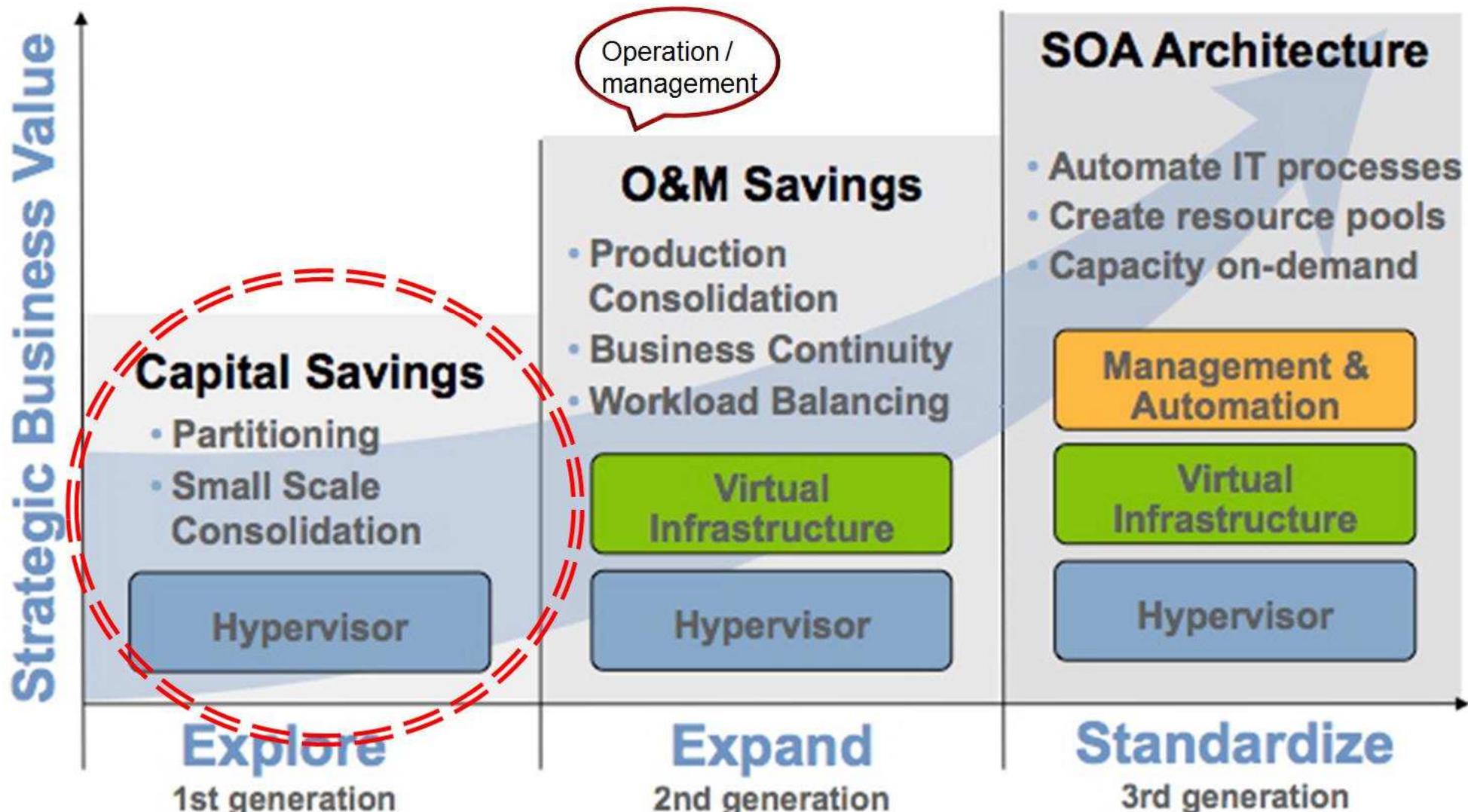
- Operate different OS's and applications *on one single server*

Production Consolidation:

- A company can achieve *greater efficiency* and *increase profitability* by *selling all or part* of its manufacturing operations.
 - The end result - *higher profitability* for the company.

Virtualization Evolution

Service-oriented architecture

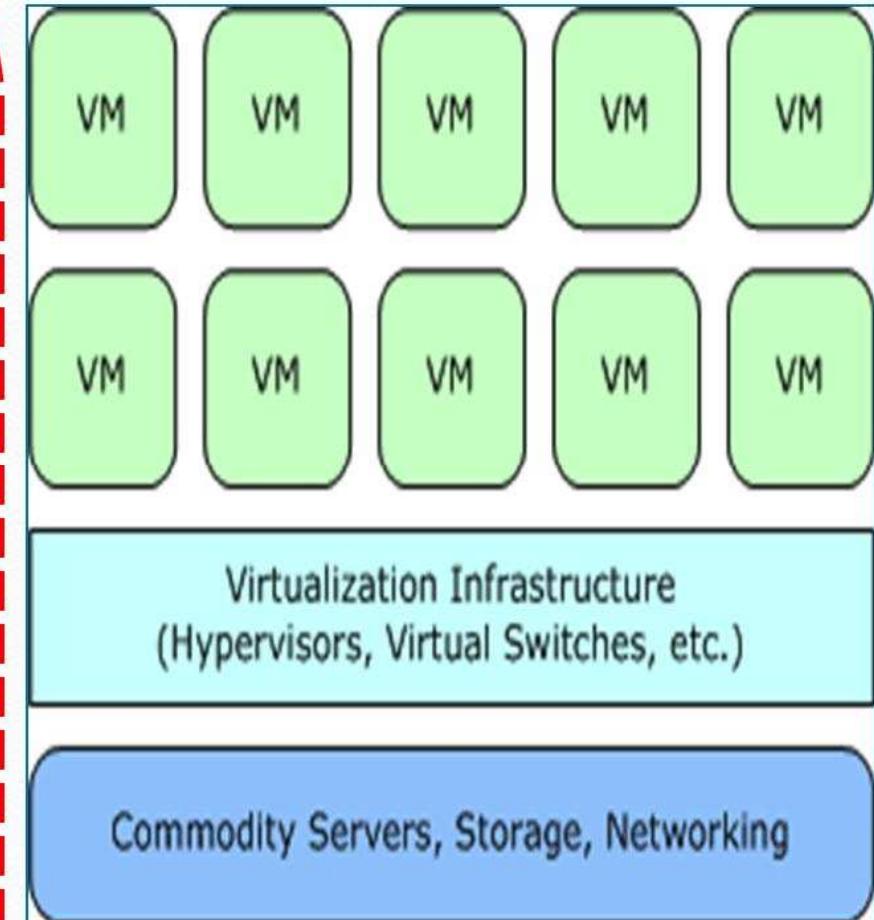


Virtualization in Cloud Computing

Virtualization in Cloud Computing

Cloud computing takes virtualization one step further:

- You don't **need** to **own hardware**
- Resources are **rented** as needed from a cloud
- Various providers allow 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
- You get **billed** only for what you used



Virtualization Over Cloud Computing

❑ Benefits of virtualization over cloud computing:

- ✓ Reduce capital expenses (CAP-EX), and
- ✓ Reduce maintenance and operation expenses (OP-EX) through server consolidation,
- ✓ Reduce physical space needed in data centers.
- ✓ Resource Management, Migration, Maintainability, High availability and Fault tolerance are other benefits.

