CMR
INSTITUTE OF
TECHNOLOGY

USN | | | | | | | | | | |

Internal Assessment Test 2 – Dec. 2021

Sub:	Cloud Computing							Sub Code:	20MCA342
Date:	17/12//2021	Duration:	90 min's	Max Marks:	50	Sem:	III	Branch:	MCA

Note : Answer FIVE FULL Questions, choosing ONE full question from each Module

		MARKS	OBE	
			CO	RBT
	PART I			
1	Explain in detail about Remote Procedure Call (RPC) with necessary diagram.	[10]	CO2	L2
	OR			
2	Explain the following terms: Distributed Object Frameworks, Object activation and lifetime.	[10]	CO2	L2
	PART II			
3	Explain the characteristics of virtualized environment in detail.	[10]	CO2	L2
	OR			
4	Explain the following terms: Execution virtualization, Storage virtualization, Network virtualisation, Desktop virtualization, application server virtualisation.	[10]	CO2	L2

CMR
INSTITUTE OF
TECHNOLOGY

USN | | | | | | | | | | |

Internal Assessment Test 2 – Dec. 2021

Sub:	Cloud Computing							Sub Code:	20MCA342
Date:	17/12//2021	Duration:	90 min's	Max Marks:	50	Sem:	III	Branch:	MCA

Note : Answer FIVE FULL Questions, choosing ONE full question from each Module

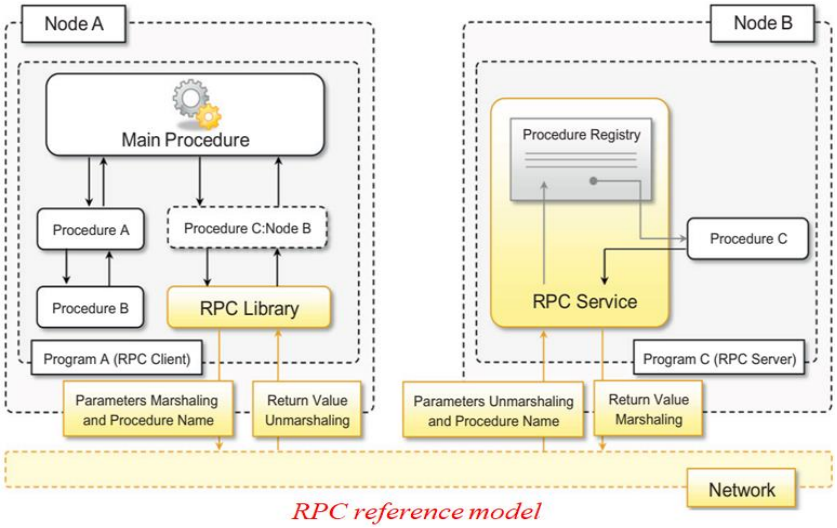
		MARKS	OBE	
			CO	RBT
	PART I			
1	Explain in detail about Remote Procedure Call (RPC) with necessary diagram.	[10]	CO2	L2
	OR			
2	Explain the following terms: Distributed Object Frameworks, Object activation and lifetime.	[10]	CO2	L2
	PART II			
3	Explain the characteristics of virtualized environment in detail.	[10]	CO2	L2
	OR			
4	Explain the following terms: Execution virtualization, Storage virtualization, Network virtualisation, Desktop virtualization, application server virtualisation.	[10]	CO2	L2

	PART III			
5	Explain in detail about the virtualization support at OS level with example.	[10]	CO2	L2
	OR			
6	Explain the process of live migration of VM from one host to another.	[10]	CO2	L2
	PART IV			
7	Explain Xen architecture with suitable diagram.	[10]	CO2	L2
	OR			
8	Explain the types of hypervisors with necessary diagrams.	[10]	CO2	L2
	PART V			
9	Explain the following with necessary diagrams: Full virtualisation, para virtualisation and partial virtualisation.	[10]	CO2	L2
	OR			
10	Illustrate Microsoft's Azure platform for cloud computing.	[10]	CO2	L2

	PART III			
5	Explain in detail about the virtualization support at OS level with example.	[10]	CO2	L2
	OR			
6	Explain the process of live migration of VM from one host to another.	[10]	CO2	L2
	PART IV			
7	Explain Xen architecture with suitable diagram.	[10]	CO2	L2
	OR			
8	Explain the types of hypervisors with necessary diagrams.	[10]	CO2	L2
	PART V			
9	Explain the following with necessary diagrams: Full virtualisation, para virtualisation and partial virtualisation.	[10]	CO2	L2
	OR			
10	Illustrate Microsoft's Azure platform for cloud computing.	[10]	CO2	L2

