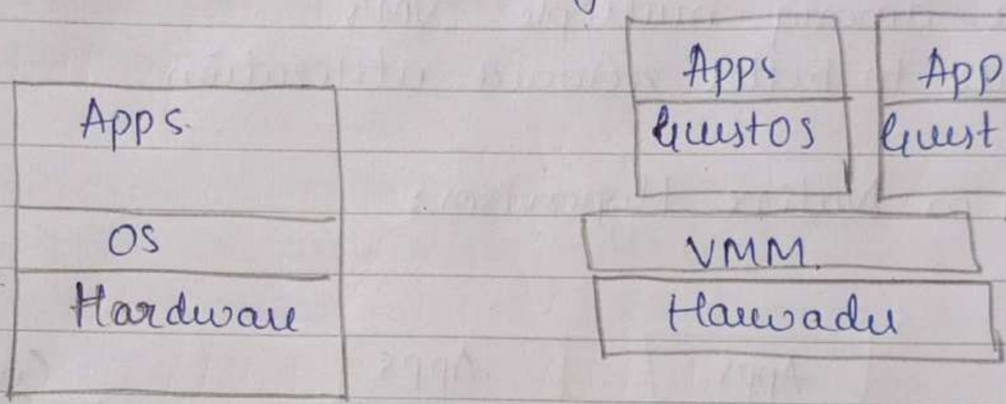
1b) Virtualization is a technology that allows sharing of the physical instance of the single resource or instance among multiple VM's.



Traditional machine.

After virtualization

→ Virtualization leads to better resource utilization. Hence, it enables cloud computing.
→ Virtualization provide separate environment to use.
→ It is logical shared but gives experience as if we are using the original resources.
→ Virtualization is managed by a middleware between hardware and the platform called as Virtual machine monitor or Hypervisor.
→ It acts as interface between hardware and virtual machine.
→ Virtualization allows multiple OS and apps to run of the same server at same time, and results in lower costs and increases efficiency.

Levels of virtualization

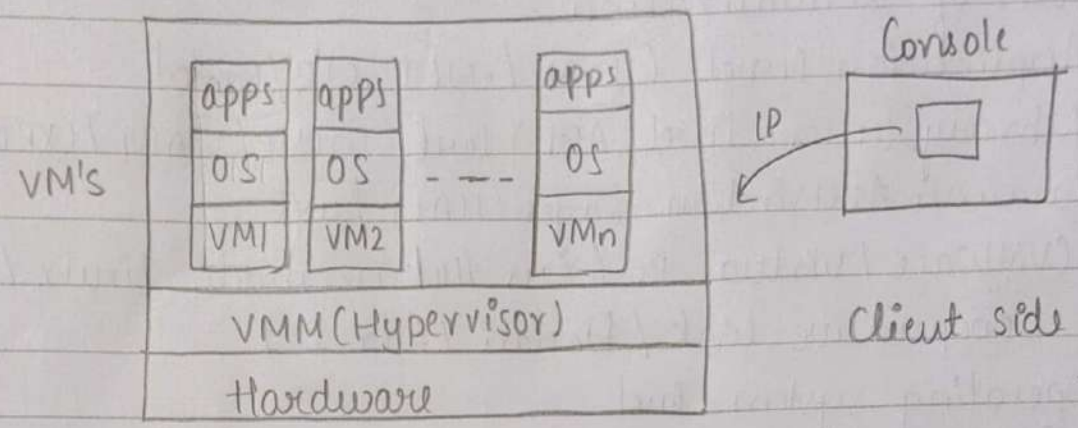
1 Application level (JVM /.NET CLR/Panot
2 Library (user - level API) level (WINE/ WABI / LXRUM/vCUDA)
3 Hardware Abstraction Layer (HAL) Level
(VMWare / Virtual PC / Xen /L4/user mode linux /
cooperative link / Denali / Plex 86)
4. Operating system level
(Jail /Virtual environment / Ensim's VPS / FVM)
5. Instruction set architecture (ISA) level.
(Bochs / Crusoe / QEMU / BIRD / Dynamo)

1. ISA level: It is performed by emulating a given ISA by the ISA of host machine the basic emulation method is through code interpretation. An interpreter program interprets the source instructions to target instructions and one source instruction may require few or 100's of native target instructions to perform it's function. Some hardware like BIRD, QEMU, Dynamo performs translation.
Suppose VM's are having RISC architecture and underlying hardware follow CISC architecture all the RISC instructions must be converted to CISC using binary translation technique.

RISC (One instruction one task)
↓ binary translation
CISC (one instruction multiple task)

Hence this requires adding processor-specific software translation layer to compiler. But this translation is time-consuming which is a demerit.