❑ Virtualization is implemented using *hypervisors*.

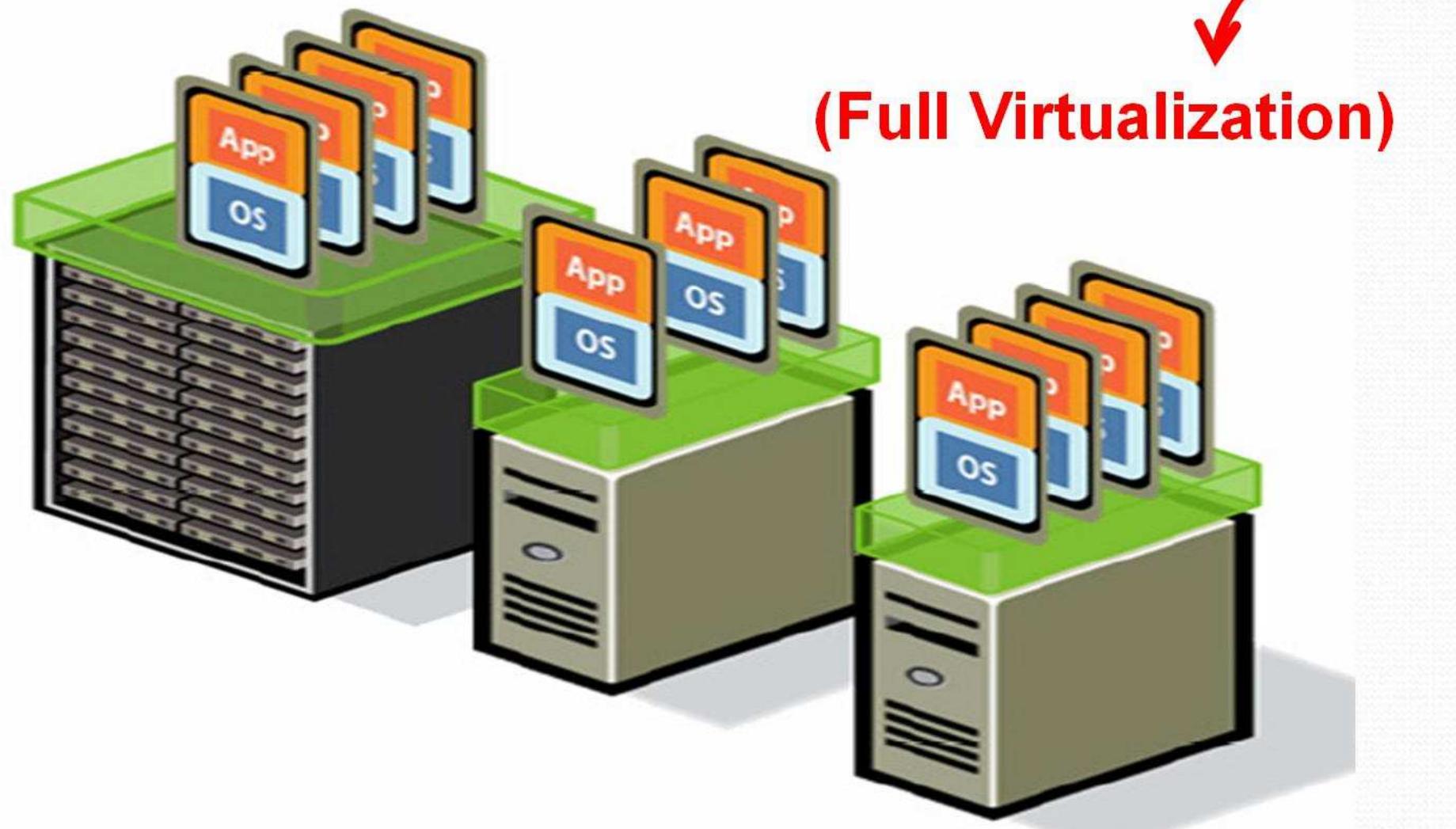
(c)Common Virtualization Allows Transformation a Server for Multiple Applications/OS

(Vertical Virtualization)

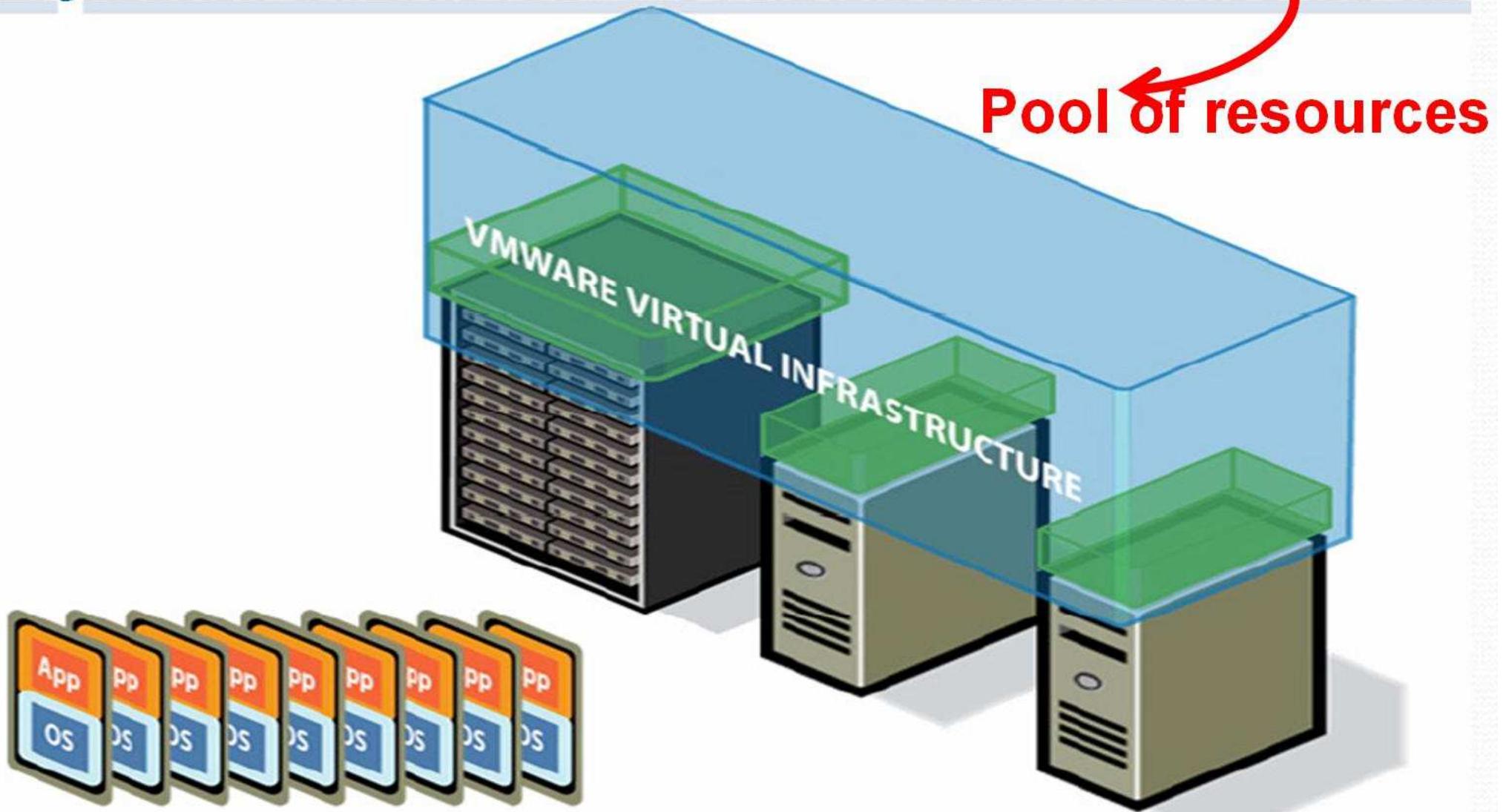
**OS-Virtualization
(Multiple Virtual servers)**