1. Explain in detail about Remote Procedure Call (RPC) with necessary diagram. (10)

- Remote procedure call
- fundamental abstraction enabling the execution of procedures on client's request.
- allows extending the concept of a procedure call beyond the boundaries of a process and a single memory address space
- The called procedure and calling procedure may be on the same system or they may be on different systems in a network
- Even though it is a quite old technology, RPC is still used today as a fundamental component for IPC in more complex systems.



RPC reference model

- Marshaling: identifies the process of converting parameter and return values into a form that is more suitable to be transported over a network through a sequence of bytes. [packing of arguments(or parameters) into a message packet]
- Unmarshaling: the opposite procedure. [unpacking of arguments(or parameters) received from the call packet].
- The RPC runtime is also for handling the request-reply interaction that happens between the client and the server process in a completely transparent manner.
- RPC for IPC consists of the following steps:
- Design and implementation of the server procedures that will be exposed for remote invocation.
- Registration of remote procedures with the RPC server on the node where they will be made available.
- Design and implementation of the client code that invokes the remote procedure(s).

2. Explain the following terms: Distributed Object Frameworks, Object activation and lifetime. (10)

- Distributed object frameworks
- extend object-oriented programming systems by allowing objects to be distributed across a heterogeneous network
- the common interaction pattern is the following:
- The server process maintains a registry of active objects that are made available to other processes.
- The client process, by using a given addressing scheme, obtains a reference to the active remote object.
- The client process invokes the methods on the active object by calling them through the reference previously obtained. Parameters and return values are marshaled

- Object activation and lifetime
- Methods live within the context of an object instance, and they can alter the internal state of the object as a side effect of their execution
- The memory allocated for them can be explicitly reclaimed by the programmer or automatically by the runtime system when there are no more references to that instance
  - Activation: the creation of a remote object
  - server-based activation: the active object is created in the server process and registered as an instance that can be exposed beyond process boundaries
  - client-based activation: the active object is created when a request for method invocation comes from a client

3. Explain the characteristics of virtualized environment in detail. (10)

(i) Increased Security:

- The ability to control the execution of a guest in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment.
- The virtual machine represents an emulated environment in which the guest is executed.
- All the operations of the guest are generally performed against the virtual machine
- Resources exposed by the host can then be hidden or simply protected from the guest
- sensitive information that is contained in the host can be naturally hidden without the need to install complex security policies.
- Increased security is a requirement when dealing with untrusted code (Example: applets downloaded from the Internet run in a sandboxed version of JVM)

(ii) Managed Execution:

- Virtualization of the execution environment not only allows increased security, but a wider range of features also can be implemented like sharing, aggregation, emulation, and isolation are the most relevant features
  - Sharing: a particularly important feature in virtualized data centers, used to reduce the number of active servers and limit power consumption.
  - Aggregation: A group of separate hosts can be tied together and represented to guests as a single virtual host
  - Emulation: a completely different environment with respect to the host can be emulated, thus allowing the execution of guest programs requiring specific characteristics that are not present in the physical host
  - Isolation: Virtualization allows providing guests 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
- Advantages of Isolation:
  - it provides a separation between the host and the guest
  - it allows multiple guests to run on the same host without interfering with each other

(iii) Portability

- Portability allows having your own system always with you and ready to use as long as the required virtual machine manager is available
- the guest is packaged into a virtual image that, in most cases, can be safely moved and executed on top of different virtual machines

- Virtual images are generally proprietary formats that require a specific virtual machine manager to be executed
- In the case of programming-level virtualization, the binary code representing application components (jars or assemblies) can be run without any recompilation on any implementation of the corresponding virtual machine

4. Explain the following terms: Execution virtualization, Storage virtualization, Network virtualisation, Desktop virtualization, application server virtualisation. (10)

