Module 2 - Virtualization : Objectives

After completing this unit you should be able to understand

- ***Characteristics of Virtualized environments***
- ***Taxonomy of Virtualization Techniques***
  - ***Execution Virtualization***
  - ***Other Types of Virtualization***
- ***Virtualization and cloud computing***
- ***Pros and Cons of Virtualization***

Introduction

- Virtualization is a large umbrella of technologies and concepts that are meant to provide an abstract environment—whether this is virtual hardware or operating system—to run applications.
- This term is often synonymous with *hardware virtualization*, which plays fundamental role in efficiently delivering *Infrastructure-as-a-Service* solutions for Cloud computing.
- virtualization technologies have a long trail in the history of computer science and have come into many flavors by providing virtual environments at operating system level, programming language level, and application level.
- Virtualization technologies not only provide a virtual environment for executing applications, but also for storage, memory, and networking.

Virtualization: reasons for renewed interest

- Virtualization technologies have gained a renewed interested recently due to the confluence of different phenomena
  - ***Increased performance and computing capacity**:* Almost all modern PCs have resources enough to host a virtual machine manager and execute a virtual machine with a by far acceptable performance.
  - ***Underutilized hardware and software resources**:* Hardware and software underutilization is occurring due to (1) the increased performance and computing capacity, and (2) effect of limited or sporadic use of resources. Using these resources for other purposes after hours could improve the efficiency of the IT infrastructure. In order to transparently provide such a service, it would be necessary to deploy a completely separate environment, which can be achieved through virtualization.
  - ***Lack of space**:* The continuous need for additional capacity, whether this is storage or compute power, makes data centers grow quickly. This condition along with hardware underutilization led to the diffusion of a technique called server consolidation, for which virtualization technologies are fundamental.

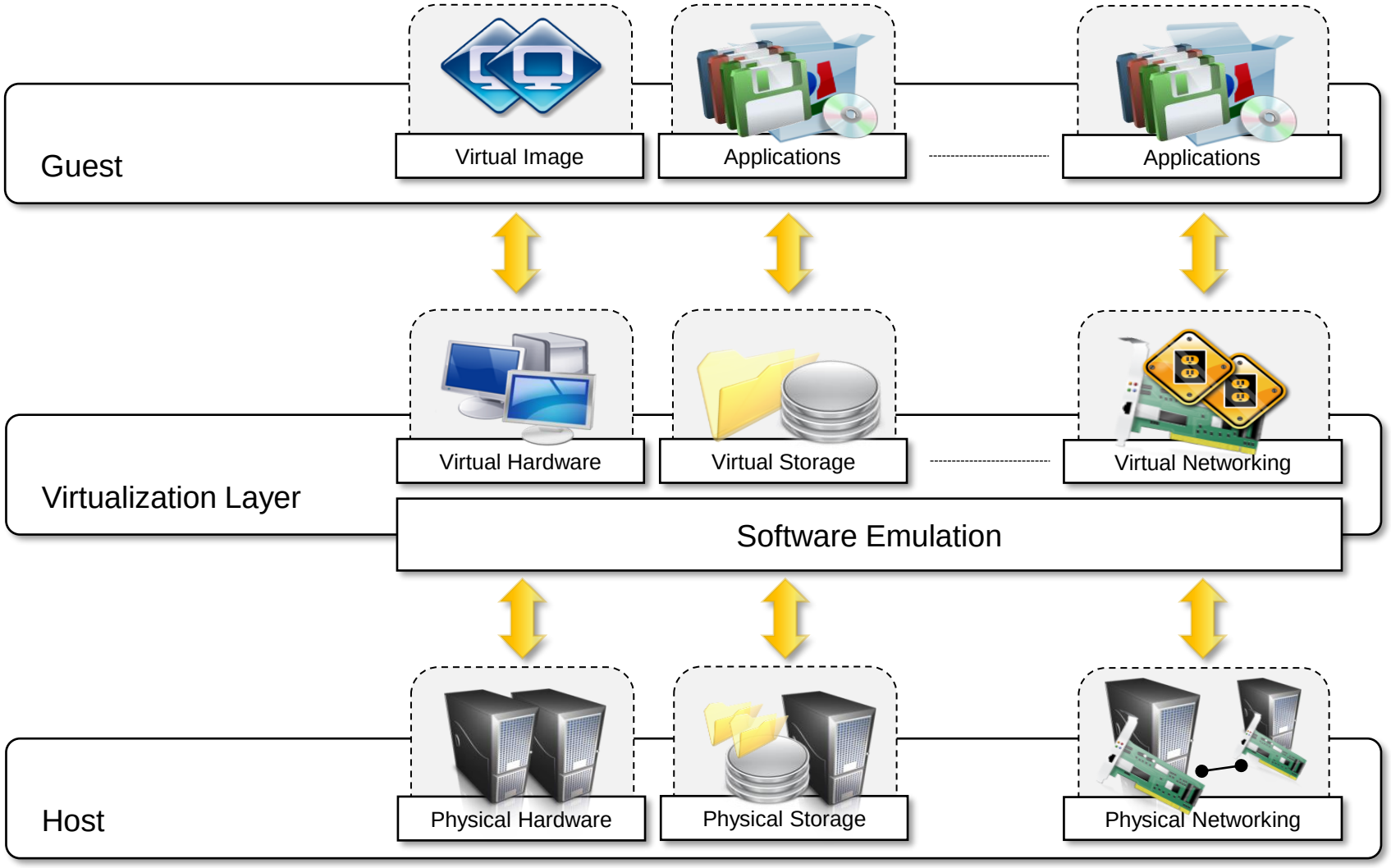
Virtualization: reasons for renewed interest

- Contd..
  - ***Greening initiatives****:* Recently, companies are increasingly looking for ways to reduce the amount of energy they consume and to reduce their carbon footprint. Hence, reducing the number of servers through server consolidation will definitely reduce the impact of cooling and power consumption of a data center. Virtualization technologies can provide an efficient way of consolidating servers.
  - ***Rise of administrative costs****:* Power consumption and cooling costs have now become higher than the cost of the IT equipment. Virtualization can help in reducing the number of required servers for a given workload, thus reducing the cost of the administrative personnel.

Virtualization reference model

- Virtualization is a broad concept and it refers to the creation of a virtual version of something, whether this is hardware, software environment, storage, or network.
- In a virtualized environment there are three major components: *guest*, *host*, and *virtualization layer*.
- The *guest* represents the system component that interacts with the virtualization layer rather than with the host as it would normally happen.
- The *host* represents the original environment where the guest is supposed to be managed.
- The *virtualization layer* is responsible for recreating the same or a different environment where the guest will operate.

Virtualization reference model



Characteristics of virtualized environments

- *Increased Security:* The ability to control the execution of a guest in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment.
  - The virtual machine represents an emulated environment in which the guest is executed. All the operations of the guest are generally performed against the virtual machine, which then translates and applies them to the host.
  - This level of indirection allows the virtual machine manager to *control* and *filter* the activity of the guest, thus preventing some harmful operations from being performed.
  - Resources exposed by the host can then be hidden or simply protected from the guest. Moreover, sensitive information that is contained in the host can be naturally hidden without the need of installing complex security policies. Increased security is a requirement when dealing with untrusted code.

Characteristics of virtualized environments contd…

- ***Managed Execution:*** Virtualization of the execution environment does not only allow the increased security but a wider range of features can be implemented. In particular, *sharing*, *aggregation*, *emulation*, and *isolation* are the most relevant.

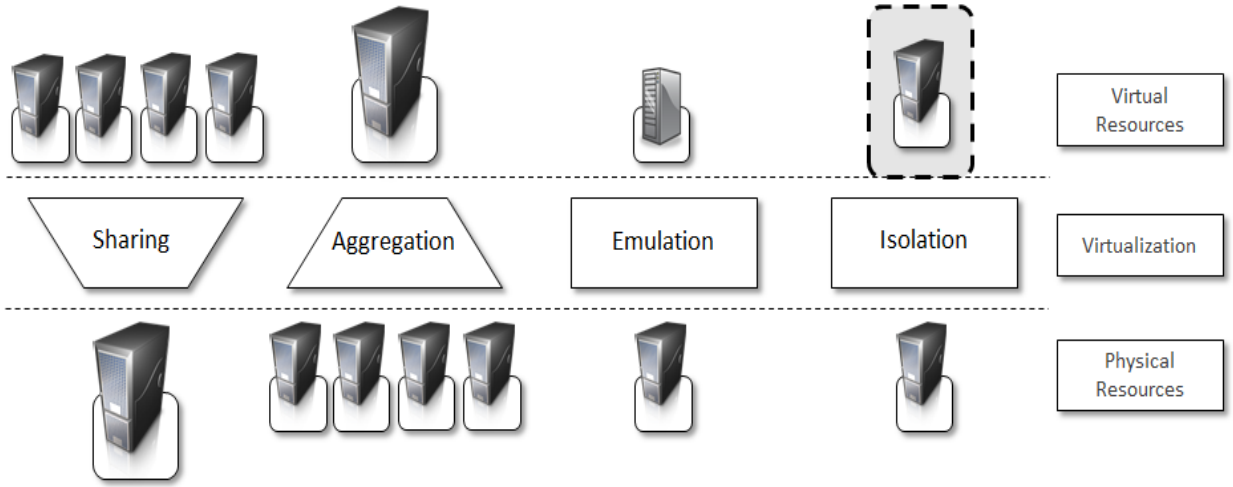


Fig- Functions Enabled by Managed Execution

Characteristics of virtualized environments contd…

- *Sharing:*
  - Virtualization allows the creation of a separate computing environment within the same host. In this way it is possible to fully exploit the capabilities of a powerful guest, which would be otherwise underutilized.
  - Sharing is a particularly important feature in virtualized data centers, where this basic feature is used to reduce the number of active servers and limit power consumption.
- *Aggregation.*
  - It is not only possible to share the physical resource among several guests, but virtualization also allows the aggregation, which is the opposite process. A group of separate hosts can be tied together and represented to guests as a single virtual host. This function is naturally implemented in middleware for distributed computing and a classical example is represented by cluster management software, which harnesses the physical resources of a homogeneous group of machines and represents them as a single resource.

Characteristics of virtualized environments contd…

- *Emulation.* Guests are executed within an environment that is controlled by the virtualization layer, which ultimately is a program. This allows for controlling and tuning the environment that is exposed to guests.
  - For instance, a complete different environment with respect to the host can be emulated, thus allowing the execution of guests requiring specific characteristics that are not present in the physical host.
  - hardware virtualization solutions are able to provide virtual hardware and emulate a particular kind of device such as *SCSI (Small Computer System Interface)* devices for file IO, without the hosting machine having such hardware installed.
  - Old and legacy software, which does not meet the requirements of current systems, can be run on emulated hardware without any need of changing their code. This is possible either emulating the required hardware architecture or within a specific operating system sandbox, such as the MS-DOS mode in Windows 95/98.

