Unit 10: Virtualization Concepts

CONTENTS

Objectives

After this lecture, you will be able to,

- Learn about virtualization concepts and the need for virtualization.
- Explore the features of virtualization and the working of virtualization in cloud.
- Know about the different types of virtualization.
- Understand the virtualized environments.
- Analyze the pros and cons of virtualization.

Introduction

In computing, virtualization or virtualisation is the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, storage devices, and computer network resources.Virtualization began in the 1960s, as a method of logically dividing the system resources provided by mainframe computers between different applications. Since then, the meaning of the term has broadened.Virtualization technology has transformed hardware into software. It allows to run multiple Operating Systems (OSs) as virtual machines (Figure 1).Each copy of an operating system is installed in to a virtual machine.

Figure 1: Virtualization Scenario

You can see a scenario over here that we have a VMware hypervisor that is also called as a Virtual Machine Manager (VMM). On a physical device, a VMware layer is installed out and, on that layer, we have six OSs that are running multiple applications over there, these can be the same kind of OSs or these can be the different kinds of OSs in it.

Why Virtualize

1. Share same hardware among independent users- Degrees of Hardware parallelism increases.
2. Reduced Hardware footprint through consolidation- Eases management and energy usage.
3. Sandbox/migrate applications- Flexible allocation and utilization.
4. Decouple applications from underlying Hardware- Allows Hardware upgrades without impacting an OS image.

Virtualization enables sharing of resources much easily, it helps in increasing the degree of hardware level parallelism, basically, there is sharing of the same hardware unit among different kinds of independent units, if we say that we have the same physical hardware and on that physical hardware, we have multiple OSs. There can be different users running on different kind of OSs. Therefore, we have a much more processing capability with us. This also helps in increasing the degree of hardware parallelism as well as there is a reduced hardware footprint throughout the VM consolidation. The hardware footprint that is overall hardware consumption also reduces out the amount of hardware that is wasted out that can also be reduced out. This consequently helps in easing out the management process and also to reduce the amount of energy that would have been otherwise consumed out by a particular hardware if we would have invested in large number of hardware machines would have been used otherwise. Virtualization helps in sandboxing capabilities or migrating different kinds of applications that in turn enables flexible allocations and utilization of the resources. Additionally, the decoupling of the applications from the underlying hardware is much easier and further aids in allowing more and more hardware upgrades without actually impacting any particular OS image.

Virtualization raises abstraction. Abstraction pertains to hiding of the inner details from a particular user. Virtualization helps in enhancing or increasing the capability of abstraction. It is very similar to how the virtual memory operates. It helps to access the larger address spaces physical memory mapping is actually hidden by an OS with the help of paging. It can be similar to hardware emulators where codes are allowed on one architecture to run on a different physical device such as virtual devices central processing unit, memory or network interface cards etc. No botheration is actually required out regarding the hardware details of a particular machine. The confinement to the excess of hardware details helps in raising out the abstraction capability through virtualization.

Basically, we have certain requirements for virtualization, first is the efficiency property. Efficiency means that all innocuous instructions are executed by the hardware independently. Then, the resource control property means that it is impossible for the programs to directly affect any kind of system resources. Furthermore, there is an equivalence property that indicates that we have a program which has a virtual machine manager or hypervisor that performs in a particular manner, indistinguishable from another program that is running on it.

Before and After Virtualization

Before virtualization, the single physical infrastructure was used to run a single OS and its applications, which results in underutilization of resources (Figure 2). The nonshared nature of the hardware forces the organizations to buy a new hardware to meet their additional computing needs. For example, if any organization wants to experiment or simulate their new idea, they have to use separate dedicated systems for different experiments. So, to complete their research work successfully, they tend to buy a new hardware that will increase the CapEx and OpEx. Sometimes, if the organization does not have money to invest more on the additional resources, they may not be able to carry out some valuable experiments because of lack of resources. So, people started thinking about sharing a single infrastructure for multiple purposes in the form of virtualization.

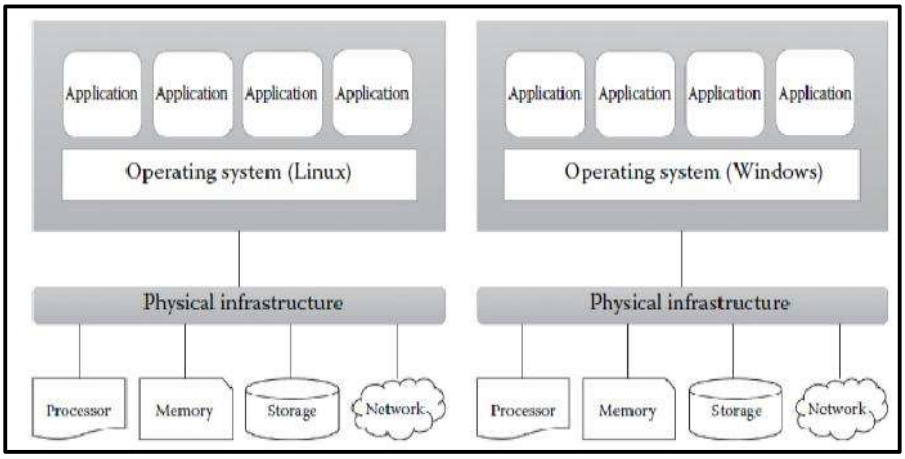


Figure 2: Before Virtualization

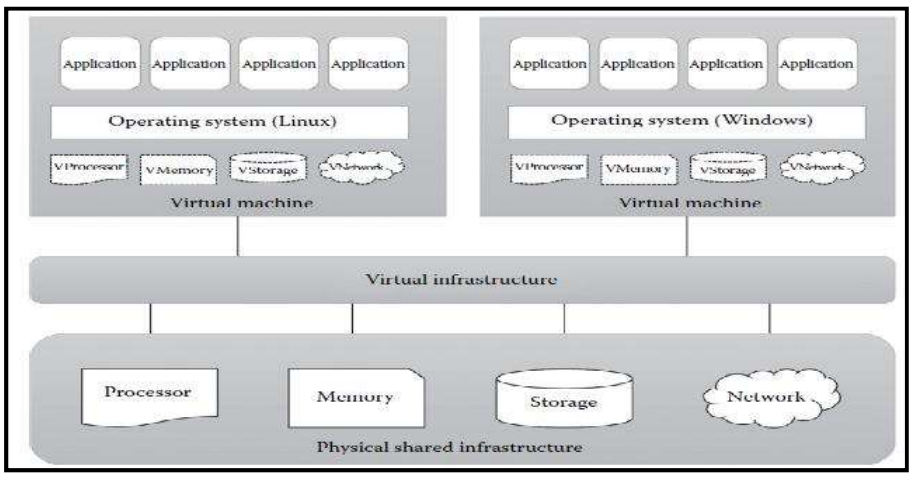


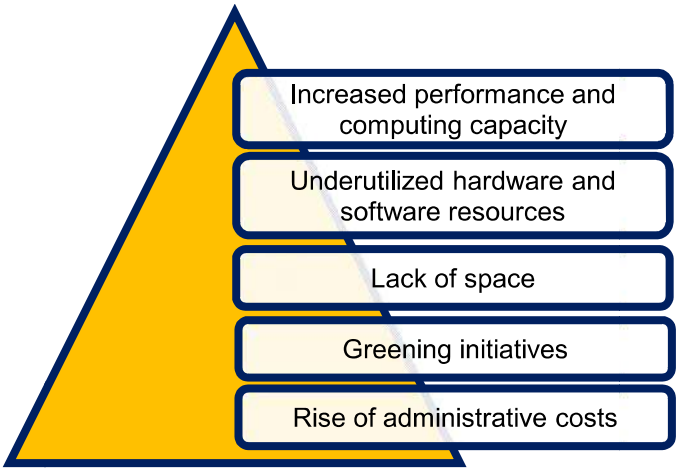
Figure 3: Post Virtualization Scenario

After virtualization was introduced, different OSs and applications were able to share a single physical infrastructure (Figure 3). The virtualization reduces the huge amount invested in buying additional resources. The virtualization becomes a key driver in the IT industry, especially in cloud computing. Generally, the terms cloud computing and virtualization are not same. There are significant differences between these two technologies.

Virtual Machine (VM):A VM involves anisolated guest OS installation within a normal host OS.From the user perspective, VM is software platform like physical computer that runs OSs and apps.VMs possess hardware virtually.

Factors Driving the Need of Virtualization

Increased Performance and Computing Capacity: PCs are having immense computing power. Nowadays, the average end-user desktop PC is powerful enough to meet almost all the needs of everyday computing, with extra capacity that Is rarely used. Almost all these PC share resources enough to host a VMM and execute a VM with by far acceptable performance. The same consideration applies to the high-end side of the PC market, where supercomputers can provide immense compute power That can accommodate the execution of hundreds or thousands of VMs.



Underutilized Hardware and Software Resources- Hardware and softwareunderutilization is occurring due to: increased performance and computing capacity, and the effect of limited or sporadic use of resources. The computers today are so powerful that in most cases only a fraction of their capacity is used by an application or the system. Moreover, if we consider the IT infrastructure of an enterprise, many computers are only partially utilized whereas they could be used without interruption on a 24/7/365 basis.For example, desktop PCs mostly devoted to office automation tasks and used by administrative staff are only used during work hours, remaining completely unused overnight. Using these resources for other purposes after hours could improve the efficiency of the IT infrastructure. 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 storage or compute power, makes data centers grow quickly. Companies such as Google and Microsoft expand their infrastructures by building data centers as large as football fields that are able to host thousands of nodes. Although this is viable for IT giants, in most cases enterprises cannot afford to build another data center to accommodate additional resource capacity. This condition, along with hardware under-utilization, hassled to the diffusion of a technique called server consolidation, for which virtualization technologies are fundamental.

Greening Initiatives: Recently, companies are increasingly looking for ways to reduce the amount of energy they consume and to reduce their carbon footprint. Data centers are one of the major power consumers; they contribute consistently to the impact that a company has on the environment. Maintaining a data center operation not only involves keeping servers on, but a great deal of energy is also consumed in keeping them cool. Infrastructures for cooling have a significant impact on the carbon footprint of a data center. 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: The power consumption and cooling costs have now become higher than the cost of IT equipment. Moreover, the increased demand for additional capacity, which translates into more servers in a data center, is also responsible for a significant increment in administrative costs. Computers—in particular, servers—do not operate all on their own, but they require care and feeding from system administrators. Common system administration tasks include hardware monitoring, defective hardware replacement, server setup and updates, server resources monitoring, and backups. These are labor-intensive operations, and the higher the number of servers that have to be managed, the higher the administrative costs. Virtualization can help reduce the number of required servers for a given workload, thus reducing the cost of the administrative personnel.

- Share same hardware among independent users.

  Degrees of hardware parallelism increases.

- Reduced hardware footprint through consolidation

  Eases management, energy usage.

- Sandbox/migrate applications

Flexible allocation & utilization.

- Decouple applications from underlying Hardware

  Allows hardware upgrades without impacting an OS image.

Features of Virtualization

- Virtualization Raises Abstraction
  - Similar to Virtual Memory: To access larger address space, physical memory mapping is hidden by OS using paging.
  - Similar to Hardware Emulators: Allows code on one architecture to run on a different physical device, such as, virtual devices, CPU, memory, NIC etc.
  - No botheration about the physical hardware details.
- Virtualization Requirements
  - Efficiency Property: All innocuous instructions are executed by the hardware.
  - Resource Control Property: It must be impossible for programs to directly affect system resources.
  - Equivalence Property: A program with a VMM performs in a manner indistinguishable from another.Except: Timing & resource availability.

Virtualized Environments

Virtualization is a broad concept that refers to the creation of a virtual version of something, whether hardware, a software environment, storage, or a network.In a virtualized environment, there are three major components (Figure 4):

- Guest: Represents the system component that interacts with the virtualization layer rather than with the host, as would normally happen.
- Host: Represents the original environment where the guest is supposed to be managed.
- Virtualization Layer: Responsible for recreating the same or a different environment where the guest will operate.