Hardware Abstraction Level (HAL) virtualization



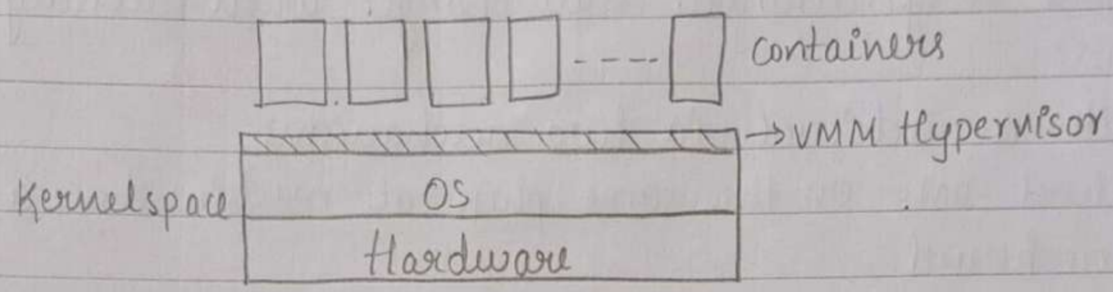
→ Virtualization is performed on top of hardware. As it is done on bare hardware it is called bare-metal VM.
→ It generates virtual hardware environments for VM's, and manages the underlying hardware through virtualization
→ To mange VM's we have client side client interacts with the system using console. through console we can access IP of the hardware and start creating, managing the VM's.
→ Typical Systems: VMware, virtual PC, Denali, Xen.

Merits:
→ High performance and good application flexibility

Demerits:
→ Each VM's OS image should be in memory. So storage is effected
→ If we perform some tasks and reboot it other applications are affected. No isolation and other VMs are affected
→ Highly expensive to implement

01fe18bcs211
Pg No. 4/19

Operating System level Virtualization



→ It is an abstraction layer between traditional OS and user applications

→ OS level virtualization creates isolated containers on a single physical server and OS instance to utilize the hardware and software in data centers. these containers are like real-servers

→ It enables multiple isolated VMs within a single operating system kernel. Therefore the Kernel space is shared among multiple VM's. this kind of VM is often called as virtual execution environment (VE), virtual private systems (VPS) or simply container.

→ Each VE or container has it's own set of processes, file-system, routing table, firewall rules, user accounts, network interfaces with IP addresses etc.

→ Although, containers can be customized for different people, they share the same operating system Kernel. Therefore OS-level virtualization is also called as single-OS image virtualization

Advantages:

(i) VM's at the OS level have minimal startups/shutdown costs, low resource requirements and high scalability

(ii) for an OS-level VM, it is possible for a VM and it's host environment to synchronize state changes when necessary

These merits are achieved via two mechanisms
(1) All OS-level VMs on the same physical machine share a single OS kernel
(2) the virtualization layer can be designed in a way that allows processes in VMs to acess as many resources of the host machine as possible, but never to modify them.

Disadvantages:
(1) All VMs at operating system level on a single container must have same kind of guest OS
(2) Poor application flexibility
(3) Access requests from VM needs to be redirected to the VMs local resource partition on the physical machine

01fe18bcs211
Pg No 6/19

Library Support level

→ It creates an execution environments for running alien programs on a platform rather than creating VM to run the entire operating system
→ It is done by API call interception and remapping
→ Typical systems: Wine, WAB, LxRun, Visual MainWin
→ Merits: Less Implementation effort
→ Demerits: Poor application flexibility and isolation
→ WINE is implemented to support windows application on top of Unix hosts

User Application level

→ It virtualizes an application as virtual environment
→ On OS, an application often runs as process so sometimes it is referred as 'process level virtualization)
→ This layer sits as an application program on top of an operating system and exports an abstraction of a VM that can run programs written and compiled to a particular abstract machine definition
Ex: Java Virtual Machine (JVM), .NET CLT, Parrot)
→ Merit: Best application isolation
→ Demerit: Low performance, low application flexibility and high implementation complexity.

01fe18bcs211
Pg No 7/19

1c) Cloud Design objectives

1) Shifting computing from desktops to data centers:
→ Computer processing, storage, and software delivery is shifted away from desktops and local servers and toward data centers over the Internet

2) Service provisioning and cloud economics:
→ Providers supply cloud services by signing SLAs with consumers and end users. The services must be efficient in terms of computing, storage, and power consumption. Pricing is based on pay-as-you-go policy.

3) Scalability in performance:
The cloud platforms and software and infrastructure services must be able to scale in performance as the number of users increases.