- Execution Virtualisation:
- techniques that aim to emulate an execution environment that is separate from the one hosting the virtualization layer providing support for the execution of programs
- Storage Virtualisation:
- system administration practice that allows decoupling the physical organization of the hardware from its logical representation
- users' data location is not revealed
- allows us to harness a wide range of storage facilities
- Network Virtualisation:
- combines hardware appliances and specific software for the creation and management of a virtual network
- aggregate different physical networks into a single logical network
- Desktop Virtualisation:
- provides the same outcome of hardware virtualization but serves a different purpose.
- makes accessible a different system as though it were natively installed on the host, but this system is remotely stored on a different host and accessed through a network connection
- Application Server Virtualisation:
- a collection of application servers that provide the same services as a single virtual application server by using load-balancing strategies and providing a high-availability infrastructure for the services hosted in the application server

5. Explain in detail about the virtualization support at OS level with example. (10)

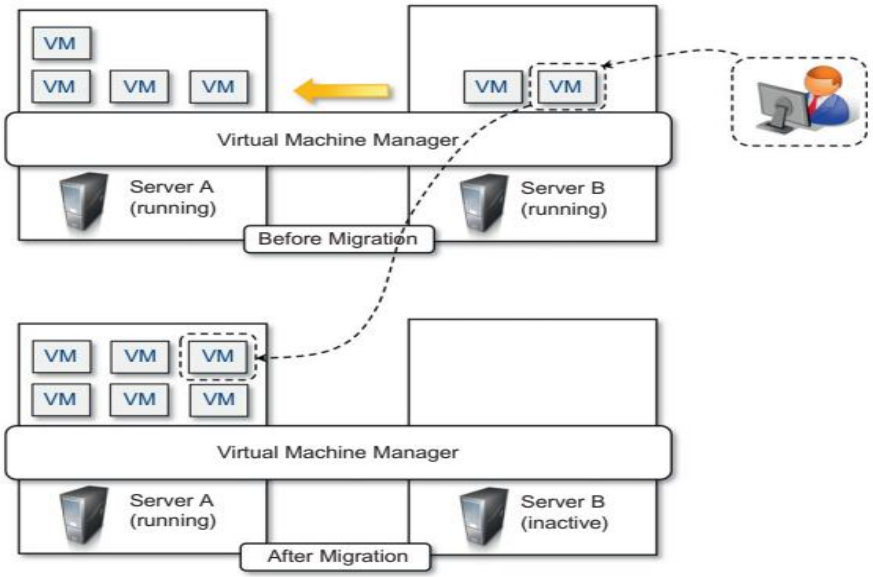
- offers the opportunity to create different and separated execution environments for applications that are managed concurrently.
- Differently from hardware virtualization, there is no virtual machine manager or hypervisor
- the virtualization is done within a single operating system, where the OS kernel allows for multiple isolated user space instances
- A user space instance in general contains a proper view of the file system, which is completely isolated, and separate IP addresses, software configurations, and access to devices
- It is an efficient solution for server consolidation scenarios

Example: *chmod* in unix (Solaris and OpenVZ)

```
[test_dev@ra-net ~]$ ls test2
alias.sh  cat.sh      loop.sh      special.sh  variable.sh
array.sh  first2.sh   sayhellow.sh test.txt
awk.sh    function.sh sed.sh       unix.sh
[test_dev@ra-net ~]$
[test_dev@ra-net ~]$
[test_dev@ra-net ~]$ chmod 755 test2/test.txt
```

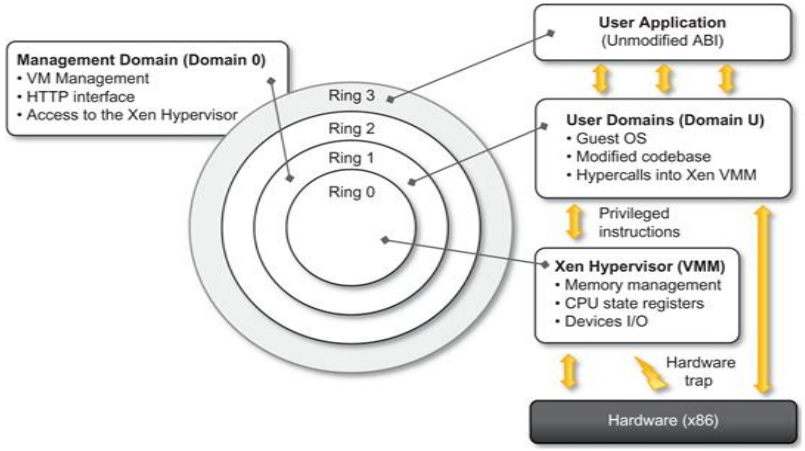
6. Explain the process of live migration of VM from one host to another. **(10)**