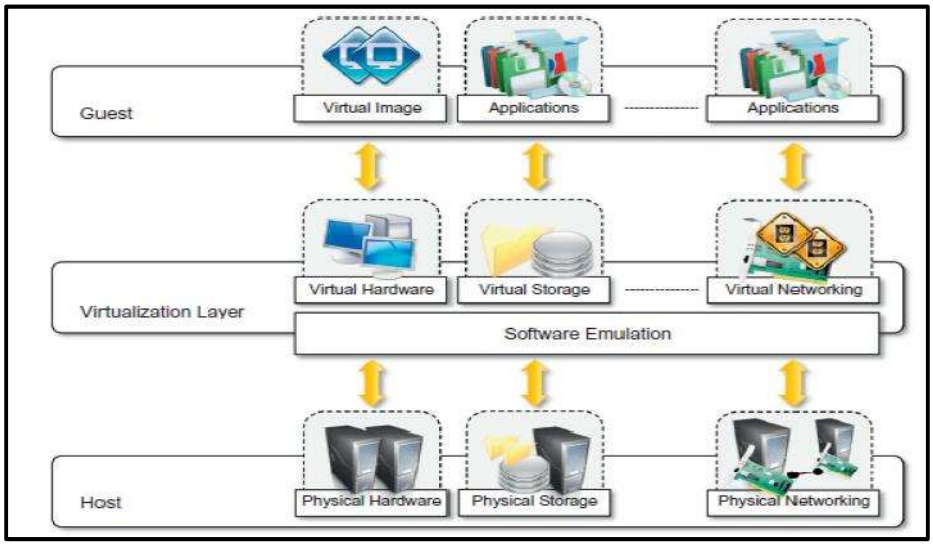


Figure 4:Virtualized Environment

The components of virtualized environments include: In the case of hardware virtualization, the guest is represented by a system image comprising an OS and installed applications. These are installed on top of virtual hardware that is controlled and managed by the virtualization layer, also called the VMM. The host is instead represented by physical hardware, & in some cases OS, that

defines an environment where VMM is running. The guest— Applications and users—interacts with a virtual network, such as a virtual private network (VPN), which is managed by specific software (VPN client) using physical network available on the node. VPNs are useful for creating an illusion of being within a different physical network & thus accessing the resources in it, which would otherwise not be available. The virtual environment is created by means of a software program. The ability to use software to emulate a wide variety of environments creates a lot of opportunities, previously less attractive because of excessive overhead introduced by the virtualization layer.

10.1 How Does Virtualization Work

For virtualizing the infrastructure, a virtualization layer is installed. It can involve the use of Bare-metal or Hosted Hypervisor architecture.It is important to understand how virtualization actually works. Firstly, in virtualization a virtual layer is installed on the systems. There are two prominent virtualization architectures, bare-metal and hosted hypervisor.

In a hosted architecture, a host OS is firstly installed out then a piece of software that is called as a hypervisor or it is called as a VM monitor or Virtual Machine Manager (VMM) (Figure 5). The VMM is installed on the top of host OS. The VMM allows the users to run different kinds of guests OSs within their own application window of a particular hypervisor. Different kinds of hypervisors can be Oracle's VirtualBox, Microsoft Virtual PC, VMware Workstation.

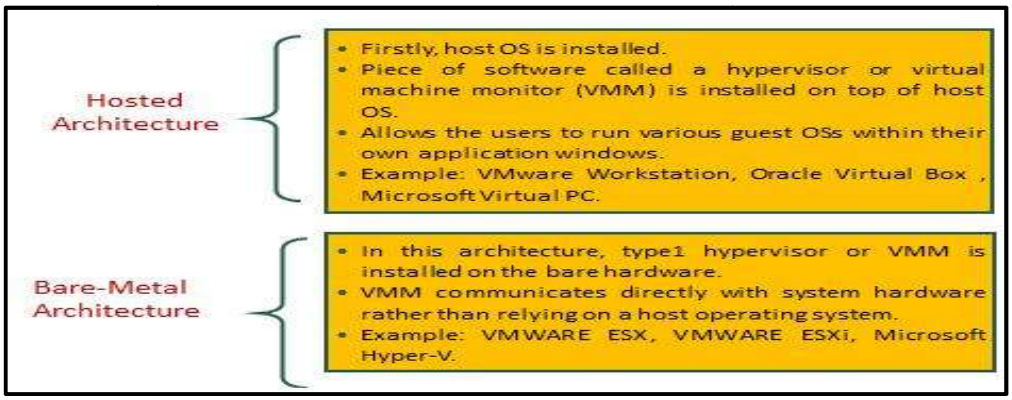


Figure 5: Hosted vs Bare-Metal Virtualization

Did you Know?

VMware server is a free application that is supported by Windows as well as by Linux OSs.

In a bare metal architecture, one hypervisor or VMM is actually installed on the bare metal hardware. There is no intermediate OS existing over here. The VMM communicates directly with the system hardware and there is no need for relying on any host OS. VMware ESXi and Microsoft Hyper-V are different hypervisors that are used for bare-metal virtualization.

A. Hosted Virtualization Architecture

A hosted virtualization architecture requires an OS (Windows or Linux) installed on the computer. The virtualization layer is installed as application on the OS.

Figure 6illustrates the hosted virtualization architecture. At the lower layer, we have the shared hardware with a host OS running on this shared hardware. Upon the host OS, a VMM is running that and is creating a virtual layer which is enabling different kinds of OSs to run concurrently. So, you can see a scenario we have a hardware then we add an operating system then a hypervisor is added and different kinds of virtual machines can run on that particular virtual layer and each virtual machine can be running same or different kind of OSs.

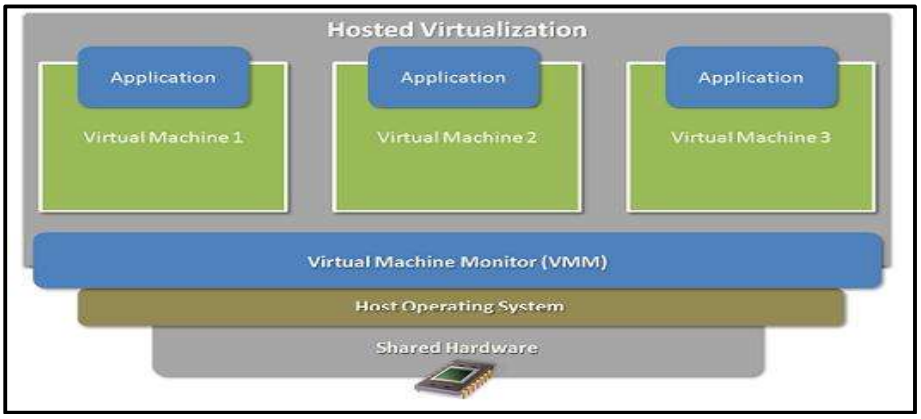


Figure 6: Hosted Virtualization Architecture

Advantages of Hosted Architecture

- Ease of installation and configuration
- Unmodified Host OS & Guest OS
- Run on a wide variety of PCs

Disadvantages of Hosted Architecture

- Performance degradation
- Lack of support for real-time OSs

B. Bare-Metal Virtualization Architecture

In a bare metal architecture, there is an underlying hardware but no underlying OS. There is just a VMM that is installed on that particular hardware and on that there are multiple VMs that are running on a particular hardware unit. As illustrated in theFigure 7, there is shared hardware that is running a VMM on which multiple VMs are running with simultaneous execution of multiple OSs.

Advantages of Bare-Metal Architecture

- Improved I/O performance
- Supports Real-time OS

Disadvantages of Bare-Metal Architecture

- Difficult to install & configure
- Depends upon hardware platform

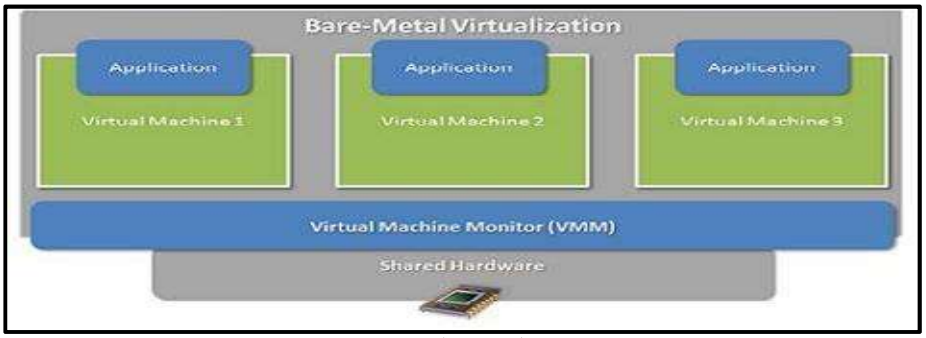


Figure 7: Bare-Metal Virtualization Scenario

10.2 Types of Virtualization

Virtualization covers a wide range of emulation techniques that are applied to different areas of computing. A classification of these techniques helps us better understand their characteristics and

use.Before discussing virtualization techniques, it is important to know about protection rings in OSs. The protection rings are used to isolate the OS from untrusted user applications. The OS can be protected with different privilege levels (Figure 8).

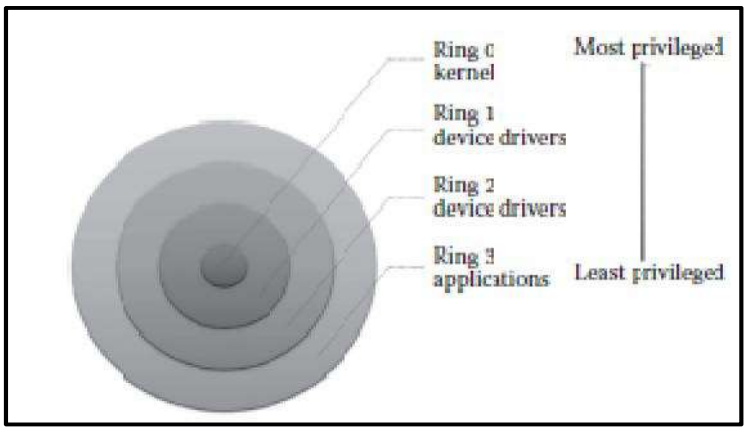


Figure 8: Protection Rings in OSs

Protection Rings in OSs

In protection ring architecture, the rings are arranged in hierarchical order from ring 0 to ring 3. The Ring 0 contains the programs that are most privileged, and ring 3 contains the programs that are least privileged. Normally, the highly trusted OS instructions will run in ring 0, and it has unrestricted access to physical resources. Ring 3 contains the untrusted user applications, and it has restricted access to physical resources. The other two rings (ring 1 & 2) are allotted for device drivers. 26 The protection ring architecture restricts the misuse of resources and malicious behavior of untrusted user-level programs. For example, any user application from ring 3 cannot directly access any physical resources as it is the least privileged level. But the kernel of the OS at ring 0 can directly access the physical resources as it is the most privileged level.Depending on the type of virtualization, the hypervisor and guest OS will run in different privilege levels. Normally, the hypervisor will run with the most privileged level, and the guest OS will run at the least privileged level than the hypervisor. There are 4 virtualization techniques namely,

- Full Virtualization (Hardware Assisted Virtualization/ Binary Translation).
- Para Virtualization or OS assisted Virtualization.
- Hybrid Virtualization
- OS level Virtualization

Full Virtualization: The VM simulates hardware to allow an unmodified guest OS to be run in isolation. There are 2 type of full virtualizations in the enterprise market, software-assisted and hardware-assisted full virtualization. In both the cases, the guest OSs source information is not modified.

The software-assisted full virtualization is also called as Binary Translation (BT) and it completely relies on binary translation to trap and virtualize the execution of sensitive, non-virtualizable instructions sets. It emulates the hardware using the software instruction sets. It is often criticized for performance issue due to binary translation. The software that fall under software-assisted (BT) include:

- VMware workstation (32Bit guests)
- Virtual PC
- VirtualBox (32-bit guests)
- VMware Server

The hardware-assisted full virtualization eliminates the binary translation and directly interrupts with hardware using the virtualization technology which has been integrated on X86 processors since 2005 (Intel VT-x and AMD-V). The guest OS's instructions might allow a virtual context execute privileged instructions directly on the processor, even though it is virtualized. There is

several enterprise software that support hardware-assisted– Full virtualization which falls under hypervisor type 1 (Bare metal) such as:

- VMware ESXi /ESX
- KVM
- Hyper-V
- Xen

Para Virtualization: The para-virtualization works differently from the full virtualization. It doesn't need to simulate the hardware for the VMs. The hypervisor is installed on a physical server (host) and a guest OS is installed into the environment. The virtual guests are aware that it has been virtualized, unlike the full virtualization (where the guest doesn't know that it has been virtualized) to take advantage of the functions. Also, the guest source codes can be modified with sensitive information to communicate with the host. The guest OSs require extensions to make API calls to the hypervisor.

Comparatively, in the full virtualization, guests issue hardware calls but in para virtualization, guests directly communicate with the host (hypervisor) using the drivers. The list of products which supports para virtualization are:

- Xen (Figure 9)
- IBM LPAR
- Oracle VM for SPARC (LDOM)
- Oracle VM for X86 (OVM)

However, due to the architectural difference between windows-based and Linux-based Xen hypervisor, Windows OS can't be para-virtualized. It does for Linux guest by modifying the kernel. VMware ESXi doesn't modify the kernel for both Linux and Windows guests.