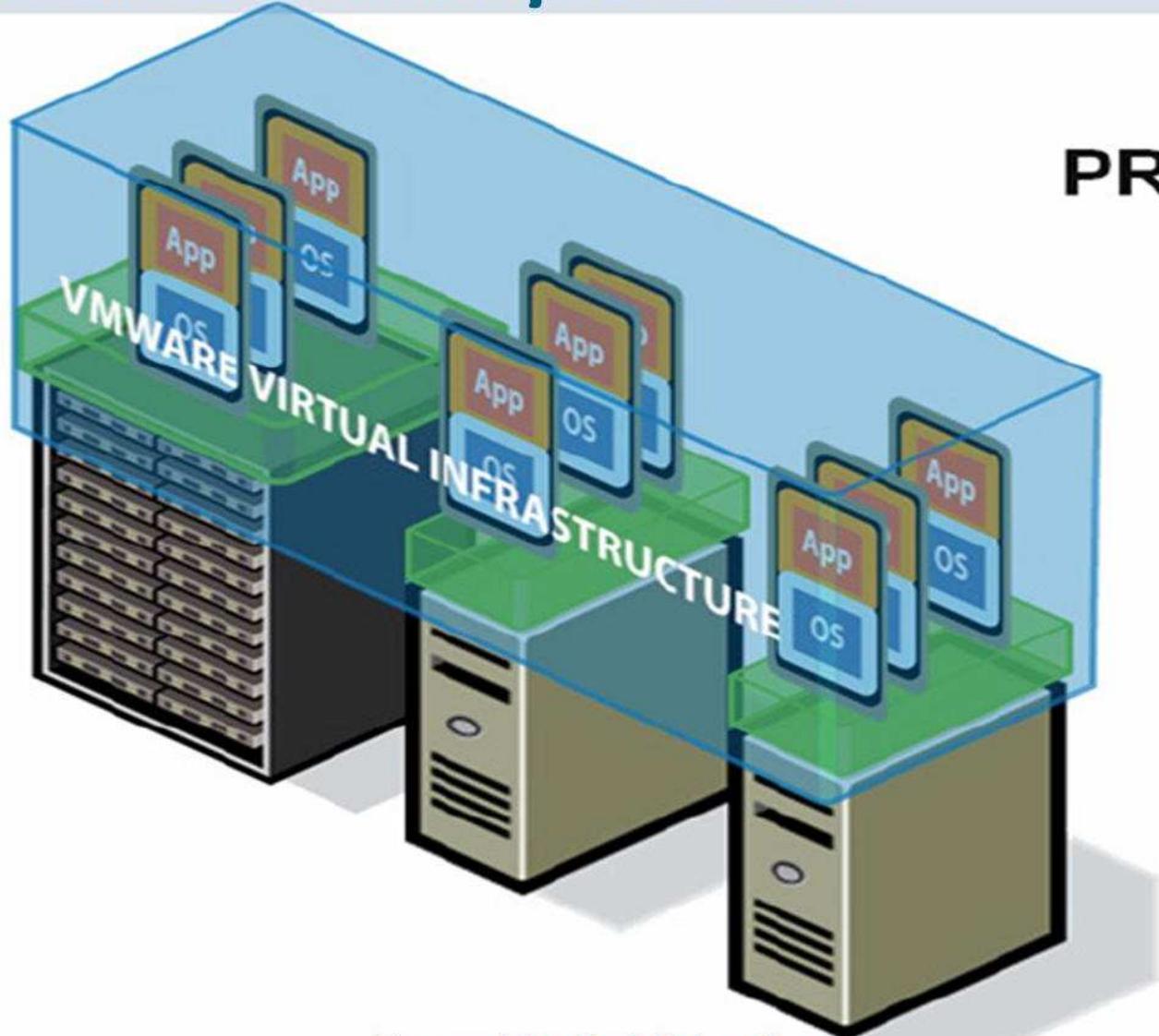
Virtual Machines Run on Any Hardware Configuration



