Amrita Vishwa Vidyapeetham

Amritapuri Campus

22AIE305: CLOUD COMPUTING

WHAT IS VIRTUALISATION?

Virtual is the antonym of “real”, but in Cloud computing it is opposite of “Physical”

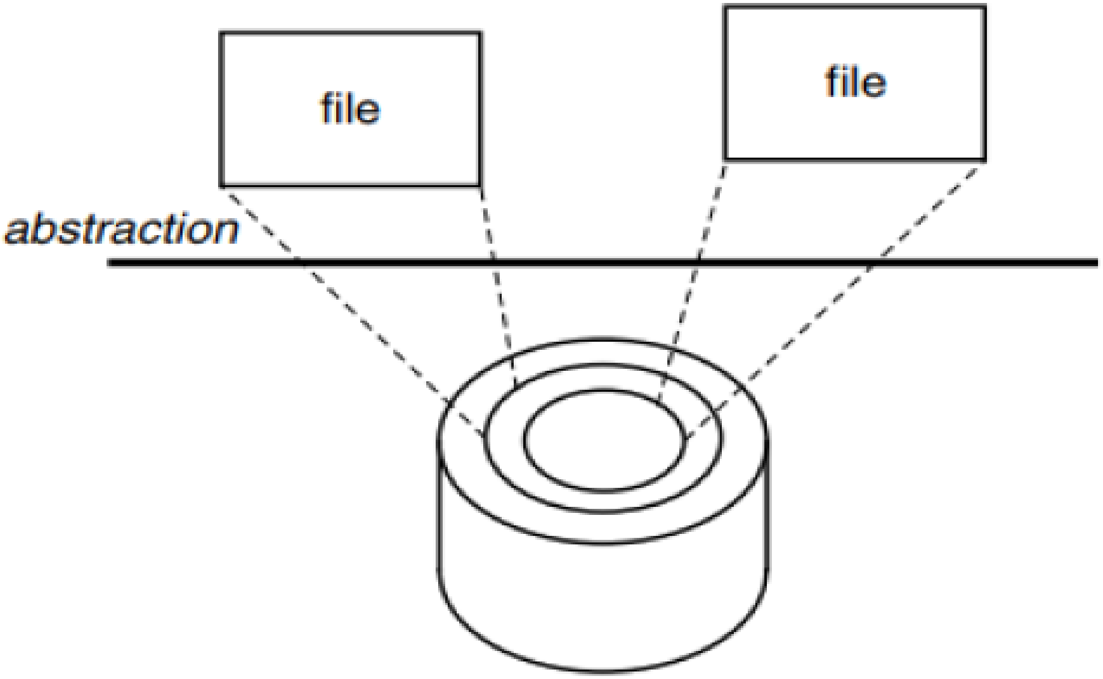
Examples are virtual memory, virtual storage, virtual disk, virtual machines, virtual node, virtual networks, etc.

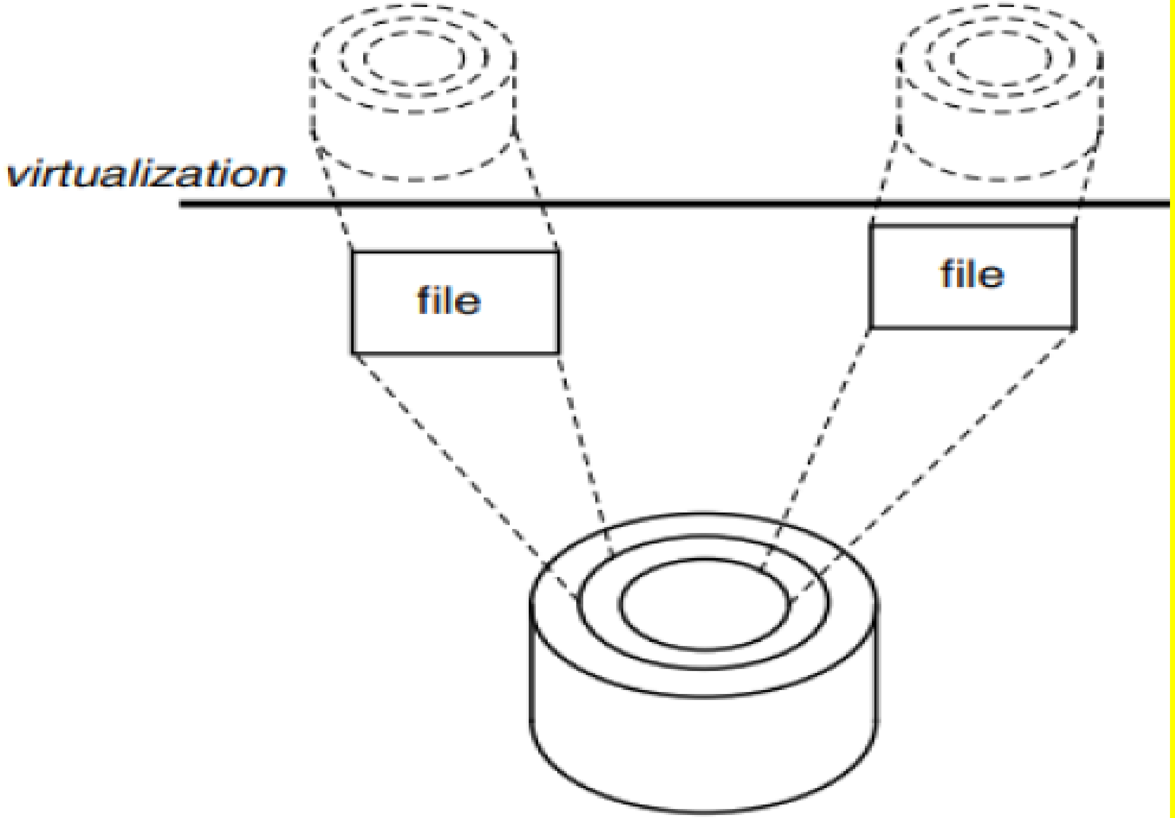
Virtualization is a fundamental technique to deliver Cloud services (XaaS) to millions of clients simultaneously.

It provides a platform to optimize complex management of Containers, client requests, accounting of services, etc.

What does virtualization mean?

An Example

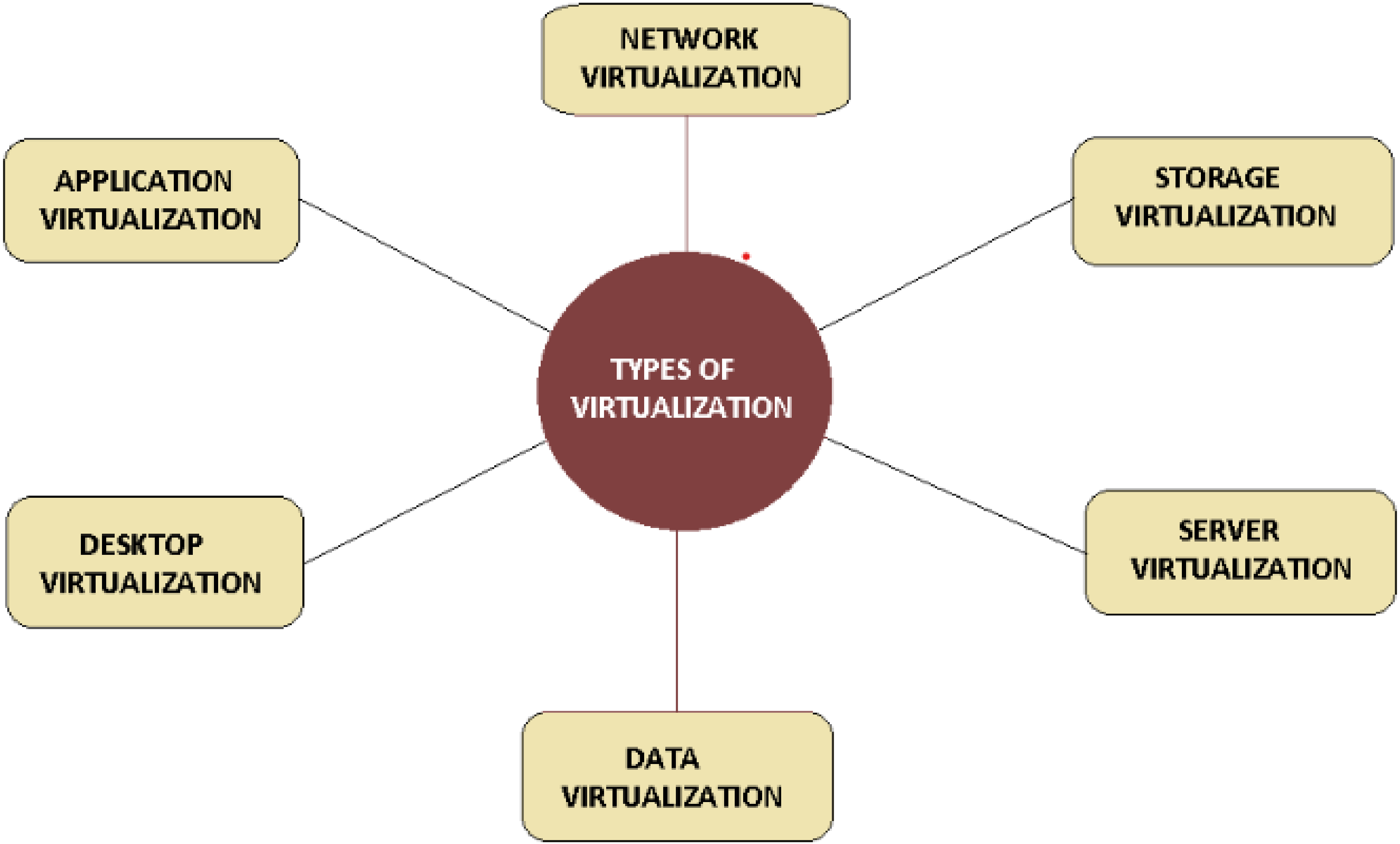




1. Files Are an Abstraction of a Disk. A level of abstraction provides a simplified interface to underlying resources.
2. Implementing Virtual Disks. Virtualization provides a different interface and/or resources at the same level of abstraction.

Why we need Virtualization?

- The need for Isolation
- Having more than one Operating System
- Low Utilization
- High costs of Scaling
- Running Legacy Applications
- Software developers need Multi-platform Systems
- The need for System Backup and Snapshots



TYPES OF VIRTUALIZATION

NETWORK VIRTUALIZATION

STORAGE VIRTUALIZATION

SERVER VIRTUALIZATION

DATA VIRTUALIZATION

DESKTOP VIRTUALIZATION

APPLICATION VIRTUALIZATION

- ✓ Desktop virtualisation
- ✓ Network virtualisation
- ✓ Storage virtualisation
- ✓ Data virtualisation
- ✓ Application virtualisation
- ✓ Data center virtualisation
- ✓ CPU virtualisation
- ✓ GPU virtualisation
- ✓ OS virtualisation
- ✓ Cloud virtualisation

VIRTUALIZATION SOFTWARE

Several virtualization software packages are available:

VMware ESX and VMware Player

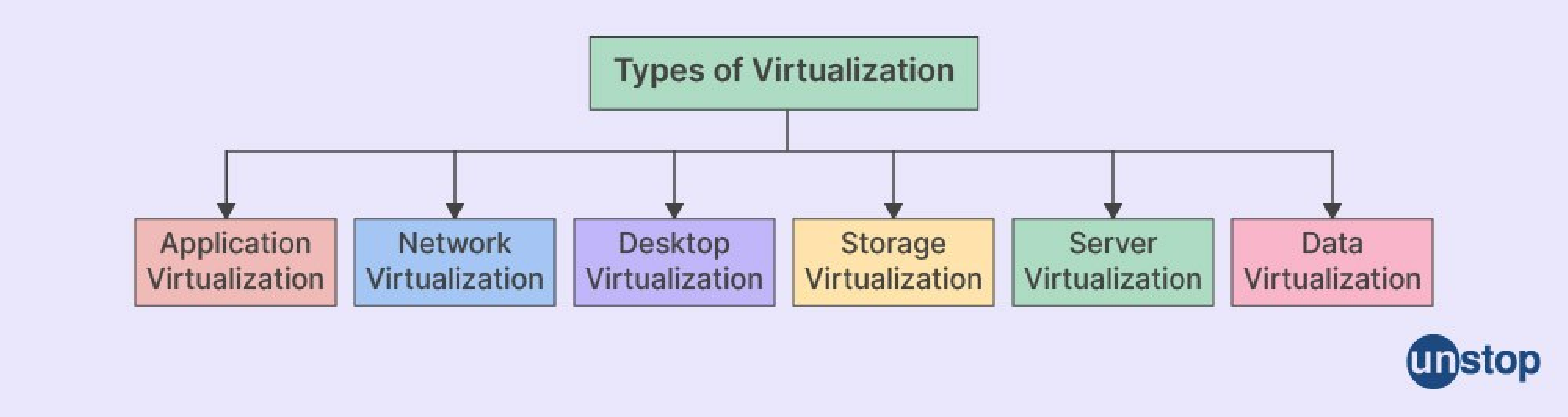
Microsoft Hyper-V

VirtualBox

Citrix XenServer

QEMU

Docker



ELEMENTS OF VIRTUALISATION

Virtual Machine: A virtual machine can be defined as a type of virtual machine that runs under the hypervisor of the computer.

