23/03/2021

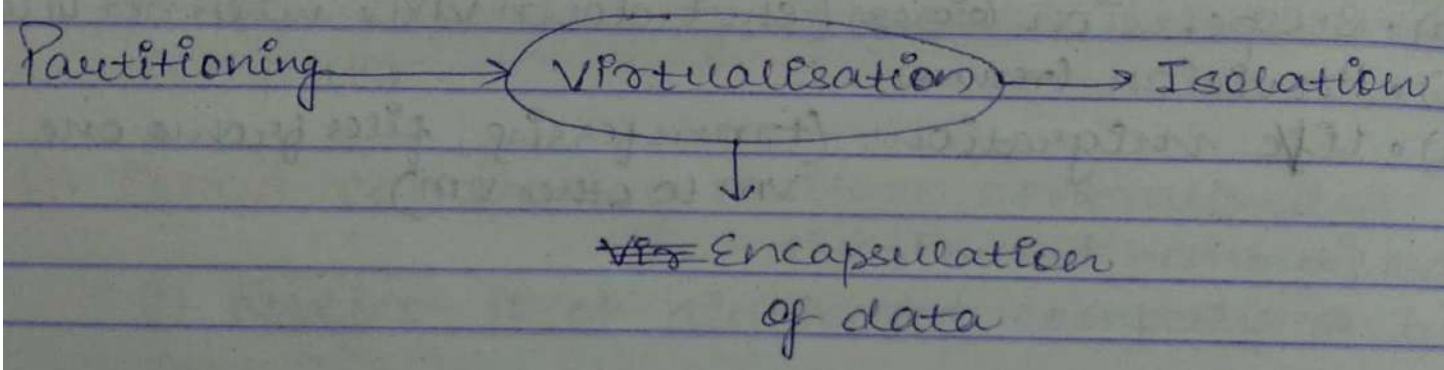
Chapter 02
Virtual Machines & Virtualization of clusters

- Virtual m/c & traditional m/c difference.

★ **Virtual machine:**
A representation of a real m/c using software that provides an operating environment which can run or host guest OS.

★ A middleware b/w hardware & virtual m/c represented in system, is known as Virtual m/c monitor (VMM) or hypervisor.

★ **Features of Virtualization:**

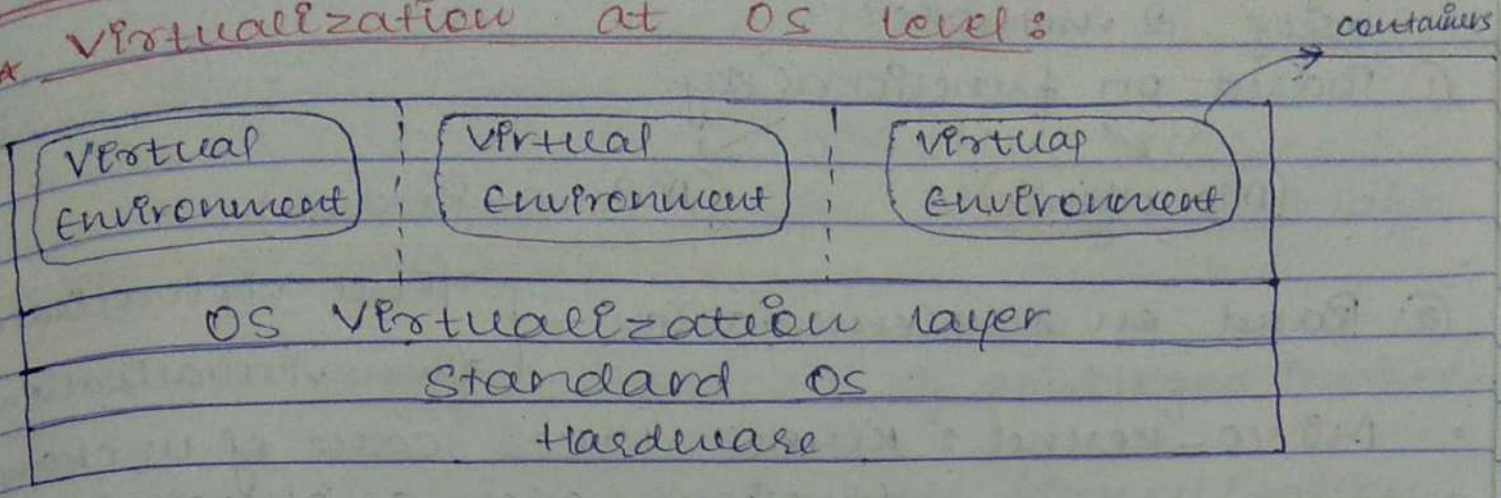


★ **Levels of Virtualization:**
- Application level
- Library
- Operating System level
- Hardware Abstraction layer (HAL)
- Introduction set architecture level (ISA)

Refer in // Bottom up approach.

MIPS: million Instruction per second.

* Virtualization at OS level:



* Three requirements for a VMM:

- VMM should provide an environment for programs which is essentially identical to the original machine.
- Programs which run in this environment should show, at worst, only minor decrease in speed.
- VMM should be in complete control of the system resources. Any program run under a VMM should exhibit a function identical to that which it runs on the original machine directly.