Virtual Machines Can Run on a Shared Infrastructure

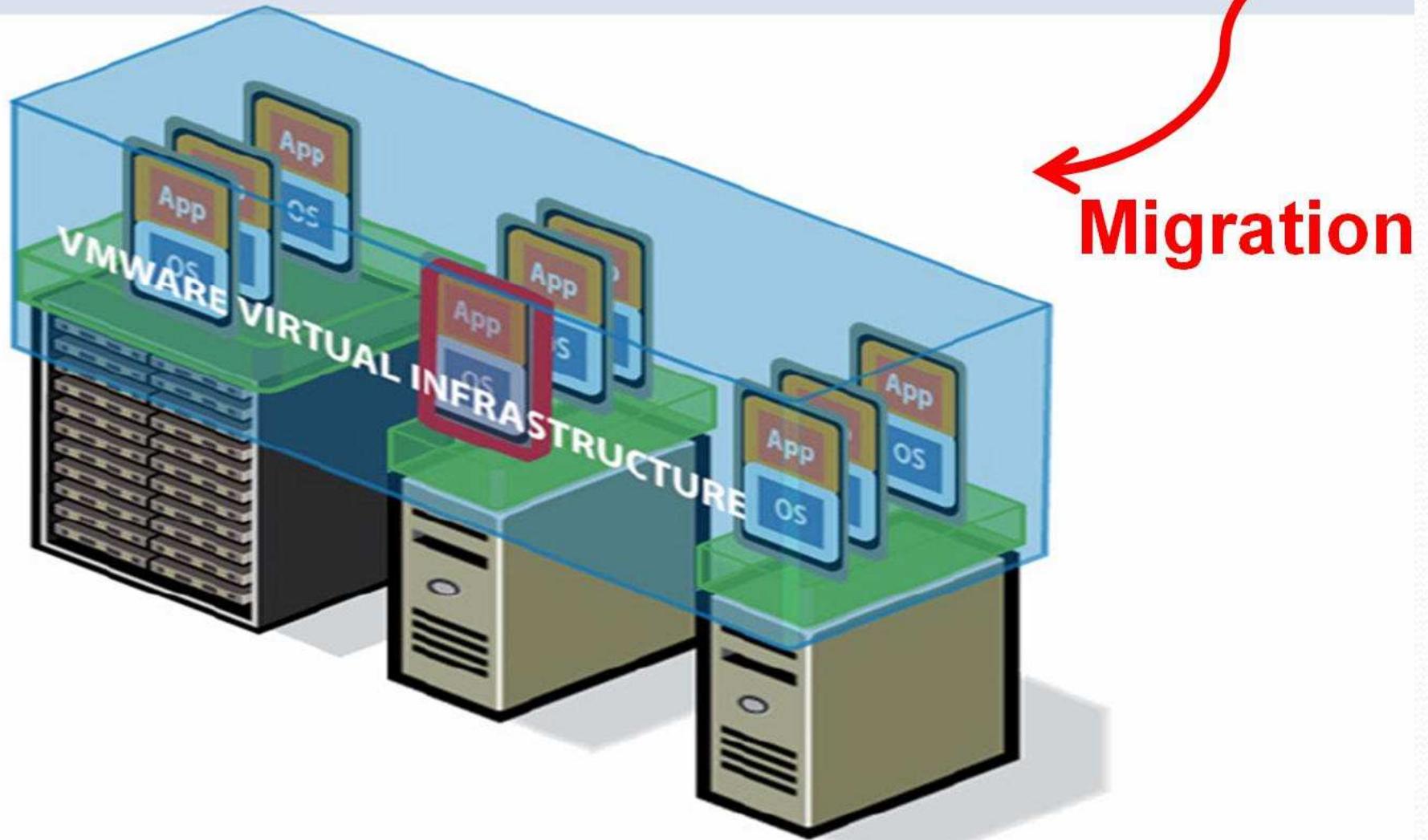


A Single Software Can Span Different Hardware Components

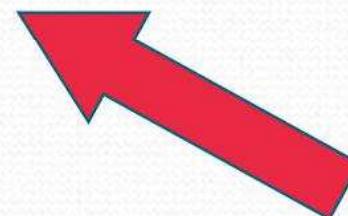
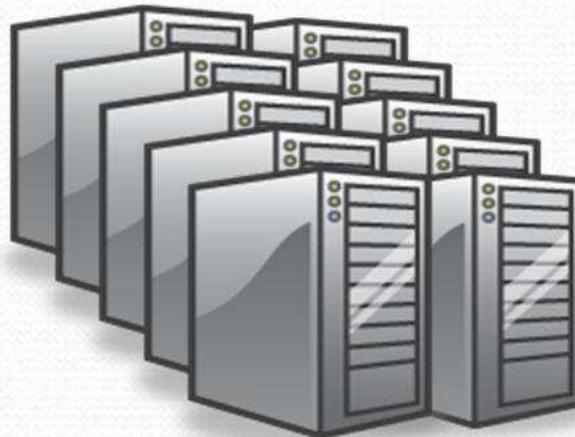


PROVISIONING

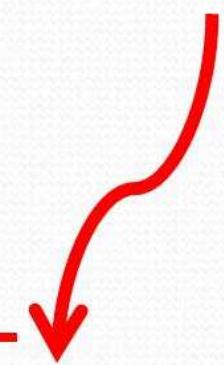
Virtualization Allows Moving Applications Without Service Interruption



Data Center Consolidation



Large Scale Consolidation



Virtualization

- Creation of a virtual version of hardware using software.
- Runs several applications at the same time on a single physical server by hosting each of them inside their own virtual machine.
 - A physical server can be utilized efficiently.

Primary approaches to virtualization

- **Platform virtualization** Ex : Server
- **Resources virtualization**
 - Ex : Storage, Network