Hypervisor: It can be defined as a process running on real hardware. The virtual counterpart of the OS is the underlying part that executes or emulates a virtual process.

Container: They can be defined as a lightweight virtual machine that is a process of the same OS (e.g. hypervisor).

Virtualization software: This type of software helps to virtualise the Cloud.

Virtual Network: It is defined as a separate network and resides on the server. This type of network can be expanded to many servers.

ADVANTAGES OF VIRTUALISATION

Safe and Seamless Data Transfer

Under virtualization, files and data can be transferred over long distances swiftly and easily. Finding data doesn't need relentless searching through hard drives; rather, it can be done with ease in the virtual cloud area.

Isolation

Virtualization's ability to offer isolation is crucial for maintaining a reliable and safe cloud environment. Every VM runs separately, isolating itself from other VMs and the fundamental hardware. As a result, if one VM malfunctions, it has no effect on the other VMs or the procedure as a whole.

LOW COST, CLONING CAPABILITY

Lower Costs

Typically, businesses set up servers to run programs that only use a small portion of their resources. These servers are never used to their maximum capacity. Additionally, these servers are entirely inactive when their applications aren't in use.

You can accurately allocate each VM in a virtualized setting with the processing power it requires to perform its tasks. After that, other VMs and their apps can use the spare resources.

Cloning a VM to create testing environments is easy. Services can be tested, and glitches can be fixed without impacting the live product. Developers can experiment with an array of flawlessly evolved virtual machines.

ADVANTAGES OF VIRTUALISATION CONT'D

Cloud Virtualizations also manage the workload by transforming traditional computing and make it more scalable, economical and efficient.

One of the important features of virtualization is that it allows sharing of applications to multiple customers and companies.

This will help the user by providing multiple machines at the same time it also allows sharing a single physical instance of resource or an application to multiple users.

Advantages of Virtualisation cont’d

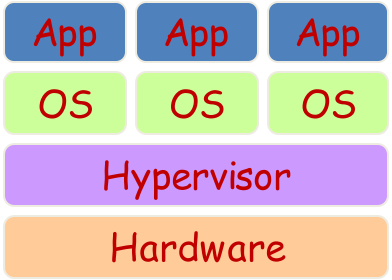
- OS and application crashes can cause downtime and disrupt user productivity.
- Admins can run multiple redundant virtual machines alongside each other and failover between them when problems arise. Running multiple redundant physical servers is more expensive.

Disadvantages of Virtualisation

- **Compatibility Issues:** Not all hardware or software can be virtualized, and some applications may not perform well in a virtual environment. This can limit the benefits of virtualization and require additional resources to manage.
- **Performance Overhead:** Virtualization introduces a performance overhead due to the need for additional layers of software and hardware abstraction. This can impact the performance of virtual machines and require additional resources to maintain performance levels.
- **Availability Issues**: Virtualization creates an availability issue since third-party providers control the ability to stay connected. If an organization cannot connect to their data for an extended period of time, they will struggle to compete in their industry.
- **High Initial Investment:** Virtualization requires a significant initial investment in hardware, software, and training. This can be a barrier for small business owners or those with limited budgets.

Virtualization

- 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

Roots of Cloud Computing

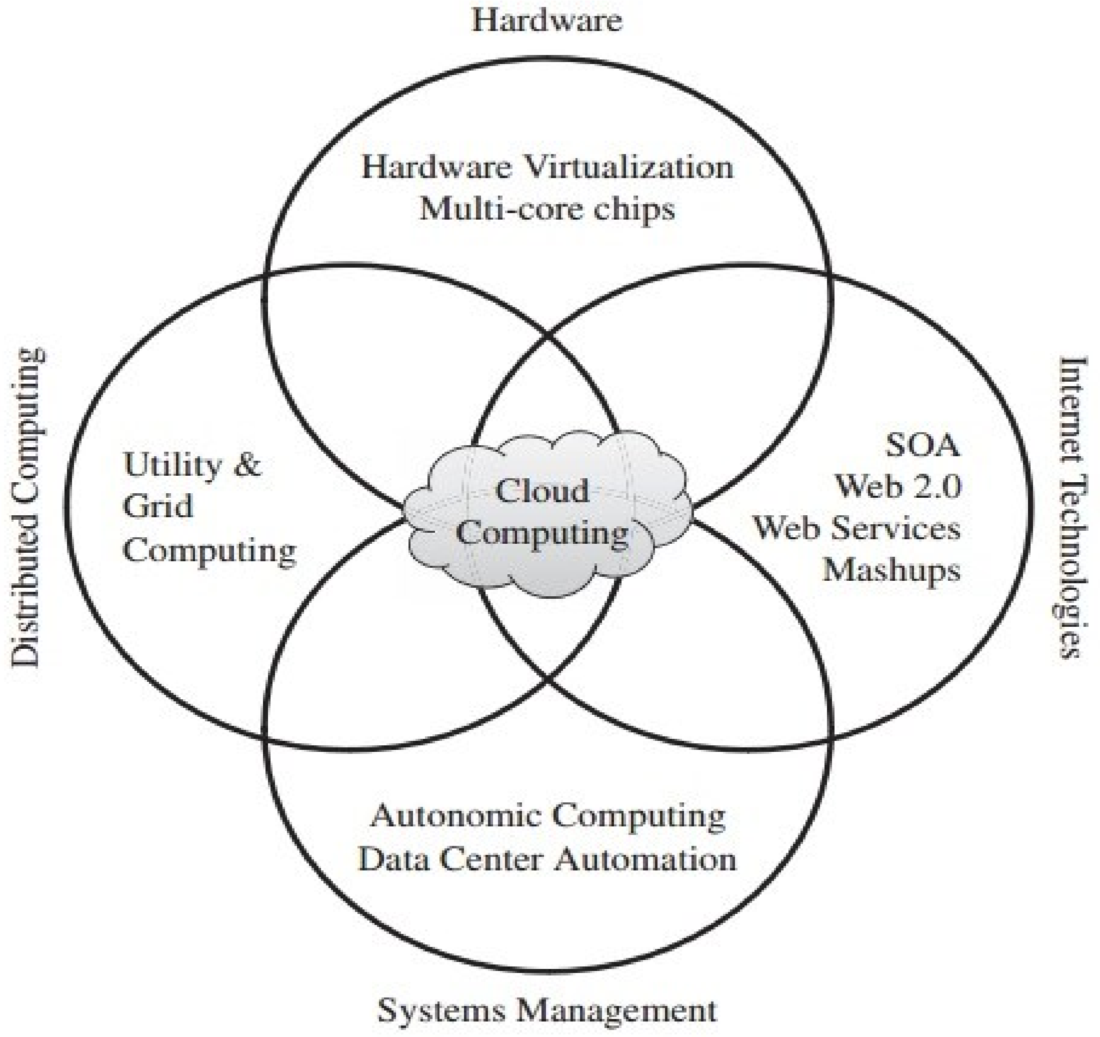
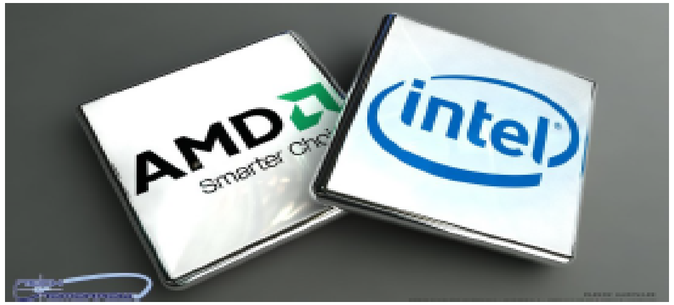
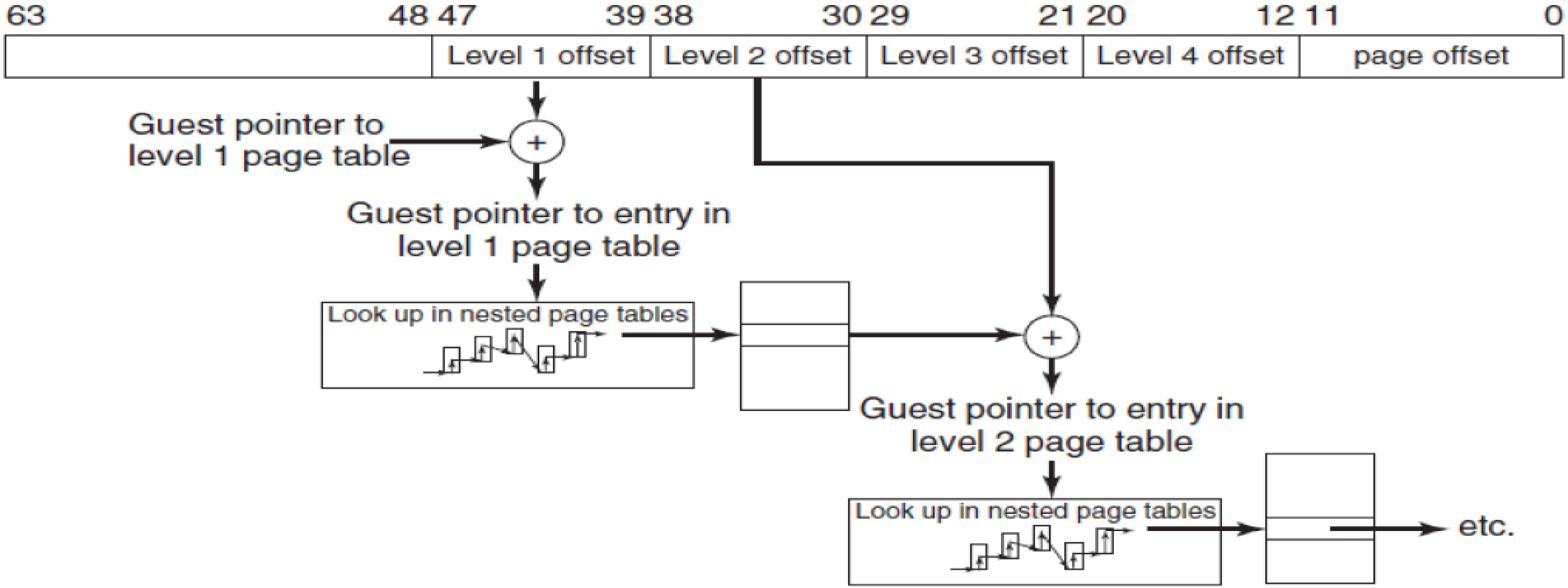