* Hypervisor: It is a hardware virtualization technique allowing multiple Os, called guests to run on a host machine.

Hardware

* OS level Virtualisation can be achieved using 2 methods:

① Based on functionality
   - Monolithic
   - Micro-Kernel

② Based on Implementation
   - Full Virtualisation
   - Para Virtualisation

- Micro-Kernel : Kernel takes care of unchanged functions. such as physical mem management & processor scheduling. E.g xen

- Monolithic : This hypervisor implements all aforementioned functions, including those of the device drivers. E.g Vmware ESXi

- Size of hypervisor code of a micro-kernel hypervisor is smaller than that of a monolithic hypervisor.

- Xen is ~~heav~~ lighter in wt than VMware wrt the functionality.

24/03/2021

* Traditional m/c vs VM

Traditional m/c can be envisioned as a straight line

VM similar to a tree. Execution can go into N different branches.

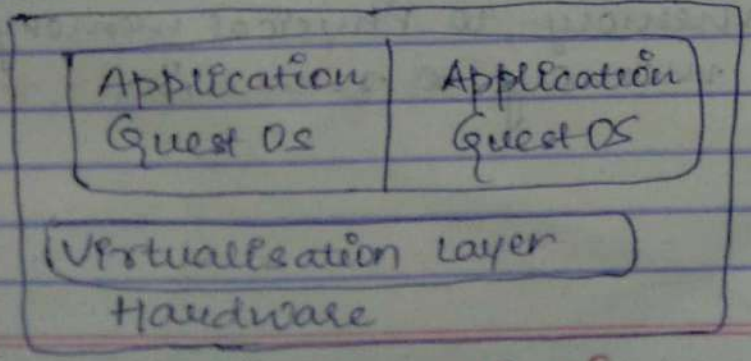
* Based on Implementation:

Full Virtualisation:

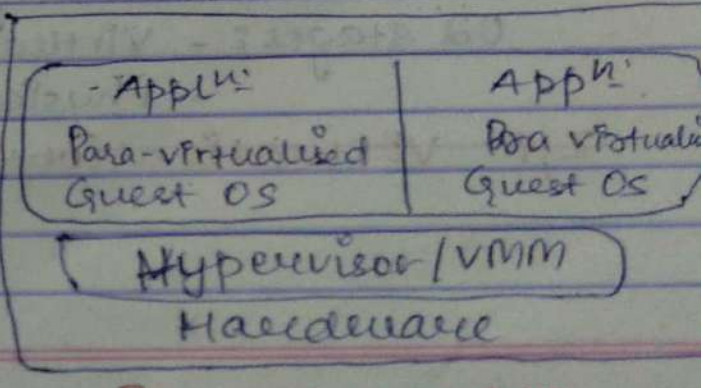
- Non privileged inst. directly accesses the hardware through binary translation of OS requests.
- VMM should take care of privileged instructions.
- Security is more in full virtualisation.
- Adv: no need to modify OS.
- Disadv: binary translation slows down performance.

Para Virtualisation: e.g. KVM

- OS is modified in such a way that below set of critical inst. are taken care by OS.
- The burden of virtualisation layer is less.
- Reduces the overhead but cost of maintaining a para virtualised OS is high.
- Improvement depends on the workload.