Characteristics of virtualized environments contd…

- *Isolation.* Virtualization allows providing guests—whether they are operating systems, applications, or other entities—with a complete separate environment, in which they are executed. The guest performs its activity by interacting with an abstraction layer, which provides access to the underlying resources.
  - Isolation brings several benefits, for example it allows multiple guests to run on the same host without each of them interfering with the other.
  - Secondly, it provides a separation between the host and the guest. The virtual machine can filter the activity of the guest and prevent harmful operations against the host.

Characteristics of virtualized environments contd…

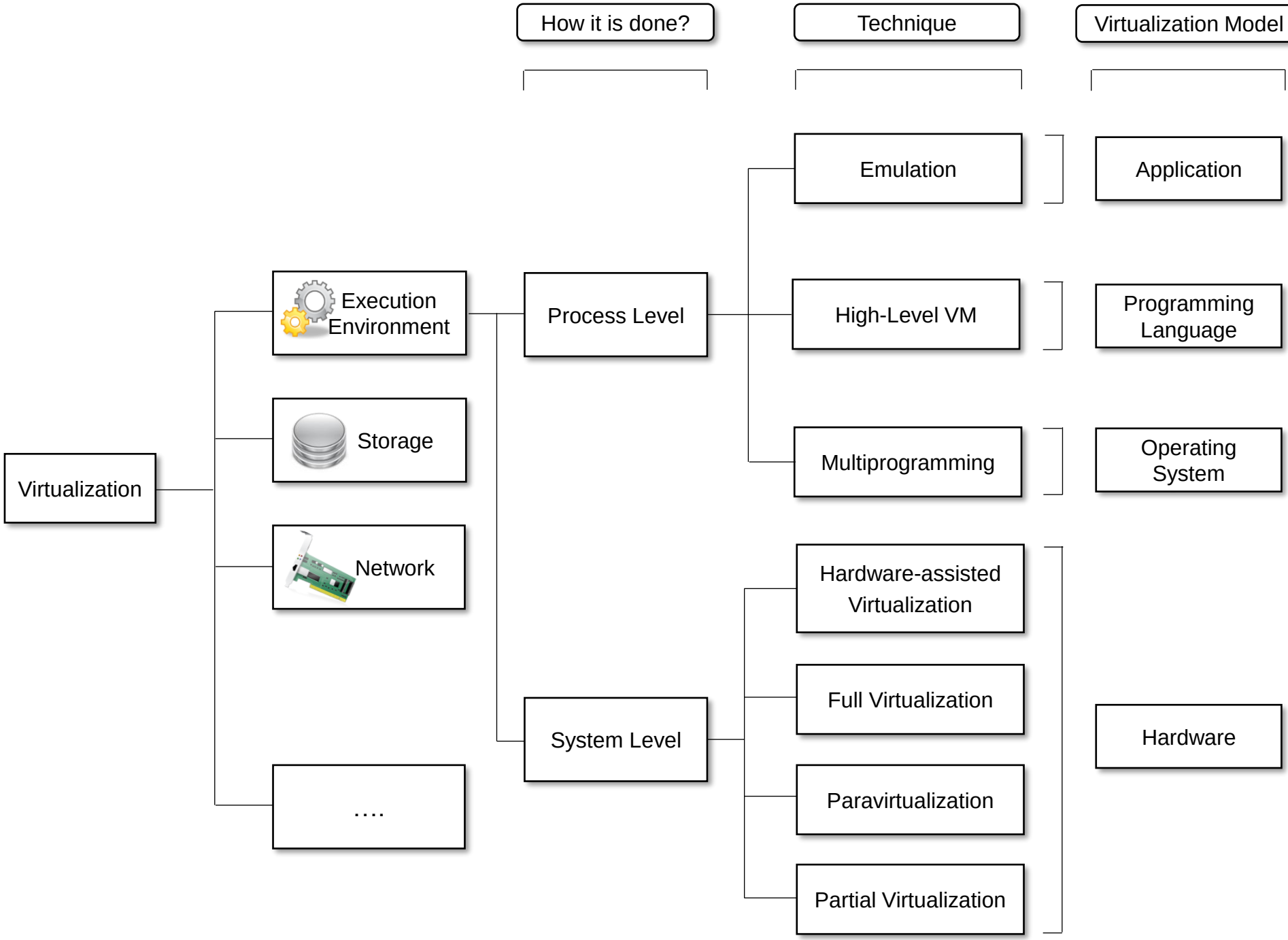
- ***Performance tuning:*** This feature is a reality at present time, given the considerable advances in hardware and software supporting virtualization. It becomes easier to control the performance of the guest by finely tuning the properties of the resources exposed through the virtual environment. This provides means to effectively implement a Quality of Service infrastructure that more easily fulfill the service level agreement established for the guest.
  - For instance software implementing hardware virtualization solutions can expose to a guest operating system only a fraction of the memory of the host machine or to set the maximum frequency of the processor of the virtual machine.
  - Another advantage of managed execution is that sometimes it allows easily capturing the state of the guest, persisting it, and resuming its execution.
    - This, for example, allows virtual machine managers such as *Xen Hypervisor* to stop the execution of a guest operating system, to move its virtual image into another machine, and to resume its execution in a completely transparent manner. This technique is called *virtual machine migration* and constitutes an important feature in virtualized data centers for optimizing their efficiency in serving applications demand.

Characteristics of virtualized environments contd…

- *Portability:* The concept of portability applies in different ways according to the specific type of virtualization considered.
  - In the case of a hardware virtualization solution the guest is packaged into a virtual image that, in most of the cases, can be safely moved and executed on top of different virtual machines.
  - In the case of programming level virtualization, as implemented by the JVM or the .NET runtime, the binary code representing application components (jars or assemblies), can be run without any recompilation on any implementation of the corresponding virtual machine.
  - This makes the application development cycle more flexible and application deployment very straightforward: one version of the application, in most of the cases, is able to run on different platforms with no changes.

Taxonomy of Virtualization Techniques

- Virtualization covers a wide range of emulation techniques that are applied to different areas of computing. A classification of these techniques helps to better understand their characteristics and use.
  - Virtualization is mainly used to emulate *execution environments*, *storage*, and *networks*.
  - Among these categories *execution virtualization* constitutes the oldest, most popular, and most developed area.
  - We can divide these execution virtualization techniques into two major categories by considering the type of host they require.
    - *Process level* techniques are implemented on top of an existing operating system, which has full control of the hardware.
    - *System level* techniques are implemented directly on hardware and do not require—or require a minimum support from—an existing operating system.


How it is done?
Technique
Virtualization Model
Virtualization
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

Execution Virtualization

- Execution virtualization includes all those techniques whose aim is to emulate an execution environment that is separate from the one hosting the virtualization layer.
- All these techniques concentrate their interest on providing support for the execution of programs, whether these are the operating system, a binary specification of a program compiled against an abstract machine model, or an application.
- Therefore, execution virtualization can be implemented directly on top of the hardware, by the operating system, an application, or libraries dynamically or statically linked against an application image.

Machine Reference Model

- Virtualizing an execution environment at different levels of the computing stack requires a reference model that defines the interfaces between the levels of abstractions, which hide implementation details.
- From this perspective, virtualization techniques actually replace one of the layers and intercept the calls that are directed towards it.
- Therefore, a clear separation between layers simplifies their implementation, which only requires the emulation of the interfaces and a proper interaction with the underlying layer.
- Modern computing systems can be expressed in terms of the reference model described in the figure.

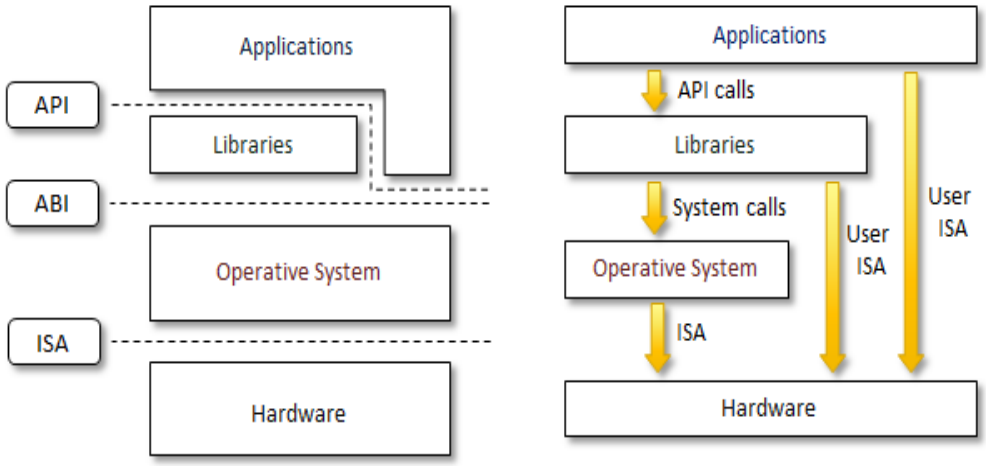


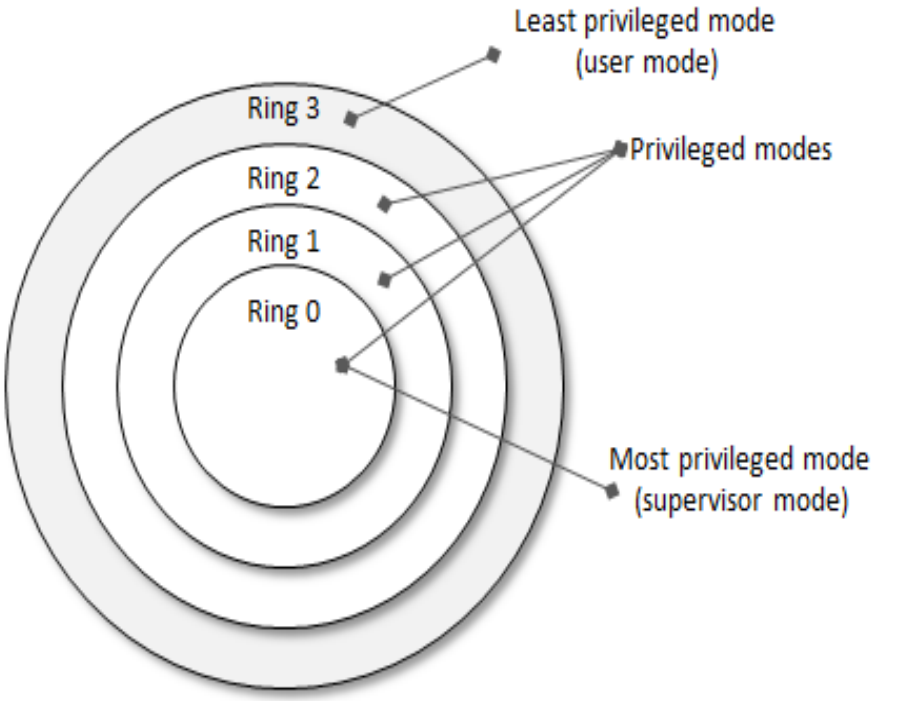
Fig: Machine reference Model

Machine Reference Model contd…

- At the bottom layer, the model for the hardware is expressed in terms of the *Instruction Set Architecture (ISA)*, which defines the instruction set for the processor, registers, memory, and interrupts management.
- ISA is the interface between hardware and software and it is important for the OS developer (*System ISA*), and developers of applications that directly manage the underlying hardware (*User ISA*).
- The *Application Binary Interface (ABI)* separates the operating system layer from the applications and libraries, which are managed by the OS.
- ABI covers details such as low-level data types, alignment, and call conventions and defines a format for executable programs. System calls are defined at this level. This interface allows portability of applications and libraries across operating systems that implement the same ABI.
- The highest level of abstraction is represented by the *Application Programming Interface (API)*, which interfaces applications to libraries and/or the underlying operating system.
- The machine level resources such as processor registers and main memory capacities are used to perform the operation in the hardware level of CPU.
- Such layered approach simplifies the development and implementation of computing systems and also simplifies the implementation of multi-tasking and the co-existence of multiple executing environments.