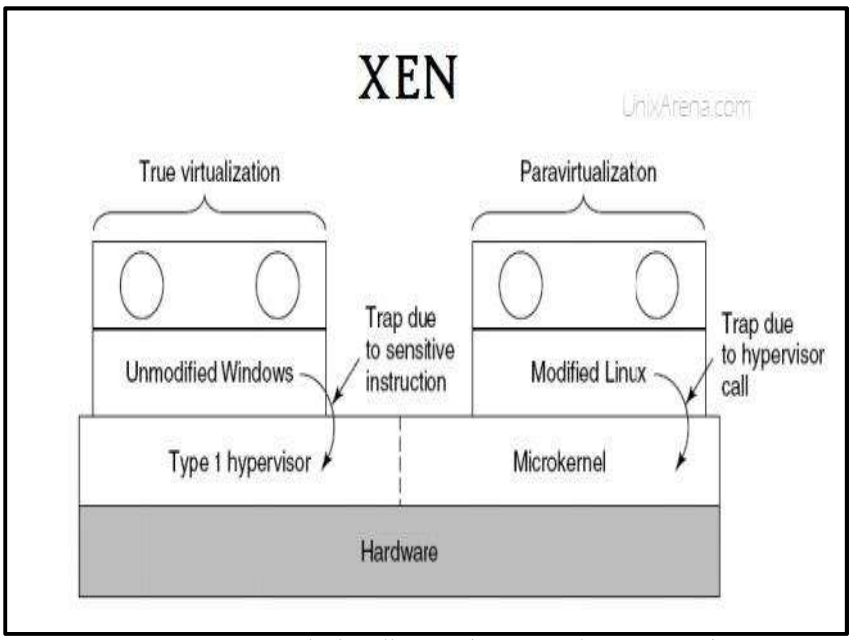


Figure 9: Xen Supports both Full-Virtualization and Para-Virtualization

Hybrid Virtualization (Hardware Virtualized with PV Drivers): In the hardware-assisted full virtualization, the guest OSs are unmodified and many VM traps occur and thus high CPU overheads which limit the scalability. Para virtualization is a complex method where guest kernel needs to be modified to inject the API. Therefore, due to the issues in full- and para- virtualization, engineers came up with hybrid paravirtualization, that is, a combination of both full and paravirtualization. The VM uses paravirtualization for specific hardware drivers (where there is a bottleneck with full virtualization, especially with I/O & memory intense workloads), and host uses full virtualization for other features. The following products support hybrid virtualization:

- Oracle VM for x86

- Xen
- VMware ESXi

OS Level Virtualization: It is widely used and is also known as "containerization". The host OS kernel allows multiple user spaces aka instance. Unlike other virtualization technologies, there is very little or no overhead since it uses the host OS kernel for execution. Oracle Solaris zone is one of the famous containers in the enterprise market. The list of other containers:

- Linux LCX
- Docker
- AIX WPAR

Processor Virtualization: It allows the VMs to share the virtual processors that are abstracted from the physical processors available at the underlying infrastructure (Figure 10). The virtualization layer abstracts the physical processor to the pool of virtual processors that is shared by the VMs. The virtualization layer will be normally any hypervisors. But processor virtualization can also be achieved from distributed servers.

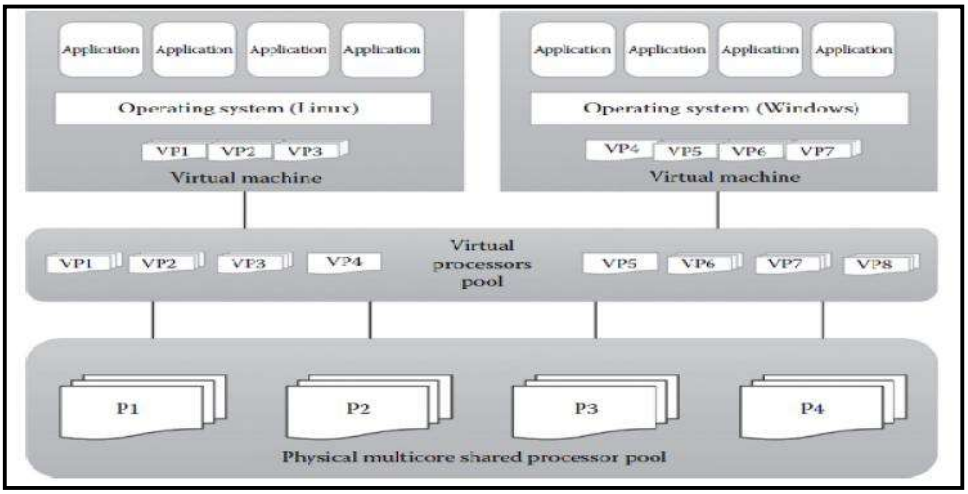


Figure 10: Processor Virtualization

Memory Virtualization: Another important resource virtualization technique is memory virtualization (Figure 11). It involves the process of providing a virtual main memory to the VMs is known as memory virtualization or main memory virtualization. In main memory virtualization, the physical main memory is mapped to the virtual main memory as in the virtual memory concepts in most of the OSs.

The main idea of main memory virtualization is to map the virtual page numbers to the physical page numbers. All the modern x86 processors are supporting main memory virtualization. The main memory virtualization can also be achieved by using the hypervisor software. Normally, in the virtualized data centers, the unused main memory of the different servers will consolidate as a virtual main memory pool and can be given to the VMs.

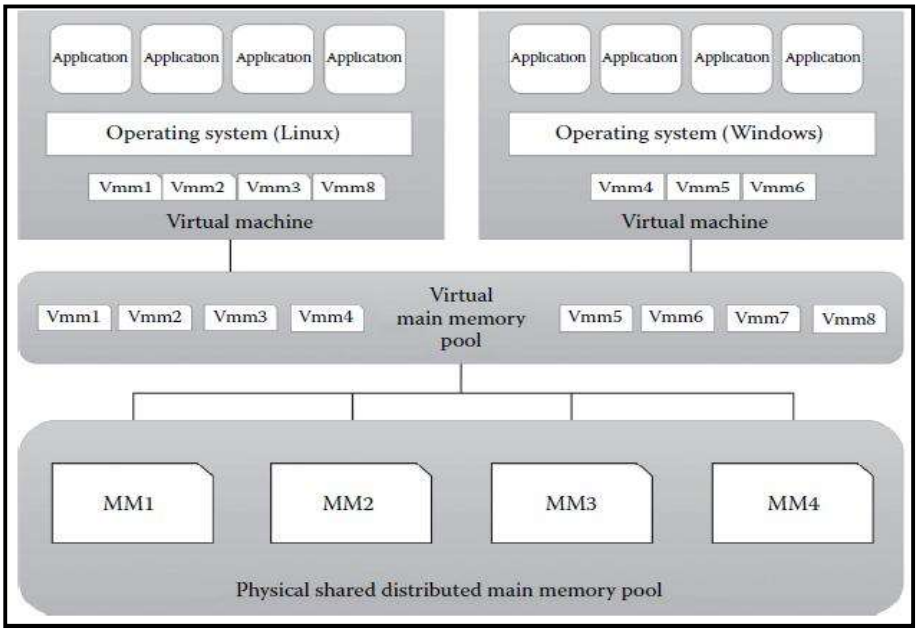


Figure 11: Memory Virtualization

Storage Virtualization: A form of resource virtualization where multiple physical storage disks are abstracted as a pool of virtual storage disks to the VMs (Figure 12). Normally, the virtualized storage will be called a logical storage.

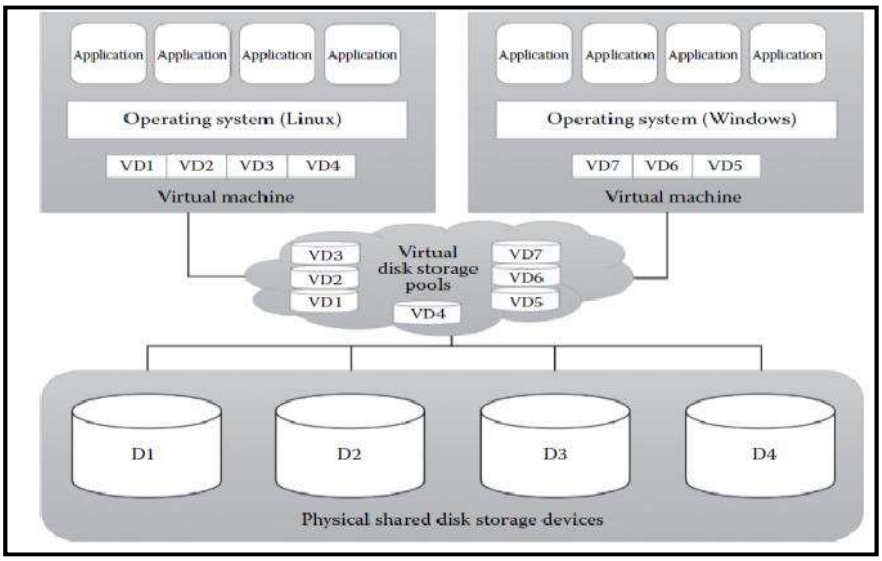


Figure 12: Storage Virtualization

Storage virtualization is mainly used for maintaining a backup or replica of the data that are stored on the VMs. It can be further extended to support the high availability of the data. It efficiently utilizes the underlying physical storage. Other advanced storage virtualization techniques are storage area networks (SAN) and network-attached storage (NAS).

Network Virtualization: It is a type of resource virtualization in which the physical network can be abstracted to create a virtual network (Figure 13).Normally, the physical network components like router, switch, and Network Interface Card (NIC) will be controlled by the virtualization software to provide virtual network components. Virtual network is a single software-based entity that contains the network hardware and software resources. Network virtualization can be achieved from internal network or by combining many external networks. It enables the communication between the VMs that share the physical network. There are different types of network access given to the VMs such as bridged network, network address translation (NAT), and host only.

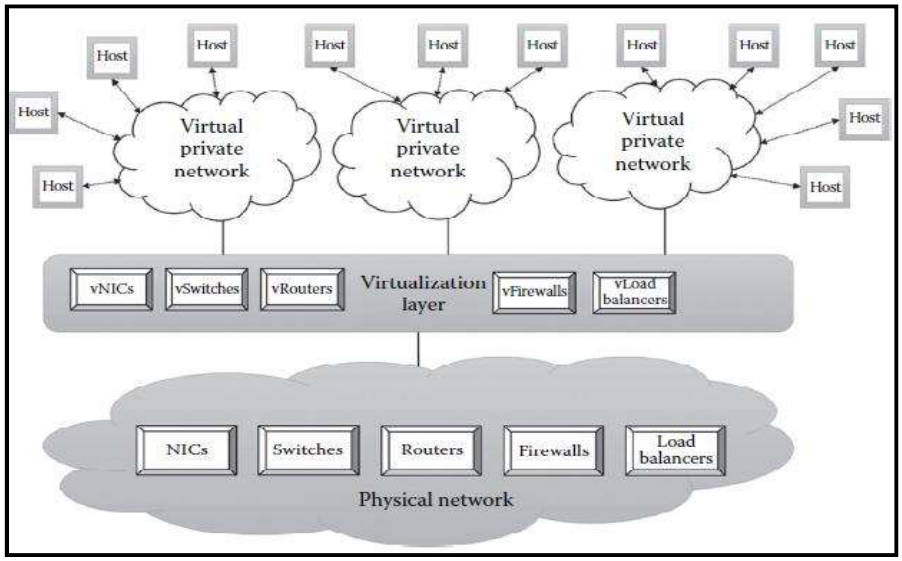


Figure 13: Network Virtualization

Data Virtualization: Data virtualization offers the ability to retrieve the data without knowing its type and the physical location where it is stored (Figure 14). It aaggregates the heterogeneous data from the different sources to a single logical/virtual volume of data. This logical data can be accessed from any applications such as web services, E-commerce applications, web portals, Software-as-a-Service (SaaS) applications, and mobile application.It hides the type of the data and the location of the data for the application that access it and ensures the single point access to data by aggregating data from different sources. It is mainly used in data integration, business intelligence, and cloud computing.

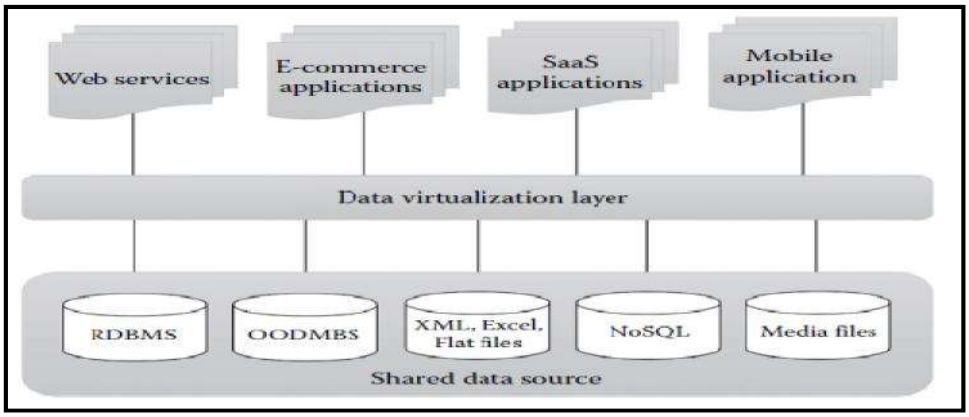


Figure 14: Data Virtualization

Application Virtualization: Application virtualization is the enabling technology for SaaS of cloud computing that offers the ability to the user to use the application without the need to install any software or tools in the machine (Figure 15). The complexity of installing the client tools or other supported software is reduced. Normally, the applications will be developed and hosted in the central server. The hosted application will be again virtualized, and the users will be given the separated/isolated virtual copy to access.