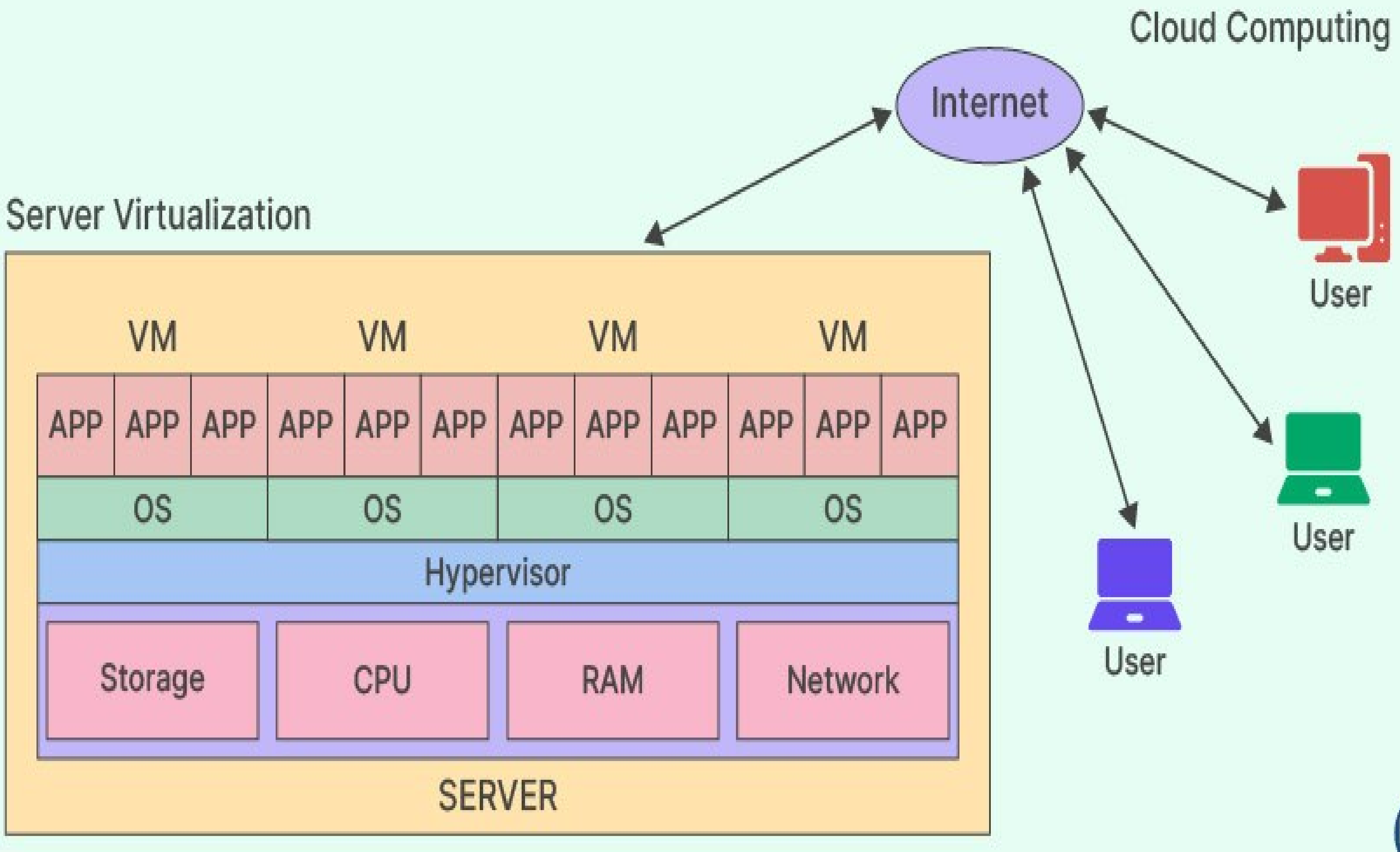
FIGURE 1.1. Convergence of various advances leading to the advent of cloud computing.

MEMORY VIRTUALIZATION



- Shadow page tables in hypervisor
- Intel's **EPT (Extended Page Tables)**


63
48 47
39 38
30 29
21 20
12 11
0
Level 1 offset
Level 2 offset
Level 3 offset
Level 4 offset
page offset
Guest pointer to level 1 page table
+
Guest pointer to entry in level 1 page table
Look up in nested page tables
+
Guest pointer to entry in level 2 page table
Look up in nested page tables
etc.



Cloud Computing

Internet

User

User

User

Server Virtualization

VM VM VM VM

APP APP APP APP APP APP APP APP APP APP APP APP

OS OS OS OS

Hypervisor

Storage CPU RAM Network

SERVER

Requirements

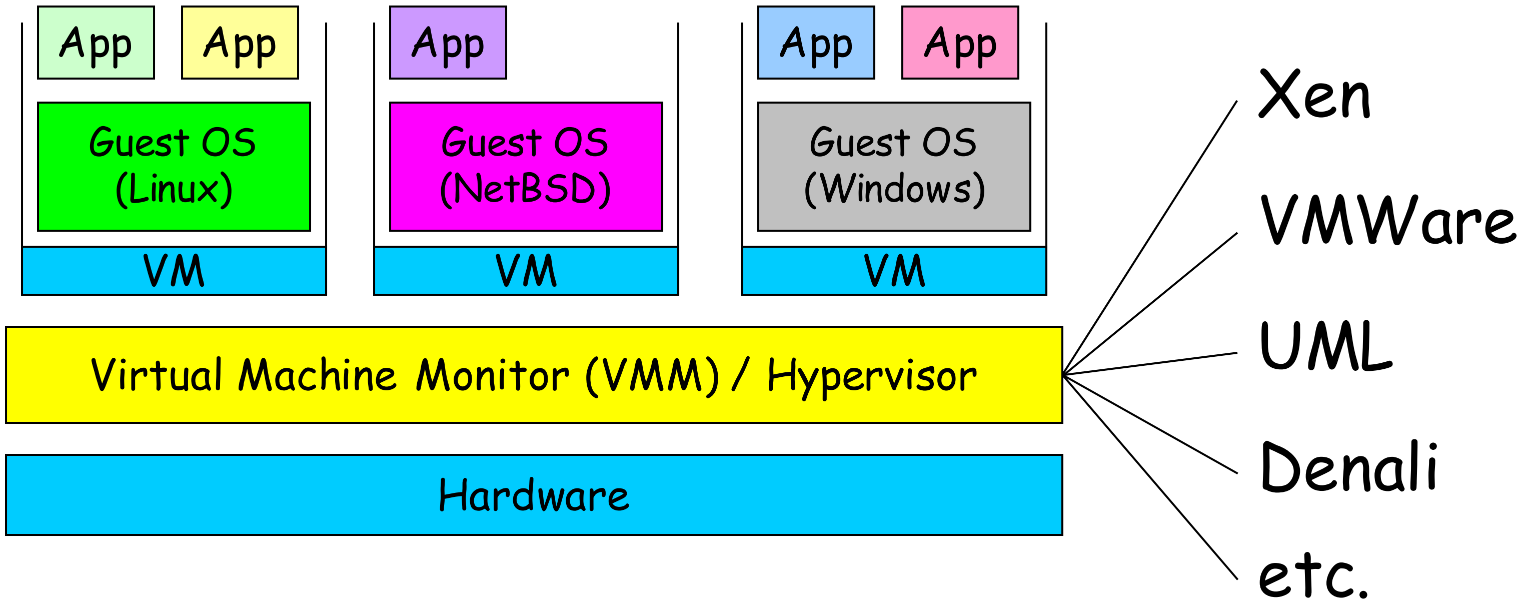
1. **Safety**: the hypervisor should have full control of the virtualized resources.

2. **Fidelity**: the behavior of a program on a virtual machine should be identical to that of the same program running on bare hardware.

3. **Efficiency**: much of the code in the virtual machine should run without intervention by the hypervisor.

Virtual Machines

- Virtual machines are virtual environments that simulate a physical computer in software form. 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!

VIRTUALISATION AND HYPERVISORS

Virtualisation is a game-changing method that involves creating virtual instances of computer components like servers, network connections, and data storage that operate identically to their actual physical counterparts using software.

Cloud runs multiple virtual machines (VMs) or containers autonomously on one physical server by severing these virtual components from the underlying hardware.

Virtualisation gives an illusion to the clients that they are working with real physical hardware.

The fundamental component of virtualization is utilizing hypervisors or virtual machine monitors (VMMs). These essential software elements act as links between virtual machines and real hardware.

CHARACTERISTICS OF VIRTUALISATION

Resource Sharing: Through virtualization, several virtual machines may share a single machine's resources, including its CPU, memory, storage, and network bandwidth. Through better hardware utilization, fewer physical servers are required.

Flexibility: Without having to buy and set up extra physical hardware, virtualization enables IT managers to rapidly and easily construct, alter, or destroy virtual computers as needed. In maintaining and growing the IT infrastructures, this offers flexibility.

Hardware Independence: Virtual machines are hardware-independent, meaning they can run on different physical machines without requiring modifications. This allows for easy migration and portability of virtual machines.

Isolation: Each virtual machine operates in its own isolated environment, separate from other virtual machines and the host operating system. This ensures that failures or issues in one virtual machine do not affect others.

Emulation: Virtualization can emulate different hardware environments, allowing virtual machines to run different operating systems and applications. This enables compatibility with legacy systems and software.

HYPERVISORS

Hypervisor is the software layer that coordinates VMs.

It serves as an interface between the VM and the underlying physical hardware, ensuring that each has access to the physical resources it needs to execute.

It also ensures that the VMs don’t interfere with each other by impinging on each other’s memory space or compute cycles.

HYPERVISOR

A hypervisor that is responsible for managing physical computing resources such as processing power, memory and storage, and allocates them to the virtual machines.

Hypervisor can be considered as the software that can create and run multiple virtual machines.

The hypervisor achieves this by creating a layer of abstraction between the physical computer infrastructure and the virtual machines. This in turn enables multiple virtual machines to run on the physical computer without any interference from each other.

There are two popular types of hypervisors called Type-1 and Type-2.

A hypervisor is an OS instance, or a software package, that creates and manages virtual machines. The hypervisor typically runs on real hardware and allows multiple virtual machines to run on the same hardware. Virtual machines are also called guests.

Modern hypervisors in the PC world include **VMware ESX, VMware Player, Xen, Citrix XenServer, Microsoft Hyper-V, VirtualBox, and QEMU**.

VMWare Player, VirtualBox, and QEMU are application-layer hypervisors (essentially like normal programs), though they rely on some specialized OS support for virtualization. Linux KVM and FreeBSD's bhyve are kernel modules that effectively convert a normal OS into a hypervisor. In fact, it is often possible to log into a hypervisor (e.g., using SSH keys and use it like a normal operating system.

TYPE 1 AND TYPE 2 HYPERVISORS CONT'D

Type 1 hypervisors: Sometimes called bare metal hypervisors, run directly on top of the host system hardware. The bare metal hypervisor provides a high level of control and management. Their direct access to system hardware provides better performance, scalability and stability. Examples include: Microsoft Hyper-V, Citrix XenServer, VMware ESXi.

Type 2 hypervisors: (also called management hypervisors) are installed on the host OS rather than directly on the hardware like Type 1 hypervisors. All guest operating systems or virtual machines run on top of the hypervisor. The simplicity of the host operation is known to facilitate the installation and management of the project. But adding a layer of layering can limit functionality and introduce a security vulnerability. Example: VMware Workstation Pro, VMware Fusion, Oracle VirtualBox, Oracle Solaris Zones, Oracle VM Server for x86.

TYPE 2 HYPERVISORS

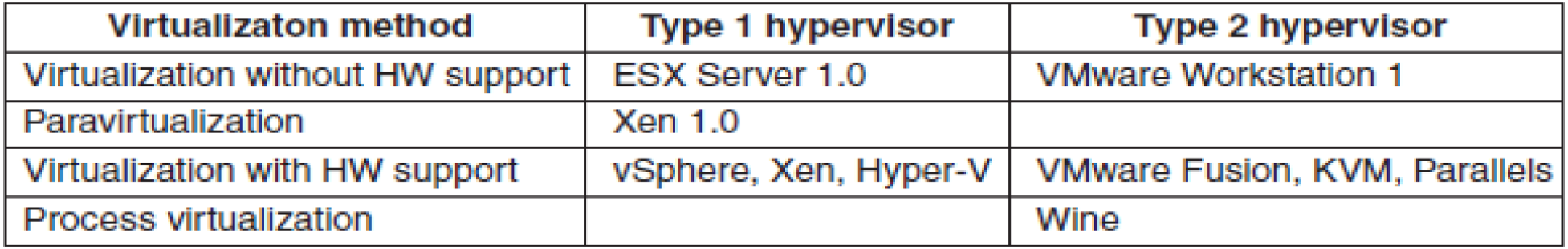
Type 2 hypervisor is also known as a hosted hypervisor, as it is installed on top of the host OS, rather than sitting directly on top of the hardware as the type 1 hypervisor does.