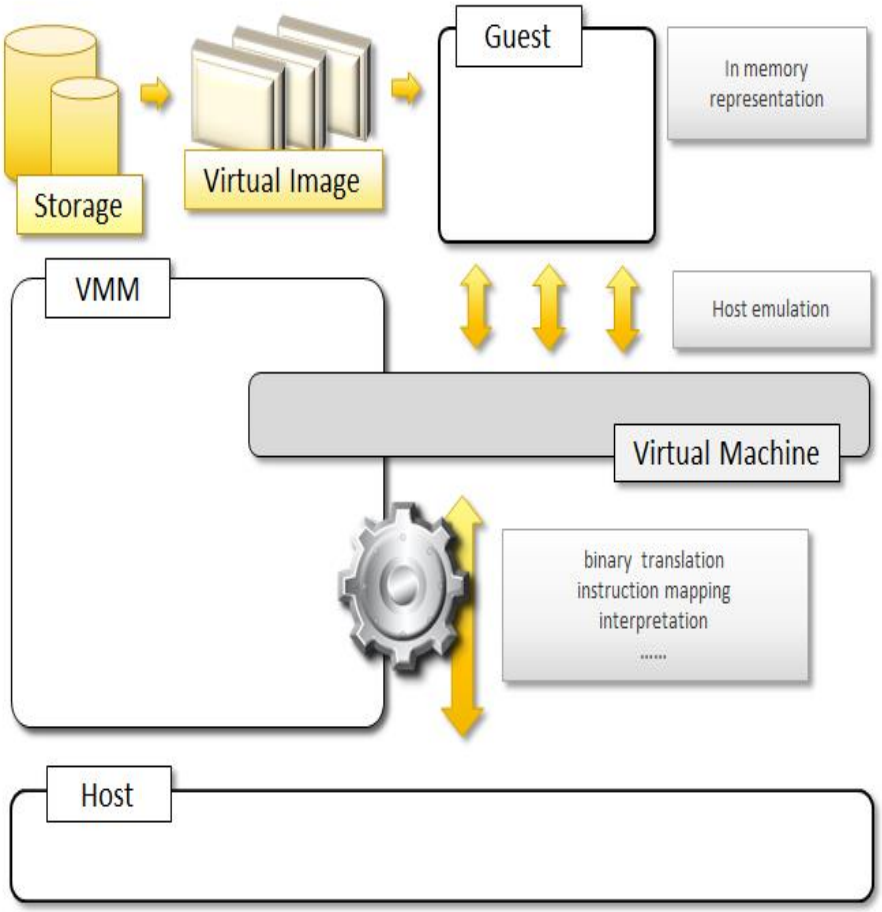
Security Rings and Privileged Modes

- Machine reference model also provides ways for implementing a minimal security model for managing and accessing shared resources.
- For this purpose, the instruction set exposed by the hardware has been divided into different security classes, which define who can operate with them.
- The first distinction can be made between *privileged* and *non-privileged* instructions.
- Non-privileged instructions are those instructions that can be used without interfering with other tasks because they do not access shared resources. This category contains, for example, all the floating, fixed point, and arithmetic instructions.
- Privileged instructions are those that are executed under specific restrictions and are mostly used for sensitive operations, which expose (*behavior sensitive*) or modify (*control sensitive*) the privileged state.
- a possible implementation features a hierarchy of privileges (see **Figure**) in the form of ring based security: *Ring 0, Ring 1, Ring 2,* and *Ring 3;* Ring 0 is in the most privileged level and the Ring 3 in the least privileged level. Ring 0 is used by the kernel of the OS and rings 1 and 2 are used by the OS level services and Ring 3 is used by the user. Recent systems support only two levels with Ring 0 for the supervisor mode and Ring 3 for user mode.



Hardware Level Virtualization

- Hardware level virtualization is a virtualization technique that provides an abstract execution environment in terms of computer hardware on top of which a guest operating system can be run.
- In this model, the guest is represented by the operating system, the host by the physical computer hardware, the virtual machine by its emulation, and virtual machine manager by the *hypervisor*.
- The hypervisor is generally a program, or a combination of software and hardware, that allows the abstraction of the underlying physical hardware.
- Hardware level virtualization is also called *system virtualization*, since it provides ISA to virtual machines, which is the representation of the hardware interface of a system. This is to differentiate from *process virtual machines*, which expose ABI to virtual machines.



Hypervisors

- A fundamental element of hardware virtualization is the hypervisor, or virtual machine manager (VMM). It recreates a hardware environment, where guest operating systems are installed. There are two major types of hypervisors: *Type I* and *Type II*.
- *Type I* hypervisors run directly on top of the hardware. Therefore, they take the place of the operating systems and interact directly with the ISA interface exposed by the underlying hardware, and emulate this interface in order to allow the management of guest operating systems. This type of hypervisors is also called *native virtual machine*, since it run natively on hardware.
- *Type II* hypervisors require the support of an operating system to provide virtualization services. This means that they are programs managed by the operating system, which interact with it through the ABI and emulate the ISA of virtual hardware for guest operating systems. This type of hypervisors is also called *hosted virtual machine*, since it is hosted within an operating system.

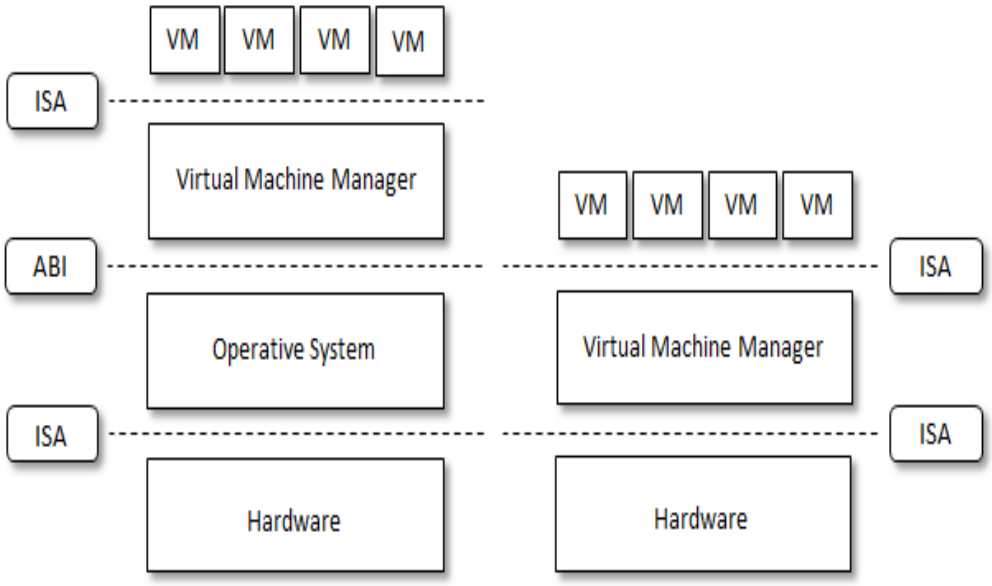


Fig-Hosted (left) and Native (right) Virtual Machine

Hypervisor Reference Architecture

- Conceptually, a virtual machine manager is internally organized as described in the Figure.
- Three main modules coordinate their activity in order to emulate the underlying hardware: *dispatcher*, *allocator*, and *interpreter*.
- The dispatcher constitutes the entry point of the monitor and reroutes the instructions issued by the virtual machine instance to one of the two other modules.
- The allocator is responsible for deciding the system resources to be provided to the VM: whenever a virtual machine tries to execute an instruction that results in changing the machine resources associated with that VM, the allocator is invoked by the dispatcher.
- The interpreter module consists of interpreter routines. These are executed whenever a virtual machine executes a privileged instruction: a trap is triggered and the corresponding routine is executed.

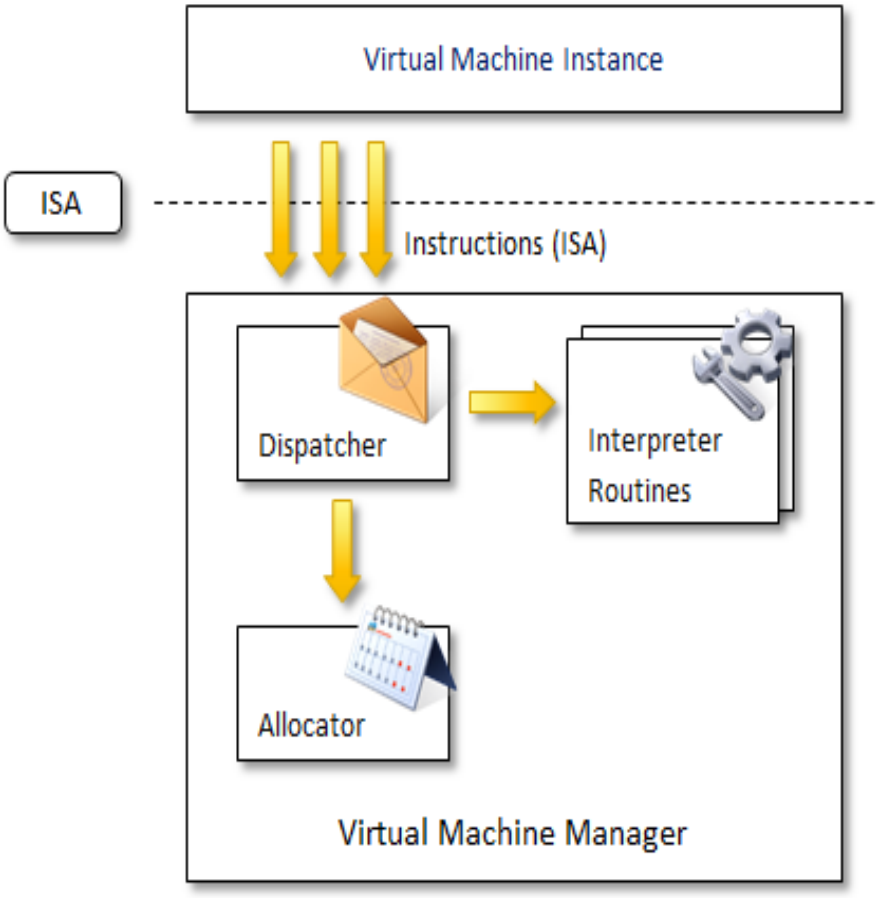


Fig- hypervisor reference architecture

Hypervisor Reference Architecture *contd…*

- The design and architecture of a virtual machine manager, together with the underlying hardware design of the host machine, determine the full realization of hardware virtualization, where a guest operating system can be transparently executed on top of a VMM as if it was run on the underlying hardware.
- The criteria that need to be met by a virtual machine manager to efficiently support virtualization were established by Goldberg and Popek in 1974 [23]. Three properties have to be satisfied:
  - *Equivalence:* a guest running under the control of a virtual machine manager should exhibit the same behavior that when it is executed directly on the physical host.
  - *Resource control:* the virtual machine manager should be in complete control of virtualized resources.
  - *Efficiency:* a statistically dominant fraction of the machine instructions should be executed without intervention from the virtual machine manager.
- The major factor that determines whether these properties are satisfied is represented by the layout of the ISA of the host running a virtual machine manager. Popek and Goldberg provided a classification of the instruction set and proposed **three theorems** that define the properties that hardware instructions need to satisfy in order to efficiently support virtualization.