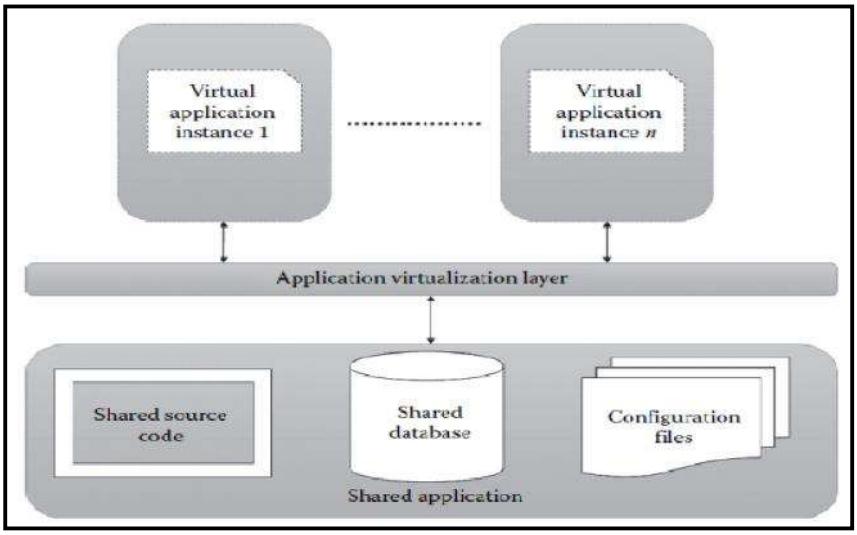


Figure 15: Application Virtualization

10.3 Pros of Virtualization

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. VM represents an emulated environment in which the guest is executed. All the operations of the guest are generally performed against the VM, which then translates & applies them to the host.By default, the file system exposed by the virtual computer is completely separated from the one of the host machines. This becomes the perfect environment for running applications without affecting other users in the environment.

Managed Execution

Virtualization of the execution environment not only allows increased security, but a wider range of features also can be implemented such as:

- Sharing: Virtualization allows creation of a separate computing environments within same host, thereby, making it possible to fully exploit capabilities of a powerful guest (that would otherwise be underutilized).
- Aggregation: 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, with a classical example represented by cluster management software, which harnesses the physical resources of a homogeneous group of machines and represents them as a single resource.
- Emulation: Guest programs 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 the guests.
- Isolation: Virtualization allows providing guests—whether they are OSs, applications, or other entities—with a completely separate environment, in which they are executed. The guest program performs its activity by interacting with an abstraction layer, which provides access to the underlying resources.

Portability

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 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 straight forward: One version of the application, in most cases, is able to run on different platforms with no changes.
- Portability allows having your own system always with you and ready to use as long as the required VMM is available. This requirement is, in general, less stringent than having all the applications and services you need available to you anywhere you go.

More Efficient Use of Resources

Multiple systems can securely coexist and share the resources of the underlying host, without interfering with each other. This is a prerequisite for server consolidation, which allows adjusting the number of active physical resources dynamically according to the current load of the system, thus creating the opportunity to save in terms of energy consumption and to be less impacting on the environment.

10.4 Cons of Virtualization

Virtualization also has downsides. The most evident is represented by a performance decrease of guest systems as a result of the intermediation performed by the virtualization layer. In addition, sub-optimal use of the host because of the abstraction layer introduced by virtualization management software can lead to a very inefficient utilization of the host or a degraded user experience.Less evident, but perhaps more dangerous, are the implications for security, which are mostly due to the ability to emulate a different execution environment.

Performance Degradation- Performance is definitely one of the major concerns in using virtualization technology. Since virtualization interposes an abstraction layer between the guest and the host, the guest can experience increased latencies. For instance, in case of hardware virtualization, where the intermediate emulates a bare machine on top of which an entire system can be installed, the causes of performance degradation can be traced back to the overhead introduced by following activities:

- Maintaining the status of virtual processors
- Support of privileged instructions (trap and simulate privileged instructions)
- Support of paging within VM
- Console functions

Inefficiency and Degraded User Experience- Virtualization can sometime lead 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 inaccessible. In the case of hardware virtualization, this could happen for device drivers: VM can sometime simply provide a default graphic card that maps only a subset of the features available in the host. In the case of programming-level VMs, some of the features of the underlying OSs may become inaccessible unless specific libraries are used. For example, in the first version of Java the support for graphic programming was very limited and the look and feel of applications was very poor compared to native applications. These issues have been resolved by providing a new framework called Swing for designing the user interface, and further improvements have been done by integrating support for the OpenGL libraries in the software development kit.

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 completely transparent manner led the way to malicious programs that are designed to extract sensitive information from the guest. In the case of hardware virtualization, malicious programs can preload themselves before the operating system and act as a thin virtual machine manager toward it. The operating system is then controlled and can be manipulated to extract sensitive information of interest to third parties.

Software Licensing Considerations- This is becoming less of a problem as more software vendors adapt to the increased adoption of virtualization, but it is important to check with your vendors to clearly understand how they view software use in a virtualized environment.

Possible Learning Curve- Implementing and managing a virtualized environment will require IT staff with expertise in virtualization. On the user side a typical virtual environment will operate similarly to the non-virtual environment. There are some applications that do not adapt well to the virtualized environment – this is something that your IT staff will need to be aware of and address prior to converting.

Summary

- Virtualization opens the door to a new and unexpected form of phishing. The capability of emulating a host in a completely transparent manner led the way to malicious programs that are designed to extract sensitive information from the guest.
- Virtualization raises abstraction. Abstraction pertains to hiding of the inner details from a particular user. Virtualization helps in enhancing or increasing the capability of abstraction.
- Virtualization enables sharing of resources much easily, it helps in increasing the degree of hardware level parallelism, basically, there is sharing of the same hardware unit among different kinds of independent units.
- In protection ring architecture, the rings are arranged in hierarchical order from ring 0 to ring 3. The Ring 0 contains the programs that are most privileged, and ring 3 contains the programs that are least privileged.
- In a bare metal architecture, one hypervisor or VMM is actually installed on the bare metal hardware. There is no intermediate OS existing over here. The VMM communicates directly with the system hardware and there is no need for relying on any host OS.
- The para-virtualization works differently from the full virtualization. It doesn't need to simulate the hardware for the VMs. The hypervisor is installed on a physical server (host) and a guest OS is installed into the environment.
- The software-assisted full virtualization is also called as Binary Translation (BT) and it completely relies on binary translation to trap and virtualize the execution of sensitive, non-virtualizable instructions sets.
- Memory virtualization is an important resource virtualization technique. In the main memory virtualization, the physical main memory is mapped to the virtual main memory as in the virtual memory concepts in most of the OSs.

Keywords

- ***Virtualization:*** Virtualization is a broad concept that refers to the creation of a virtual version of something, whether hardware, a software environment, storage, or a network.
- ***Hardware-assisted full virtualization:*** Hardware-assisted full virtualization eliminates the binary translation and directly interrupts with hardware using the virtualization technology which has been integrated on X86 processors since 2005.
- ***Data Virtualization:*** Data virtualization offers the ability to retrieve the data without knowing its type and the physical location where it is stored.
- ***Application Virtualization:*** Application virtualization is the enabling technology for SaaS of cloud computing that offers the ability to the user to use the application without the need to install any software or tools in the machine.
- ***Memory Virtualization:*** It involves the process of providing a virtual main memory to the VMs is known as memory virtualization or main memory virtualization.

- ***Network Virtualization:*** It is a type of resource virtualization in which the physical network can be abstracted to create a virtual network.

Self Assessment

1. _____ technology allows to run multiple operating systems as virtual machines.

A. Operating System
B. Virtualization
C. Fostering
D. Embellishment

2. In a _____ virtualization, the VMM communicates directly with system hardware rather than relying on a host operating system.

A. Bare-metal
B. Hybrid
C. Storage
D. Hosted

3. Which of the following is an example of Hosted virtualization hypervisors?

A. VMware Workstation
B. Oracle Virtual Box
C. Microsoft Virtual PC
D. All the above

4. Cloud computing is an abstraction based on the notion of pooling physical resources and presenting them as a _____ resource.

A. limited
B. restricted
C. virtual
D. homogeneous

5. _____ virtualization requires an operating system installed on the computer with the virtualization layer installed as an application on the OS.

A. Bare-metal
B. Hybrid
C. Storage
D. Hosted

6. CapEx stands for

A. Capacity expenses
B. Capital expenditure
C. Capital explosion
D. Capacity explosion

7. Which of the following is/are benefits of virtualizing the resources?

A. To increase the resource utilization
B. To increase the returns on investment

C. Ability to transform hardware into software
D. All of the above

8. What is most commonly used for managing the resources for every virtual system?
A. Hypervisor
B. Router
C. Cloud

9. Which is not a benefit of virtualization?
A. Run on single operating system
B. Flexible and efficient allocation of resources
C. Lowers the cost of IT infrastructure
D. Remote access and rapid scalability
10. Which of the following properties does not specify the virtualization requirements?
A. Efficiency property
B. Resource control property
C. Cluster property
D. Equivalence property

11. _____ are used to isolate the OS from untrusted user applications.
A. Usage rings
B. Protection rings
C. Privacy rings
D. Trustful rings

12. An operating system running on a Type _____ VM is full virtualization.
A. 1
B. 2
C. 3
D. 4

13. In _____ the virtual machine simulates hardware, so it can be independent of the underlying system hardware.
A. Paravirtualization
B. full virtualization
C. emulation
D. None of the above

14. The software that supports virtual machine is called?
A. VMM
B. Hypervisor
C. Kernal
D. Both A and B

15. Which of the following type of virtualization is also characteristic of cloud computing?
A. Storage
B. CPU

C. Application
D. All of the above

Answers for Self Assessment

1.	B	2.	A	3.	D	4.	C	5.	D
6.	B	7.	D	8.	A	9.	A	10.	C
11.	B	12.	A	13.	C	14.	D	15.	D

Review Questions

1. What are the pros and cons of virtualization?
2. Explain the concept of virtualization?
3. What are the different types of virtualization?
4. Write a short note on:
   (a) Para-virtualization
   (b) OS virtualization
   (c) Storage virtualization
5. Differentiate storage and data virtualization?
6. Discuss about virtualized environments?

Further Readings

- Mastering Cloud Computing by "Rajkumar Buyya, Christian Vecchiola, S ThamaraiSelvi, Tata McGraw-Hill Education, 2021.
- Cloud Computing: Concepts, Technology and Architecture by Thomas Erl, Pearson Education.
- Cloud Computing Black Book by Kailash Jayaswal,Jagannath Kallakurchi, Donald J. Houde, Deven Shah, Kogent Learning Solutions, DreamTech Press.
- Virtualization: A Manager's Guide by Dan Kusnetzky, O'Reilly, 2011.

Web Links

1. What is Virtualization? (tutorialspoint.com)
2. What is Virtualization? Definition from SearchServerVirtualization (techtarget.com)
3. Virtualization - Wikipedia
4. What is Virtualization? How does it Work? [Understanding Virtualization] (kuberty.io)
5. What is Virtualization? - Definition from Techopedia
6. What is virtualization? (redhat.com)
7. https://youtu.be/iBI31dmqSX0
8. https://youtu.be/Pl45CQYN3zI

Unit 11: Virtual Machine

CONTENTS

Objectives

After this lecture, you will be able to,

- Discover about Virtual Machine (VM)and their properties.
- Explore the different types of Virtual Machines.
- Understand the concept of interpretation and binary translation.
- Learn about hypervisors and their types.
- Know about different hypervisors such as: HLL VM: Xen, KVM, VMware, Virtual Box, Hyper-V.

Introduction

A software that creates a virtualized environment between the computer platform and the end-user in which the end user can operate software. It provides an interface identical to the underlying bare hardware. The Operating System (OS) creates the illusion of multiple processes, each executing on its own processor with its own (virtual) memory.Virtual machines are "an efficient, isolated duplicate of a real machine"- Popek and Goldberg. Popek and Goldberg introduced conditions for computer architecture to efficiently support system virtualization.

Virtual machine is a software that creates a virtualized environment between the computer platform and the end user in which the end user can operate software. The concept of virtualization applied to the entire machine involves:

- mapping of virtual resources or state to real resources.
- use of real machine instructions to carry out actions specified by the virtual machine instructions.
- Implemented by adding a layer of software to a real machine to support the desired VMs architecture.