- Server Consolidation: allows reducing the number of active resources by aggregating virtual machines over a smaller number of resources that become fully utilized
- Virtual Machine Migration: the movement of virtual machine instances
- Live Migration: the process of moving a running virtual machine or application between different physical machines without disconnecting the client or application



7. Explain Xen architecture with suitable diagram. **(10)**

- an open-source initiative implementing a virtualization platform based on para-virtualization, under GPL2
- Xen now has a large open-source community backing it
- It is used for either desktop virtualization or server virtualization
- Virtualisation system supporting both para-virtualisation and hardware-assistant full virtualisation
  - Para-virtualisation: High performance, high scalability and uses a modified OS
  - Hardware-assisted full virtualisation: enhance virtualisation in x86, uses an unmodified OS
- Offers high performance and secure architecture



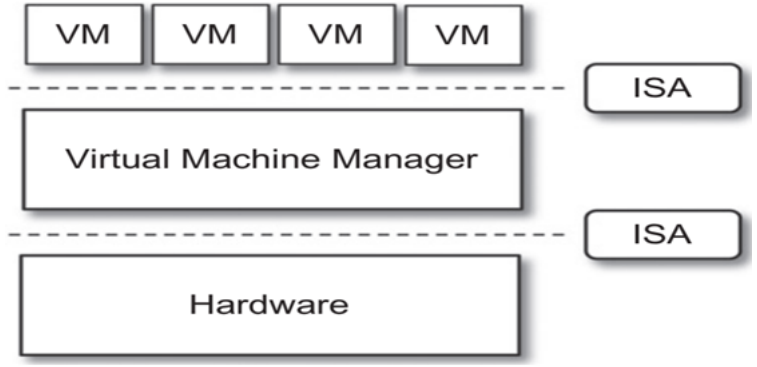
Xen architecture and guest OS management

- Xen hypervisor: runs in the highest privileged mode and controls the access of guest operating system to the underlying hardware

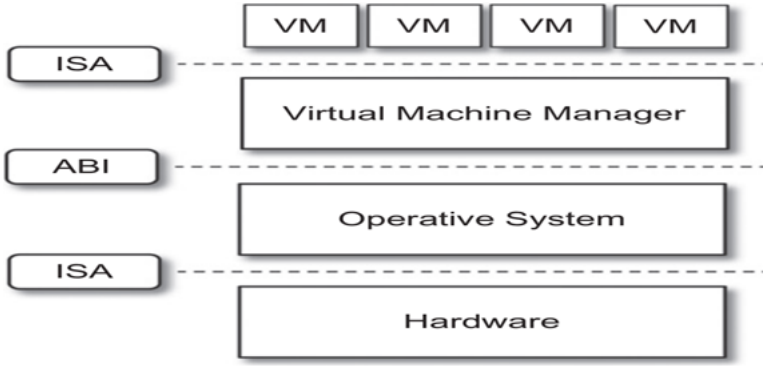
- Guest OS are executed within domains, which represent virtual machine instances
- All guest OSs are maintained and executed in a special domain called Domain 0
- HTTP server serves requests for virtual machine creation
- Xen implemented virtualization by executing the hypervisor in Ring 0, Domain 0

8. Explain the types of hypervisors with necessary diagrams. (10)

- Hypervisors: allows an OS to run independently from the underlying hardware thro' VM.
- run in supervisor mode
  - It is expected that all the sensitive instructions will be executed in privileged mode, which requires supervisor mode in order to avoid traps.
  - Without this assumption it is impossible to fully emulate and manage the status of the CPU for guest operating systems
- Type-1 Hypervisors (also called Native Virtual Machine / Bare Metal Hypervisor):
  - run directly on top of the hardware
  - they take the place of the operating systems and interact directly with the ISA interface exposed by the underlying hardware
  - they emulate this interface in order to allow the management of guest operating systems.
  - Example: Microsoft Hyper-V, Xen
- Type-2 (also called a Hosted Virtual Machine / Hosted Hypervisor):
- hosted within an OS
  - require the support of an OS to provide virtualization services.
  - they are programs managed by the operating system, which interact with it through the ABI and emulate the ISA of virtual hardware for guest OS.
  - Example: Oracle VM