Popek and Goldberg theorems

- Theorem-1*: For any conventional third-generation computer, a VMM may be constructed if the set of sensitive instructions for that computer is a subset of the set of privileged instructions.*
- Theorem 2: *A conventional third-generation computer is recursively virtualizable if*
  - *It is virtualizable and.*
  - *A VMM without any timing dependencies can be constructed for it.*
- Theorem 3: *A hybrid VMM may be constructed for any conventional third generation machine, in which the set of user sensitive instructions are a subset of the set of privileged instructions.*

Hardware Virtualization Techniques

- **Hardware-assisted virtualization:**
  - This term refers to a scenario in which the hardware provides architectural support for building a virtual machine manager able to run a guest operating system in complete isolation.
  - This technique was originally introduced in the IBM System/370. At present, examples of hardware-assisted virtualization are the extensions to the x86-64 bit architecture introduced with *Intel VT*
- **Full virtualization**
  - Full virtualization refers to the ability of running a program, most likely an operating system, on top of a virtual machine directly and without any modification, as if it were run on the raw hardware.
  - In order to make this possible, virtual machine managers are required to provide a complete emulation of the entire underlying hardware.
  - The principal advantage of full virtualization is complete isolation, which leads to enhanced security, ease of emulation of different architectures, and coexistence of different systems on the same platform. Whereas it is a desired goal for many virtualization solutions, it poses important concerns on performance and technical implementation.

Hardware Virtualization Techniques

- **Paravirtualization**
  - This is a not transparent virtualization solution that allows implementing thin virtual machine managers.
  - Paravirtualization techniques expose a software interface to the virtual machine that is slightly modified from the host and, as a consequence, guests need to be modified.
  - The aim of paravirtualization is to provide the capability to demand the execution of performance critical operation directly on the host.
  - solutions using paravirtualization include: *VMWare*, *Parallels*, and some solutions for embedded and real time environment such as *TRANGO*, *Wind River*, and *XtratuM*.
- **Partial virtualization**
  - Partial virtualization provides a partial emulation of the underlying hardware, thus not allowing the complete execution of the guest operating system in complete isolation.
  - Partial virtualization allows many applications to run transparently but not all the features of the operating system can be supported as happens with full virtualization.
  - An example of partial virtualization is address space virtualization used in time sharing systems.
  - Partial virtualization was implemented on the experimental *IBM M44/44X*. Address space virtualization is a common feature of contemporary operating systems.

Operating System Level Virtualization

- Operating System level virtualization offers the opportunity to create different and separated execution environment for applications that are managed concurrently.
- Differently from hardware virtualization, there is no virtual machine manager or hypervisor and the virtualization is done within a single operating system, where the OS kernel allows for multiple isolated user space instances.
- The kernel is also responsible for sharing the system resources among instances and for limiting the impact of instances on each other.
- A user space instance in general contains a proper view of the file system which is completely isolated, separate IP addresses, software configurations, and access to devices.
- Examples of operating system level virtualizations are: *FreeBSD Jails*, *IBM Logical Partition (LPAR)*, *SolarisZones* and *Containers*, *Parallels Virtuozzo Containers*, *OpenVZ*, *iCore Virtual Accounts*, *Free Virtual Private Server (FreeVPS)* and others.
- The services offered by each of these technologies differ and most of them are available on Unix based systems.

Programming Language Level Virtualization

- Programming language level virtualization is mostly used for achieving ease of deployment of applications, managed execution, and portability across different platforms and operating systems.
- It consists of a virtual machine executing the byte code of a program, which is the result of the compilation process.
- Compilers implemented used this technology produce a binary format representing the machine code for an abstract architecture.
- The main advantage of programming-level virtual machines, also called process virtual machines, is the ability of providing a uniform execution environment across different platforms.
- Programs compiled into byte code can be executed on any operating system and platform for which a virtual machine able to execute that code has been provided.
- As an example, both Java and .NET provide an infrastructure for pluggable security policies and code access security frameworks.

Application Level Virtualization

- Application level virtualization is a technique allowing applications to be run on run-time environments which do not natively support all the features required by such applications.
- In this scenario, applications are not installed in the expected run time environment, but run as if they were.
- In general, these techniques are mostly concerned with partial file systems, libraries, and operating system component emulation.
- Such emulation is performed by a thin layer—a program or an operating system component—that is in charge of executing the application.
-

Application Level Virtualization

- Emulation can also be used to execute program binaries compiled for different hardware architectures. In this case, one of the following strategies can be implemented:
  - *Interpretation.* In this technique every source instruction is interpreted by emulator for executing native ISA instructions leading to poor performance. Interpretation has a minimal startup cost but a huge overhead since each instruction is emulated.
  - *Binary Translation.* In this technique every source instruction is converted to native instructions with equivalent functions. After a block of instructions is translated it is cached and reused. Binary translation has a large initial overhead cost but over time it is subject to a better performance, since previously translated instruction blocks are directly executed.
- Application virtualization is a good solution in the case of missing libraries in the host operating system: in this case a replacement library can be linked with the application or library calls can be remapped to existing functions available in the host system.
- Another advantage is that in this case the virtual machine manager is much lighter since it provides a partial emulation of the run time environment if compared to hardware virtualization.
- One of the most popular solution implementing application virtualization is ***Wine***, which is a software application allowing Unix-like operating systems to execute programs written for the Microsoft Windows platform.

Storage Virtualization

- Storage virtualization is a system administration practice that allows decoupling the physical organization of the hardware from its logical representation.
- By using this technique users do not have to be worried about the specific location of their data, which can be identified by using a logical path.
- Storage virtualization allows harnessing a wide range of storage facilities and representing them under a single logical file system.
- There are different techniques for storage virtualization one of the most popular includes network based virtualization by means of *Storage Area Networks (SANs)*.
- Storage Area Networks use a network accessible device through a large bandwidth connection to provide storage facilities.

Network Virtualization

- Network virtualization combines hardware appliances and specific software for the creation and management of a virtual network.
- Network virtualization can aggregate different physical networks into a single logical network (*external* network virtualization) or provide network like functionality to an operating system partition (*internal* network virtualization).
- The result of external network virtualization is generally a *Virtual LAN (VLAN)*. A *VLAN* is an aggregation of hosts that communicate with each other as if they were located under the same broadcasting domain.
- Internal network virtualization is generally applied together with hardware and operating system level virtualization in which the guests obtain a virtual network interface to communicate with.
- There are several options for implementing internal network virtualization: the guest can share the same network interface of the host and use NAT to access the network; the virtual machine manager can emulate, and install on the host, an additional network device together with the driver; or the guest can have a private network only with the guest.

Desktop Virtualization

- Desktop virtualization abstracts the desktop environment available on a personal computer in order to provide access to it by using a client server approach.
- Desktop virtualization provides the same outcome of hardware virtualization but serves a different purpose.
- Similarly to hardware virtualization it makes accessible a different system as if it was natively installed on the host, but this system is remotely stored on a different host and accessed through a network connection.
- Moreover, desktop virtualization addresses the problem of making the same desktop environment accessible from everywhere.
- While the term desktop virtualization strictly refers to the ability to remotely access a desktop environment, generally, the desktop environment is stored in a remote server or a data center which provides a high availability infrastructure and ensures the accessibility and the persistence of the data.
- The basic services for remotely accessing a desktop environment are implemented in software components such as: *Windows Remote Services*, *VNC*, and *X Server*.
- Infrastructures for desktop virtualization based on Cloud computing solutions are: *Sun Virtual Desktop Infrastructure (VDI)*, *Parallels Virtual Desktop Infrastructure (VDI)*, *Citrix XenDesktop* and others.

Application Server Virtualization

- Application server virtualization abstracts a collection of application servers that provide the same services as a single virtual application server by using load balancing strategies and providing a high availability infrastructure for the services hosted in the application server.
- This is a particular form of virtualization and serves the same purpose of storage virtualization: providing a better quality of service rather than emulating a different environment.

Virtualization and Cloud Computing

- Virtualization plays an important role in Cloud computing, since it allows for the appropriate degree of customization, security, isolation.
- Virtualization technologies are primarily used to offer configurable computing environments and storage.
- Particularly important is the role of virtual computing environment and execution virtualization techniques. Among these, hardware and programming language virtualization are the techniques adopted in Cloud computing systems.
- virtualization also gives the opportunity of designing more efficient computing systems by means of consolidation
- Server consolidation and virtual machine migration are principally used in case of hardware virtualization even though technically possible also in case of programming language virtualization.
- Storage virtualization constitutes an interesting opportunity given by virtualization technologies, often complimentary to the execution virtualization.
- Finally, Cloud computing revamps the concept of desktop virtualization, initially introduced in the mainframe era.