VMs are a number of discrete identical execution environments on a single computer, each of which runs an OS (Figure 1). These allow applications written for one OS to be executed on a machine which runs a different OS which provide a greater level of isolation between processes than is achieved when running multiple processes on the same instance of an OS.

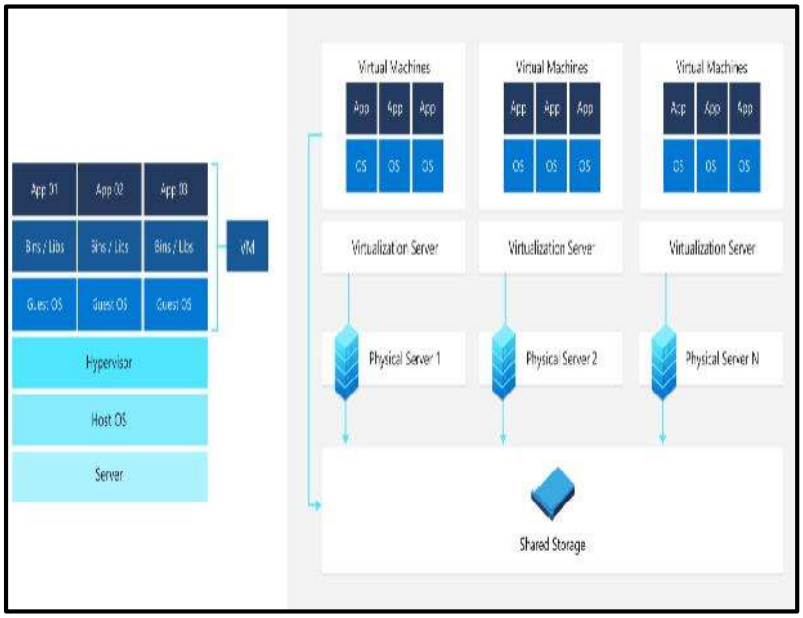


Figure 1: Virtual Machine Scenario

Virtual Machines: Virtual Computers Within Computers

VM is no different than any other physical computer like a laptop, smart phone or server. It has a CPU, memory, disks to store your files and can connect to the internet if needed. While the parts that make up your computer (called hardware) are physical and tangible, VMs are often thought of as virtual computers or software-defined computers within physical servers, existing only as code.

11.1 Virtualization

In computing, virtualization or virtualisation is the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, storage devices, and computer network resources.Virtualization began in the 1960s, as a method of logically dividing the system resources provided by mainframe computers between different applications. Since then, the meaning of the term has broadened.Virtualization technology has transformed hardware into software. It allows to run multiple Operating Systems (OSs) as virtual machines.Each copy of an operating system is installed in to a virtual machine.

Figure 2: Virtualization Scenario

You can see a scenario (Figure 2), we have a VMware hypervisor that is also called as a Virtual Machine Manager. On a physical device, a VMware layer is installed out and, on that layer, we have six OSs that are running multiple applications over there, these can be the same kind of OSs or these can be the different kinds of OSs in it.

Why Virtualize

1. Share same hardware among independent users- Degrees of Hardware parallelism increases.

2. Reduced Hardware footprint through consolidation- Eases management and energy usage.
3. Sandbox/migrate applications- Flexible allocation and utilization.
4. Decouple applications from underlying Hardware- Allows Hardware upgrades without impacting an OS image.

Virtualization enables sharing of resources much easily, it helps in increasing the degree of hardware level parallelism, basically, there is sharing of the same hardware unit among different kinds of independent units, if we say that we have the same physical hardware and on that physical hardware, we have multiple OSs. There can be different users running on different kind of OSs. Therefore, we have a much more processing capability with us. This also helps in increasing the degree of hardware parallelism as well as there is a reduced hardware footprint throughout the VM consolidation. The hardware footprint that is overall hardware consumption also reduces out the amount of hardware that is wasted out that can also be reduced out. This consequently helps in easing out the management process and also to reduce the amount of energy that would have been otherwise consumed out by a particular hardware if we would have invested in large number of hardware machines would have been used otherwise. Virtualization helps in sandboxing capabilities or migrating different kinds of applications that in turn enables flexible allocations and utilization of the resources. Additionally, the decoupling of the applications from the underlying hardware is much easier and further aids in allowing more and more hardware upgrades without actually impacting any particular OS image.

Virtualization raises abstraction. Abstraction pertains to hiding of the inner details from a particular user. Virtualization helps in enhancing or increasing the capability of abstraction. It is very similar to how the virtual memory operates. It helps to access the larger address spaces physical memory mapping is actually hidden by an OS with the help of paging. It can be similar to hardware emulators where codes are allowed on one architecture to run on a different physical device such as virtual devices central processing unit, memory or network interface cards etc. No botheration is actually required out regarding the hardware details of a particular machine. The confinement to the excess of hardware details helps in raising out the abstraction capability through virtualization.

Basically, we have certain requirements for virtualization, first is the efficiency property. Efficiency means that all innocuous instructions are executed by the hardware independently. Then, the resource control property means that it is impossible for the programs to directly affect any kind of system resources. Furthermore, there is an equivalence property that indicates that we have a program which has a virtual machine manager or hypervisor that performs in a particular manner, indistinguishable from another program that is running on it.

11.2 Virtual Machine Attributes

Virtual machine is a software that creates a virtualized environment between the computer platform and the end user in which the end user can operate software.The section below discusses the different characteristics for the same.

History of Virtual Machine

Virtualization was first introduced in the 1960s to allow partitioning of large, mainframe hardware.In the 1990s, researchers began to see how virtualization could solve some of the problems associated with the proliferation of less expensive hardware, including under-utilization, escalating management costs and vulnerability. The concept of a VM was introduced around 1960. The evolution of time-sharing technique, where each program has full access to all computer resources but at a time, only one program will be executed. The system switch between programs in time slices while saving and restoring program states each time. With the use of the time-sharing method, multiple users can use the computer system concurrently. IBM research centers evolved the time-sharing method as VMs. CP-67 was the first available VM architecture. Evidently, the systems with multiple VMs on a single host and single VM on multiple hosts were developed.

Properties of a Virtual Machine

Virtual Hardware

- Each VM has its own set of virtual hardware (e.g., RAM, CPU, NIC, etc.) upon which an operating system and applications are loaded.

- OS sees a consistent, normalized set of hardware regardless of the actual physical hardware components.

Partitioning

- Multiple applications and OSs can be supported within a single physical system.
- There is no overlap amongst memory as each Virtual Memory has its own memory space.

Isolation

- VMs are completely isolated from host machine and other VMs. If a VM crashes, all others are unaffected.
- Data does not leak across VMs.

Identical Environment

- VMs can have a number of discrete identical execution environments on a single computer, each of which runs an OS.

Other VM Features

- Each VM has its own set of virtual hardware (e.g., RAM, CPU, NIC, etc.) upon which an operating system and applications are loaded.
- OS sees a consistent, normalized set of hardware regardless of the actual physical hardware components.
- Host system resources are shared among the various VMs. For example, if a host system has 8GB memory where VMs are running, this amount will be shared by all the VMs, depending upon the size of the allocation.
- One of the best features of using Virtual machines is we can run multiple OSs/VMs in parallel on one host system.
- VMs are isolated from one another, thus secure from malware or threat from any other compromised VM running on the same host.
- Direct exchange of data and mutual influencing are prevented.
- Transfer of VMs to another system can be implemented by simply copying the VM data since the complete status of the system is saved in a few files.
- VMs can be operated on all physical host systems that support the virtualization environment used.

Virtual Machine Architecture

- Runtime software is the virtualization software that implements the Process VM. It is implemented at the API level of the computer architecture above the combined layer of OS and Hardware. This emulates the user-level instructions as well as OS or library calls.
- For the system VM, the virtualization software is called Virtual Machine Monitor(VMM).
- This software is present between the host hardware machine and the guest software.
- VMM emulates the hardware ISA allowing the guest software to execute a different ISA.

Virtual Machine Taxonomy

Figure 3 depicts the taxonomy for the virtual machines. Let us discuss each one of them below:

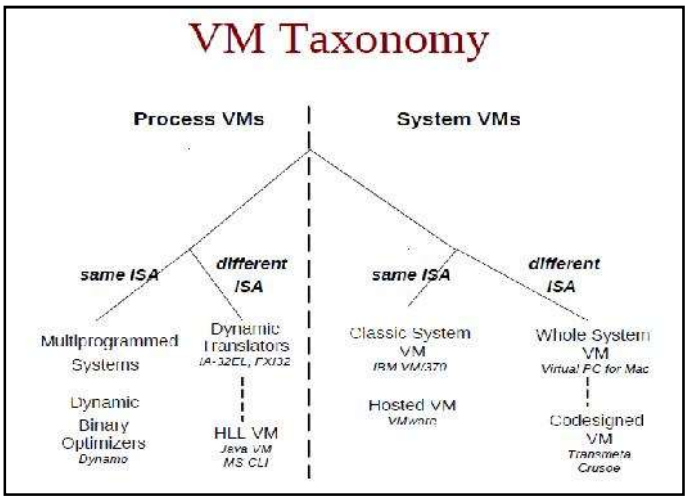


Figure 3: Taxonomy of Virtual Machines

Process Virtual Machines: These are also known as Application VM (Figure 4). The virtualization below the API or ABI, providing virtual resources to a single process executed on a machine is called as the process virtualization. It is created for the process alone, destroyed when process finishes.

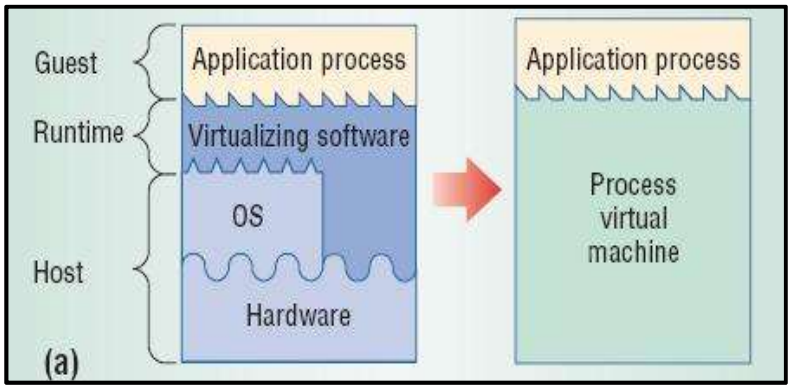


Figure 4: Process VM

Multiprogrammed Systems: Each application is given effectively separate access to resources, managed by the OS.

Emulators and Translators:

- Executes program binaries compiled for different instruction sets.
- Slower, requiring hardware interpretation.
- Optimization through storing blocks of converted code for repeated execution.

Optimizers, same ISA: Perform code optimization during translation and execution.

High-Level-Language VM:

- Cross-platform compatibility.
- Programs written for an abstract machine, which is mapped to real hardware through a VM.
  - Sun Micro systems Java VM
  - Microsoft Common Language Infrastructure, .NET framework.

System Virtual Machines: These correspond to the virtualized hardware below the ISA. The single host can run multiple isolated OSs (Figure 5). The servers running different OSs but in isolation between concurrent systems. The hardware managed by the Virtual Machine Manager (VMM).Classically, the

VMM runs on bare hardware, directly interacting with resources. It intercepts and interprets guest OS actions.

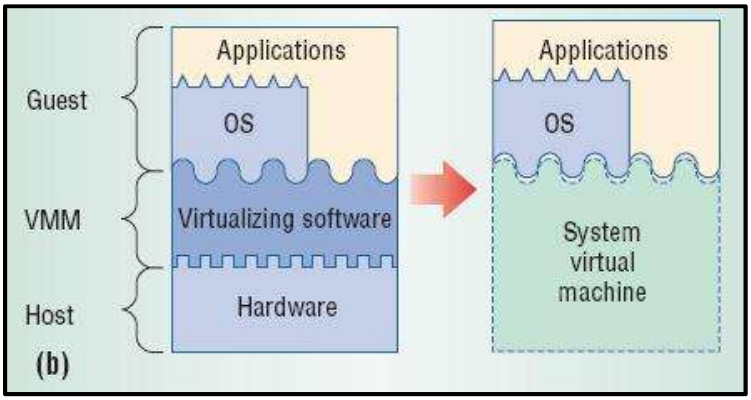


Figure 5: System Virtual Machines

Uses of Virtual Machines

- Building and deploying apps to the cloud.
- Trying out a new operating system (OS), including beta releases.
- Spinning up a new environment to make it simpler and quicker for developers to run dev-test scenarios.
- Backing up your existing OS.
- Accessing virus-infected data or running an old application by installing an older OS.
- Running software or apps on operating systems that they were not originally intended for.