4) High Quality of cloud services:
The QoS of cloud computing must be standardized to make cloud interoperable among multiple providers

5) Data privacy protection: This should be addressed by CSP's (Cloud service providers) to make clouds successful as trusted service

6) New Standards and interfaces: It refers to solving the data lock-in problem associated with data-centers and cloud providers. Universally accepted API's and access protocols are needed to provide high portability and flexibility of virtualized application

01fe18bcs211
Pg No:

01fe18bcs211
Pg No. 8/19

1c Design challenges

1> Service availability and Data lock-in problem.
2> Data privacy and security concerns : As the consumers data can be critical it must be very clear that CSP's provide strict security and make it as a trusted service.

3> Unpredictable performance and bottlenecks : Owing to huge resources they own CSP's should not allocate everything. The optimization of resource allocation plays a major role. There should be a backup in case of emergency.

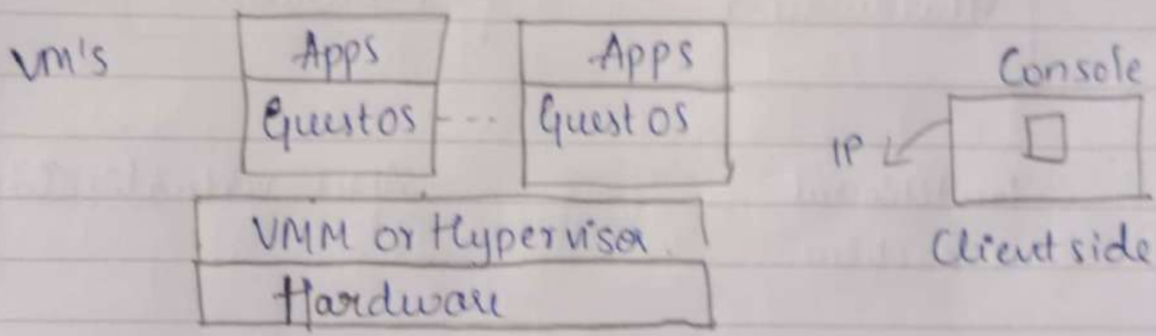
4> Distributed storage and widespread software bugs.

5) Cloud scalability, Intreoperability and standardization. The QoS parameters should be standarized so that CSP's are interoperable. It should be scalable to meet unpredictable demands

6> Software Licensing and Reputation sharing

(a) Virtualization is a technology that allows sharing of the physical instance of the single resource instance among multiple VM's
It leads to better resource utilization

Type-1 or Native Hypervisor:



→ Hypervisor is a form of virtualization software used in cloud hosting to divide and allocate the resources on various pieces of hardware.
→ It is a piece of hardware used to create, delete, manage and monitor VM's.
→ In Native there is hardware level virtualization
→ Hypervisor is called as virtual machine monitor.
→ In type-1 the hypervisor runs directly on the bare hardware. So sometimes it is also called as bare metal VM.
→ This does not require any server operating system

01fe18bcs211

→ It has direct access to hardware resources whereas in hosted VM or fully virtualized VM the privileged instructions needs to undergo binary translation after getting trapped. Hence it is very slower. An in dual mode OS needs to be modified to accomodate some part of VMM into it this is not the case in Type-1 hypervisor.

→ Hypervisor sits on the bare metal computer hardware like CPU memory etc.