Each guest OS or VM runs above the hypervisor.

The convenience of a known host OS can ease system configuration and management tasks.

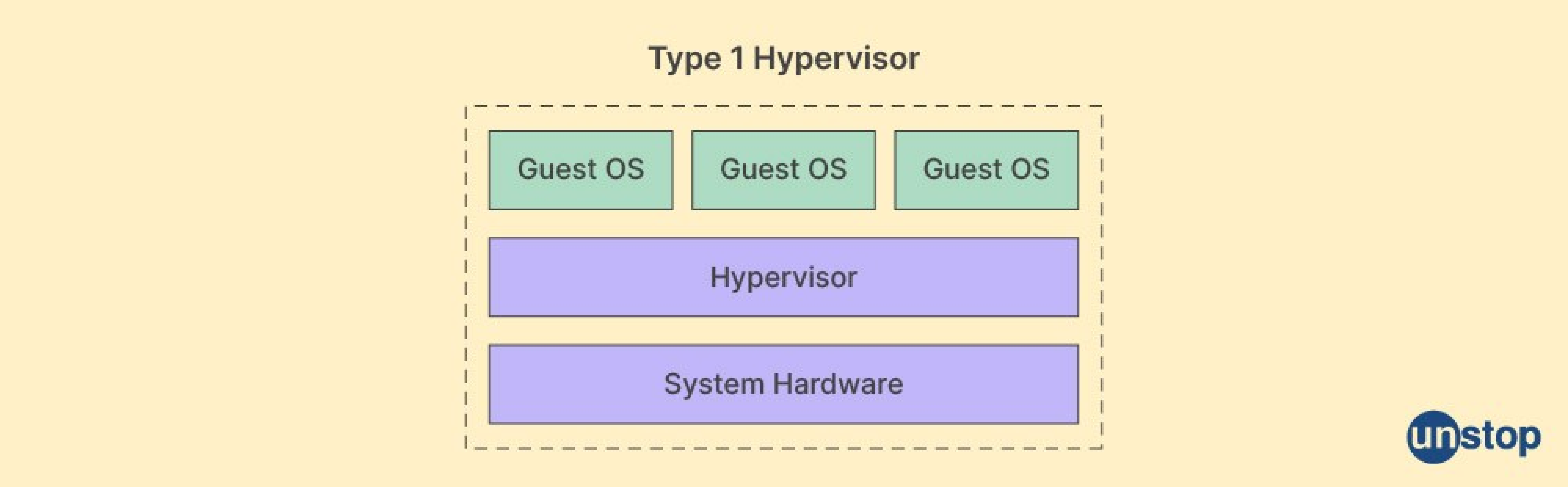
However, the addition of a host OS layer potentially can limit performance and expose possible OS security flaws.

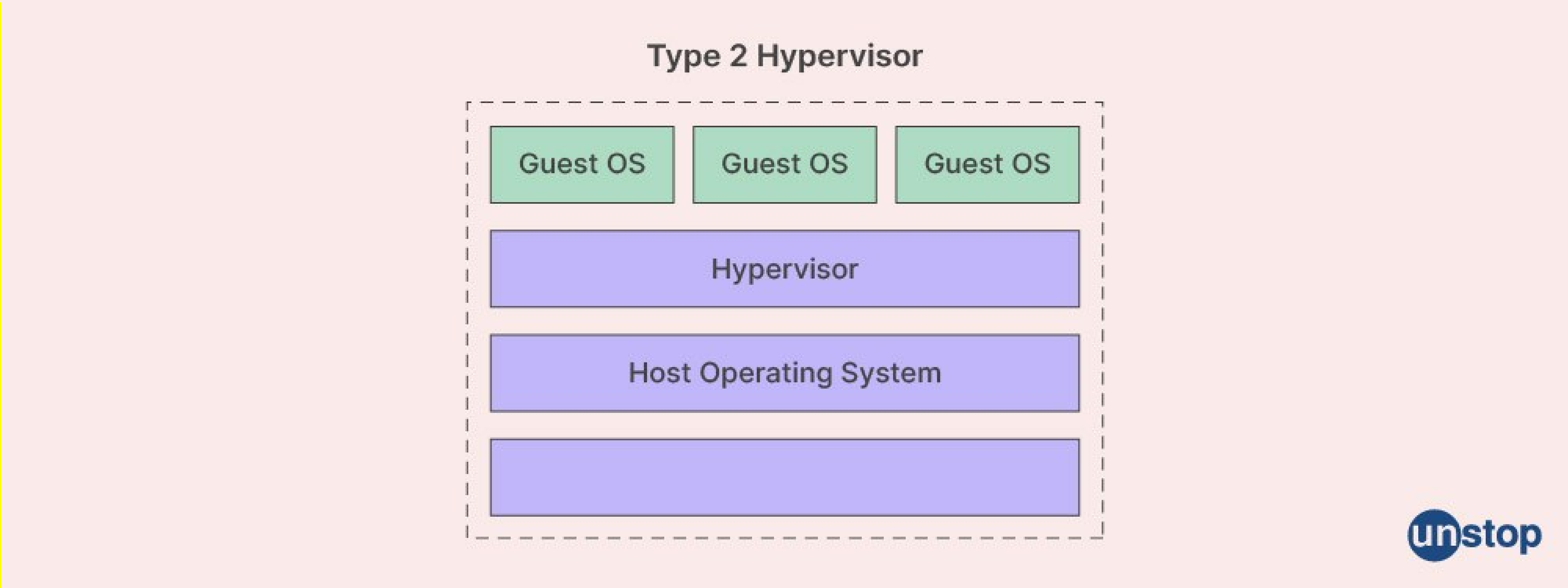
Examples of Hypervisors



Virtualizaton method	Type 1 hypervisor	Type 2 hypervisor
Virtualization without HW support	ESX Server 1.0	VMware Workstation 1
Paravirtualization	Xen 1.0	
Virtualization with HW support	vSphere, Xen, Hyper-V	VMware Fusion, KVM, Parallels
Process virtualization		Wine

Figure 7-2. Examples of hypervisors. Type 1 hypervisors run on the bare metal whereas type 2 hypervisors use the services of an existing host operating system.


Type 1 Hypervisor
Guest OS
Guest OS
Guest OS
Hypervisor
System Hardware
unstop


Type 2 Hypervisor
Guest OS
Guest OS
Guest OS
Hypervisor
Host Operating System
unstop

ISOLATION

The hypervisor isolates the OS and applications from the underlying computer hardware so the host machine can run multiple VMs as guests that share the system's physical compute resources, such as processor cycles, memory, disk space and network bandwidth.

Simply put, the hypervisor separates the available power, memory or storage resources and assigns some of these resources to each VM as needed in the virtualization architecture

OS VIRTUALIZATION

OS virtualization is a container-based kernel virtualization method.

In this architecture, an OS is adapted so it functions as multiple discrete systems, making it possible to deploy and run distributed applications without launching an entire VM for each one.

Instead, multiple isolated systems, called containers, run on a single control host and all access a single kernel.

CONTAINER

A container is a special kind of virtual machine that has no guest OS.

It is a group of processes isolated from other processes on the same host using special namespaces. In Linux, these are called cgroups or control groups.

A container engine manages all containers.

CONTAINER IS A LIGHTWEIGHT VM

Like a VM, a container is a way to create self-contained virtual packages.

Essentially, a container is a lightweight VM that is a subset of the same OS instance or the hypervisor.

However, where virtualization is a way to run multiple OS's on the hardware of a single physical server, containerization is a way to deploy multiple applications using the same OS on a single VM.

VIRTUALIZATION VS CONTAINERIZATION

The alternative to hypervisor-based virtualization in cloud computing is containerization. For example, OS virtualization is an important volume-based virtualization approach. In this architecture, the OS is set up to act as multiple separate systems, allowing distributed applications to be deployed and run without having to turn on the entire virtual machine for each system. Instead, multiple isolated machines (called volumes) run on a single host, each accessing a single key.

Virtualization done by pooling

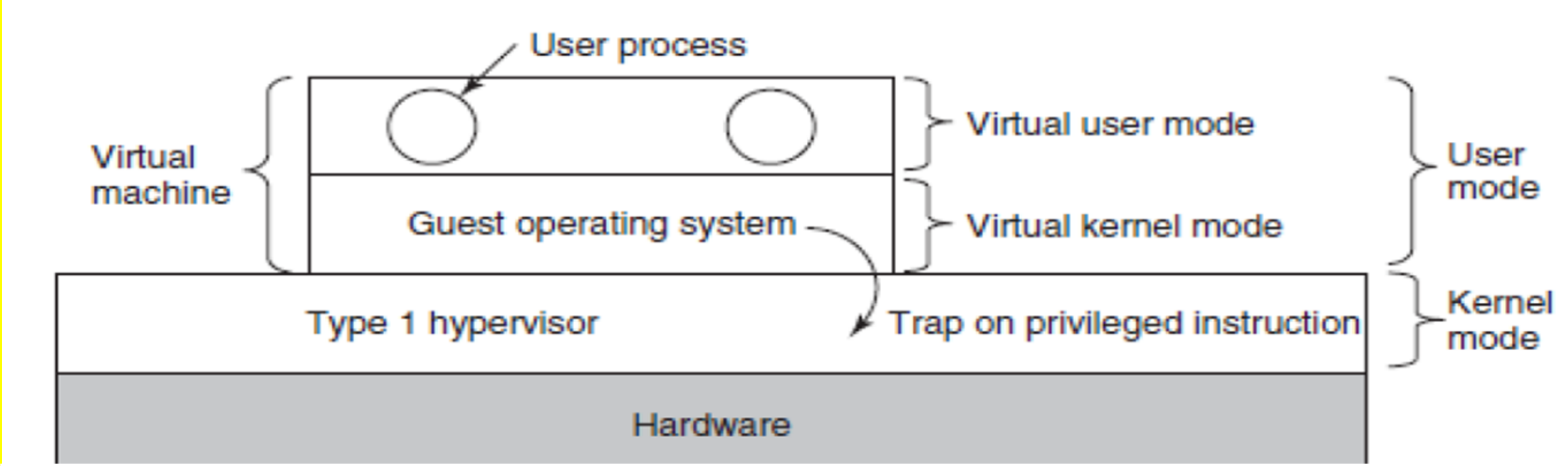
- Cloud computing virtualizes systems by pooling and sharing resources (memory, CPU (Cores), Disk-space, SSD, network, etc)
- Systems and storage can be provisioned as needed from a centralized infrastructure pool
- costs are assessed on a metered basis
- multi-tenancy is enabled
- and resources are scalable with agility.