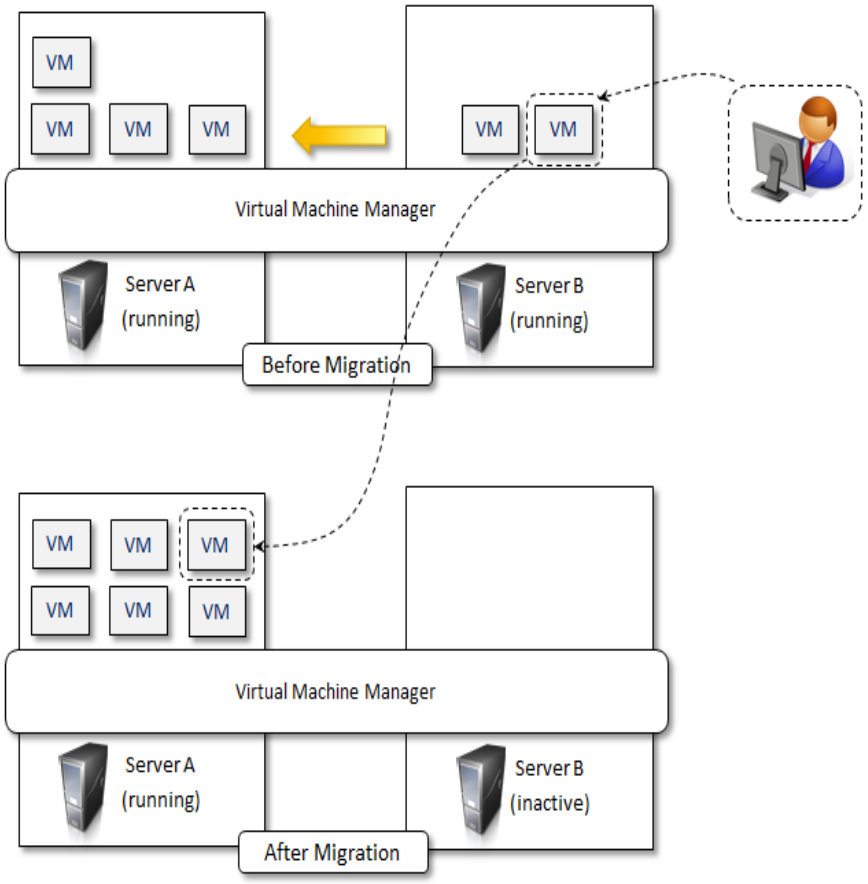


Fig- . Live Migration and Server Consolidation

Pros and Cons of Virtualization

- ***Advantages of Virtualization***
  - Managed execution and isolation are perhaps the most important advantages of virtualization.
  - these two characteristics allow building secure and controllable computing environments. A virtual execution environment can be configured as a sandbox, thus preventing any harmful operation to cross the borders of the virtual host.
  - Moreover, allocation of resources and their partitioning among different guests is simplified, being the virtual host controlled by a program.
  - Portability is another advantage of virtualization, especially for execution virtualization techniques.
  - Portability and self-containment also contribute to reduce the costs for maintenance, since the number of hosts is expected to be lower than the number of virtual machine instances.
  - Finally, by means of virtualization it is possible to achieve a more efficient use of resources. Multiple systems can securely coexist and share the resources of the underlying host, without interfering with each other.

Pros and Cons of Virtualization

- ***Disadvantages of Virtualization***
  - ***Performance Degradation***
    - Performance is definitely one of the major concerns when using virtualization technology. Since virtualization interposes an abstraction layer between the guest and the host, increased latencies and delays can be experienced by the guest.
    - Also, when hardware virtualization is realized through a program that is installed or executed on top of the host operating systems, a major source of performance degradation is represented by the fact that the virtual machine manager is executed and scheduled together with other applications, thus sharing with them the resources of the host.
  - ***Inefficiency and Degraded User Experience***
    - Virtualization can sometime led to an inefficient use of the host. In particular, some of the specific features of the host cannot be exposed by the abstraction layer and then become not accessible.
  - ***Security Holes and New Threats***
- Virtualization opens the door to a new and unexpected form of *phishing.* The capability of emulating a host in a complete transparent manner, has led the way to malicious programs which are designed to extract from the guest sensitive information.

Technology examples

A wide range of virtualization technology is available especially for virtualizing computing environments.

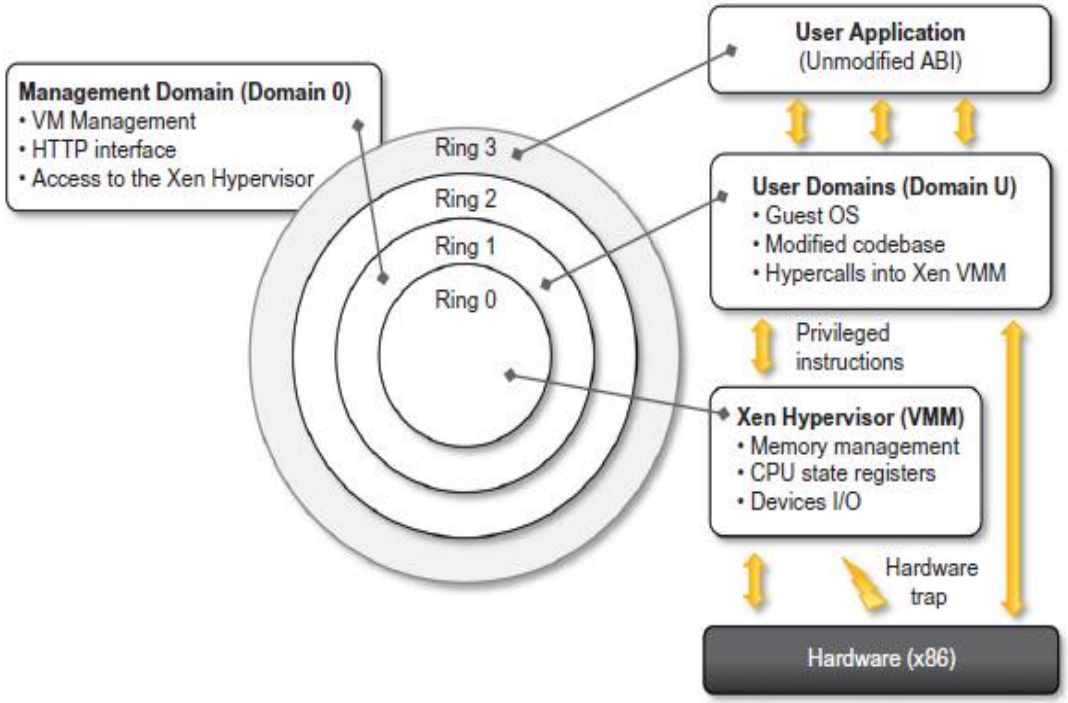
- ***Xen: para virtualization***
- ***VMware: full virtualization***
- ***Microsoft Hyper-V***

Technology examples

- ***Xen: para virtualization***
  - Xen is an open-source initiative implementing a virtualization platform based on paravirtualization.
  - Initially developed by a group of researchers at the University of Cambridge in the United Kingdom, Xen now has a large open-source community backing it.
  - Citrix also offers it as a commercial solution, XenSource.
  - Xen-based technology is used for either desktop virtualization or server virtualization, and recently it has also been used to provide cloud computing solutions by means of Xen Cloud Platform (XCP).
  - Recently Xen has been advanced to support full virtualization using hardware-assisted virtualization

Technology examples - Xen: para virtualization

- Figure Describes the architecture of Xen and its mapping on to a classic x86 privilege model.
- A Xen-based system is managed by the Xen hypervisor, which runs in the highest privileged mode and controls the access of guest operating system to the underlying hardware.



Technology examples

- Operating systems are executed within domains, which represent virtual machine instances.
- Moreover, specific control software, which has privileged access to the host and controls all the other guest operating systems, is executed in a special domain called Domain0.
- This is the first one that is loaded once the virtual machine manager has completely booted, and it hosts a Hyper Text Transfer Protocol(HTTP) server that serves requests for virtual machine creation, configuration, and termination.
- This component constitutes the embryonic version of a distributed virtual machine manager, which is an essential component of cloud computing systems providing Infrastructure-as-a-Service(IaaS)solutions.

Technology examples

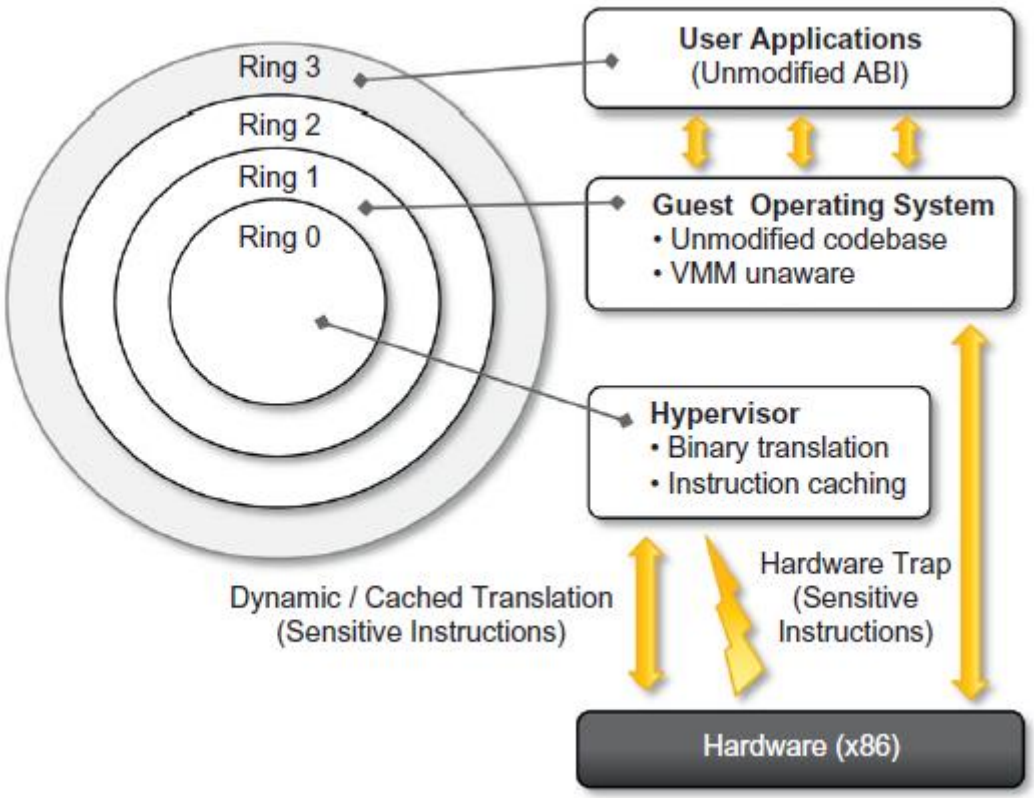
- Ring 0 represent the level with the highest privileges and Ring 3 the level with the lowest ones
- Para virtualization needs the operating system codebase to be modified, and hence not all operating systems can be used as guests in a Xen-based environment.
- Open-source operating systems such as Linux can be easily modified, since their code is publicly available and Xen provides full support for their virtualization, whereas components of the Windows family are generally not supported by Xen unless hardware-assisted virtualization is available.