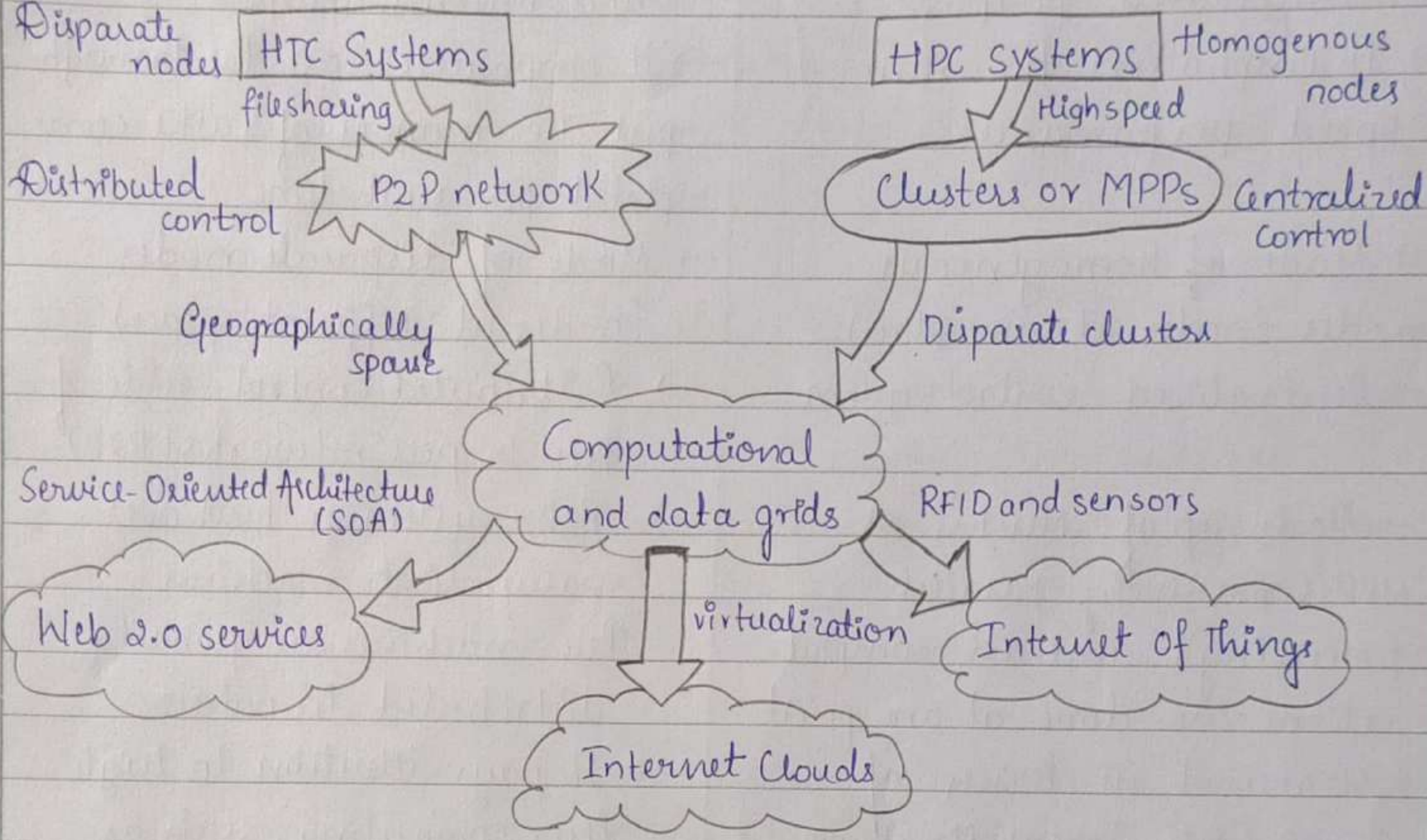
→ Type1 hypervisor is very efficient because they are having direct access to hardware which boots there performance. No extra translation as in hosted or conversion from hypercalls to system calls is needed.

→ The guest OS are layer above the hypervisor.

→ This causes the empowerment of security because there is nothing any kind of third party resource that could attack hardware. whereas in Hosted VM security is less as OS is not in Kernel mode and in dual mode OS is in kernel mode but is modified to certain extent.

01fe18bcs211
Pg No 11 /19

2b)



In 1950's to 1970's there were mainframes that were built to satisfy large business and government organization needs. From 1970's to 1990's use of personal computers increased with VLSI processors and integrated chips. the hardware costs started decreasing.
Then in later 1995-2000 Emerged www and Internet. Today we exchange huge data between the systems via Internet.

Earlier HPC and HTC systems were used.
HPC stands for high performance computing
HTC stands for high throughput computing.

HPC	HTC
→ It is used for scientific and research purpose	→ It is mainly used for market and buisness purpose.
→ It emphasize the raw speed performance	→ It emphasizes on the through-put i.e number of tasks completed per unit time.
→ Made of homogeneous nodes (nodes at same place)	→ Made of disparate nodes (nodes at different place)
→ Centralized control system	→ Distributed control made of Peer-to-peer networks (P2P)
→ Made up of clusters or MPPs (massively parallel processing) and all computations are done at one point	→ P2P system is built over many client machines. Peer machines are globally distributed in nature.
→ Measured in terms of GFLOP (Gigo point floating point operations)	→ It pays attention to high flux computing such as Internet searches and Web Services.

With SOA, Web 2.0 services are available. Advances in virtualization led to growth of Internet Clouds that proliferated a new computing pardigm. The maturity of radio-frequency identification (RFID), Global Positioning System (GPS) and sensors triggered to the development of Internet of things (IoT)

2c

Computing Paradigms.