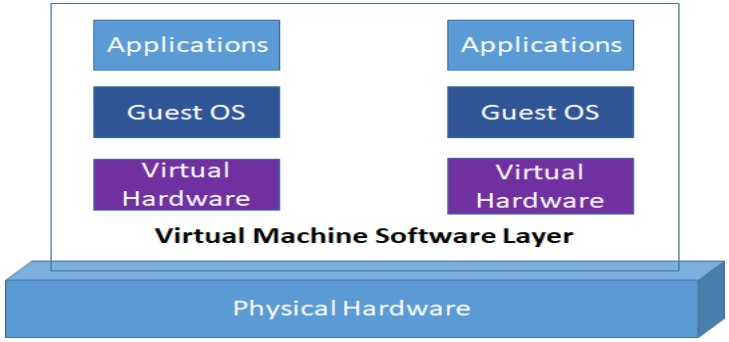
Type-1 Hypervisor: Native Virtual Machine



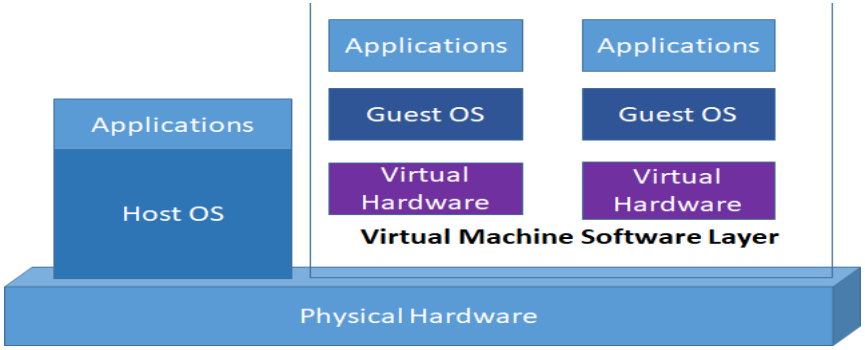
Type-2 Hypervisor: Hosted Virtual Machine

9. Explain the following with necessary diagrams: Full virtualisation, para virtualisation and partial virtualisation. (10)

- Full virtualization
- the ability to run a program, most likely an operating system, directly on top of a virtual machine and without any modification, as though it were run on the raw hardware.
- Virtual Machine Managers (VMM) are required to provide a complete emulation of the entire underlying hardware



- Para-Virtualisation
- expose a software interface to the virtual machine that is slightly modified from the host and, as a consequence, guest OS needs to be modified.
- Aim: to provide the capability to demand the execution of performance-critical operations directly on the host, thus preventing performance losses that would otherwise be experienced in managed execution.
- Example: VMWare, Parallels
- Embedded and real-time environments: TRANGO, WindRiver, and XtratuM

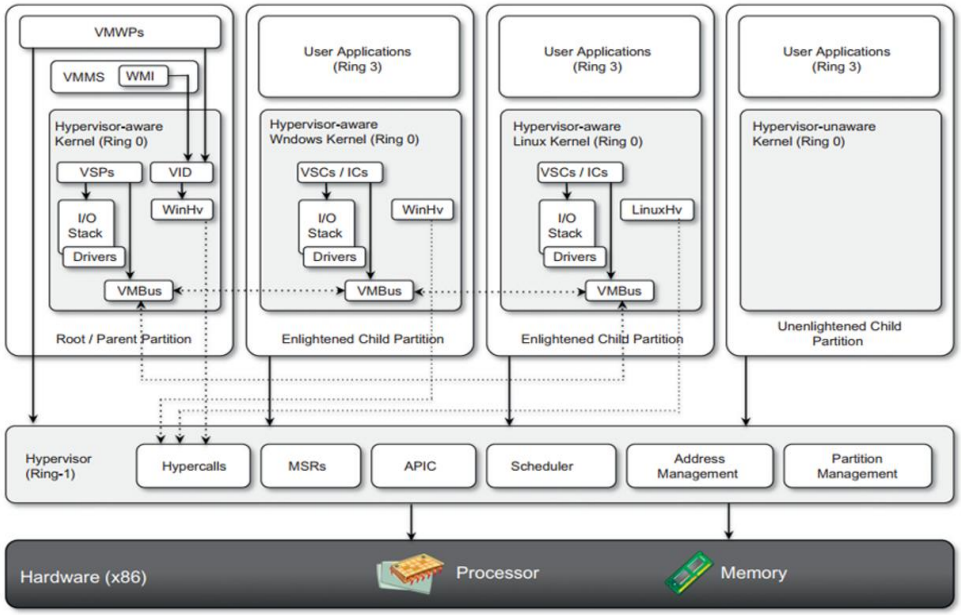


- Partial virtualization
- Many different operating systems can run simultaneously
- provides a partial emulation of the underlying hardware, thus not allowing the complete execution of the guest operating system in complete isolation.
- allows many applications to run transparently, but not all the features of the operating system can be supported, as happens with full virtualization.
- Example: IBM M44/44X: Partial virtualisation as realized for full virtualisation
  - Address space virtualization is a common feature of contemporary operating systems.