VMware: fullvirtualization- Full virtualization and binary translation

- Here the underlying hardware is replicated and made available to the guest operating system
- VMware implements full virtualization either in the desktop environment, by means of Type II hypervisors, or in the server environment, by means of Type I hypervisors.
- VMware provides additional tools and software that simplify the use of virtualization technology either in a desktop environment, with tools enhancing the integration of virtual guests with the host, or in a server environment, with solutions for building and managing virtual computing infrastructures
- As discussed before, x86 architecture design does not satisfy the first theorem of virtualization since the set of sensitive instructions is not a subset of the privileged instructions

VMware: fullvirtualization- Full virtualization and binary translation

- This causes a different behavior when such instructions are not executed in Ring 0, which is the normal case in a virtualization scenario where the guest OS is run in Ring 1
- In the case of dynamic binary translation, the trap triggers the translation of the offending instructions into an equivalent set of instructions that achieves the same goal without generating exceptions.



VMware: fullvirtualization- Full virtualization and binary translation

- This approach has both advantages and disadvantages
- The major advantage is that guests can run unmodified in a virtualized environment.
- Binary translation is applied to only a subset of the instruction set, whereas the others are managed through direct execution on the underlying hardware. This somehow reduces the impact on performance of binary translation
- CPU virtualization is only a component of a fully virtualized hardware environment
- VMware achieves full virtualization by providing virtual representation of memory and I/O devices
- Memory virtualization constitutes another challenge of virtualized environments and can deeply impact performance without the appropriate hardware support.
- VMware also provides Desktop virtualization and Server Virtualization

Microsoft Hyper-V

- Hyper-V supports multiple and concurrent execution of guest operating systems by means of partitions. A partition is a completely isolated environment in which an operating system is installed and run.

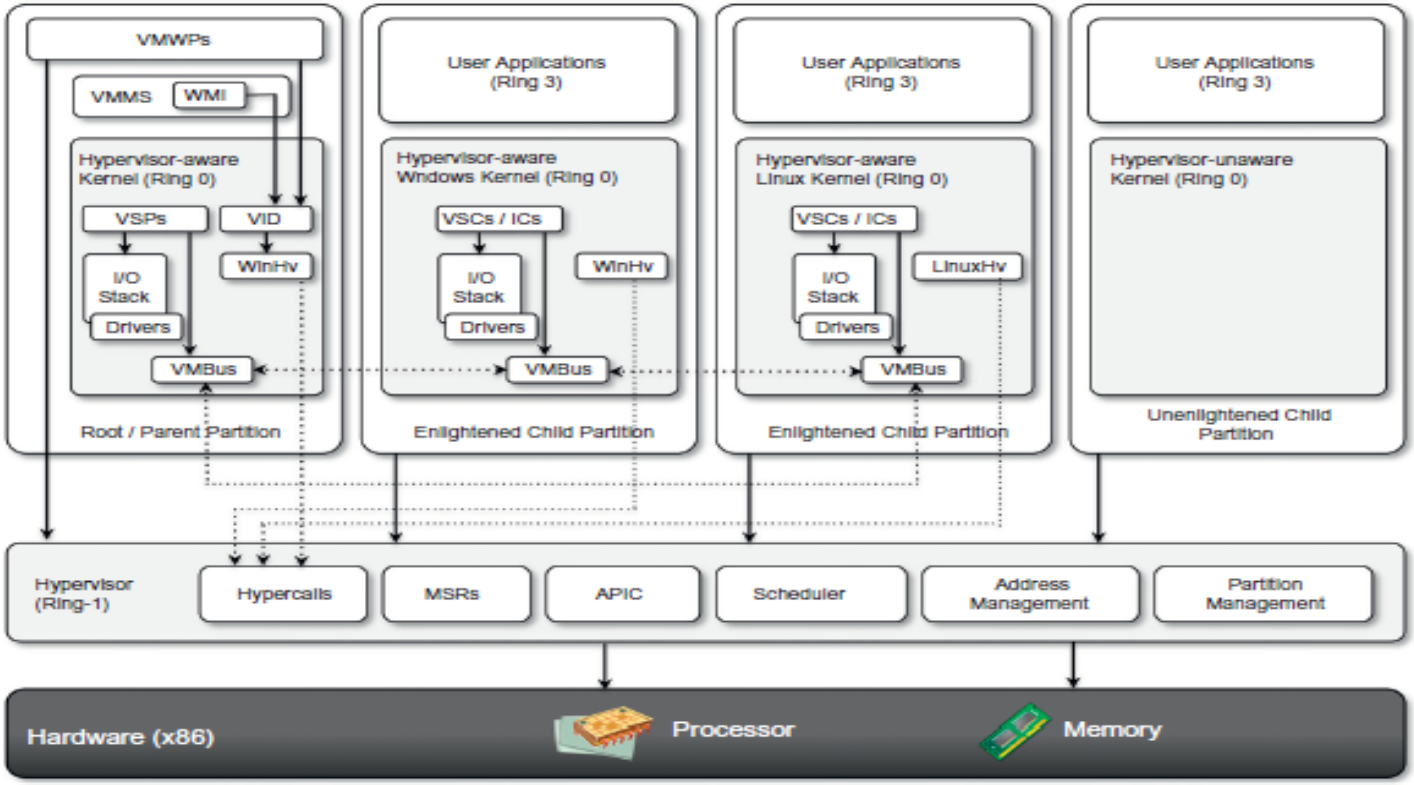


FIGURE 3.17
Microsoft Hyper-V architecture.

Microsoft Hyper-V

- Hyper-V takes control of the hardware, and the host operating system becomes a virtual machine instance with special privileges, called the parent partition -has direct access to the hardware.
- Root Partition runs the virtualization stack, hosts all the drivers required to configure guest operating systems, and creates child partitions through the hypervisor
- **Hypervisor**
- It is logically defined by the following components: .
- **Hyper calls interface**- This is the entry point for all the partitions for the execution of sensitive instructions.
- **Memory service routines (MSRs) -** These are the set of functionalities that control the memory and its access from partitions
- **Advanced programmable interrupt controller (APIC)** -

Microsoft Hyper-V

- Scheduler
- Address manager.
- Partition manager.
- The hypervisor runs in Ring -1 and therefore requires corresponding hardware technology that enables such a condition
- .The hypervisor can support legacy operating systems that have been designed for x86 hardware

Microsoft Hyper-V

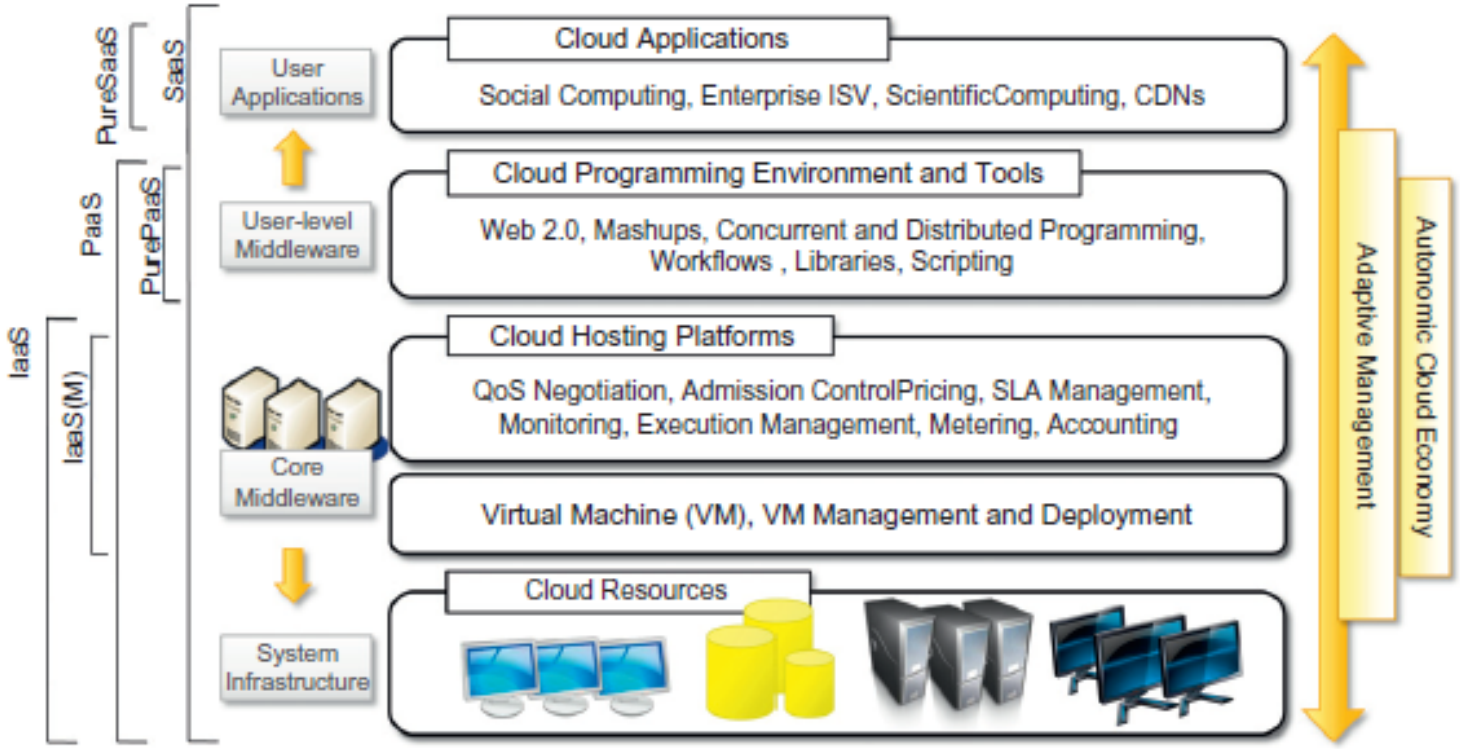
- Another component that provides advanced management virtual machines is System Center Virtual Machine Manager (SCVMM)
- SCVMM complements the basic features offered by Hyper-V
  - Management portal for the creation and management of virtual instances
  - Virtual to Virtual (V2V) and Physical to Virtual (P2V) conversions
  - Delegated administration •
  - Library functionality and deep PowerShell integration •
  - Intelligent placement of virtual machines in the managed environment
  - Host capacity management

Microsoft Hyper-V - Observations

- Compared with Xen and VMware, Hyper-V is a hybrid solution because it leverages both para virtualization techniques and full hardware virtualization
- The advantages reside in a flexible virtualization platform supporting a wide range of guest operating systems
- The disadvantages are represented by both hardware and software requirements.
- Hyper-V is compatible only with Windows Server
- Hyper-V is a role that can be installed on a existing operating system, while vSphere and Xen can be installed on the bare hardware

Cloud Computing Architecture

- The cloud reference model
- Architecture



Architecture

- Cloud infrastructure can be heterogeneous in nature because a variety of resources, such as clusters and even networked PCs, can be used to build it
- The physical infrastructure is managed by the core middleware, the objectives of which are to provide an appropriate runtime environment for applications and to best utilize resources
- At the bottom of the stack, virtualization technologies are used to guarantee runtime environment customization, application isolation, sandboxing, and quality of service.
- Hardware virtualization is most commonly used at this level.
- Hypervisors manage the pool of resources and expose the distributed infrastructure as a collection of virtual machines.