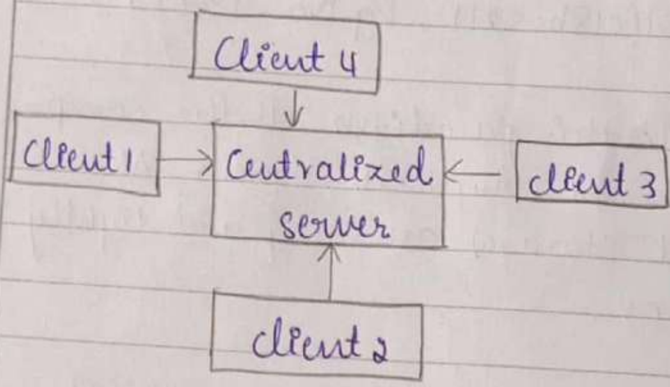
1> Centralized computing: In this paradigm all the computer resources are centralized in one physical system. All resources (processors, memory and storage) are fully and tightly coupled within one integrated OS

2> Parallel Computing: In parallel computing, all processors are either tightly coupled with centralized shared memory or loosely coupled with distributed memory. Interprocessor communication is accomplished through shared memory or via message passing.

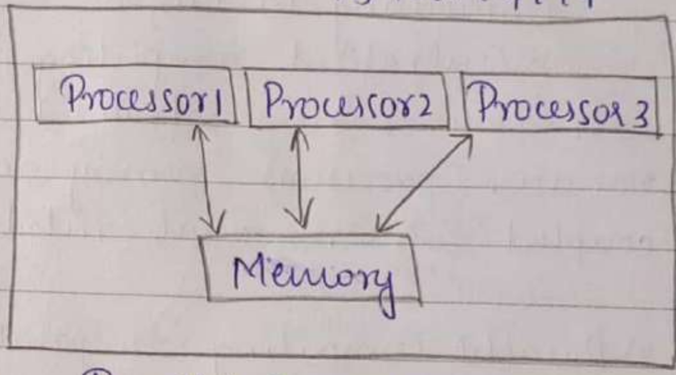
3> Distributed computing
Field of computer science/engineering that studies distributed systems. A distributed system consists of multiple autonomous computers each having its own private memory, communicating through a computer network. Information exchange in the distributed system is accomplished through message passing.

4> Cloud Computing
An Internet cloud of resources can be either be a centralized or distributed computing system. The cloud applies parallel or distributed computing, or both. Clouds can be built with physical or virtualized resources over large data centers that are centralized or distributed