Machine Stack showing virtualization opportunities

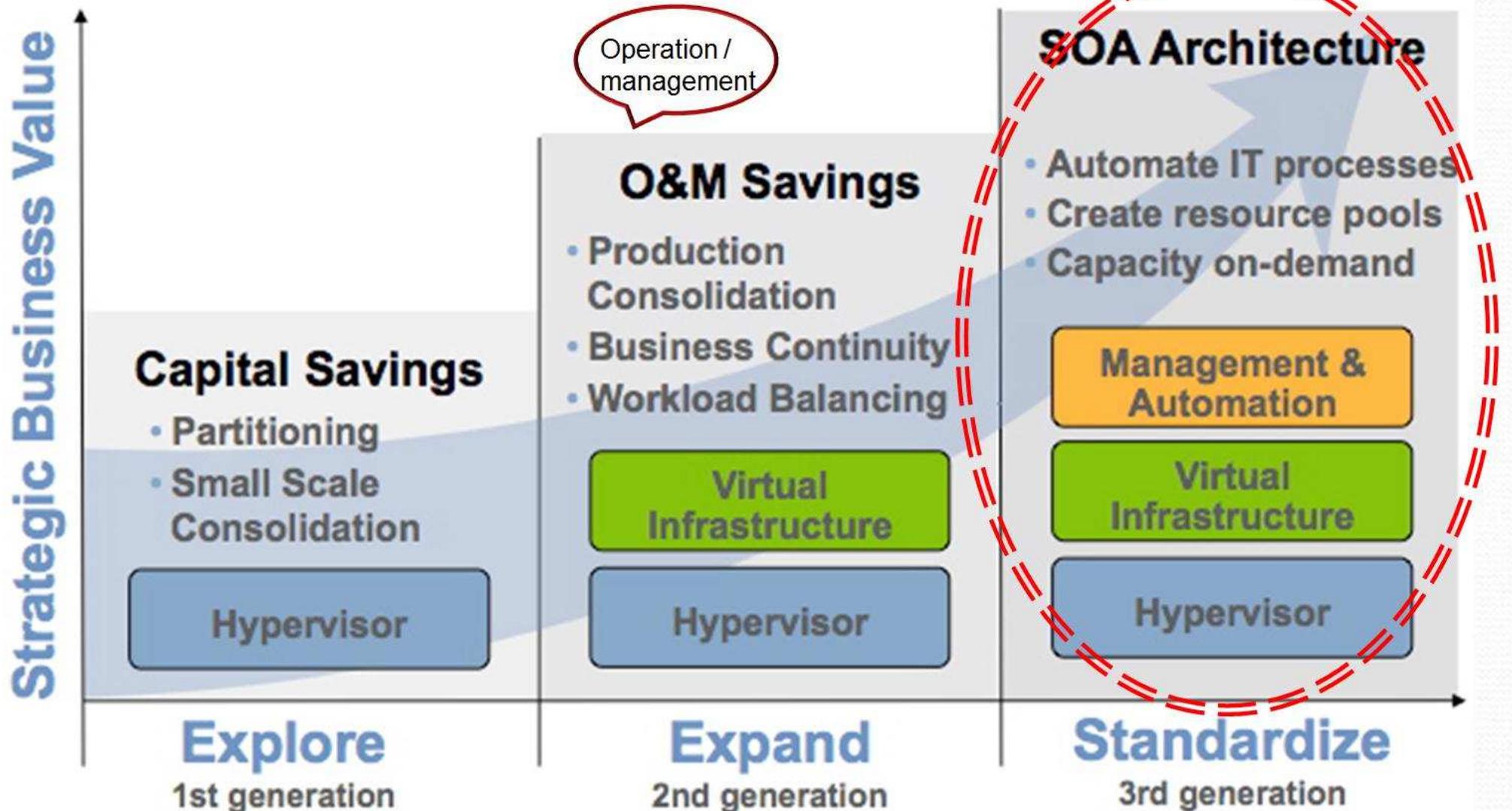
Application

Libraries

Operating System

Hardware

Virtualization Evolution



Advantages of Virtualization Over Cloud Computing

- Zero downtime maintenance
- Freedom from vendor-imposed upgrade cycles
- Instant provisioning
- Pooling hardware resource
- Virtual hardware supports legacy operating systems efficiently
- Dynamic resource sharing
- Security and fault isolation
- Business continuity, backups, and automated restoration

Impact of Virtualization Over Man Hours

Operations Require One Staff per 200-400 Virtual Machines

Note: Without virtualization one staff can handle up to 30 VMs

Before

From 20–40 hrs to build a server and re-load application...

- Build and configure hardware
- Load operating system
- Load configuration tools (Backup, Resource Kit, Monitoring, etc...)
- Assign 2 IP addresses
- Build 3 network connections, copper or fiber
- Turn over to applications team to re-load and re-configure software
- Test applications
- Coordinate outage/data migration

After

...To 15–30 min to copy a virtual machine and restart



333 servers replaced per year = ~ 10,000 man/hrs saved



Impact of Virtualization

Hard cost savings

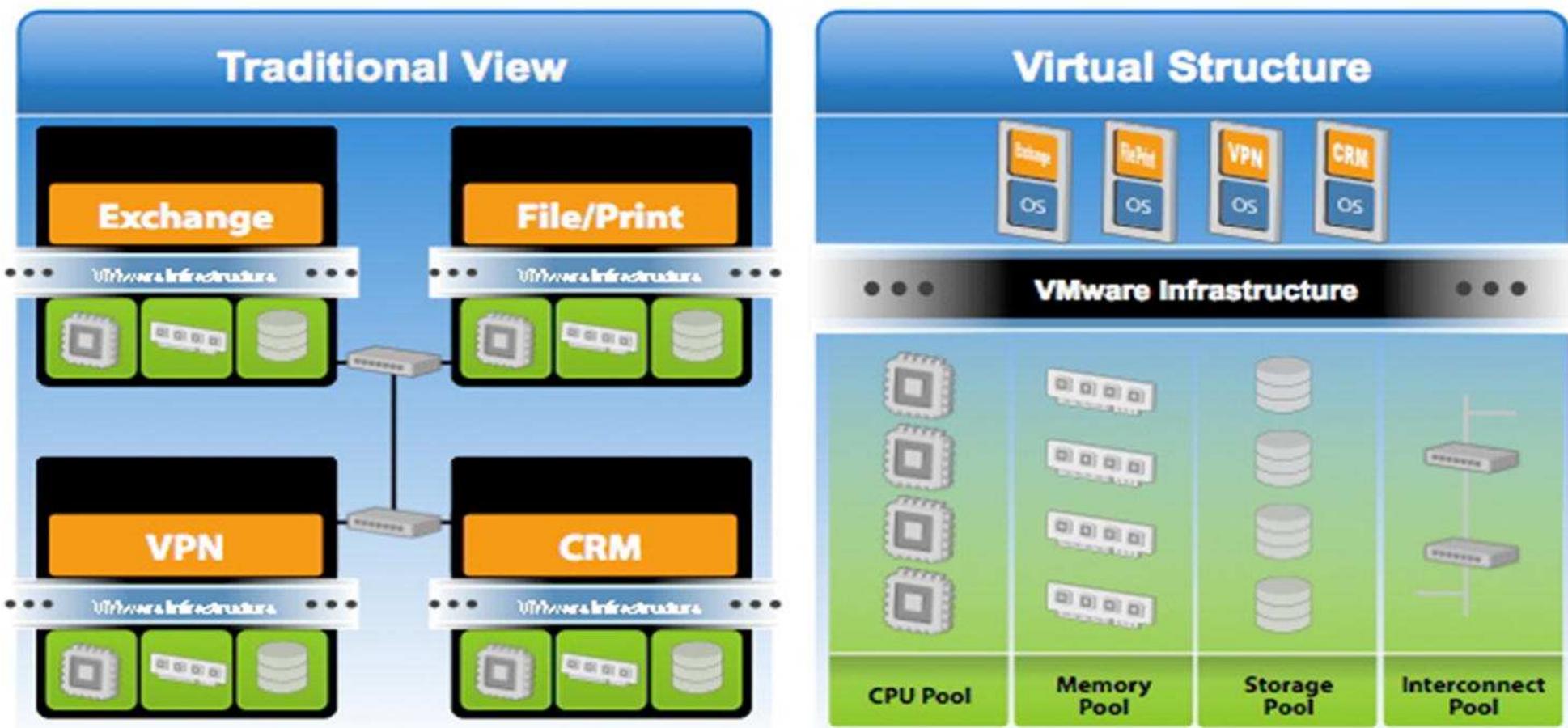
- > 70-80% reduction in data center space, power infrastructure
- > \$8M cumulative savings since 2003

Operational efficiency

- > Server rebuild and application load went from 20-40 hrs => 15-30 min
- > 10,000 man hours saved per year

Deploying Virtualization

From Dedicated Processing to Pooled Processing •



System Virtual Machines Applications

- Implementing Multiprogramming (**HPC**)
- Multiple single-application virtual machines (**SPMD**)
- Multiple **secure** environments
- Managed application environments
- Mixed-OS environments
- Legacy applications
- Multiplatform application development
- New system transition

System Virtual Machines Applications (cont.)