Benefits of Virtual Machines

While VMs run like individual computers with individual operating systems and applications, they have the advantage of remaining completely independent of one another and the physical host machine. A piece of software called a hypervisor or virtual machine manager, lets you run different operating systems on different virtual machines at the same time. This makes it possible to run Linux VMs, for example, on a Windows OS or to run an earlier version of Windows on more current Windows OS. As the VMs are independent of each other, they are also extremely portable. You can move a VM on a hypervisor to another hypervisor on a completely different machine almost instantaneously.Because of their flexibility and portability, virtual machines provide many benefits, such as:

Cost Savings- Running multiple virtual environments from one piece of infrastructure means that you can drastically reduce your physical infrastructure footprint. This boosts your bottom line—decreasing the need to maintain nearly as many servers and saving on maintenance costs and electricity.

Agility and Speed- Spinning up a VM is relatively easy and quick and is much simpler than provisioning an entire new environment for your developers. Virtualisation makes the process of running dev-test scenarios a lot quicker.

Lowered Downtime- VMs are so portable and easy to move from one hypervisor to another on a different machine—this means that they are a great solution for backup, in the event the host goes down unexpectedly.

Scalability- VMs allow you to more easily scale your apps by adding more physical or virtual servers to distribute the workload across multiple VMs. As a result you can increase the availability and performance of your apps.

Security Benefits- Because VMs run in multiple OSs, using a guest operating system on a VM allows you to run apps of questionable security and protects your host OS. VMs also allow for better security forensics and are often used to safely study computer viruses, isolating the viruses to avoid risking their host computer.

Isolated environment provided by VMs- If you are a tester or security analyst then VMs will be a good idea to run multiple applications and services in an isolation using VMs because they do not affect each other.

Easy to Backup and Clone- All the VMs are stored on the physical hard drive of our host or physical machine in the file format. Thus, they can be easily back up, moved, or cloned in real-time is one of the popular benefits we get from running a virtual machine.

Faster Server Provisioning- VMs are easy to install, eliminating the cumbersome and time-consuming installation of applications on servers. For example, if you want a new server to run some application then it is very easy and fasts to deploy pre-configured VM templates instead of installing a new server OS on a physical machine. The same goes for cloning existing applications to try something new.

Beneficial in Disaster Recovery- As VM doesn't depend upon the underlying hardware, thus they are independent of the hardware or CPU model on which it is running. Hence, we can easily replicate VMs to cloud or offsite, so in some disaster situations, it would be easy to recover and get online in less span of time as we don't need to care for some particular server manufacturer or server model.

Use Older Applications for a Longer Time- Well, still many companies are using old applications but crucial to them and couldn't support modern hardware or operating system. In such situations, even the company wants, the IT would never prefer to touch them. However, we can pack such applications in a VM with the compatible old operating system and old virtual hardware. In this way, it will be possible to switch to modern hardware while keeping the old software stack intact.

Virtual Machine is Easily Portable- A single server running with some particular operating system software is not easy to move from one place to another, whereas if we have virtualized the same, then it becomes very easy to move data and OS from one physical server to another, situated somewhere else with the minimal workforce and without heavy transportation requirements.

Better Usage of Hardware Resources- Our modern computer or server hardware is quite powerful, using a single operating system and a couple of applications can't churn out the maximum juice of it. Thus, using VMs not only efficiently use the power of the CPU but allows the companies to save hundreds of bucks from spending on hardware.

Made Cloud Computing Possible- Yes, without VMs there will be no cloud computing because the whole idea behind it to provide an instant provision of machines running either Windows or Linux OS; it is only possible with the help of pre-build templates ready to deploy as VMs on some remote data center hardware. For example, Digital Ocean, AWS, and Google Cloud. So, next time whenever you heard "Cloud hosting" or "Virtual Private Server" hosting, remember it is a VM running on data center hardware.

11.3 Interpretation and Binary Translation

Emulation is required for implementing many VMs. It is the process of implementing the interface and functionality of one (sub)system on a (sub)system having a different interface and functionality. There are terminal emulators, such as for VT100, xterm, putty.There are various methods for enabling a (sub)system to present the same interface and characteristics as other ways of implementing emulation.

Interpretation: Interpretation involves relatively inefficient instruction-at-a-time.

Instruction Set Emulation- Binaries in source instruction set can be executed on machine implementing target instruction set. e.g., IA-32 execution layer.

Binary Translation: Binary translation involves block-at-a-time optimization for repeated. Example: Execution of programs compiled for instruction set A on a machine that executes instruction set B. The other features of binary translation include:

- Intercept OS code: Run-time translation of some OS instructions.
- User-level code is directly executed on the real hardware.
- No modifications to the OS are needed: the guest OS is not aware of virtualization.
- Specific device drivers are required.

Table 1 shows the difference between Interpretation vs Binary Translation.

Table 1: Interpretation vs Binary Translation

Interpretation	**Binary Translation**
Simple and easy to implement, portable	Complex implementation
Low performance	High initial translation cost, small execution cost
Threaded interpretation	Selective compilation

11.4 Hypervisors

VMs are widely used instead of physical machines in the IT industry today. The VMs support green IT solutions, and its usage increases resource utilization, making the management tasks easier. Since the VMs are mostly used, the technology that enables the virtual environment also gets attention in industries and academia. The virtual environment can be created with the help of a software tool called hypervisors.

Hypervisors are the software tool that sits in between VMs and physical infrastructure and provides the required virtual infrastructure for VMs.Hypervisors are also called as Virtual Machine Manager (VMM) (Figure 6). These are the key drivers in enabling virtualization in cloud data centers. Different hypervisors are being used in the IT industry. Some of the examples are VMware, Xen, Hyper-V, KVM, and OpenVZ.

The virtual infrastructure means virtual CPUs (vCPUs), virtual memory, virtual NICs (vNICs), virtual storage, and virtual I/O devices. The fundamental element of hardware virtualization is the hypervisor, or VMM that helps to recreate a hardware environment in which Guest Operating Systems (OSs) are installed.

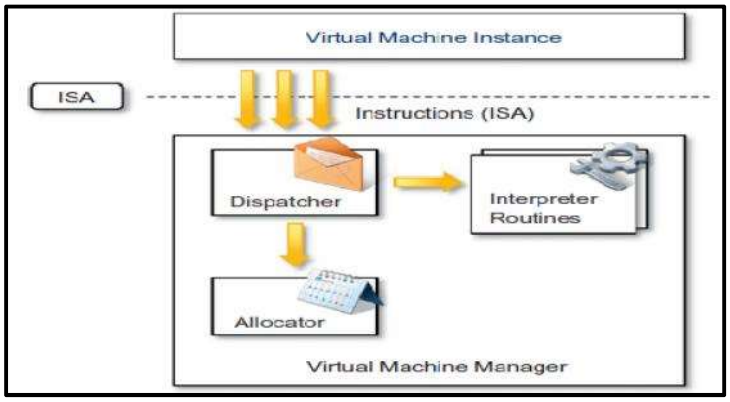


Figure 6: Internal Organization of a Virtual Machine Manager

There are three main modules, dispatcher, allocator, and interpreter, coordinate their activity in order to emulate the underlying hardware. 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 when ever a VM executes a privileged instruction: a trap is triggered and the corresponding routine is executed.

The design and architecture of a VMM, together with the underlying hardware design of the host machine, determine the full realization of hardware virtualization, where a guest OS can be transparently executed on top of a VMM as though it were run on the underlying hardware.

The criteria that need to be met by a VMM to efficiently support virtualization were established by Goldberg and Popekin 1974. The three properties have to be satisfied:

- Equivalence: A guest running under the control of a virtual machine manager should exhibit the same behavior as when it is executed directly on the physical host.
- Resource control: VMM should be incomplete control of virtualized resources.
- Efficiency: A statistically dominant fraction of the machine instructions should be executed without intervention from the VMM.

Before the hypervisors are introduced, there was a one-to-one relationship between hardware and OSs. This type of computing results in underutilized resources.

After the hypervisors are introduced, it became a one-to-many relationship. With the help of hypervisors, many OSs can run and share a single hardware.

Types of Hypervisors

Hypervisors are generally classified into two categories:

- Type 1 or bare metal hypervisors
- Type 2 or hosted hypervisors

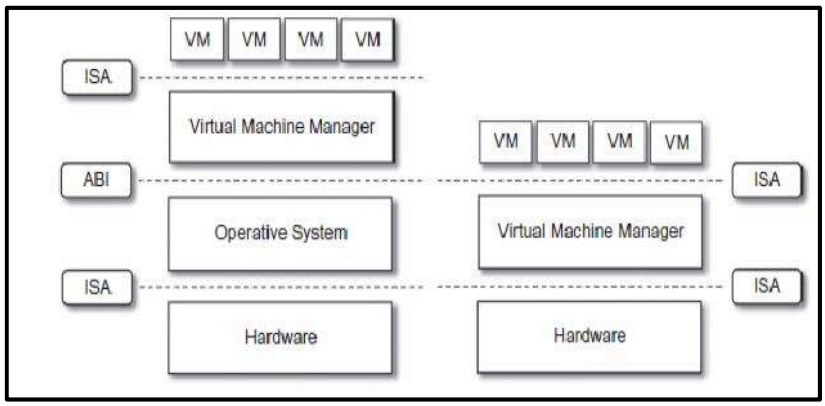


Figure 7: Hosted (left) and Native (right) VMs

Type I Hypervisors run directly on top of the hardware. Therefore, they take the place of the OSs and interact directly with the ISA interface exposed by the underlying hardware, and they emulate this interface in order to allow the management of guest OSs. These are also called a native VM since it runs natively on the hardware. The other characteristics of Type I hypervisors include:

- Can run and access physical resources directly without the help of any host OS.
- Additional overhead of communicating with the host OS is reduced and offers better efficiency when compared to type 2 hypervisors.
- Used for servers that handle heavy load and require more security.
- Examples- Microsoft Hyper-V, Citrix XenServer, VMWare ESXi, and Oracle VM Server for SPARC.

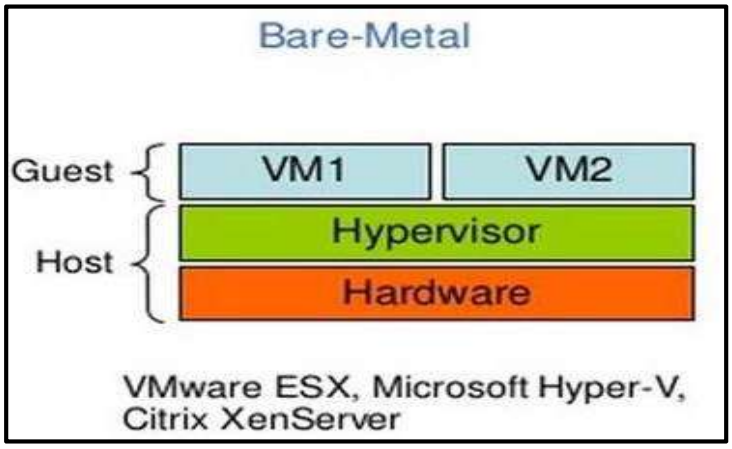


Figure 8: Bare-Metal Virtualization

Type II Hypervisors require the support of an operating system to provide virtualization services (Figure 9). This means that they are programs managed by the OS, which interact with it through the ABI and emulate the ISA of virtual hardware for guest OSs.This type of hypervisor is also called a hosted or embedded VM since it is hosted within an OS (Figure 10). Hosted virtualization requires the host OS and does not have direct access to the physical hardware. The host OS is also known as physical host, which has the direct access to the underlying hardware. However, the major disadvantage of this approach is if the host OS fails or crashes, it also results in crashing of VMs. So, it is recommended to use type 2 hypervisors only on client systems where efficiency is less critical.Examples- VMWare Workstation and Oracle Virtualbox.