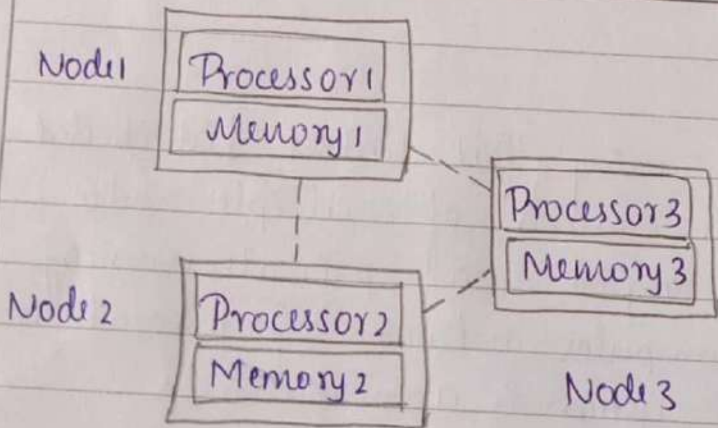
01fe18bcs211
Pg No. 14/19



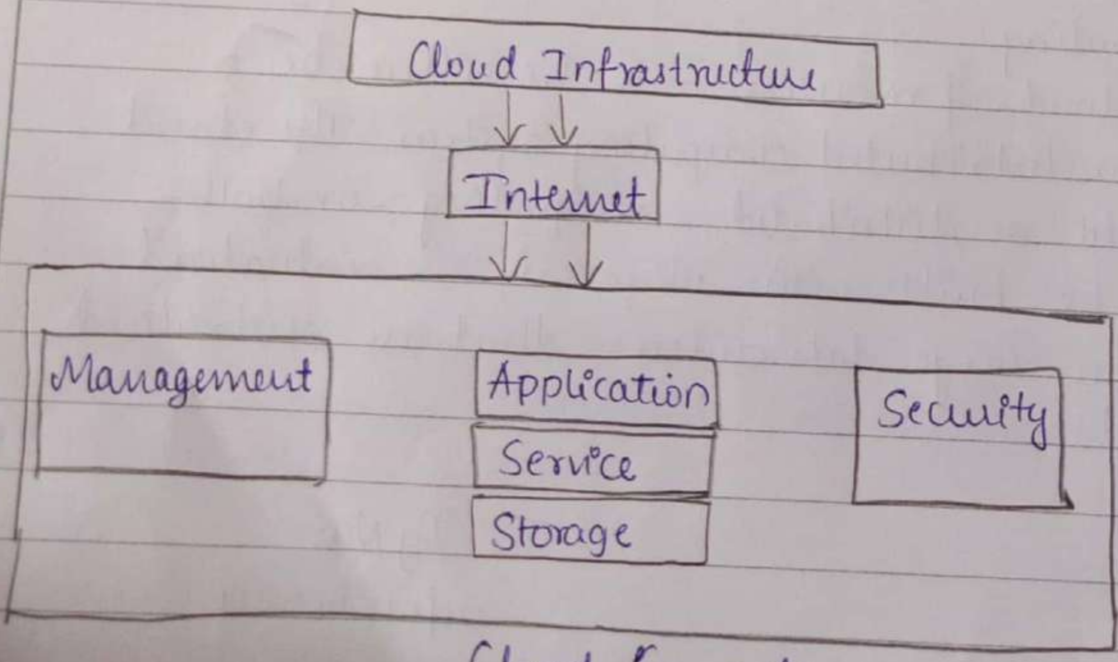
Centralized Computing



Parallel Computing

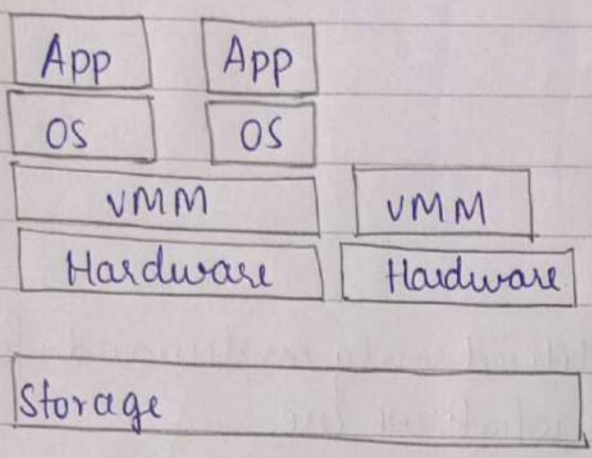


Distributed Computing

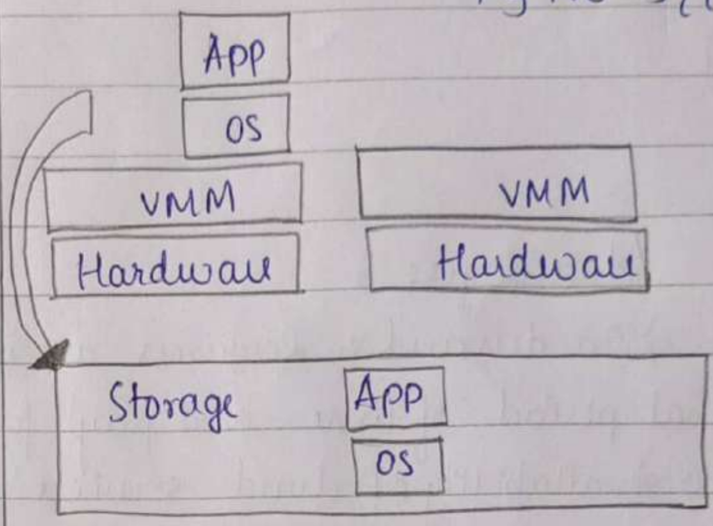


Cloud Computing

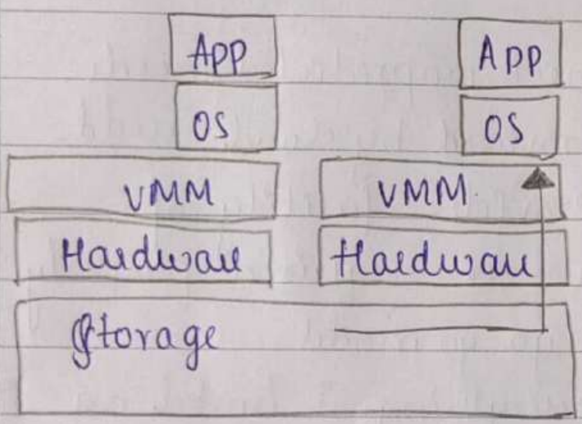
1c)



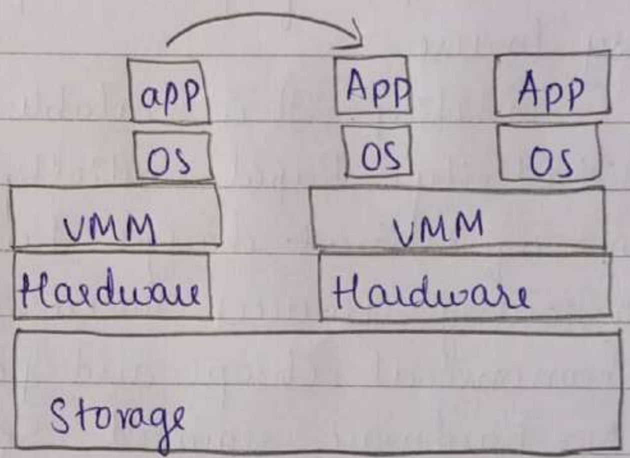
(a) Multiplexing



(b) Suspension (Storage)