Architecture

- The combination of cloud hosting platforms and resources is generally classified as a Infrastructure-as-a-Service(IaaS) solution.
- IaaS into two two categories: Some of them provide both the management layer and the physical infrastructure; others provide only the management layer(IaaS (M))

Table 4.1 Cloud Computing Services Classification

Category	Characteristics	Product Type	Vendors and Products
SaaS	Customers are provided with applications that are accessible anytime and from anywhere.	Web applications and services (Web 2.0)	SalesForce.com (CRM) Clarizen.com (project management) Google Apps
PaaS	Customers are provided with a platform for developing applications hosted in the cloud.	Programming APIs and frameworks Deployment systems	Google AppEngine Microsoft Azure Manjrasoft Aneka Data Synapse
IaaS/HaaS	Customers are provided with virtualized hardware and storage on top of which they can build their infrastructure.	Virtual machine management infrastructure Storage management Network management	Amazon EC2 and S3 GoGrid Nirvanix

Infrastructure as a Service Reference implementation

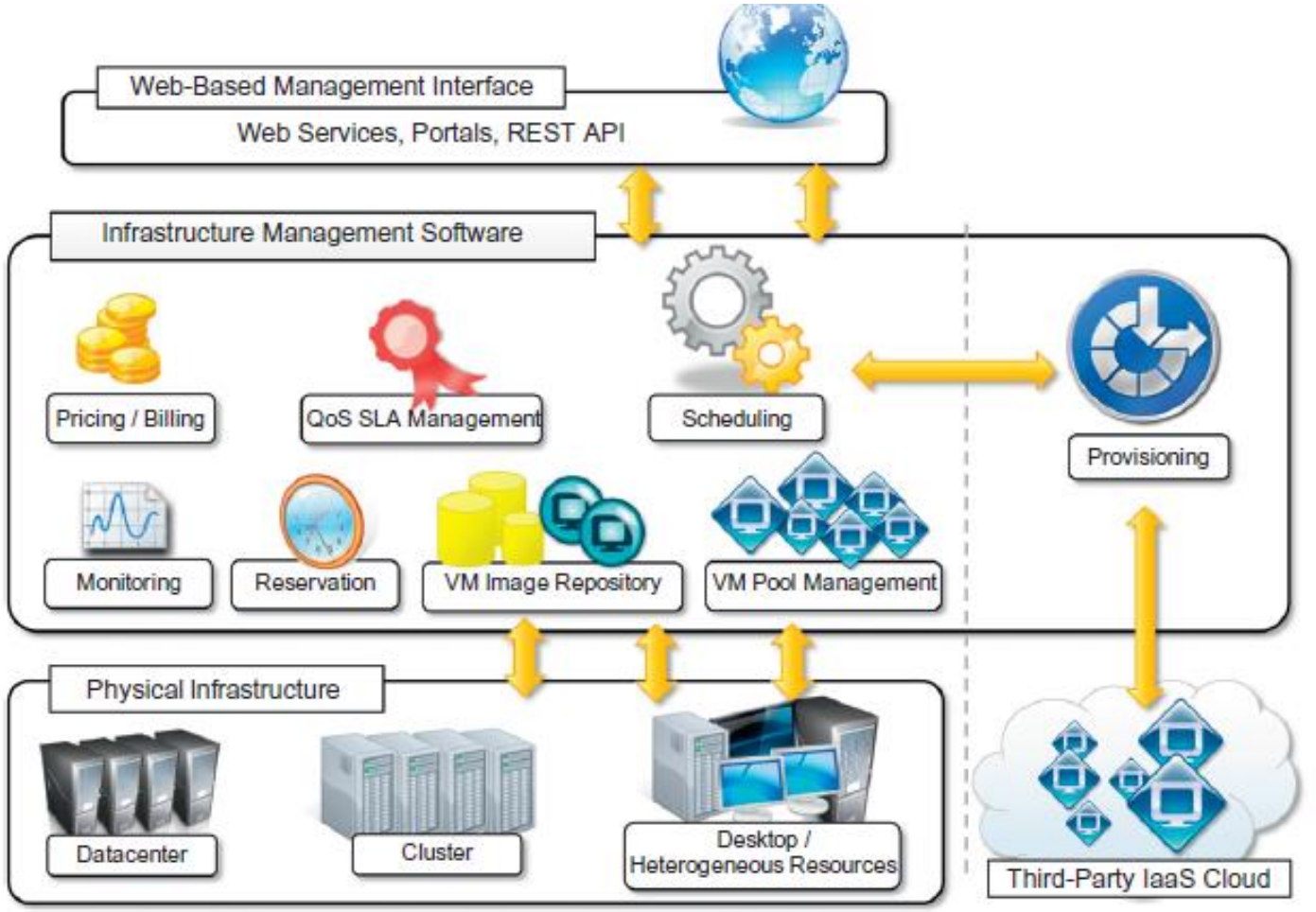


FIGURE 4.2

Infrastructure-as-a-Service reference implementation.

Platform as a Service Reference Model

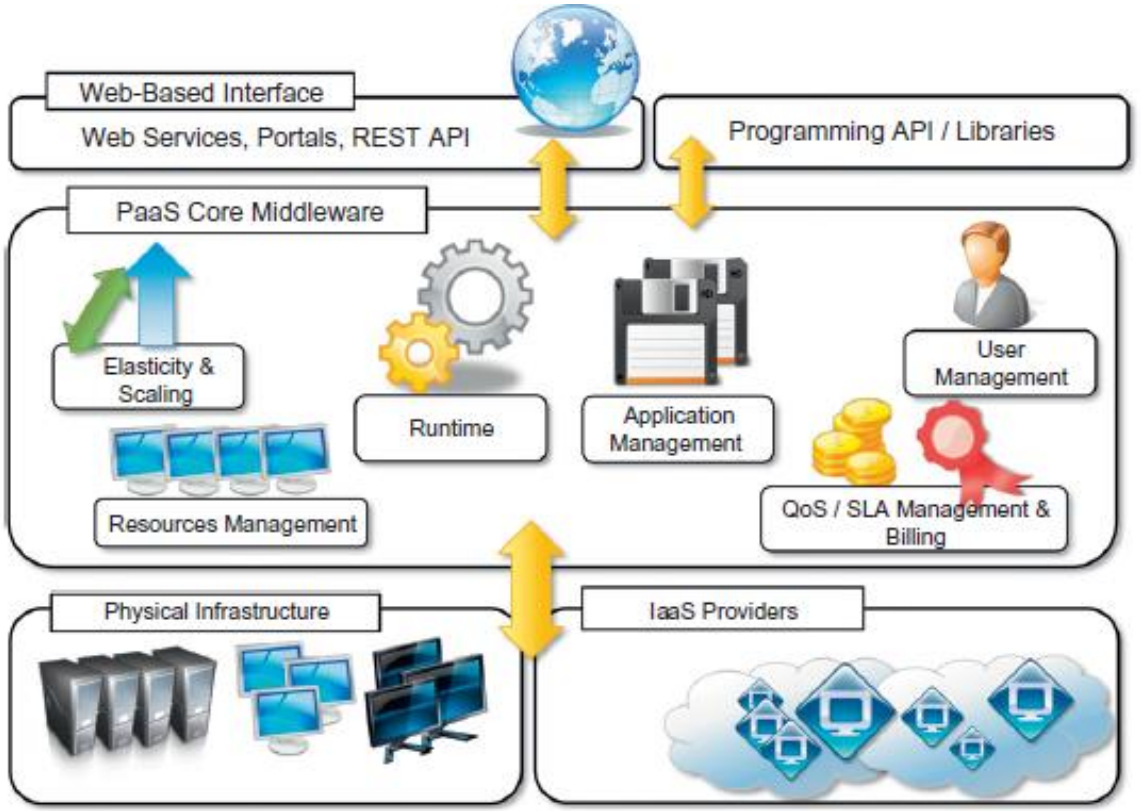


FIGURE 4.3

The Platform-as-a-Service reference model.

Platform as a Service Reference Model

Table 4.2 Platform-as-a-Service Offering Classification

Category	Description	Product Type	Vendors and Products
PaaS-I	Runtime environment with Web-hosted application development platform. Rapid application prototyping.	Middleware + Infrastructure Middleware + Infrastructure	Force.com Longjump
PaaS-II	Runtime environment for scaling Web applications. The runtime could be enhanced by additional components that provide scaling capabilities.	Middleware + Infrastructure Middleware Middleware + Infrastructure Middleware + Infrastructure Middleware + Infrastructure Middleware	Google AppEngine AppScale Heroku Engine Yard Joyent Smart Platform GigaSpaces XAP
PaaS-III	Middleware and programming model for developing distributed applications in the cloud.	Middleware + Infrastructure Middleware Middleware Middleware Middleware Middleware	Microsoft Azure DataSynapse Cloud IQ Manjrasof Aneka Apprenda SaaSGrid GigaSpaces DataGrid

Software as a Service

- Software-as-a-Service (SaaS) is a software delivery model that provides access to applications through the Internet as a Web-based service.
- It provides a means to free users from complex hardware and software management by offloading such tasks to third parties, which build applications accessible to multiple users through a Web browser.
- In this scenario, customers neither need install anything on their premises nor have to pay considerable up-front costs to purchase the software and the required licenses.
- On the provider side, the specific details and features of each customer's application are maintained in the infrastructure and made available on demand

Types of Cloud

- Public clouds. The cloud is open to the wider public.
- Private clouds. The cloud is implemented within the private premises of an institution and generally made accessible to the members of the institution or a subset of them
- Hybrid or heterogeneous clouds.
- Community clouds - The cloud is characterized by a multi-administrative domain involving different deployment models (public, private, and hybrid), and it is specifically designed to address the needs of a specific industry.