- Management simplification
 - Dynamic provisioning
 - Workload management/isolation
 - Virtual machine migration
 - Reconfiguration
- Virtualization protects IT investment
- Virtualization is a true scalable multi-core work load

Virtualization - Green ICT

The Reality:

- Most servers only use 5-15% of their capabilities on average, while consuming 60-90% of their peak power.

The Solution - Virtualization:

- Use one server to host multiple applications.
- Reduce energy consumption
- Reduce CO2 emissions

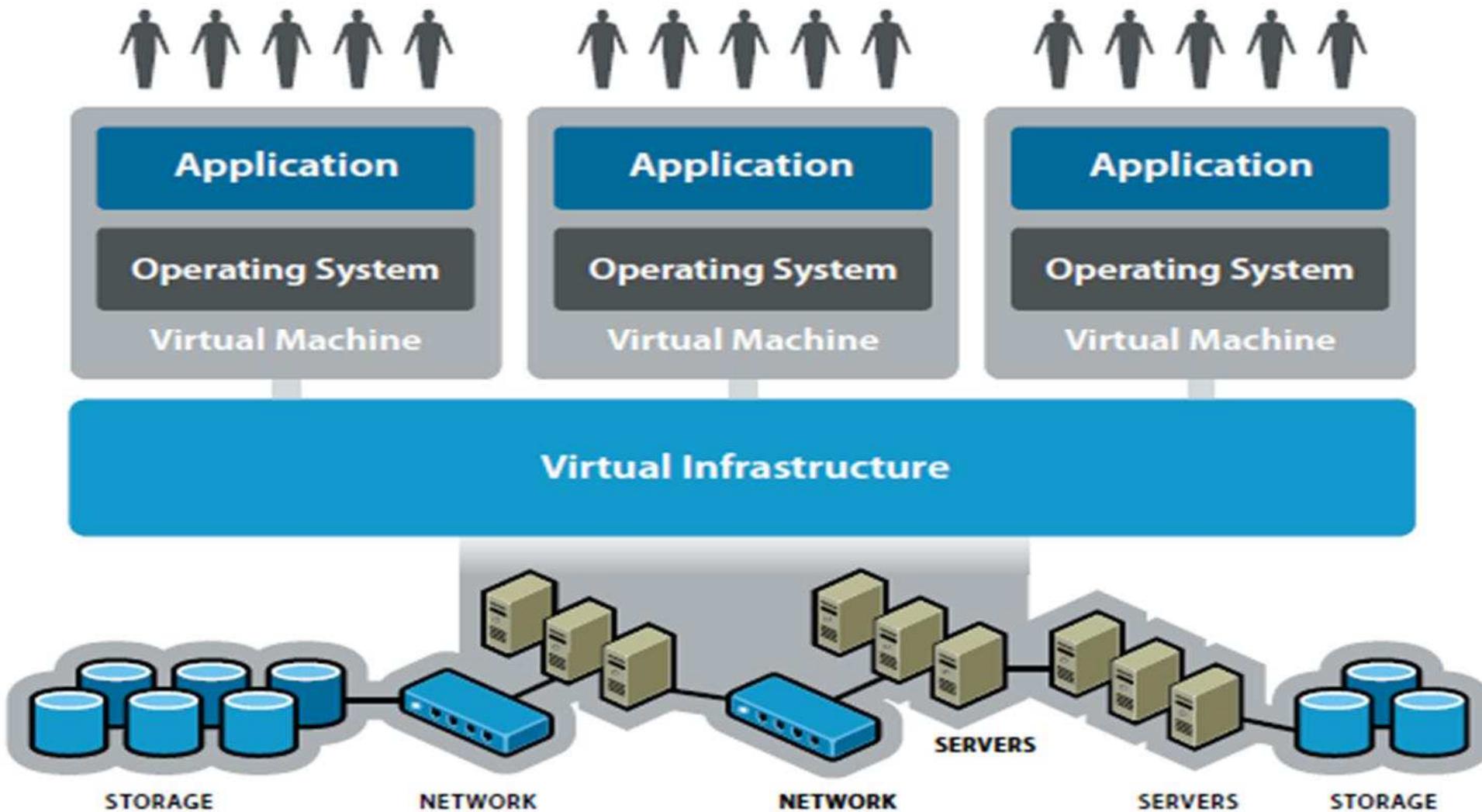
Running fewer, highly utilized servers:

- Free up space and Reduce power.
- Less space and power is better for Environment and saves money.



Massively Virtualized Model - Cloud

Infrastructure as Service (IaaS) Model



IaaS Model

- Cloud Computing offers ***infrastructure as a service*** which is based on:
 - Pay-as-you-use model
 - On-demand computing model
- To provide ***infrastructure as service (IaaS)***
 - ***Provisioning*** of the cloud infrastructure in data centers is a prerequisite
- However, provisioning for ***systems and applications*** on a **large number of physical machines** is traditionally a ***time consuming process***
- There are ***two core services*** enable the users to get the best out of the ***IaaS model***:
 - **Virtual machine provisioning**
 - **Migration services**

IaaS Model- Cont..

Virtualization Services:

Historically

- ❑ When it needs to install a ***new server*** for a certain workload to provide a particular service for a client
 - ***lots of effort*** needed by ***IT administrator***, and ***much time*** was spent to install and provision a new server because:
 - The administrator has to follow specific checklist and procedures to perform this task on hand

Now

- ❑ By emerging of virtualization technology and the cloud computing IaaS model,
 - it is just ***a matter of minutes (15-30 min)*** to achieve the same task

IaaS Model- Cont...

Migration Services:

Previously

- ❑ whenever there was a need for performing a *server's upgrade* or *performing maintenance* tasks, you would exert a lot of *time* and *effort*
 - Because it is an expensive operation to maintain or upgrade a *main server that has lots of applications and users*

Now

- ❑ With the advance of *revolutionized virtualization technology and migration services associated with hypervisors' capabilities*
 - *Server's upgrade* or *performing maintenance*, etc.) are very easy and need no time to accomplish

IaaS Model- Cont...