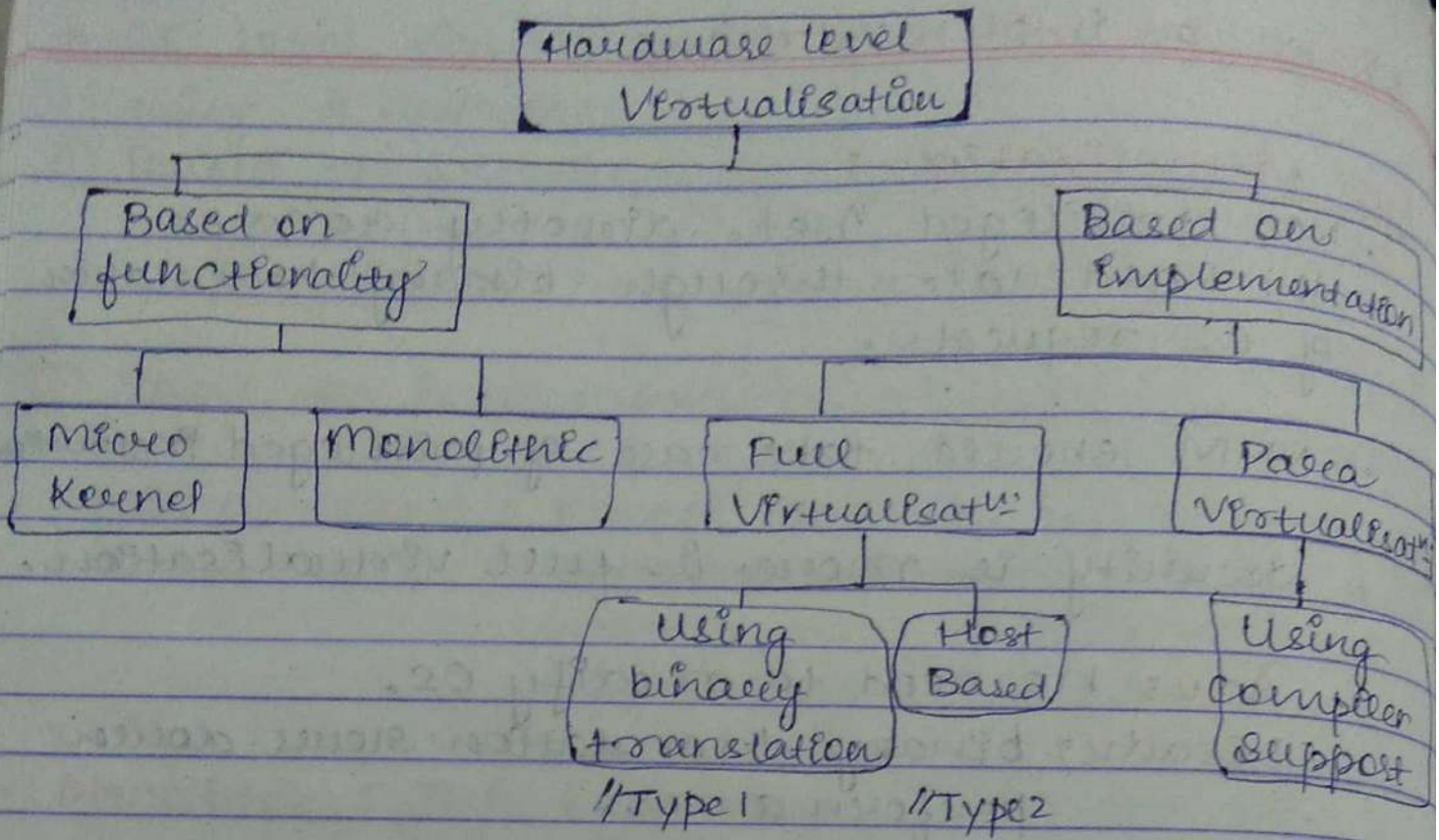


Full Virtualisation



Para Virtualisation
(Type 1)

Overview



* Virtualization of CPU, memory and I/O devices :-

CPU virtualization -

03 types of instructions :-
- Privileged
- Control-sensitive
- Behavior-sensitive

} these 3 set of insts. need to be virtualized.

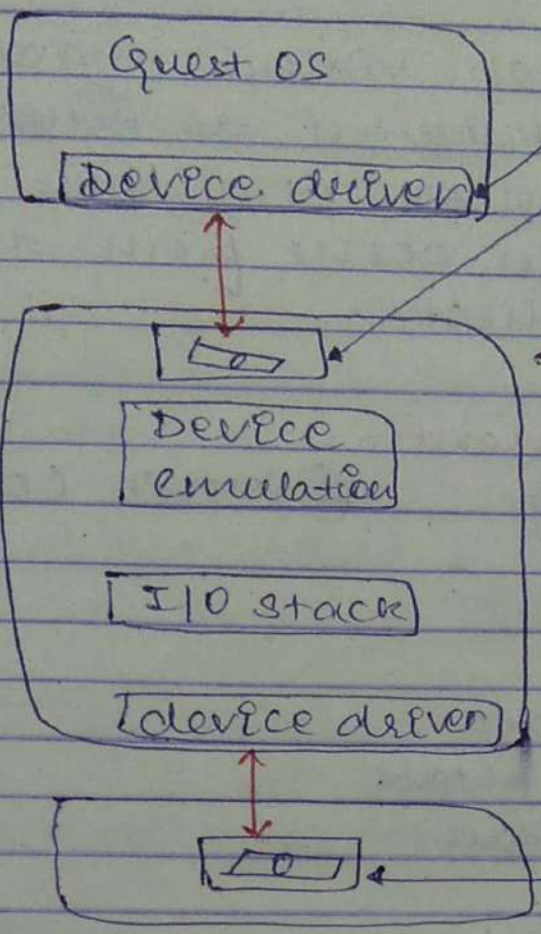
Memory virtualization -

Os maintains mapping of virtual memory to main memory.

02 stages : - Virtual memory to Physical memory
- Physical memory to machine

I/O virtualization -

I/O Virtualization :



- Guest device driver
- Virtual device
- Virtualization layer
  - emulates the virtual device
  - remaps guest & real I/O addresses
  - multiplexes & drives the phy device
  - I/O features e.g. COW disks
- Real device
  - may be different from virtual device.

- Also known as split driver model consisting of a frontend driver & backend driver.

25/03/2021

* Virtual Clusters
* Live Migration

P.T.O

* Physical Cluster: group of machines that are performing same task

* Virtual Cluster: Each virtual machine can be considered ~~as cluster etc~~ as cluster. And those virtual m/c can come from different physical cluster.

* Resource Management -
  ① Load Balancing ② Server Consolidation

* Live Migration

- States - (i) Active State
  (ii) Inactive State
  (iii) Pause State
  (iv) Suspend

- Live migration of a VM consists of following six steps:-

Step 0	Pre-migration	} VM running normally on Host A
Step 1	Reservation	
Step 2	Iterative pre-copy	// Overhead due to copying
Step 3	Stop & copy	} Downtime (VM out of service)
Step 4	Commitment	
Step 5	Activation	// VM running normally on Host B

CHAPTER-2 (LP)

Virtual Machines & Virtualization of Clusters

1). Explain the differences between hypervisor and para-virtualization and give an example VMM that was built in each of the two categories.