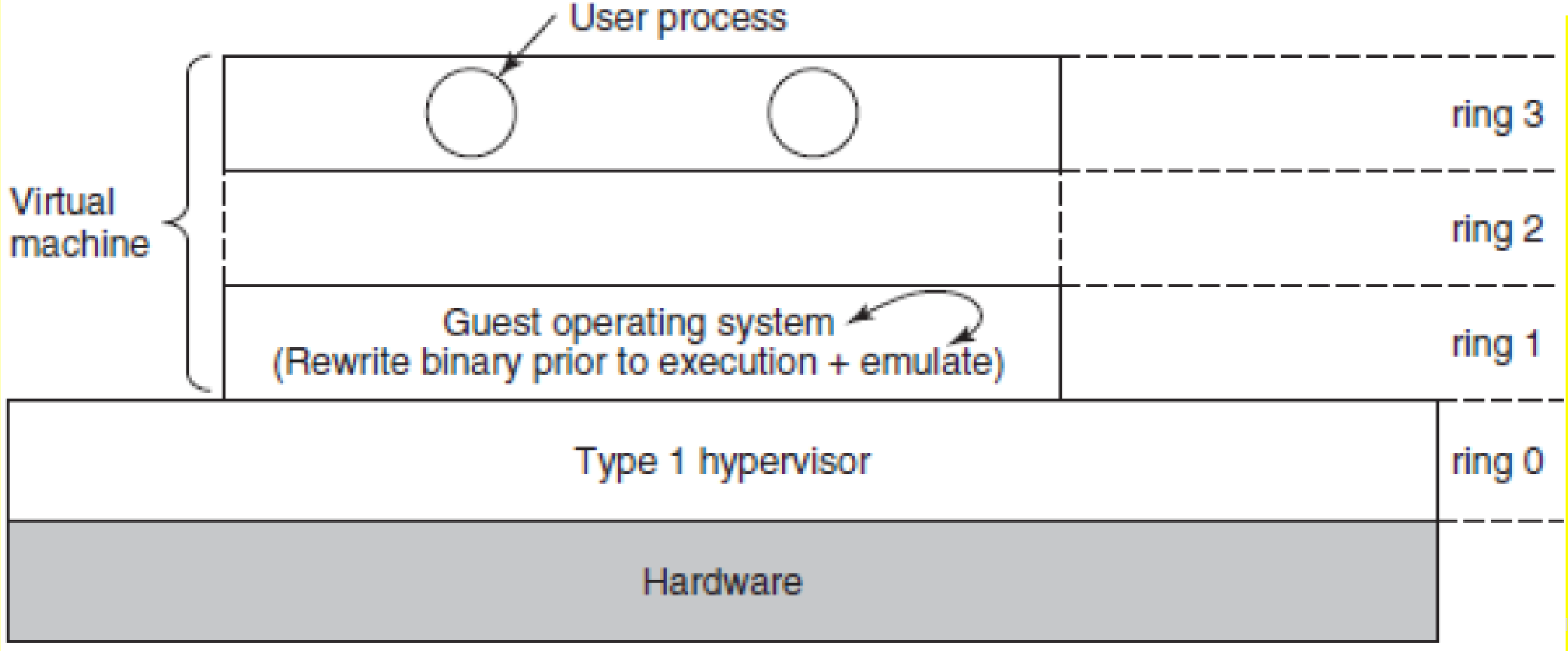
Virtual Machines

- A virtual machine is a virtual OS instance run under a hypervisor. A virtual machine may run in a cloud service or may be running on a user's desktop or some server hardware.
- Generally, a virtual machine looks like just a normal computer and OS to anyone logging into it over a network. A virtual machine will usually have virtual disks, virtual network interfaces, and often a virtual display.
- A typical server will run up to several dozen virtual machines. Some servers may run hundreds or even thousands of virtual machines.

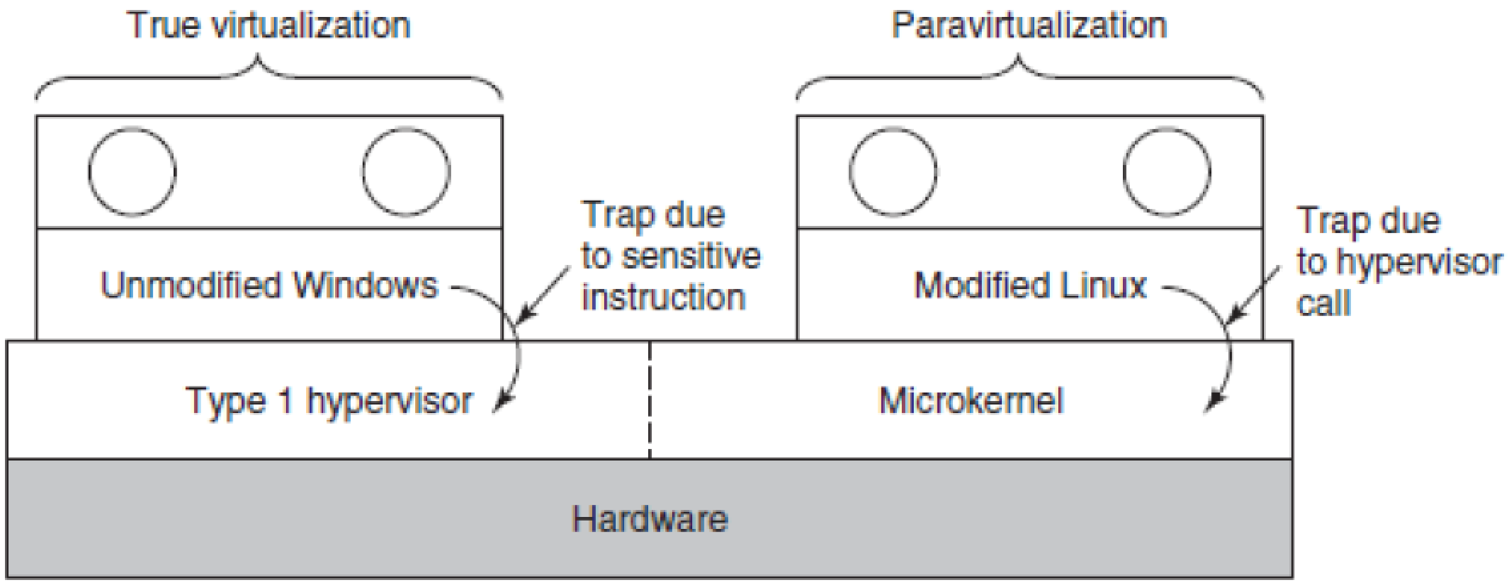
Virtual Machine Device Access

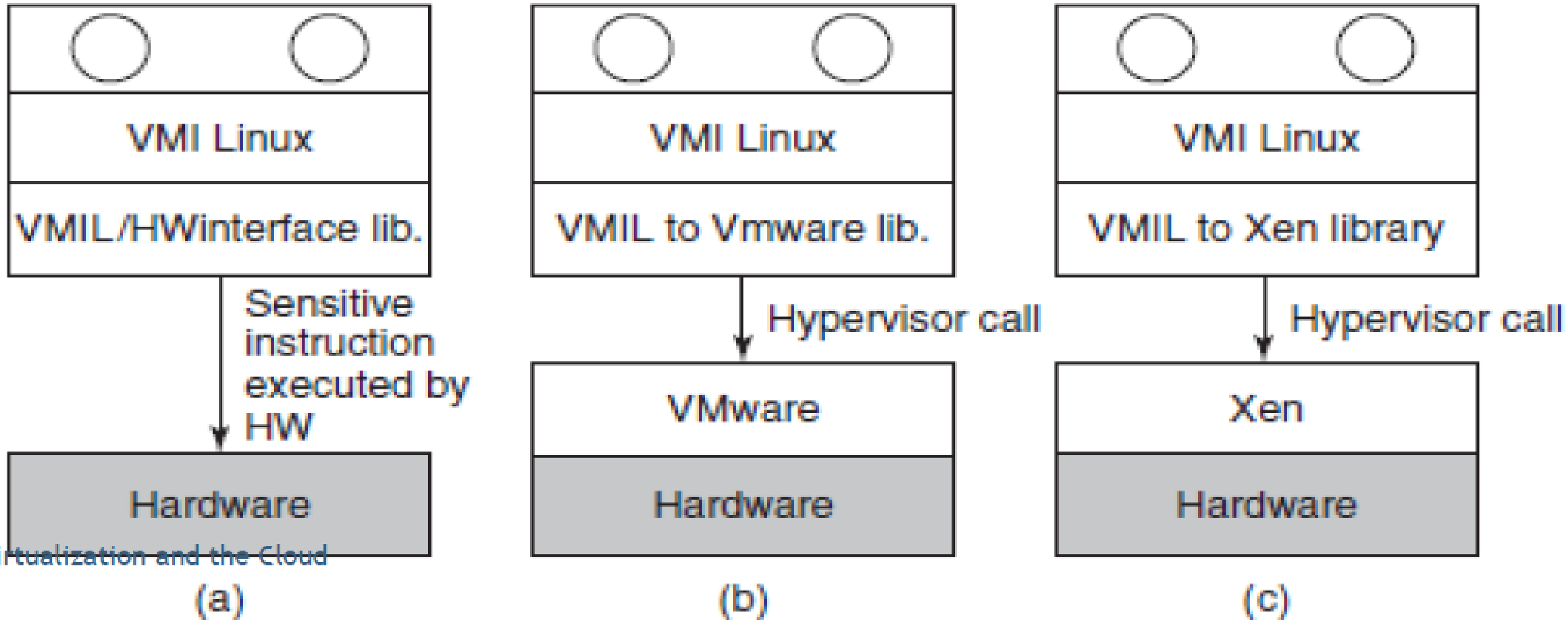
- A virtual machine uses virtual devices to interact with the host operating system. The host OS may emulate actual hardware to allow the guest OS (i.e., the OS running on the virtual machine) to access storage, network, and other resources.
- Many hypervisors and guest OS support para-virtualized device drivers. This means that the guest OS driver does not actually access emulated hardware; instead, it recognizes it is running on a particular hypervisor, and uses a hypervisor-specific communication mechanism to access the hypervisor's device drivers. The hypervisor may control and map access to its device drivers to enforce security and direct the virtual machine to its own virtual storage.


User process
Virtual machine
Virtual user mode
Virtual kernel mode
User mode
Guest operating system
Type 1 hypervisor
Trap on privileged instruction
Kernel mode
Hardware


User process
ring 3
Virtual machine
ring 2
Guest operating system
(Rewrite binary prior to execution + emulate)
ring 1
Type 1 hypervisor
ring 0
Hardware

HYPERVISORS as MICROKERNELS





HOW DOES VIRTUALIZATION REDUCE COSTS?

Instead of running a single application on a single server, single server can be used to support multiple virtual machines, each running its own operating system and applications.

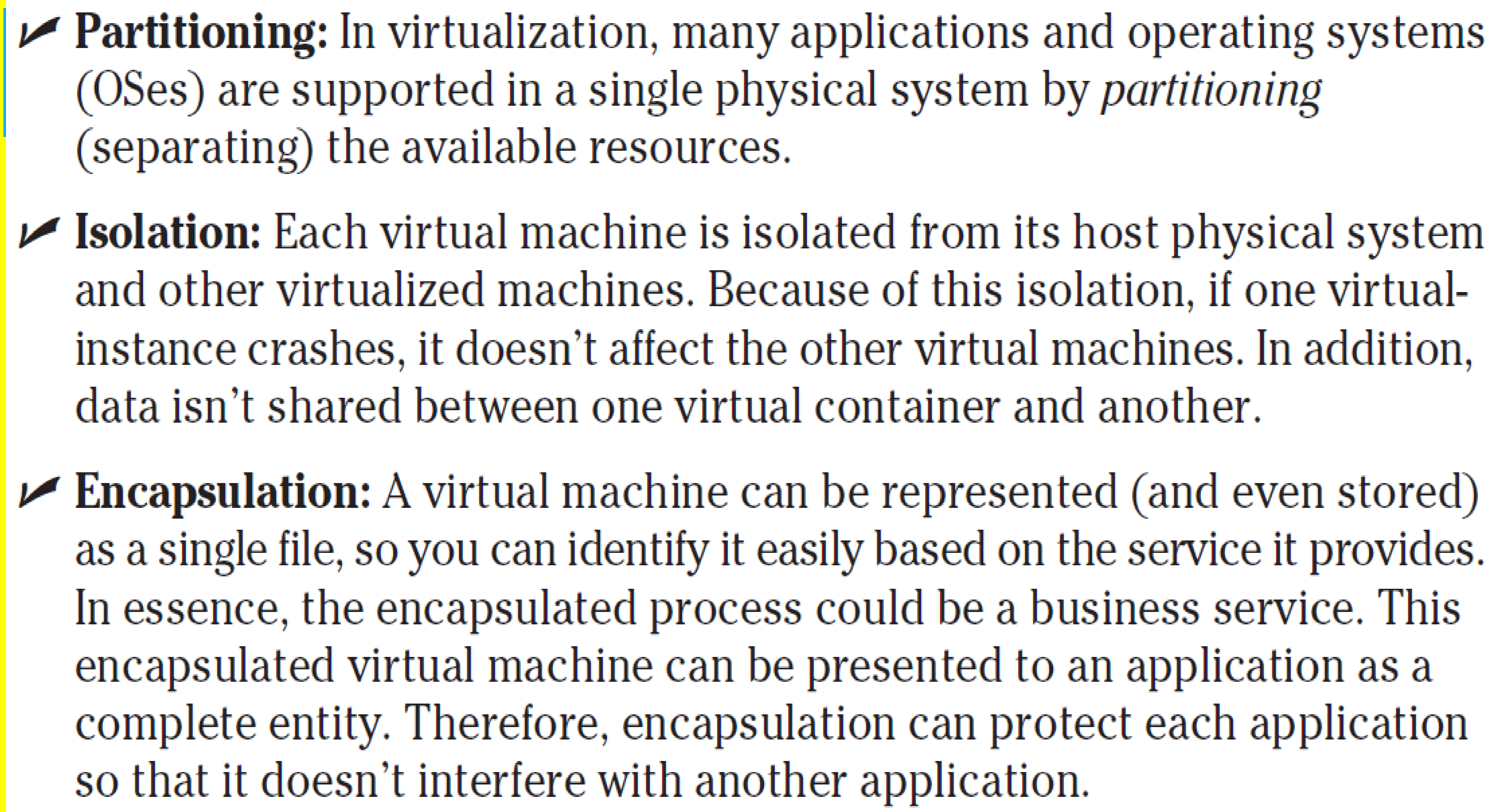
This results in significantly higher hardware recycling rates and lower maintenance costs.