Private Cloud

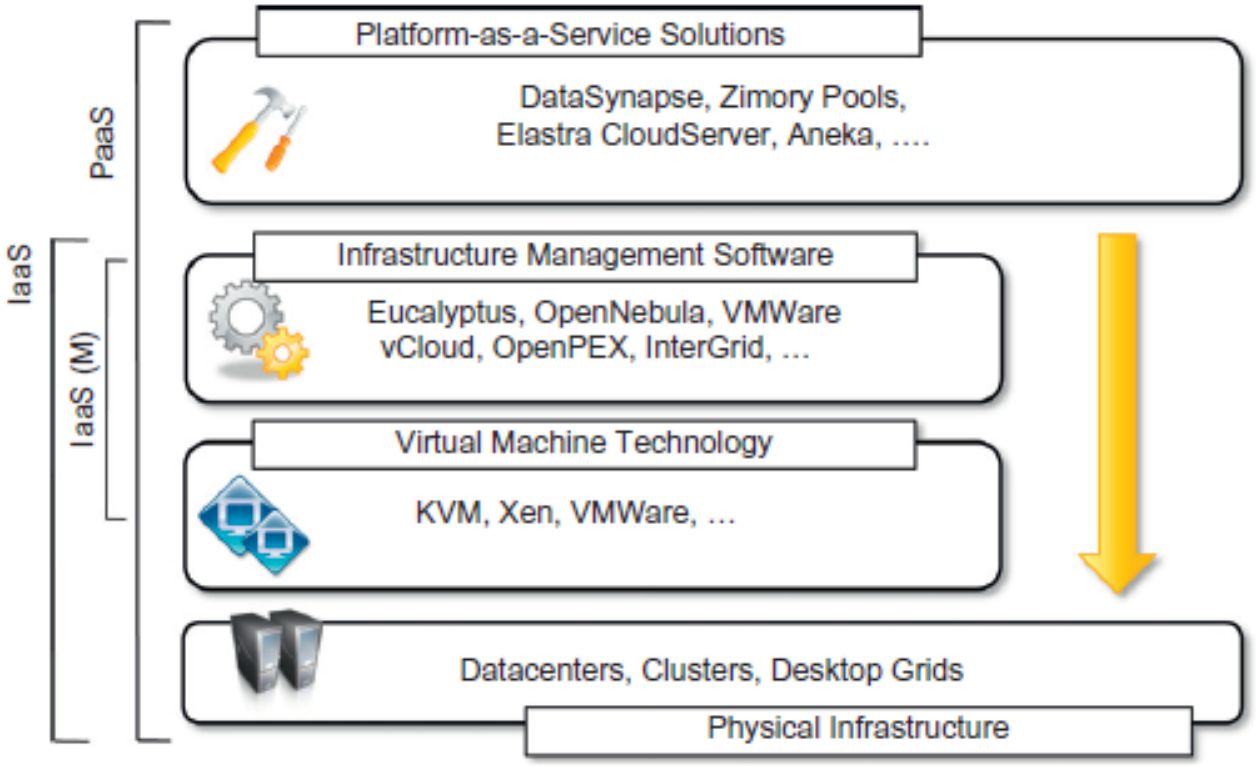


FIGURE 4.4

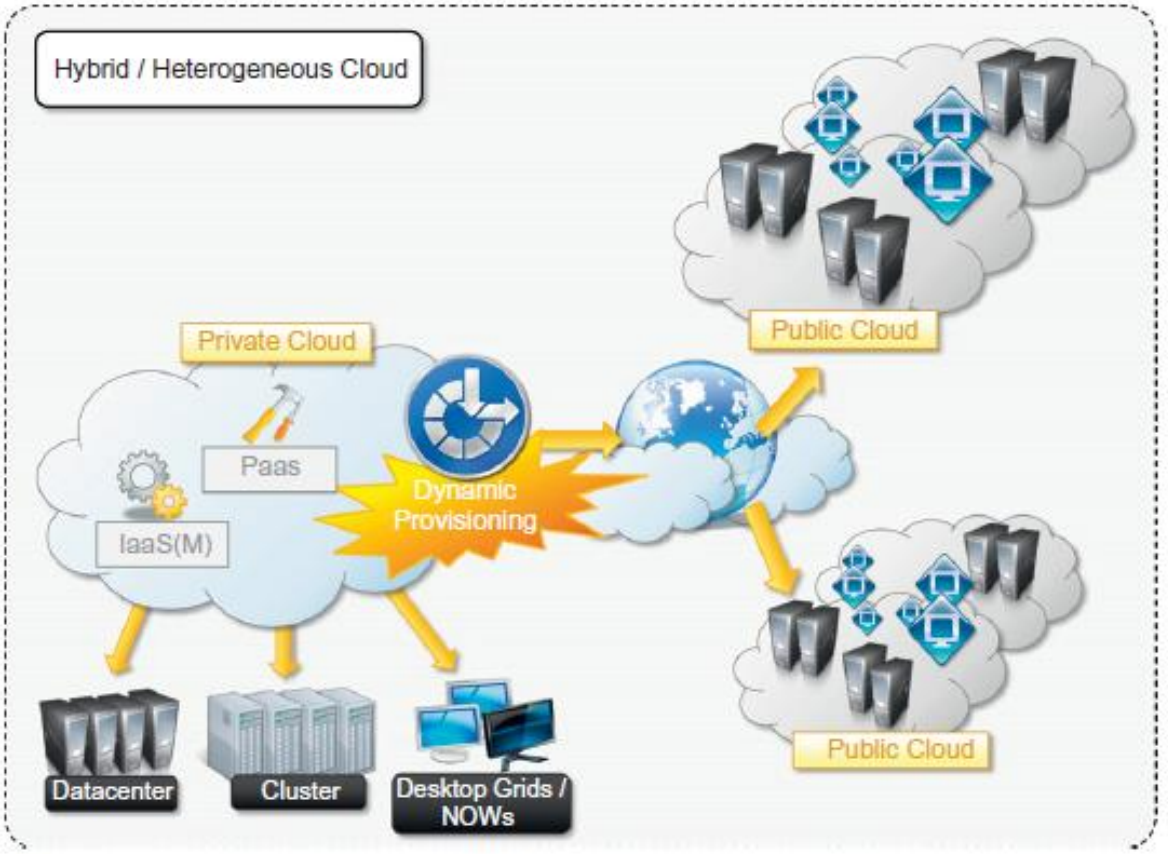
Private clouds hardware and software stack.

Private Cloud

- Key Advantages using Private Cloud
  - Customer information protection .
  - Infrastructure ensuring SLAs.
  - Compliance with standard procedures and operations.

.

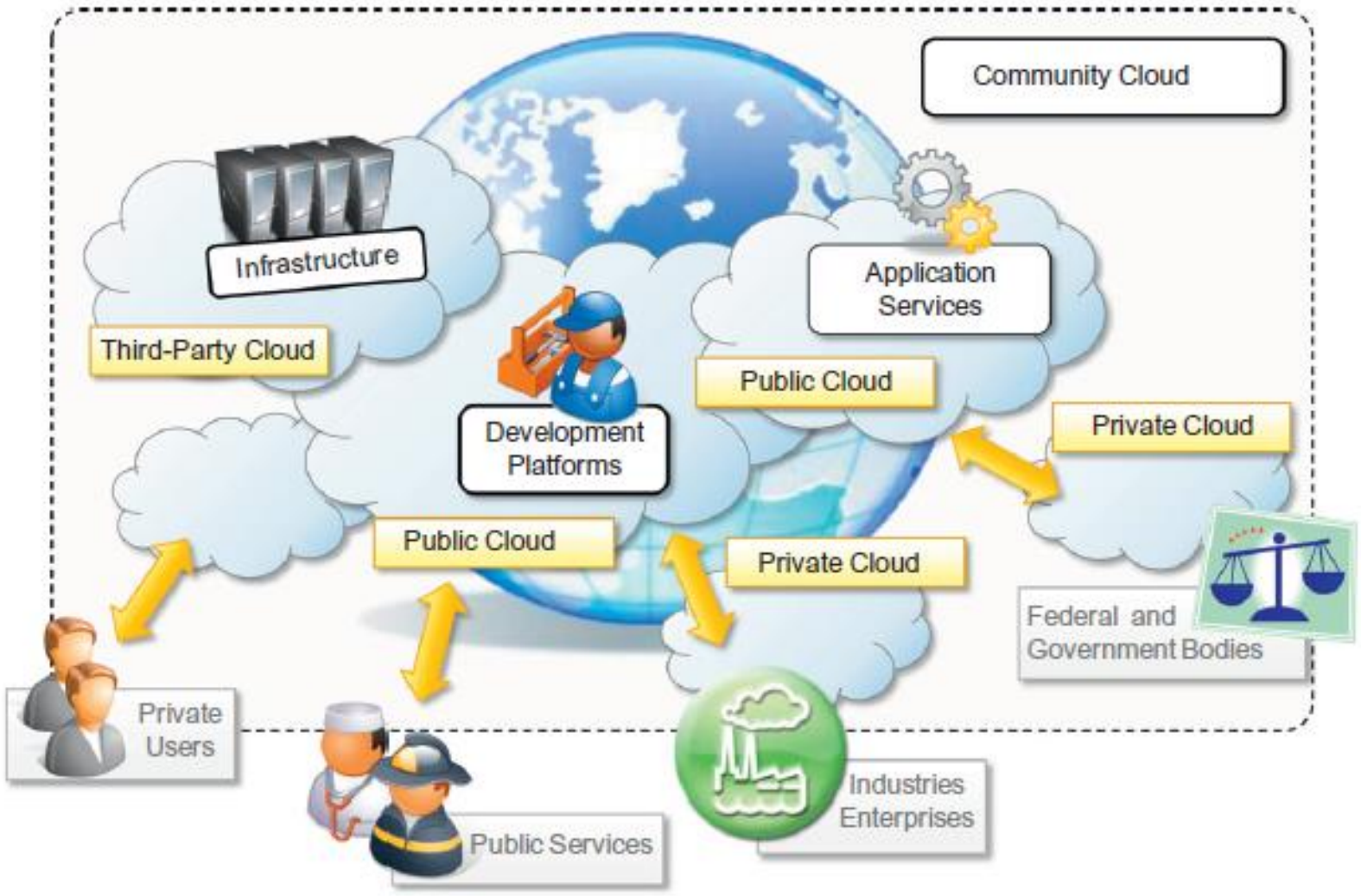
Hybrid Cloud



Community Cloud

- Community clouds are distributed systems created by integrating the services of different clouds to address the specific needs of an industry, a community, or a business sector
- NIST Define –
- The infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.

Community Cloud



Community Cloud

Candidate Sector for Community Cloud are-

- Media industry
- Healthcare industry
- Energy and other core industries
- Public sector
- Scientific research

Benefits –

- Openness
- Community.
- Graceful failures.
- Convenience and control

Economics of Cloud

- The main drivers of cloud computing are economy of scale and simplicity of software delivery and its operation.
- cloud computing allows:
  - Reducing the capital costs associated to the IT infrastructure
  - Eliminating the depreciation or lifetime costs associated with IT capital assets
  - Replacing software licensing with subscriptions
  - Cutting the maintenance and administrative costs of IT resources

Open Challenges

- Cloud definition
- Cloud interoperability and standards
- Scalability and fault tolerance
- Security, trust, and privacy
- Organizational aspects

Review questions (Virtualization)

- What is virtualization and what are its benefits?
- What are characteristics of virtualized environments?
- Discuss classification or taxonomy of virtualization at different levels.
- Discuss machine reference model of execution virtualization.
- What are hardware virtualization techniques?
- List and discuss different types of virtualization.
- What are benefits of virtualization in the context of Cloud computing?
- What are disadvantages or cons of virtualization?

References

- Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, **Mastering Cloud Computing**, McGraw Hill, ISBN-13: 978-1-25-902995-0, New Delhi, India, 2013.
  - Chapter 3- Virtualization
    - Section 3.1 to 3.5
  - This chapter slides text is compiled by:
    - Dr. Sounak Paul, BIT Mesra, Deoghar, India