Hypervisor :- — **E.g :** Xen

- It is a software structure used to fully emulate computer hardware in software.
- The software layer creates virtual CPUs and virtual memory by intercepting the input/output from the operating system before it reaches the physical hardware.
- Virtual Hard Disk Drives are created by mapping disk read & writes to a single file.
- In the end, the key factor is that guest OS does not know that it runs an abstracted environment & the system administrator should not perceive any significant difference between an OS running on a hypervisor or on a dedicated physical server.

ParaVirtualisation :-

- The concept of ParaVirtualisation is very similar to that of the hypervisor principle.

- A software hypervisor is installed on a physical server & a guest OS is installed into the environment.
- The difference is that the guest OS needs to know that it is virtualized to take advantage of the functions. Operating systems require extensions to make API calls to the hypervisor.

Eg KVM

2). Explain the various levels in implementation of virtualization. Analyze the benefits & disadv. of each virtualization level.

Various levels of virtualization implementation:

Application level : JVM / .NET CLR / Panot

Library (User-level API) level :
WINE / WABI / LxRun / Visual Maincain / vCUDA

Operating System level :
Jail / Virtual environment / Ensim's VPS / FVM

Hardware abstraction layer (HAL) level :
VMware / Virtual PC / Denali / Xen / L4 / Plex86 / User mode Linux

Instruction set architecture (ISA) level :
Bochs / Crusoe / QEMU / BIRD / Dynamo

*Advantage & disadvantage are avl in ppt.

6). Difference between para & full virtualization.

Full Virtualization :- It is the first software solution of server virtualization & uses binary translation and direct approach technique.

In full virtualization, guest OS is completely isolated by the virtual machine from the virtualization layer and hardware.

Para Virtualization :- It is the category of CPU virtualization which uses hypercalls for operations to handle instructions at compile time.

In para virtualization, guest OS is not completely isolated but it is partially isolated by the virtual machine from the virtualization layer & hardware.

Full Virtualization	Para Virtualization
1). In Full Virtualization, virtual machine permit the execution of the instructions with running of unmodified OS in an entire isolated way.	1). In para virtualization, virtual machine does not implement full isolation of OS but rather provides a different API which is utilized when OS is subjected to alteration.
Full Virtualization is less secure.	While the para virtualization is more secure than full virtualization.

3) Full virtualization uses binary translation & direct approach as a technique for operations.	3) while this uses hypercalls at compile time for operations.
4). It is slower than para virtualizn in operation.	4) It is faster in operatn
5) It is more portable & compatible.	5) It is less portable & compatible.
6) Examples of full virtualization are Microsoft & Parallel systems.	6) Examples of para virtualn are VMware & Xen.

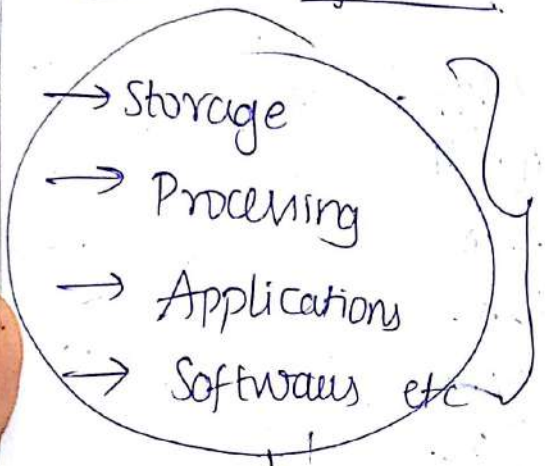
17/5/21

Radhakrishna

DCC

Chapter-1

Features of cloud

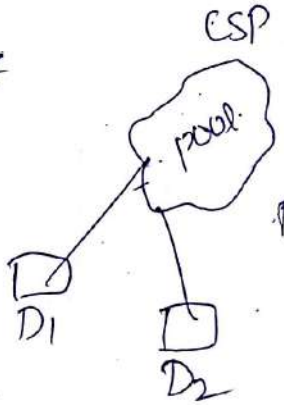


Connected to Internet

The entity who takes care of all these is CSP

CSP → cloud service provider.

provider service plan



access resources
money
processing
applications
utilities.

* Internet away from device not local
* Access anywhere anytime, device independent (when user storage is full, then he can push data into cloud & use, whereas he want from anywhere through internet)

* On demand Services:- Wtever is required for you, we opt those only (Pay only for the services those you are interested in)

* Unlimited resources: User can buy storage unlimitedly & he can use it how much ever wants (not only storage, many other resources are also available)

* Measured services:- Pay how much you use. (based on us plan we can say)

Advantages of cloud

1) Irrespective of device, OS, platform we are using, we can make use of cloud.

2) On demand:- Whenever required, we use only that time & pay for it & take plans or services acc to demand.

3) Availability of cloud services across the globe is there for 24/7

4) Easy Maintaince:-

→ Easy to maintain, need not worry about backup, safety is also provided for our data & easy to use.

5) Scalability :- As per requirements, scalability is provided.

* If we want more memory in our local device, it is time consuming & expensive too. or even connecting some other storage device is also expensive.