IS A VIRTUALIZATION PLATFORM NEEDED TO USE THE CLOUD?

Virtualization is used to deliver cloud service models including Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

Installing a virtualization platform is easier, more cost-effective, and more efficient.

3 CHARACTERISTICS OF VIRTUALISATION



- **Partitioning:** In virtualization, many applications and operating systems (OSes) are supported in a single physical system by *partitioning* (separating) the available resources.

- **Isolation:** Each virtual machine is isolated from its host physical system and other virtualized machines. Because of this isolation, if one virtual-instance crashes, it doesn't affect the other virtual machines. In addition, data isn't shared between one virtual container and another.

- **Encapsulation:** A virtual machine can be represented (and even stored) as a single file, so you can identify it easily based on the service it provides. In essence, the encapsulated process could be a business service. This encapsulated virtual machine can be presented to an application as a complete entity. Therefore, encapsulation can protect each application so that it doesn't interfere with another application.

Hardware Virtualisation

- Hardware virtualization allows running multiple operating systems and software stacks on a single physical platform

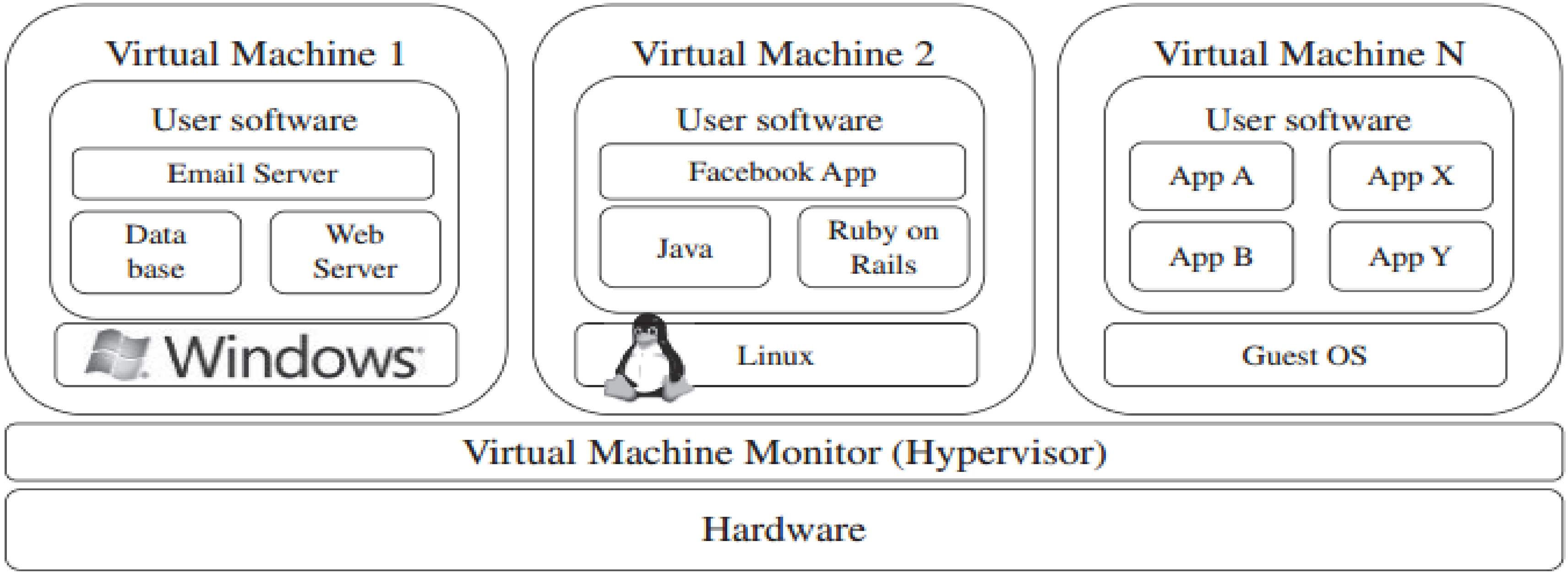
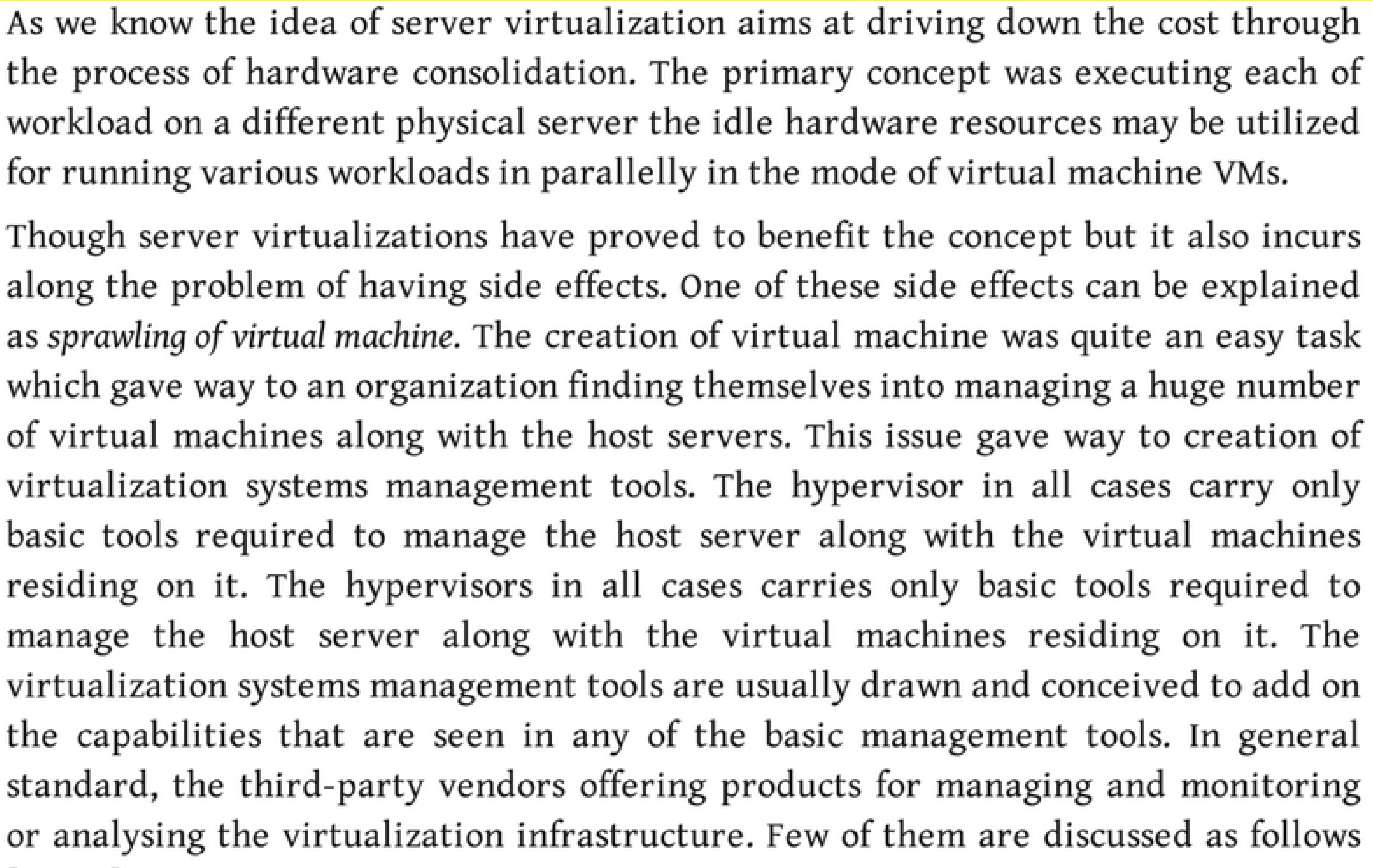


FIGURE 1.2. A hardware virtualized server hosting three virtual machines, each one running distinct operating system and user level software stack.

- 3 basic capabilities related to management of workload: isolation, Consolidation and Migration

Container vs Virtualiser

- A container is a lightweight virtual machine that is part of the same OS instance or hypervisor. But virtualization is a way to run multiple OS on a single physical server. Containerization, on the other hand, is a way of running multiple applications on the same machine in a virtual machine.
- VMs are best suited for applications that require full-scale performance. Containers are a better choice for short-term projects that need to reduce the number of servers used by multiple applications.



As we know the idea of server virtualization aims at driving down the cost through the process of hardware consolidation. The primary concept was executing each of workload on a different physical server the idle hardware resources may be utilized for running various workloads in parallelly in the mode of virtual machine VMs.