10. Illustrate Microsoft's Azure platform for cloud computing. (10)

- infrastructure virtualization solution developed by Microsoft for server virtualization.
- it uses a hypervisor-based approach to hardware virtualization
- It leverages several techniques to support a variety of guest operating systems.
- Hyper-V is currently shipped as a component of Windows Server 2008 R2 that installs the hypervisor as a role within the server

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



Microsoft Hyper-V architecture

- Hypervisor
- the component that directly manages the underlying hardware (processors and memory).
- It is logically defined by the following components:
  - Hypercalls interface: the entry point for all the partitions for the execution of sensitive instructions; it is an implementation of the para-virtualization approach; used by drivers in the partitioned OS to contact the hypervisor using the standard Windows calling convention
  - Memory Service Routines (MSRs): the set of functionalities that control the memory and its access from partitions; the hypervisor uses the Input/Output Memory Management Unit (IOMMU) to fast-track access to devices from partitions by translating virtual memory addresses
  - Advanced programmable interrupt controller (APIC): represents the interrupt controller, which manages the signals coming from the underlying hardware when some event occurs (timer expired, I/O ready, exceptions and traps)
  - Enlightened I/O and synthetic devices
  - Enlightened I/O provides an optimized way to perform I/O operations, allowing guest operating systems to leverage an inter-partition communication channel rather than traversing the hardware emulation stack provided by the hypervisor.
- There are three fundamental components:

i. VMBus: an inter-partition communication channel that is used to exchange data between partitions; implements the channel and defines the protocol for communication between partitions

ii. Virtual Service Providers (VSPs): kernel-level drivers that are deployed in the parent partition and provide access to the corresponding hardware devices

iii. Virtual Service Clients (VSCs): VSPs interacts with VSCs which represent the virtual device drivers

- OS supported by Hyper-V utilize this preferred communication channel to perform I/O for storage, networking, graphics, and input subsystems
  - Parent partition

- executes the host operating system and implements the virtualization stack that complements the activity of the hypervisor in running guest operating systems
- hosts an instance of the Windows Server 2008 R2, which manages the virtualization stack made available to the child partitions
- manages the creation, execution, and destruction of child partitions by means of the Virtualization Infrastructure Driver (VID)
- For each child partition created, a Virtual Machine Worker Process (VMWP) is instantiated in the parent partition, which manages the child partitions by interacting with the hypervisor through the VID.

▪ Child partitions

- used to execute guest operating systems.
- These are isolated environments that allow secure and controlled execution of guests.
- Two types of child partition exist, they differ on whether the guest operating system is supported by Hyper-V or not.

i. Enlightened partitions: can benefit from Enlightened I/O and
ii. Unenlightened partitions: executed by leveraging hardware emulation from the hypervisor.

▪ Cloud computing and infrastructure management

- Hyper-V constitutes the basic building block of Microsoft virtualization infrastructure.
- Other components contribute to creating a fully featured platform for server virtualization

(i) Windows Server Core (WSC):

- To increase the performance of virtualized environments
- It is a specific version of the OS with a reduced set of features and a smaller footprint
- it allows for reduced maintenance, reduced attack surface, reduced management, and less disk space

(ii) System Center Virtual Machine Manager (SCVMM):

- a component of the Microsoft System Center suite, which brings into the suite the virtual infrastructure management capabilities from an IT lifecycle point of view.
- Management capabilities of SCVMM including:
  - Management portal for the creation and management of virtual instances
  - Virtual to Virtual (V2V) and Physical to Virtual (P2V) conversions
  - Delegated administration
  - Library functionality and deep PowerShell integration
  - Intelligent placement of virtual machines in the managed environment
  - Host capacity management
  - SCVMM has also been designed to work with other virtualization platforms such as VMware vSphere (ESX servers) but benefits most from the virtual infrastructure management implemented with Hyper-V.