→ Therefore whenever extra memory is required, two or more devices are connected by csp (not knowing to user) & when not required (usage is done resources are released.

increasing resources when required (highly scalable)

→ It can accomodate resources dynamically. & managing properly

* 6) Elasticity :- Rapid elasticity is provided by cloud, suddenly requirement changed

then cloud provides facility of providing resources when & where required rapidly.

7) Economical :- Cheap & pay only for the time you used.

8) Pay as you go :- Pay while using based on your usage (resources / storage etc)

Disadvantages

1) User has to connect to internet to use cloud services, if he cant get proper internet connection, he cant use services

2) There are some offline services available but they pay load on local machines itself so not soo - helpful.

3) Highly internet dependent

4) Data security is not guarentee. as all users are putting data in same cloud.

→ Though they provide safety rules but to an extent, it not safe.

→ Safety is given for private users.

Computing paradigms

Centralized

* All computer resources are centralized & they reside in same physical system.
* The pgm, data everything resides in the physical system, but one control will be there centrally with single P.S (PS)
* Computer centrally control everything (peripherals & tasks etc)
* Since, pgm, data everything resides inside single computer, access becomes easy
* Data security also provided as data is inside device & bounded inside single PS. & even computing is happening internally
* Ease of control is also there here.

disadv

* Since all the data / pgms resides in one single system, if central system fails, the entire system fails
* Increasing capability w.r.t data or anything dynamically, we have

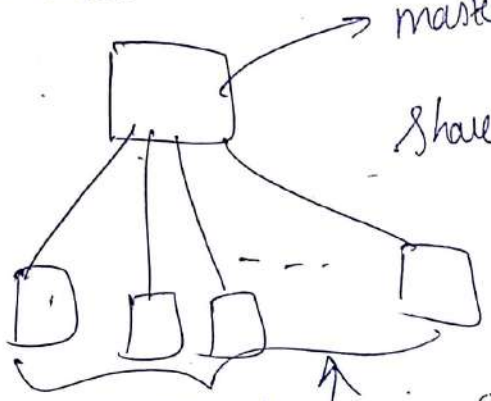
to increase memory resources like memory processors etc

* Scaling can't be done easily

TO overcome these problems we are going with distributed / decentralized computing

Distributed

* centralized work is broken & distributed to no. of nodes.



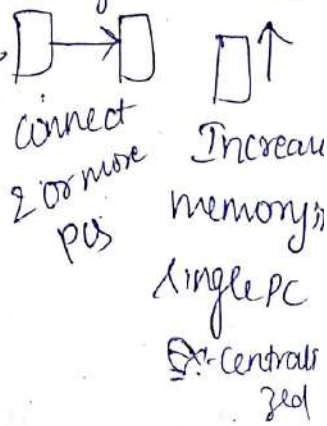
* Once work assigned to nodes is done, consoli-date & give it to master.

* Workload & data can be distributed.

* Distribute tasks across various nodes if work can be done parallely

* As tasks can be done parallely, computation is done fastly.

2 types of Scaling



* Data can be split & given to different workers nodes & can be done work ||y (files can be split & can be given)

* Workers can be of same-type or different.
  - same-type → homogeneous
  - different → like fileserver, data server

* We require some connection b/w master & slave.
  - & even synchronization is must
    - common clock (Global clock)

* even if one worker fails, entire system won't fail. (only it may degrade)

* Failure is less & maintained by master

disadv

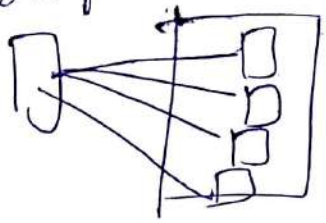
* If master or slave changes in node it has to reflect to overall system.
* It takes time.

* Communications becomes hard if there is any damage in links

Parallel

* Several tasks are ~~dist~~ done lly
* larger / complex problems are broken into smaller problems & work is done lly
* Distributed PC can accoumate this
* This can be done only if there is Scope of ||ism in pgm. (bit level / task / instruction levels)

no dependency



Adv

* It saves time & money

disAdv

* Same as distributed
* Identify the level / extent of ~~max~~ parallism to be done on pgm is challenge

Cloud Computing

→ centralized
or
→ distributed

} or ||el too

All together constitutes C.C

DATE : / /

Distributed cloud computing

→ Utility computing (pay for what you use)
→ cloud - remote storage / On demand provisioning
→ manipulation done on data - computing of resources
→ CSP - cloud service providers.
  i) storage
  ii) any where any place access.
  iii) Build Application.
  iv) Scalability
  v) Pay as you go (economical)
  vi) agility.

Demerits
i> Security
ii> Internet / Bandwidth

The Platform Evaluation

Before cloud,
HPC - high performance computing
HTC - high throughput computing

performance + throughput

cloud

ps - physical system.

PAGE NO.:
DATE: / /

→ HPC is centralized process. where all computations are done at same point
→ MPP - massively parallel process clusters

→ Homogenous nodes - nodes at same place

→ Grids - types of network.