(c) Provision (resume)

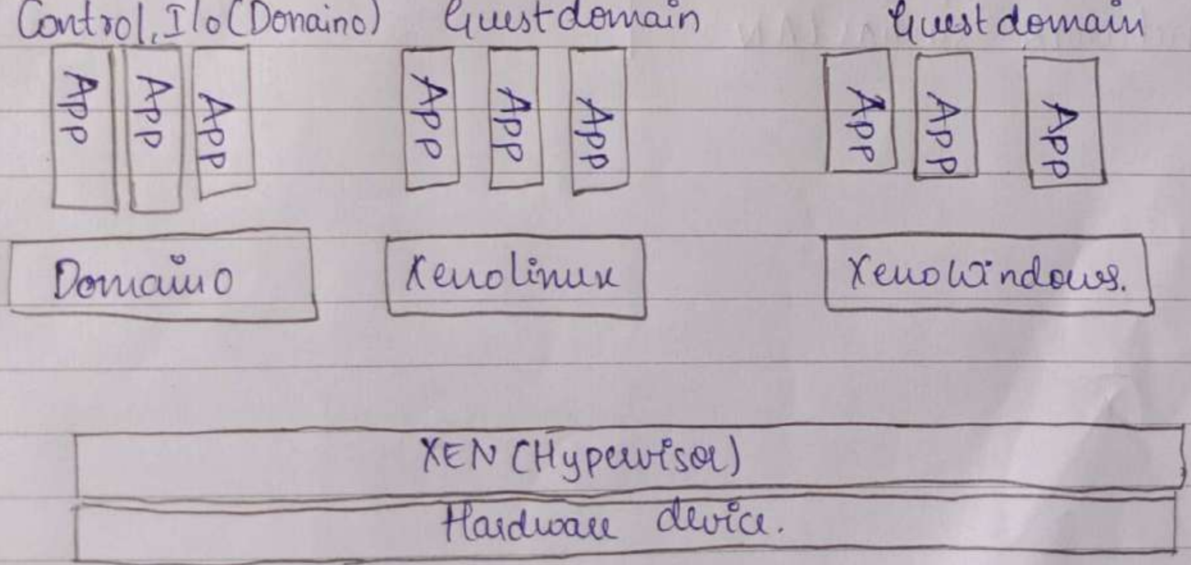


(d) Life migration

(a) VM's can be multiplexed between hardware machines
(b) VM can be suspended and stored in stable storage
(c) Suspended VM can be resumed or provisioned to a new hardware platform
(d) VM can be migrated from one hardware Platform to another

2a

b) Micro-kernel	Monolithic hypervisor.
→ Ex: Xen Architecture	→ Ex: VMware ESXCi
→ Policies and mechanism are separated	→ Policies and mechanism are in VMM.
→ Policies are in domain0 a default VM it takes care of creating, deleting and managing VM	→ Creating, deleting and managing VMs are in hands of VMM
→ Microsoft Hyper V	
→ Kernel takes care of the functions such as physical VM management and processor scheduling	→ This hypervisor are aforementiontioned functions including those of device drivers
→ The code size smaller than monolithic hypervisor.	→ The code size is more than micro-kernel.