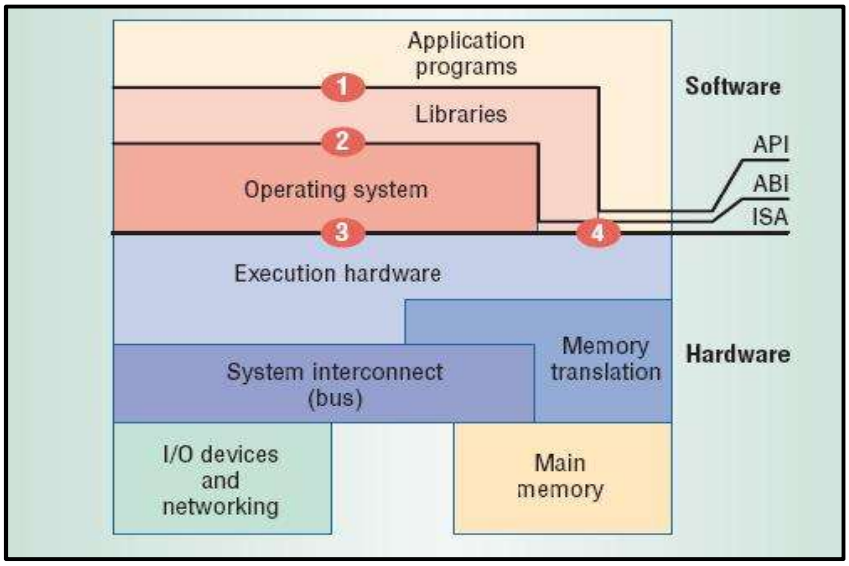


Figure 9: Type II Hypervisor

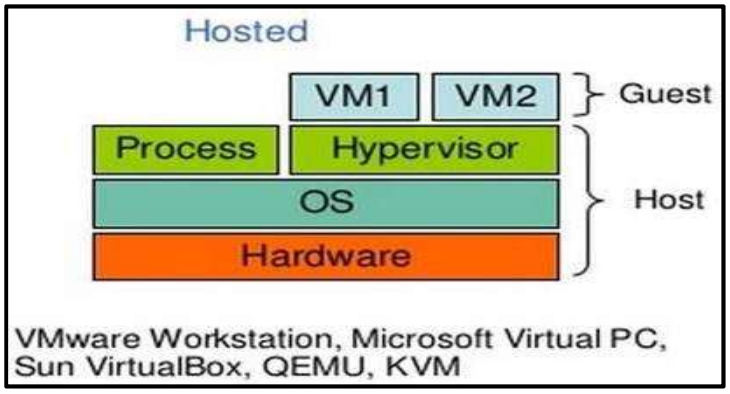


Figure 10: Hosted Virtualization

Summarized Implementation of Hypervisors

Vary greatly, with options including:

- Type 0 Hypervisors- Hardware-based solutions that provide support for virtual machine creation and management via firmware. Example: IBM LPARs and Oracle LDOMs are examples.
- Type 1 Hypervisors- Operating-system-like software built to provide virtualization. Example: Including VMware ESX, JoyentSmartOS, and Citrix XenServer.
- Type 1 Hypervisors– Also includes general-purpose operating systems that provide standard functions as well as VMM functions. Example: Microsoft Windows Server with HyperV and RedHat Linux with KVM.
- Type 2 Hypervisors- Applications that run on standard OSs but provide VMM features to guest OSs. Example: VMware Workstation and Fusion, Parallels Desktop, and Oracle VirtualBox.

Other Variations Include: Much variation exists due to breadth, depth and importance of virtualization in modern computing.

Para Virtualization- Technique in which the guest operating system is modified to work in cooperation with the VMM to optimize performance.

Programming-environment Virtualization- VMMs do not virtualize real hardware but instead create an optimized virtual system. It is used by Oracle Java and Microsoft.Net.

Emulators– Allow applications written for one hardware environment to run on a very different hardware environment, such as a different type of CPU.

Application Containment- Not virtualization at all but rather provides virtualization-like features by segregating applications from the operating system, making them more secure, manageable. It is included in Oracle Solaris Zones, BSD Jails, and IBM AIX WPARs.

Xen

An open-source initiative implementing a virtualization platform based on paravirtualization. Xen is a VMM for IA-32 (x86, x86-64), IA-64 and PowerPC 970 architectures. It allows several guest OSs to be executed on the same computer hardware concurrently. It was initially created by University of Cambridge, Computer Laboratory and is now developed and maintained by Xen community as free software, as well as Citrix XenServer Commercial version variant. It is the central part of Amazon.com's cloud computing platform, EC2 (Elastic Compute Cloud) that allows the users to rent virtual computers on which to run their own computer applications.

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. The most popular implementation of paravirtualization, which, in contrast with full virtualization, allows high performance execution of guest OSs. This is made possible by eliminating the performance loss while executing instructions that require special management.This is done by

modifying portions of the guest OSs run by Xen with reference to the execution of such instructions.Therefore, it is not a transparent solution for implementing virtualization. This is particularly true for x86, which is the most popular architecture on commodity machines and servers.

Xen Architecture

Figure 11 depicts the Xen architecture consisting of three different layers as discussed below:

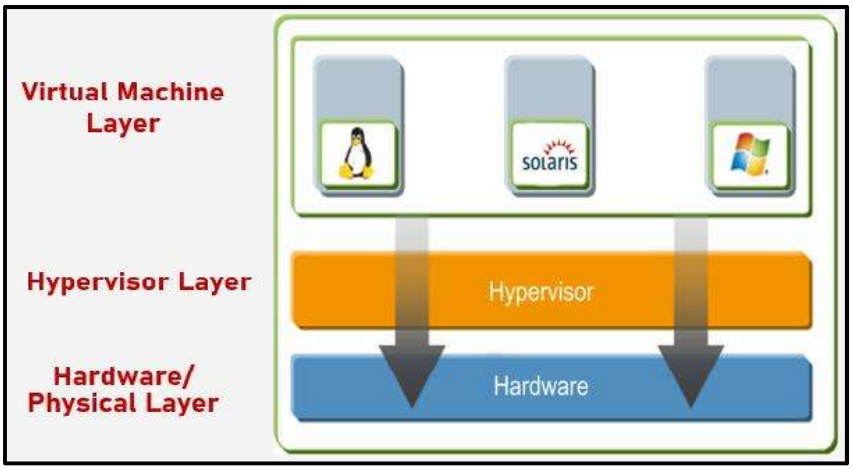


Figure 11: Xen Architecture

- Hardware or Physical Layer: Physical hardware components including memory, CPU, network cards, and disk drives.
- Hypervisor Layer: Thin layer of software that runs on top of the hardware. The Xen hypervisor gives each virtual machine a dedicated view of the hardware.
- Virtual Machine Layer: OS hosted on the hypervisor and appearing to the user as a separate physical computer. However, the machine shares physical resources with other virtual machines, and it is portable because the virtual machine is abstracted from the physical hardware.

A Xen-based system is managed by the Xen hypervisor, which runs in the highest privileged mode and controls the access of guest OS to the underlying hardware. Guest OSs are executed within domains, which represent VM instances. Moreover, specific control software, which has privileged access to the host and controls all the other guest OSs, is executed in a special domain called Domain 0. This is the first one that is loaded once the VMM has completely booted, and it hosts a Hyper Text Transfer Protocol (HTTP) server that serves requests for VM creation, configuration, and termination. This component constitutes the embryonic version of a distributed VMM, which is an essential component of cloud computing systems providing Infrastructure-as-a-Service (IaaS) solutions.

KVM (Kernel-based Virtual Machine (KVM))

An open-source virtualization technology built into Linux®. It is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V). It consists of a loadable kernel module, kvm.ko, that provides the core virtualization infrastructure and a processor specific module, kvm-intel.ko or kvm-amd.ko.KVM lets you turn Linux into a hypervisor that allows a host machine to run multiple, isolated virtual environments called guests or virtual machines (VMs).

KVM was first announced in 2006 and merged into the mainline Linux kernel version a year later.As KVM is part of existing Linux code, it immediately benefits from every new Linux feature, fix, and advancement without additional engineering.KVM converts Linux into a type-1 (bare-metal) hypervisor.All hypervisors need some operating system-level components—such as a memory manager, process scheduler, input/output (I/O) stack, device drivers, security manager, a network stack, and more—to run VMs.

How does KVM work?

- KVM has all these components because it's part of the Linux kernel.
- Every VM is implemented as a regular Linux process, scheduled by the standard Linux scheduler, with dedicated virtual hardware like a network card, graphics adapter, CPU(s), memory, and disks.

Implementing KVM

You have to run a version of Linux that was released after 2007 and it needs to be installed on X86 hardware that supports virtualization capabilities. If both of those boxes are checked, then all you have to do is load 2 existing modules (a host kernel module and a processor-specific module), an emulator, and any drivers that will help you run additional systems.But implementing KVM on a supported Linux distribution—like Red Hat Enterprise Linux—expands KVM's capabilities, letting you swap resources among guests, share common libraries, optimize system performance, and a lot more.

KVM Features

Security- KVM uses a combination of security-enhanced Linux (SELinux) and secure virtualization (sVirt) for enhanced VM security and isolation. SELinux establishes security boundaries around VMs. sVirt extends SELinux's capabilities, allowing Mandatory Access Control (MAC) security to be applied to guest VMs and preventing manual labeling errors.

Storage- KVM is able to use any storage supported by Linux, including some local disks and network-attached storage (NAS). Multipath I/O may be used to improve storage and provide redundancy. KVM also supports shared file systems so VM images may be shared by multiple hosts. Disk images support thin provisioning, allocating storage on demand rather than all up front.

Hardware Support- KVM can use a wide variety of certified Linux-supported hardware platforms. Because hardware vendors regularly contribute to kernel development, the latest hardware features are often rapidly adopted in the Linux kernel.

Memory Management- KVM inherits the memory management features of Linux, including non-uniform memory access and kernel same-page merging. The memory of a VM can be swapped, backed by large volumes for better performance, and shared or backed by a disk file.

Live Migration- KVM supports live migration, which is the ability to move a running VM between physical hosts with no service interruption. The VM remains powered on, network connections remain active, and applications continue to run while the VM is relocated. KVM also saves a VM's current state so it can be stored and resumed later.

Performance and Scalability- KVM inherits the performance of Linux, scaling to match demand load if the number of guest machines and requests increases. KVM allows the most demanding application workloads to be virtualized and is the basis for many enterprise virtualization setups, such as data centers and private clouds (via OpenStack®).

Scheduling and Resource Control- In the KVM model, a VM is a Linux process, scheduled and managed by the kernel. The Linux scheduler allows fine-grained control of the resources allocated to a Linux process and guarantees a quality of service for a particular process. In KVM, this includes the completely fair scheduler, control groups, network name spaces, and real-time extensions.

Lower Latency and Higher Prioritization- The Linux kernel features real-time extensions that allow VM-based apps to run at lower latency with better prioritization (compared to bare metal). The kernel also divides processes that require long computing times into smaller components, which are then scheduled and processed accordingly.

VMware

VMware Workstation is the most dependable, high-performing, feature-rich virtualization platform for your Windows or Linux PC (Figure 12 and Figure 13). It allows one physical PC to run multiple operating systems at the same time. Actually, no restarting or hard-drive partitioning is required. The software developers rely on workstation to develop and test client-server, Web and cloud applications in a replica of their production environments.

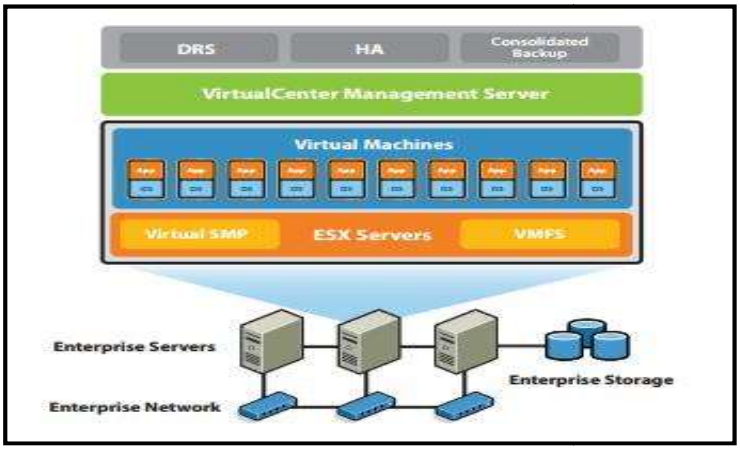


Figure 12: VMware Scenario

VMware's technology is based on the concept of full virtualization, where the underlying hardware is replicated and made available to the guest OS, which runs unaware of such abstraction layers and does not need to be modified.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. In both cases, full virtualization is made possible by means of direct execution (for non sensitive instructions) and binary translation (for sensitive instructions), thus allowing the virtualization of architecture such as x86. Besides these two core solutions, 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.

Figure 13: VMware Workstation Layout

Key Benefits of VMware

- Access anytime, anywhere
- Run applications in Windows, Linux and other systems at the same time without restarting.
- Remotely access VMs running on VMware.
- Run as a server to host applications for your team, department or anyone in your organization.

- Create VMs that are encrypted, block USB devices and have read-only settings.

Virtual Box

It is a powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use. An extremely feature rich, high performance product for enterprise customers also, the only professional solution that is freely available as Open Source Software under the terms of the GNU General Public License (GPL) version 2. Presently, it runs on Windows, Linux, Macintosh, and Solaris hosts and supports a large number of guest OSs including but not limited to Windows (NT 4.0, 2000, XP, Server 2003, Vista, Windows 7, Windows 8, Windows 10), DOS/Windows 3.x, Linux (2.4, 2.6, 3.x and 4.x), Solaris and OpenSolaris, OS/2, and OpenBSD.

Oracle VM VirtualBox-

- Cross-platform virtualization application.
- Installs on your existing Intel or AMD-based computers, whether they are running Windows, Mac OS X, Linux, or Oracle Solaris operating systems (OSes).
- Extends the capabilities of your existing computer so that it can run multiple OSes, inside multiple virtual machines, at the same time.
- Oracle VM VirtualBox is deceptively simple yet also very powerful.
- Can run everywhere from small embedded systems or desktop class machines all the way up to data center deployments and even Cloud environments.