HPC	HTC
→ Clusters of MPP's	→ P2P network
→ GFLOP's	→ distributed control
→ measured with giga floating point operations/sec	↓ physically distributed but connected geographically sparse.
→ Disparate clusters	→ Business purpos
→ Scientific	

Computing paradigms.
⇒ Centralized computing
→ all resources are on one ps and computations
⇒ Parallel computing : In centralized but parallely executed - tightly loosely - different memory locations

it

PAGE NO.:
DATE: / /

3) Distributed computing
4) Cloud computing.

Hype cycle.
→ Graph of technology or its hype in this year
Gartener company contributes to hype cycle

Virtualization
→ cloud can be with/without virtualization

help us to manage virtual machine-
hypervisor is a software helps in VMM

VMM - virtual machine management.

Native -or- type 1 hypervisor
Hosted -or- type 2 hypervisor

ESXI - Dual mode VM.

Virtual machines
i) Physcial machine
ii) Native / Type 1 VM
iii) Hosted / Type 2 VM
iv) Dual mode VM

Docker & containers → OS level virtualization.

PAGE NO.:
DATE: / /

(a) Multiplexing
1:n
1 hardware acts as many hardware create n VM's from one hardware.

(b) Suspension
Shutdown VM not in use.

(c) Resume
resume the idle VMs.

(d) Live migration
transfer app's from 1 VM to another VM on same hardware.

⇒ Services.
Infrastructure → EC2, GCP
Platform → colab, codeboard,
Software → Slides, Drive

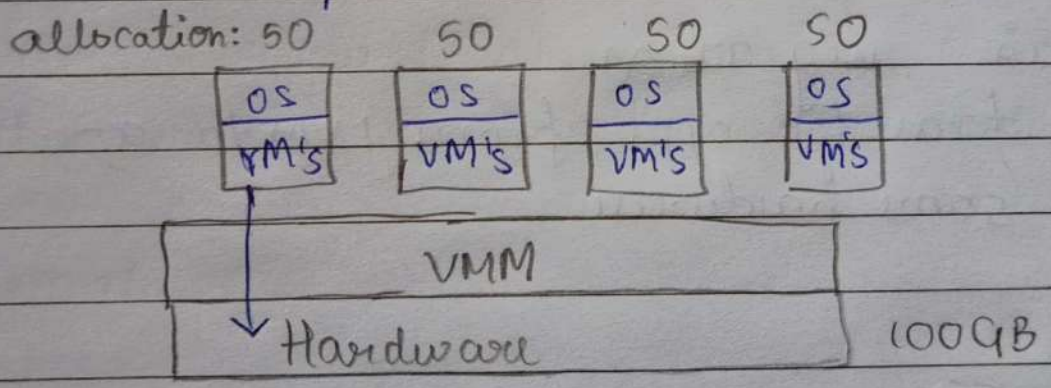
XAAS - Everything as service.

DATE : / /

→ Deployment
Public → Google Slides.
Private → Organization
Community → 2/more organization
Hybrid

Why virtualization?
→ Efficient usage of underlying hardware needs virtualization

→ Allocation ≠ utilization.



→ All VM's don't run simultaneously
→ Even if all are running each can utilize @ max 25gb
→ Ex: for hard
VMM - virtual machine monitor / hypervisor
piece of software that manages underlying

DATE : / /

hardware
→ VM's have their own OS

→ Levels of Virtualization
1. Instruction Set Architecture (ISA) level.
At instruction level.
If VM's is RISC and underlying hardware is CISC
Some hardware like Bochs / Crusoe / QEMU / BIRD / DYNAMO does internal conversion.
→ Binary translation technique converts RISC to CISC instruction.

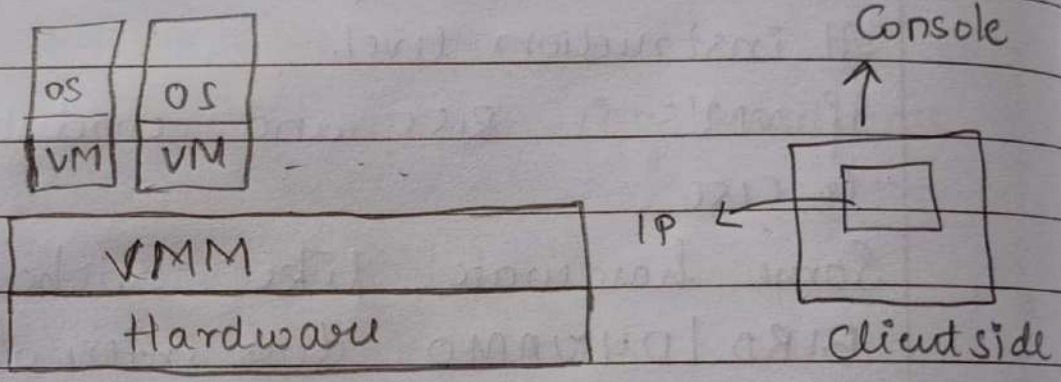
RISC - one instruction one task
↓
CISC - one instruction multiple task.

→ This conversion is time-consuming

→ Any piece of code needs to reach hardware to get executed.