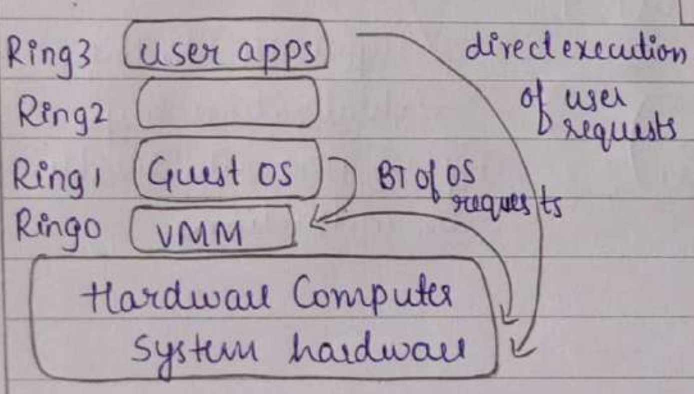
2a

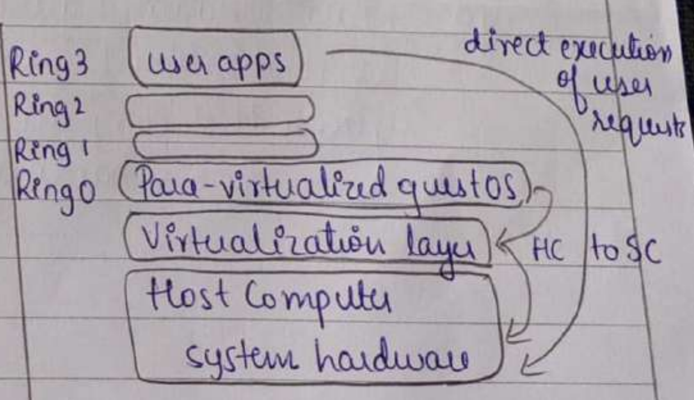
i) Physical Cluster	Virtual Cluster.
→ The all machines are physically connected at one place.	→ the cluster or group of machines forming cluster virtually.
→ The nodes must be Physical machine.	→ The nodes can either be physical or virtual machines.
→ Poor resource management	→ Better resource management.
→ Provisioning is static.	→ Provisioning is dynamic
→ No load balancing technique	→ Load balancing with live migration
→ Connected by physical network such as LAN	→ Connected by internet

2a) a)

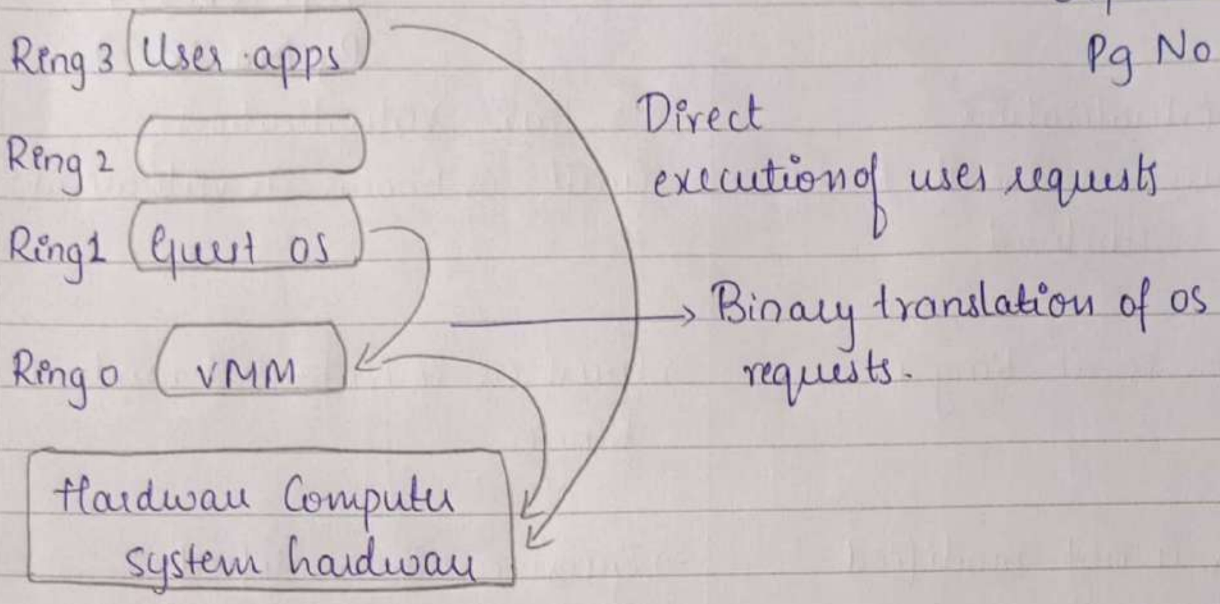
01fe18bcs21
Pg No 18/19

Full virtualization	Para- Virtualization
→ the guest OS doesn't know it is virtualized	→ Guest OS knows it's virtualized
→ Guest OS is at Ring 1	→ Guest OS is in kernel mode at Ring 0
→ Guest OS is not modified	→ Guest OS is modified
→ It is slower than para-virtualization	→ It is faster in operations
→ It uses binary translation (BT) and direct approach as a technique for operation	→ This uses hypercalls at compile time for operations
→ It is less secure as OS is away from hardware	→ It is more secure as OS is part of Kernel
→ It is more portable and compatible	→ It is less portable and compatible
→ Ex: Microsoft and parallel System	→ Ex: Xen Architecture, VMware
→ Previliged instructions are binary translated and then executed	→ Priveledged instruction are trapped as hypercalls and converted to system calls





01fe18bcs211.
Pg No 19



Full virtualization