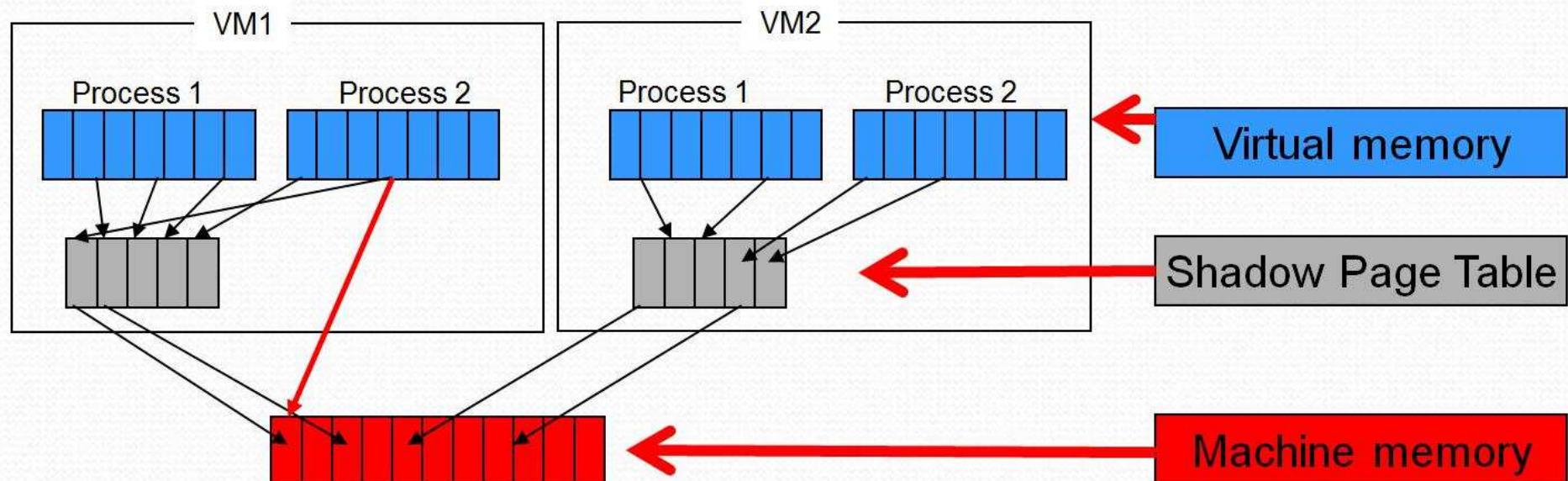
- The virtualization layer will partition the physical resource of the underlying physical server into multiple *virtual machines* with *different workloads*
- The *main issue about this virtualization layer* is that:
 - **Schedules, allocates** physical resource, and
 - Make **each virtual machine** think that it totally owns the whole underlying hardware's physical resource
 - E.g., processor, disks, RAMs, etc.

Virtualization Security Requirements

- **Scenario:** A client uses the service of a cloud computing company to **build a remote VM**, need:
 - A secure network interface
 - **T**ransport **I**ayer **S**ecurity (TLS)
 - A secure secondary storage
 - **N**etwork **F**ile **S**ystem (NFS)
 - A secure run-time environment
 - ✓ All **cryptographic algorithms** and **security protocols** reside in the run-time environment
 - ✓ Build, save, restore, destroy

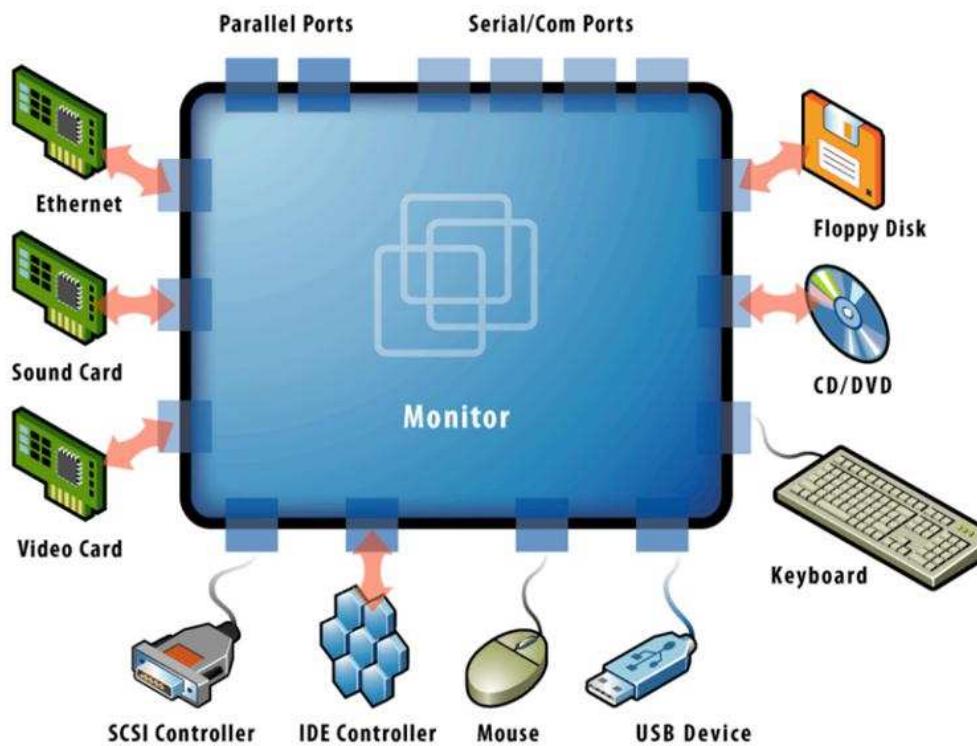
Memory Virtualization

- To run multiple VMs on *a single system*, another level of *memory virtualization* is required.
- The VMM is responsible for *mapping guest Virtual memory to the actual host machine memory, and it uses shadow page tables to accelerate the mappings.*