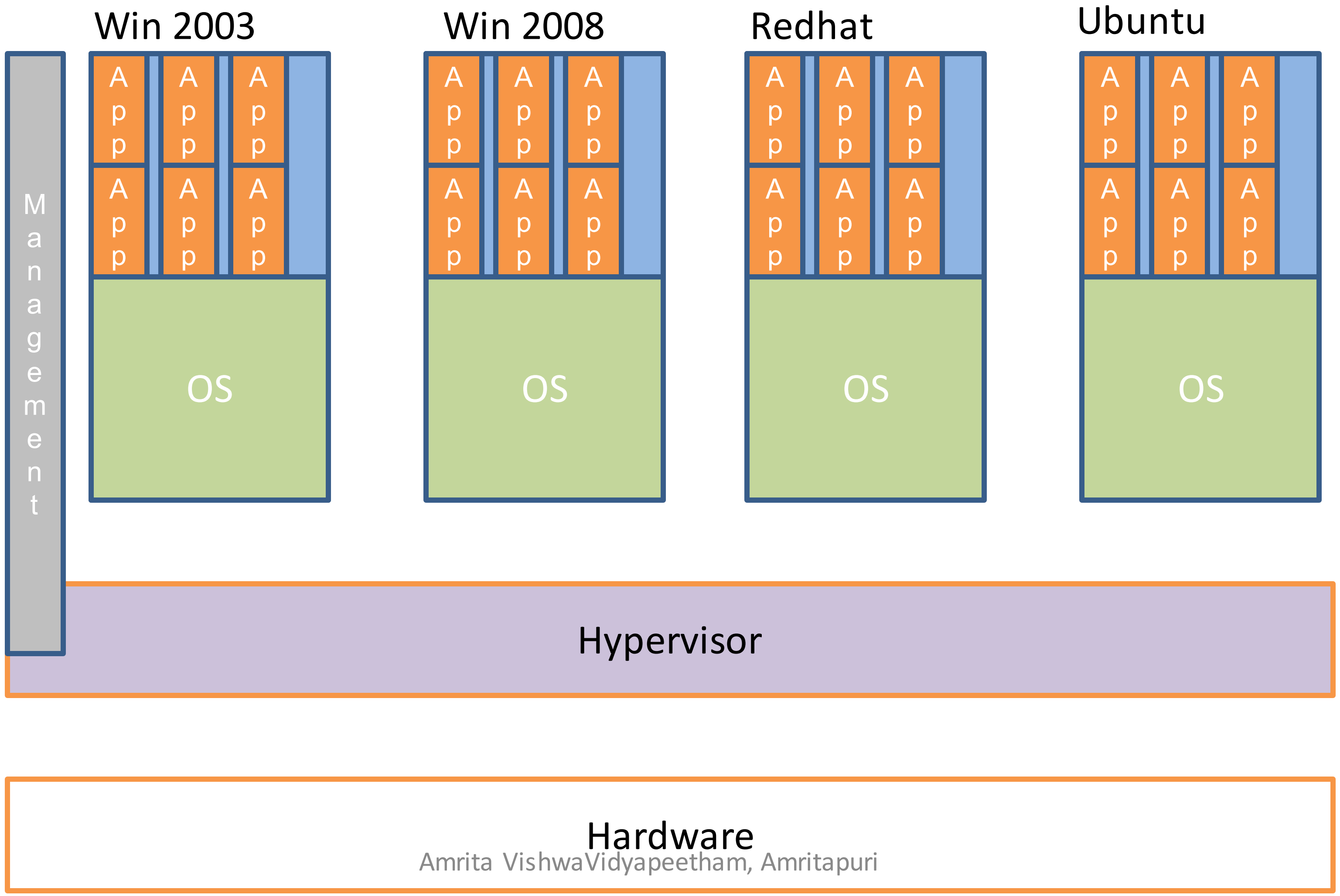
Though server virtualizations have proved to benefit the concept but it also incurs along the problem of having side effects. One of these side effects can be explained as *sprawling of virtual machine*. The creation of virtual machine was quite an easy task which gave way to an organization finding themselves into managing a huge number of virtual machines along with the host servers. This issue gave way to creation of virtualization systems management tools. The hypervisor in all cases carry only basic tools required to manage the host server along with the virtual machines residing on it. The hypervisors in all cases carries only basic tools required to manage the host server along with the virtual machines residing on it. The virtualization systems management tools are usually drawn and conceived to add on the capabilities that are seen in any of the basic management tools. In general standard, the third-party vendors offering products for managing and monitoring or analysing the virtualization infrastructure. Few of them are discussed as follows

- Server virtualization enables you to run several applications—each on its own VM with its own OS—on a single physical computer
- kernel-based virtual machine (KVM) is a Linux virtualization subsystem
- Is has been part of the mainline Linux kernel since version 2.6.20, thus being natively supported by several distributions.
- In addition, activities such as memory management and scheduling are carried out by existing kernel
- KVM leverages hardware-assisted virtualization, which improves performance and allows it to support unmodified guest operating systems

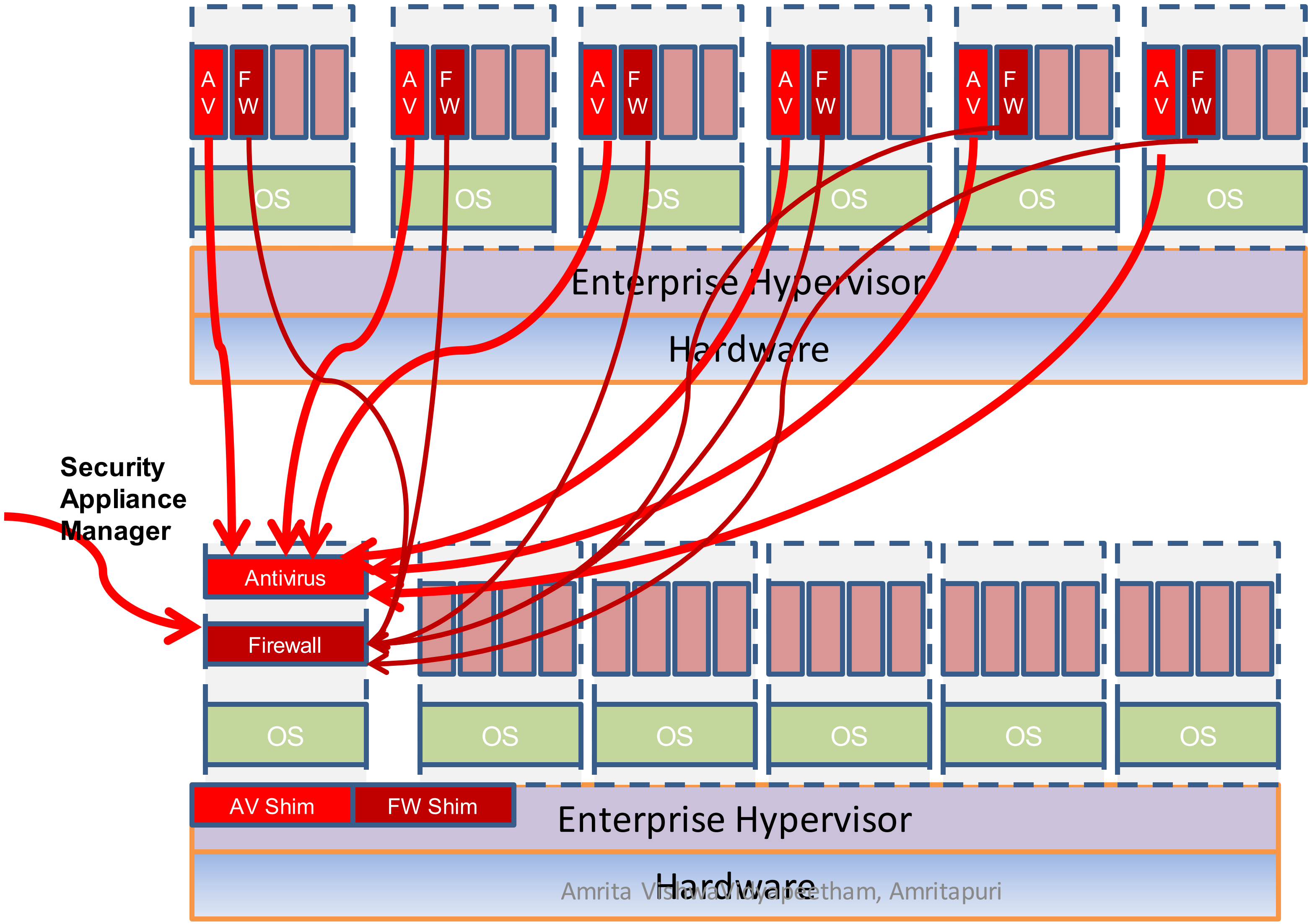
Virtual Appliance and OVF(open virtual format)

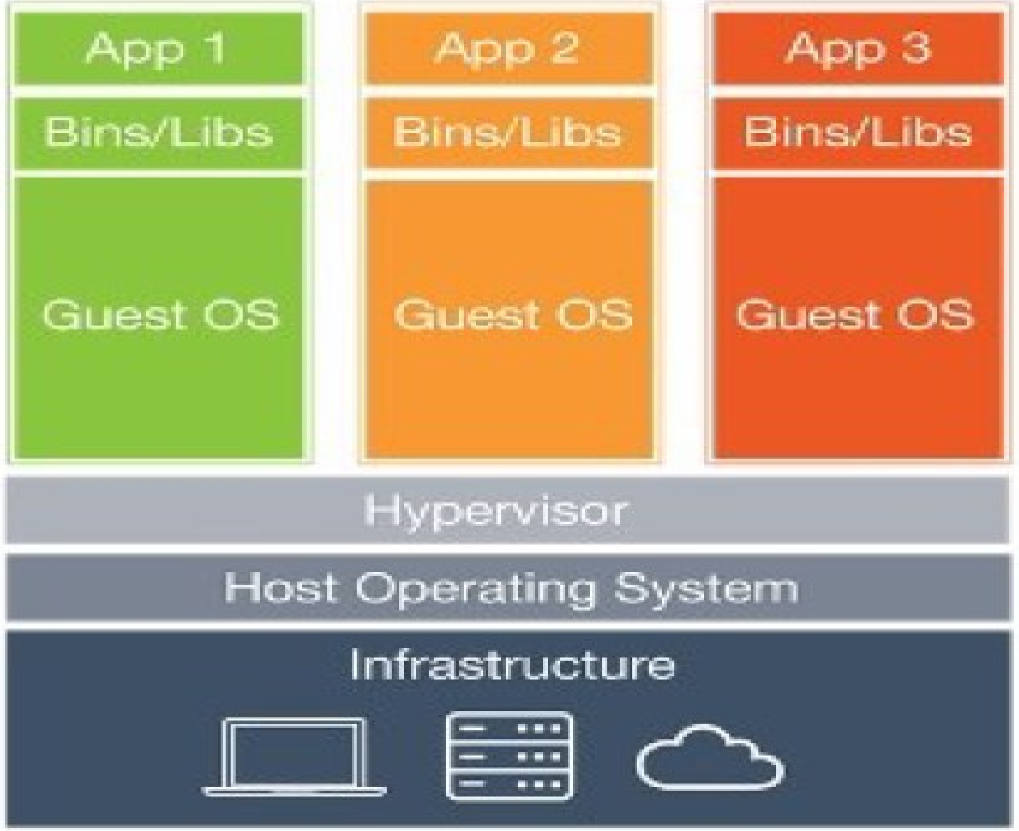
- An application combined with the environment needed to run it (operating system, libraries, compilers, databases, application containers, and so forth) is referred to as a “virtual appliance.”
- In a multitude of hypervisors, where each one supports a different VM image format and the formats are incompatible with one another, a great deal of interoperability issues arises.
- For instance, Amazon has its Amazon machine image (AMI) format, made popular on the Amazon EC2 public cloud. Other formats are used by Citrix XenServer, several Linux distributions that ship with KVM, Microsoft Hyper-V, and VMware ESX