2) Hardware abstraction level (HAL)

Type-1



→ VMM - VMWare ESXI is a hypervisor.
→ to manage VM's we have client side console
→ from console we access IP of hardware and create VM's

Advantage:-

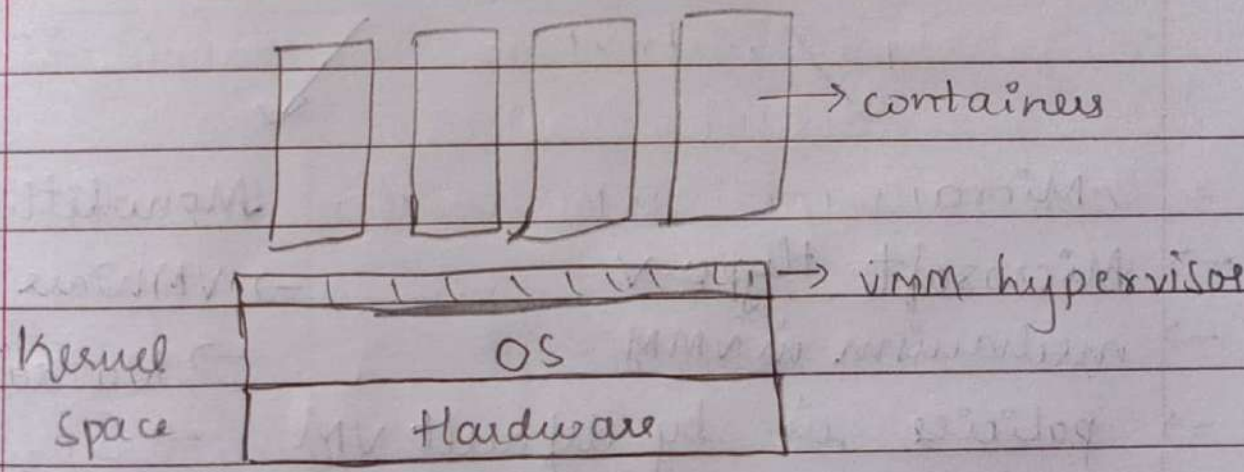
Disadvantage:
→ Each VM's os image should be in memory So storage is affected

PAGE NO.:
DATE: / /

→ If we perform some task and reboot it other apps are affected. No isolations. other VM's are affected.

3) Operating System level virtualization.

→ for one template for every OS. So image of every OS is not used.
→ Hypervisor is in OS itself.



→ Kernel space is shared among multiple containers
→ Applications run isolatedly.
→ Isolation b/w containers.
→ Reboot containers at any time.
→ Containers and host os should be same Windows 10, & but now windows containers can have linux.

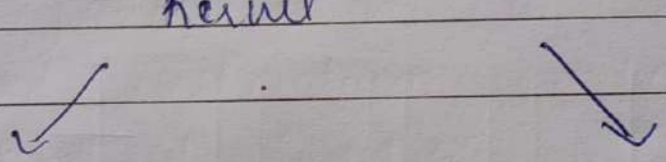
DATE : / /

4) Library Support Level

Windows → Wine ⟶ Linux

5) Application level

Ex: JVM



Micro

→ Microsoft Hyper-V
→ mechanism in VMM
→ policies in by default VM
→ policies & mechanisms are separated

→ Dom0 allocates resources to others VM's any create, delete of VM's is done by Dom0
→ XEN architecture.
→ Hypervisor is smaller in size.
→ Security is less

Monolithic

→ VMWare ESX
→ No domain
→ Size of Software hypervisor is large
→ Policies + Mechanism = Hypervisor

Implement fundion

para & mondathic → VMWARE ESXi

functionalities → micro
→ macro

Implementation → para
→ full

Full Virtualization

In OS

Ringo

User mode
Kernel mode
H/w

Kernel interrupt handler handles system call.
So when privileged instruction is called/execute from user the " " trigger or is trapped by OS and triggers interrupt handler.

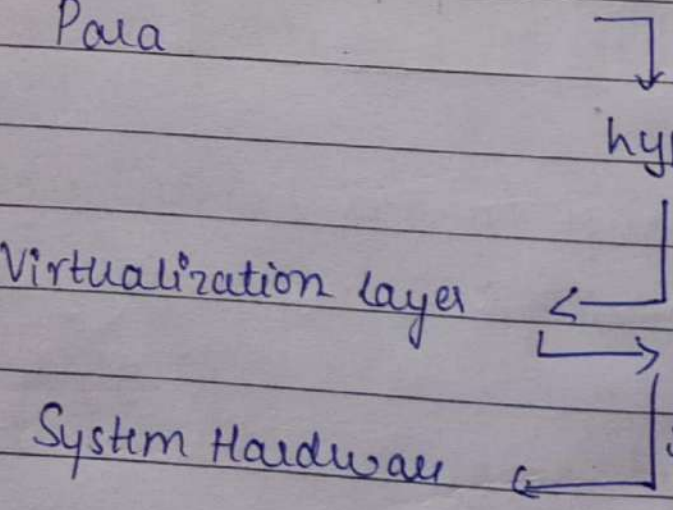
In virtualization it is trapped by VMM and binary translation takes place & execution on hardware is carried out.

DATE: / /

Guest os doesn't know it's virtualized. This binary translation is time-consuming. So non-privelged are directly executed on hardware as they don't effect hardware / security.

Para-virtualized

→ Guest os knows that it is virtualized
→ Guest os is in kernel mode and is modified to be compatible with hardware
→ So privelged instructions are trapped at Para-virtualized Guest os
→ Hyper-calls to virtualization layer is made when trap occurs



hypercalls are converted to system calls. No binary translation is required.