Microsoft Hyper-V

Hyper-V is a primary engine that drives Windows Server 2008 "beyond virtualization" initiative. The primary responsibility of Windows Server 2008 Hyper-V is to provide the tool kit that organizations will use to create a shared pool of compute, network, and storage resources where servers and applications can be virtualized for consolidation, scalability, and mobility purposes.

Microsoft Hyper-V Architecture

Hyper-V supportshypervisor-based virtualization platform and an enabling technology for one of Windows Server 2008 R2's marquee features, Live Migration. The guest OSs running in a Hyper-V VM provide performance approaching the performance of an OS running on physical hardware if the necessary virtual server client (VSC) drivers and services are installed on the guest OS.

Hyper-V virtual server client (VSC) code, is also known as Hyper-V enlightened I/O, enables direct access to the Hyper-V "Virtual Machine Bus" and is available with the installation of Hyper-V integration services. Both Windows Server 2008 R2 and Windows 7 support Hyper-V enlightened I/O with Hyper-V integration services. Hyper-V supports isolation in terms of a partition. The Microsoft hypervisor must have at least one parent, or root, partition, running Windows Server 2008 R2. The root partition then creates the child partitions which host the guest OSs. The child partitions also do not have direct access to other hardware resources and are presented a virtual view of the resources, as virtual devices (VDevs).

Despite its straightforward installation as a component of the host OS, Hyper-V takes control of the hardware, and the host OS becomes a VM instance with special privileges, called the parent partition.

- Parent partition (also called the root partition) is the only one that has direct access to the hardware. It runs the virtualization stack, hosts all the drivers required to configure guest OSs, and creates child partitions through the hypervisor.
- Child partitions are used to host guest OSs and do not have access to the underlying hardware, but their interaction with it is controlled by either the parent partition or the hypervisor itself.

Hypervisor is the component that directly manages the underlying hardware (processors and memory). 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. This is an implementation of the Para virtualization approach discussed with Xen. This interface is used by drivers in the partitioned OS to contact the Hypervisor using the standard Windows calling convention. The parent partition also uses this interface to create child partitions.

Memory Service Routines (MSRS): These are the set of functionalities that control the memory and itsaccessfrompartitions.Byleveraginghardware-assistedvirtualization,thehypervisor uses the Input/Output Memory Management Unit (I/O MMU or IOMMU) to fast-track access to devices from partitions by translating virtual memory addresses.

Advanced Programmable Interrupt Controller (APIC): This component represents the interrupt controller,whichmanagesthesignalscoming from the underlyinghardwarewhensomeevent occurs (timer expired, I/O ready, exceptions and traps). Each virtual processor is equipped with a synthetic interrupt controller (SynIC), which constitutes an extension of the local APIC. The hypervisor is responsible of dispatching, when appropriate, the physical interrupts to the synthetic interrupt controllers.

Scheduler: This component schedules the virtual processors to run on available physical processors. The scheduling is controlled by policies that are set by the parent partition. • Address manager. This component is used to manage the virtual network addresses that are allocated to each guest OS.

Partition Manager: This component is in charge of performing partition creation, finalization, destruction, enumeration, and configurations. Its services are available through the hypercalls interface API.

Microsoft Hyper-V Performance Characteristics

Improved Hardware Sharing Architecture: Hyper-V provides improved access and utilization of core resources, such as disk, networking, and video when running guest OSs with a hypervisor-aware kernel and which are equipped with requisite virtual server client (VSC) code (known as Hyper-V enlightened I/O).

Critical Disk Performance for I/O Intensive Applications: Disk performance is critical for disk I/O intensive enterprise applications such as Microsoft BizTalk Server and in addition to Hyper-V enlightened I/O; Hyper-V provides "Passthrough" disk support which provides disk performance on par with physical disk performance.

Processor Hardware-assisted Virtualization Support: Hyper-V takes full advantage of processor hardware-assisted virtualization support that is available with recent processor technology.

Multi-core (SMP) Guest OS Support: Hyper-V provides the ability to support up to four processors in a virtual machine environment, which allows applications to take full advantage of multi-threading functionality in a VM.

Both 32-bit and 64-bit Guest OS Support: Hyper-V provides broad support for simultaneously running different types of OSs, including 32-bit and 64-bit systems across different server platforms, such as Windows, Linux, and others.

Advantages of Microsoft Hyper-V Architecture

Consolidation of Hardware Resources: Multiple physical servers can be easily consolidated into comparatively fewer servers by implementing virtualization with Hyper-V. Consolidation accommodates full use of deployed hardware resources. Hyper-V in Windows Server 2008 R2 can now access up to 64 logical CPUs on host computers.

Ease of Administration:

- Consolidation and centralization of resources simplifies administration.
- Implementation of scale-up and scale out is accommodated with much greater ease.

Fault Tolerance Support through Hyper-V Clustering: Because Hyper-V is a cluster aware application, Windows Server 2008 SP2 provides native host clustering support for virtual machines created in a Hyper-V virtualized environment.

Ease of deployment and management:

- Consolidation of existing servers into fewer physical servers simplifies deployment.

- A comprehensive Hyper-V management solution is available with System Center VMM.

Proven Track Record- Key Microsoft web sites MSDN (http://msdn.microsoft.com) and TechNet (http://technet.microsoft.com) are hosted in Hyper-V environments.

Comprehensive Product Support– Because Microsoft enterprise applications (such as Exchange Server and SQL Server) are fully tested running in Hyper-V, Microsoft provides code fix support for these applications when deployed and run in a Hyper-V environment.

Scalability– Additional processing power, network bandwidth, and storage capacity can be accomplished quickly and easily by apportioning additional available resources from the host computer to the guest VM(s).

Summary

- Virtualization raises abstraction. Abstraction pertains to hiding of the inner details from a particular user. Virtualization helps in enhancing or increasing the capability of abstraction.
- Virtualization enables sharing of resources much easily, it helps in increasing the degree of hardware level parallelism, basically, there is sharing of the same hardware unit among different kinds of independent units.
- In a bare metal architecture, one hypervisor or VMM is actually installed on the bare metal hardware. There is no intermediate OS existing over here. The VMM communicates directly with the system hardware and there is no need for relying on any host OS.
- Type I Hypervisors run directly on top of the hardware. Therefore, they take the place of the OSs and interact directly with the ISA interface exposed by the underlying hardware, and they emulate this interface in order to allow the management of guest OSs.
- Type II Hypervisors require the support of an operating system to provide virtualization services. This means that they are programs managed by the OS, which interact with it through the ABI and emulate the ISA of virtual hardware for guest OSs.
- Xen is an open-source initiative implementing a virtualization platform based on paravirtualization. Xen is a VMM for IA-32 (x86, x86-64), IA-64 and PowerPC 970 architectures.
- KVM is part of existing Linux code, it immediately benefits from every new Linux feature, fix, and advancement without additional engineering. KVM converts Linux into a type-1 (bare-metal) hypervisor.
- VMware Workstation is the most dependable, high-performing, feature-rich virtualization platform for your Windows or Linux PC.

Keywords

- ***Virtualization:*** Virtualization is a broad concept that refers to the creation of a virtual version of something, whether hardware, a software environment, storage, or a network.
- ***Type 0 Hypervisors-*** Hardware-based solutions that provide support for virtual machine creation and management via firmware. Example: IBM LPARs and Oracle LDOMs are examples.
- ***Type 1 Hypervisors-*** Operating-system-like software built to provide virtualization. Example: Including VMware ESX, JoyentSmartOS, and Citrix XenServer. It also includes general-purpose operating systems that provide standard functions as well as VMM functions. Example: Microsoft Windows Server with HyperV and RedHat Linux with KVM.
- ***Type 2 Hypervisors-*** Applications that run on standard OSs but provide VMM features to guest OSs. Example: VMware Workstation and Fusion, Parallels Desktop, and Oracle VirtualBox.
- ***Interpretation:*** Interpretation involves relatively inefficient instruction-at-a-time.
- ***Binary Translation:*** Binary translation involves block-at-a-time optimization for repeated.

- ***Para Virtualization-*** Technique in which the guest operating system is modified to work in cooperation with the VMM to optimize performance.
- ***Programming-environment Virtualization-*** VMMs do not virtualize real hardware but instead create an optimized virtual system. It is used by Oracle Java and Microsoft.Net.
- ***Emulators–***Emulators allow the applications written for one hardware environment to run on a very different hardware environment, such as a different type of CPU.

Self Assessment

1. _____ is software that creates a virtualized environment between the computer platform and the end-user in which the end user can operate software.

A. Virtual machine
B. FogSim
C. Firmware
D. Freeware

2. Which of the following is characteristic of virtualization concept?

A. Mapping of virtual resources or state to real resources
B. Use of real machine instructions to carry out actions specified by the virtual machine instructions
C. Implemented by adding a layer of software to a real machine to support the desired VM's architecture
D. All of the above

3. Process virtual machines are also known as _____ virtual machines.

A. Primary
B. Hype
C. Application
D. Sigma

4. _____ is the process of implementing the interface and functionality of one (sub)system on a (sub)system having a different interface and functionality.

A. Inheritance
B. Emulation
C. Multitenancy
D. Interfacing

5. Which of the following is not true about Interpretation?

A. Simple and easy to implement
B. High performance
C. Threaded interpretation
D. Portability

6. _____ is a form of binary recompilation where sequences of instructions are translated from a source instruction set to the target instruction set.

A. Compilation
B. Blocking

C. Binary translation
D. Coordination

7. Which of the following is/are significant feature(s) of binary translation?
A. Run-time translation of some OS instructions
B. User-level code is directly executed on the real hardware
C. Guest OS is not aware of virtualization
D. All of the above

8. Virtualization allows multiple virtual machines, with heterogeneous OSs to run in isolation, side-by-side on the _____ physical machine(s).
A. Same
B. Different
C. One
D. Two

9. In virtualization, the virtual environment can be created with the help of a software tool called _____.
A. Hypervisors
B. Emulators
C. Compilers
D. Interpreters

10. A _____ is a fundamental element of hardware virtualization.
A. Bare-metal device
B. Virtual machine manager
C. Storage
D. Software stick

11. In a hypervisor, the _____constitutes the entry point of the monitor and reroutes the instructions issued by the virtual machine instance to one of the two other modules.
A. Virtual layer
B. Interpreter
C. Dispatcher
D. Translator

12. Which of the following property(s) specify the virtualization requirements?
A. Efficiency property
B. Resource control property
C. Equivalence property
D. All of the above

13. Type II hypervisors require the support of an operating system to provide virtualization services. Such hypervisors are also called as
A. Hybrid
B. Hosted
C. Bare-metal

D. Native

14. Which of the following is an example of hosted virtualization hypervisors?

A. VMware Workstation

B. Oracle Virtual Box

C. Microsoft Virtual PC

D. All the above

15. In a _____ virtualization, the VMM communicates directly with system hardware rather than relying on a host operating system.

A. Bare-metal

B. Hybrid

C. Storage

D. Hosted

Answers for Self Assessment

1.	A	2.	D	3.	C	4.	B	5.	B
6.	C	7.	D	8.	A	9.	A	10.	B
11.	C	12.	D	13.	B	14.	D	15.	A

Review Questions

1. Differentiate Native and Hosted virtualization?
2. Explain the concept of Hosted virtualization?
3. Write a short note on:
   (a) Xen
   (b) KVM
   (c) Hyper-V
4. What are the benefits of using Microsoft Hyper-V?
5. Elaborate the working on VMware?
6. Discuss about Interpretation and Binary Translation?
7. What is Virtualization and why it is required?
8. How does Virtual Machines work? Discuss the characteristics.
9. Give an overview of Virtual Machine Taxonomy.
10. Compare the Process VM with System VM?

Further Readings

- Mastering Cloud Computing by "Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi, Tata McGraw-Hill Education, 2021.
- Cloud Computing: Concepts, Technology and Architecture by Thomas Erl, Pearson Education.
- Cloud Computing Black Book by Kailash Jayaswal,Jagannath Kallakurchi, Donald J. Houde, Deven Shah, Kogent Learning Solutions, DreamTech Press.

- Virtualization: A Manager's Guide by Dan Kusnetzky, O'Reilly, 2011.
- Virtual Machines, by James Edward Smith, Elsevier, 2021.

Web Links

1.What is Virtualization? (tutorialspoint.com)

2.What is Virtualization? How does it Work? [Understanding Virtualization] (kuberty.io)

3.What is a Virtual Machine? | VMware Glossary

4.What Is a Virtual Machine and How Does It Work | Microsoft Azure

5. Virtual Machine - Javatpoint

6.Types of Virtual Machines - GeeksforGeeks

7.Home - Xen Project

8.What Is Hyper-V & How Do You Use It? A Beginner's Guide (cloudwards.net)

https://youtu.be/d7J9p2uHkEU