Server Virtualization

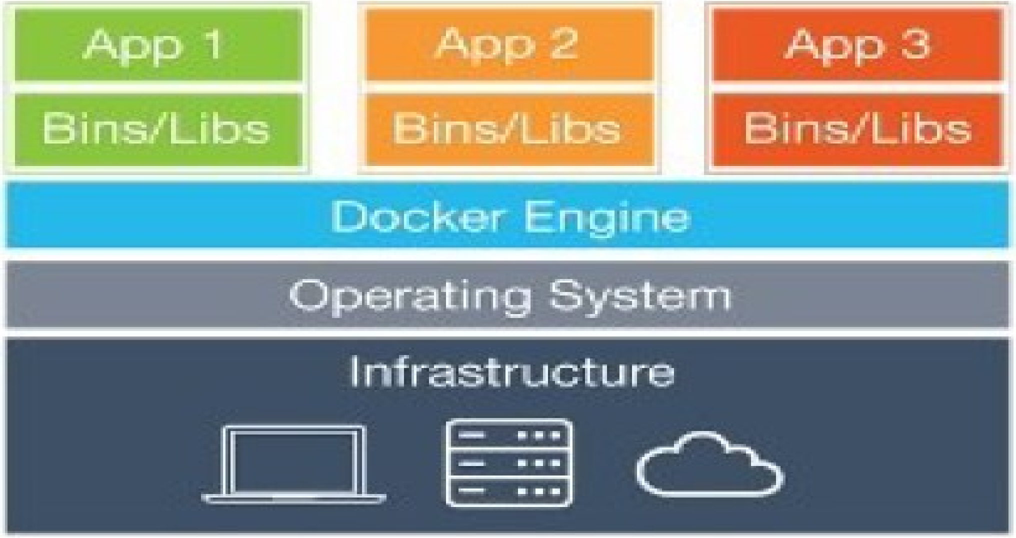


Early Trends – Consolidate common security functions to security appliance





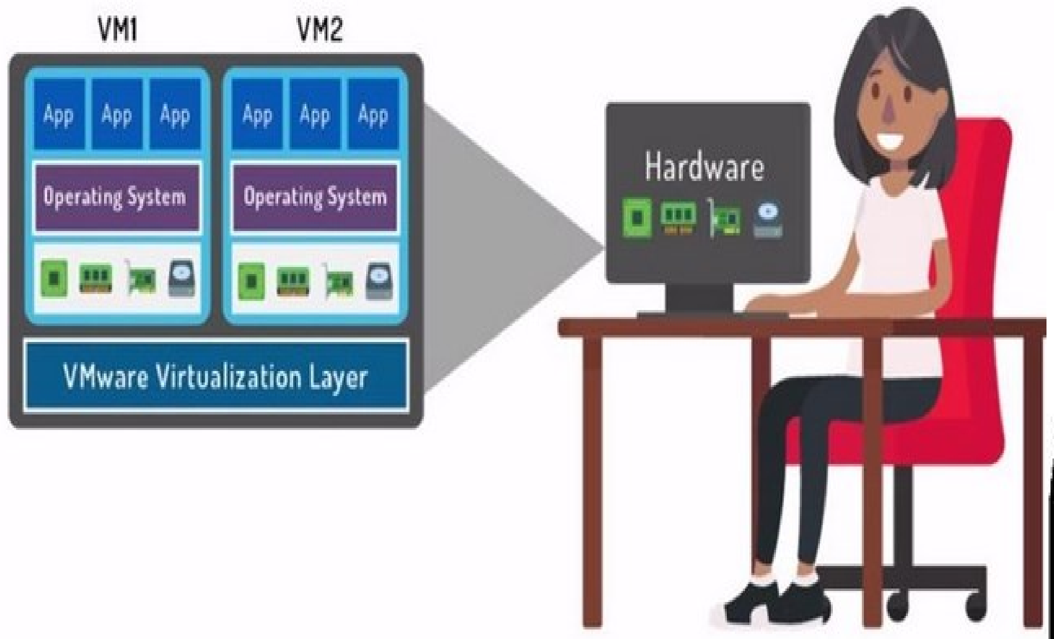
Virtual Machines

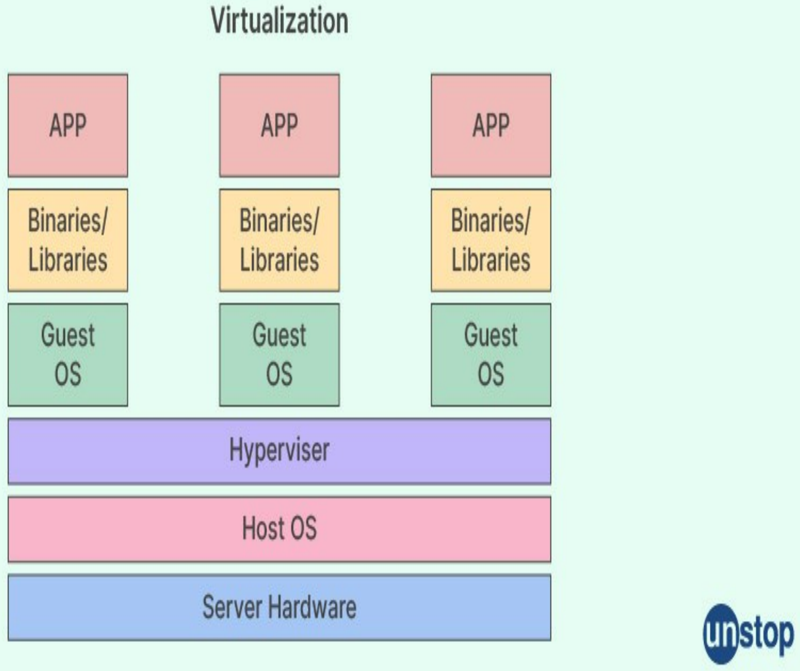


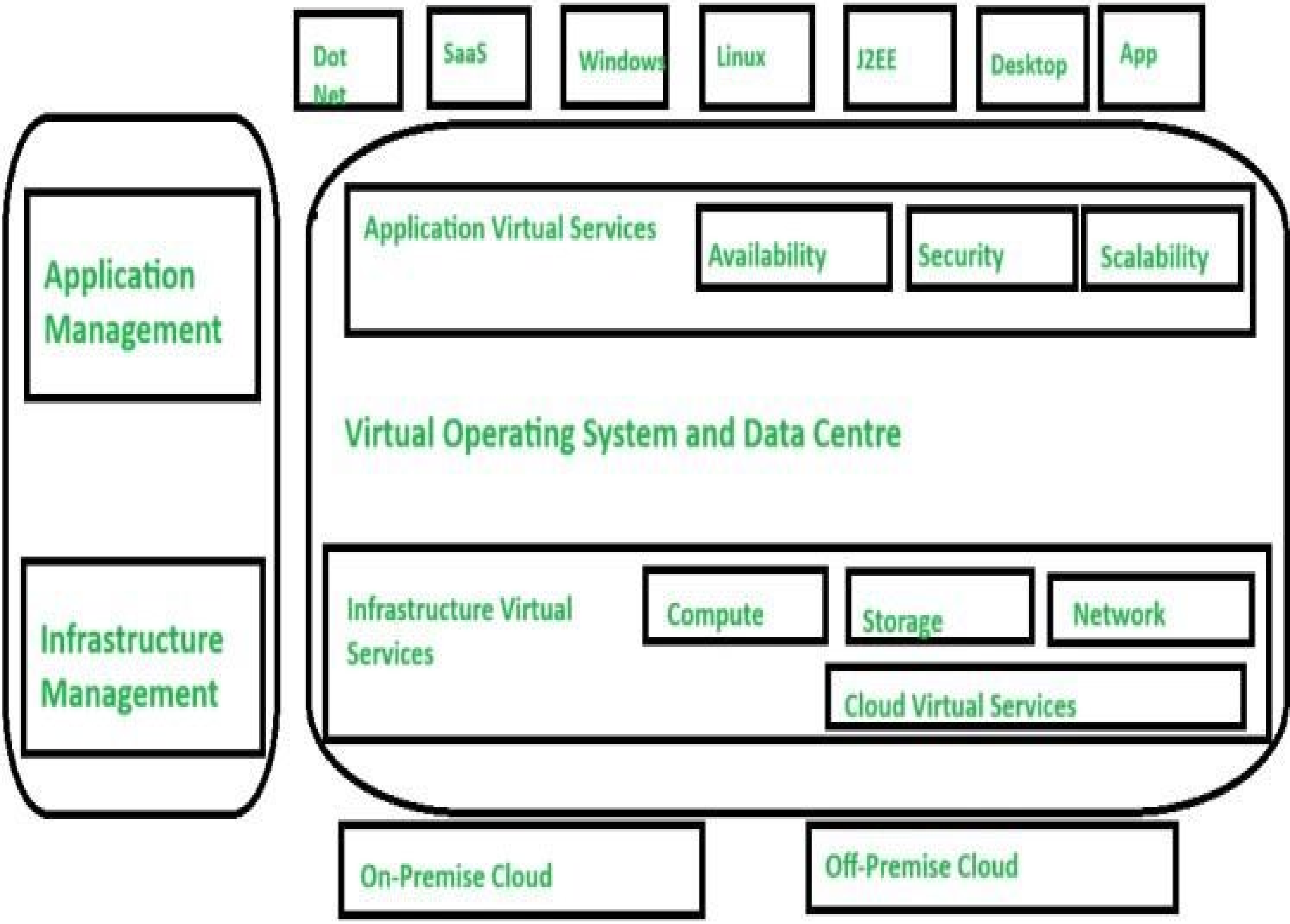
Containers

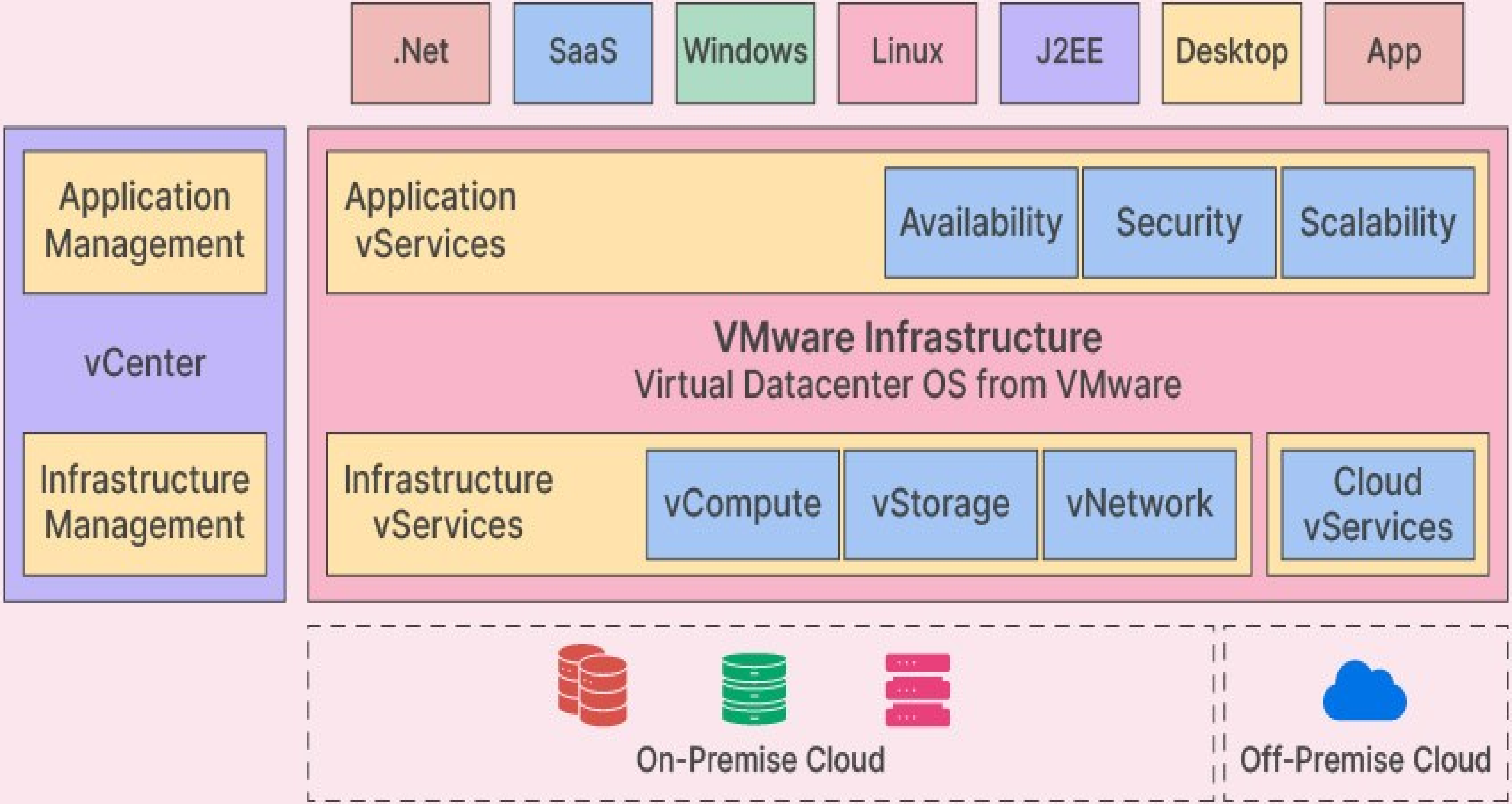
docker

LXC


VM1
VM2
App
App
App
App
App
App
Operating System
Operating System
VMware Virtualization Layer
Hardware


Virtualization
APP
APP
APP
Binaries/ Libraries
Binaries/ Libraries
Binaries/ Libraries
Guest OS
Guest OS
Guest OS
Hyperviser
Host OS
Server Hardware
unstop


Dot Net
SaaS
Windows
Linux
J2EE
Desktop
App
Application Management
Infrastructure Management
Application Virtual Services
Availability
Security
Scalability
Virtual Operating System and Data Centre
Infrastructure Virtual Services
Compute
Storage
Network
Cloud Virtual Services
On-Premise Cloud
Off-Premise Cloud


.Net
SaaS
Windows
Linux
J2EE
Desktop
App
Application Management
vCenter
Infrastructure Management
Application vServices
Availability
Security
Scalability
VMware Infrastructure
Virtual Datacenter OS from VMware
Infrastructure vServices
vCompute
vStorage
vNetwork
Cloud vServices
On-Premise Cloud
Off-Premise Cloud