PAGE NO.:
DATE: / /

Memory Virtualization

Shadow Paging → manipulations done in virtualized memory that will be reflected in actual physical memory

Shadow tables.

I/O virtualization

Ex- VMWare ESXI - full virtualization of I/O.
Xen hypervisor - para-virtualization of I/O

1. full-device emulation
   emulation of devices using software

2. Para-virtualization
   In domain 0 all I/O requests are propagated and device drivers in Domain-0 gives the access. All device drivers are in default Domain 0 VM

→ Multiple VM's can access same port
→ All VM's have direct access to virtualization layer. VM's access emulated I/O device.

3) Direct I/O
→ Not preferred as it can harm hardware.

DATE : / /

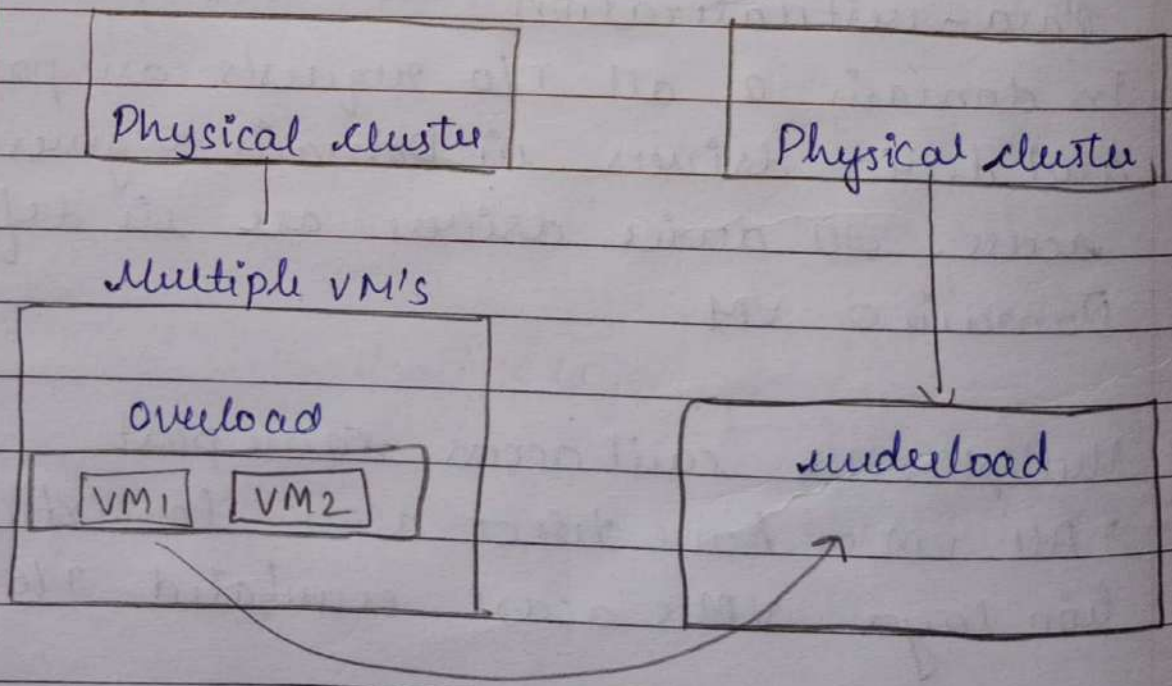
Live VM migration
We migrate application

→ One cluster might be overloaded so we migrate from overloaded to underloaded VM to attain balance and improve efficiency. And also for efficient power usage and better resource utilization
Cold / dead

→ Application is shut down and migrated to another VM

Hot / live

→ Application is up and running and we migrate to other VM's



→ for efficient power usage migrate one application to another cluster and shut down all

other VM's.

Live VM migration Steps:

→ Cloud manager takes care of loads, power usage and migrations.

→ for better services live migration is better else application will not be accesible to users and users don't emperience down time which would be caused by shutdown

→ Scenario: In hostA active VM is to be migrated to host B.

Step1: Slt the destination VM in target host

Step2: Take snapshot of memory in hostA (continously)

Step3: Continously copy pages from one VM to another until we find least dirty VM final state is copied. Shutdown source for few ms and take final snapshot and copy to destination VM and up it.

Demerit: Overhead of copying

Merit: Negligible down time
Redirect traffic to hostB using ARP
Applications starts on Host B.

Type1 is more secure than Type-2

PAGE NO.:
DATE: / /

Cloud Platform Architecture over Virtualized Data Center

Services
AS - as service.

1) Infrastructure (IAS) Ex: EC2 instance
2) Plat form (PAS) Ex: IDE's
3) Software (SAS) Ex: Google doc's

Deployment
1) Public cloud → Demerit: Security / Privacy
   → hazel-free (not worried abt maintainm
   → Easy to use, flexible, scalable, reliable
2) Private cloud → more capital, maintainenace (demerit)
   → secured.
3) Community cloud → more secured than public
4) Hybrid cloud → combination
   → critical is kept private
   → else public.

Two / more organizations build it.
→ complex to implement
→ Secured

Public ← Community ← Hybrid ← Private

SLA - service level agreements