Device and I/O Virtualization

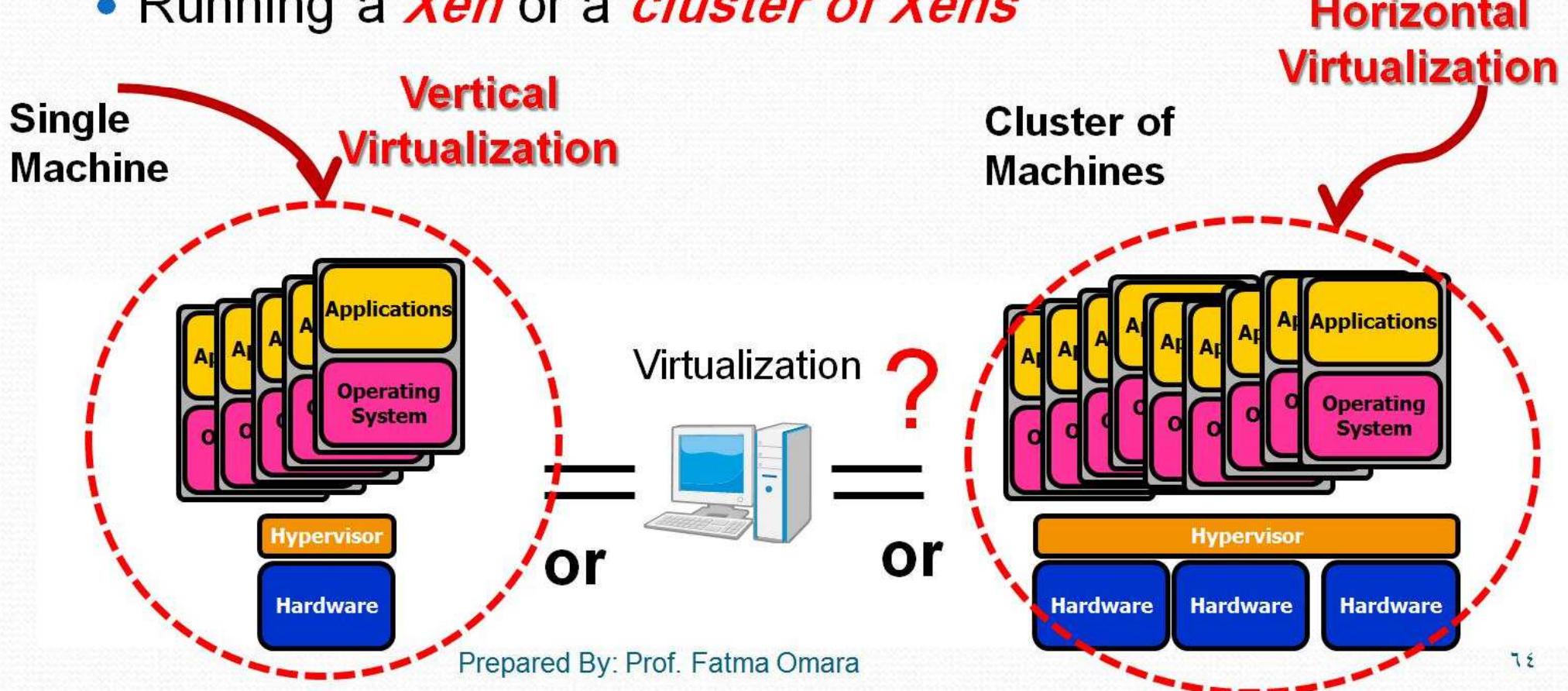
- VMM supports all device I/O drivers
- Physically/virtually existed



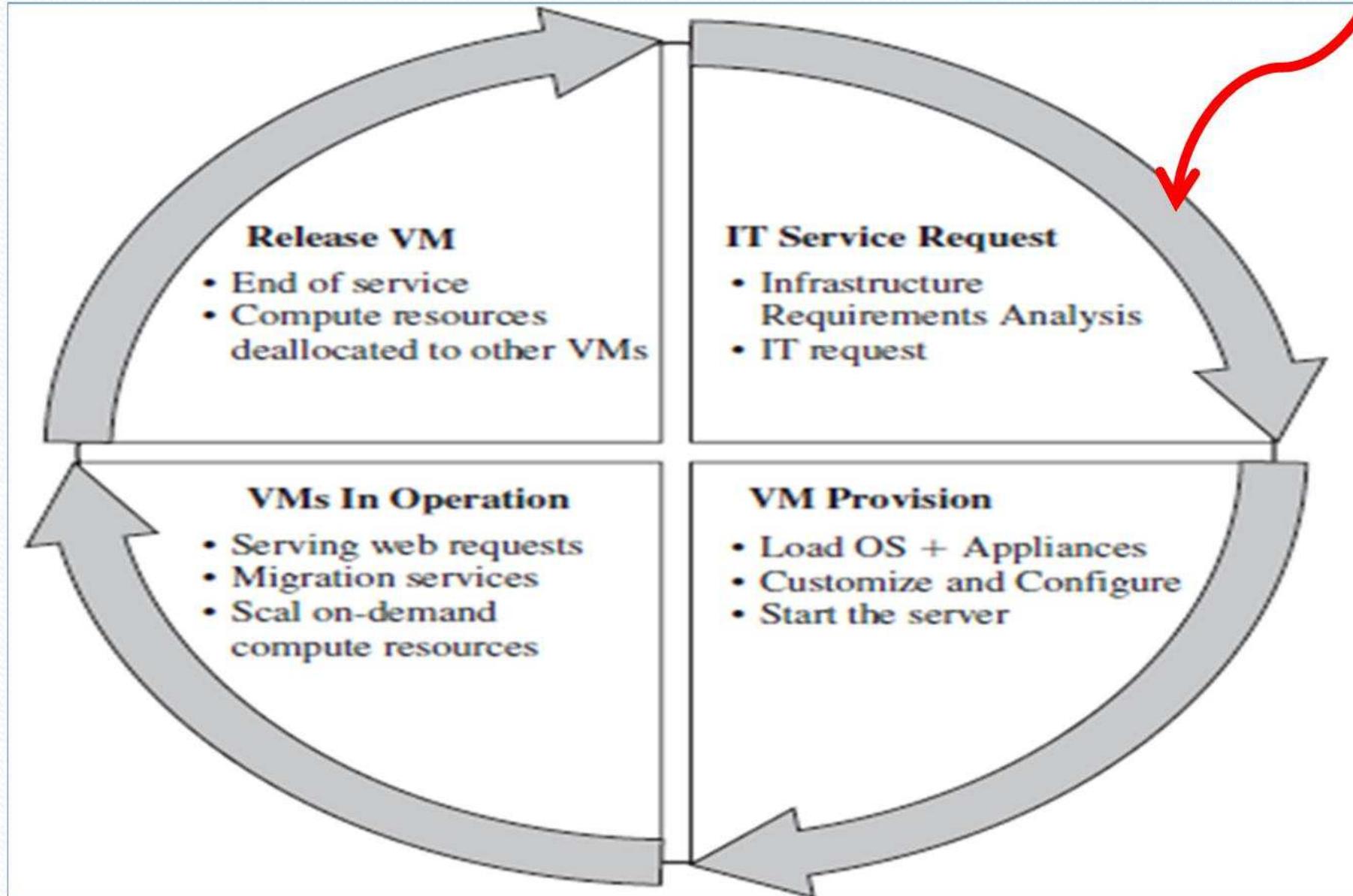
Source: VMware white paper, "Understanding Full Virtualization, Paravirtualization, and Hardware Assist"

Issues in Virtualization for Cloud-Computing

- Virtualization implemented on
 - A single machine (with multi-core CPUs)
 - a cluster of machines (with multi-core CPUs)
- The state-of-the-art
 - Running a *Xen* or a *cluster of Xens*



Virtual Machines Provisioning Life Cycle



Provisioning in Public Cloud

- Resources are dynamically provisioned via publicly accessible Web applications/Web services (SOAP or RESTful interfaces) from an off-site third-party provider, who **shares resources** and **bills** on a fine-grained utility computing basis

Public Cloud Providers:

- Amazon EC2
- GoGrid
- RackSpace
- AppNexus
- FlexiScal
- ...

Provisioning in Private Cloud

- Providing public cloud functionality on private resources, while maintaining control over an organization's data and resources to meet security and governance's requirements in an organization.
- Private cloud exhibits a highly virtualized cloud data center located inside your organization's firewall.
- Private Cloud Frameworks:
 - Eucalyptus
 - OpenNebula