University of Applied Sciences and Arts Northwestern Switzerland
School of Computer Science



# Cloud Computing
(cloud)

## Module 2: Virtualization

Prof. Dr. Sebastian Graf (sebastian.graf@fhnw.ch)
Norwin Schnyder (norwin.schnyder@fhnw.ch)

# Input of this Module

- https://learning.oreilly.com/playlists/43827098-7a87-435e-823e-736586b5694c (02-virtualization):
  - Smith and Nair, Virtual Machines: Versatile Platforms for Systems and Processes (2005)
  - Chisnall, The Definitive Guide to the Xen Hypervisor (2007)
  - Portnoy, Virtualization Essentials, 3rd Edition (2023)

- Research papers
  - Popek and Goldberg, Formal requirements for virtualizable third generation architectures (1974), Communications of the ACM, *(Class Materials)*
  - Randal, The Ideal Versus the Real: Revisiting the History of Virtual Machines and Containers (2020), ACM Computing Surveys, *(Class Materials)*

- Several Blogposts and White papers
  - RedHat, What's the difference between cloud and virtualization? (2018)
    https://www.redhat.com/en/topics/cloud-computing/cloud-vs-virtualization
  - VMware, Understanding Full Virtualization, Paravirtualization, and Hardware Assist (2019), *(Class Materials)*
  - De Gelas, Hardware Virtualization: the Nuts and Bolts (2008)
    https://www.anandtech.com/show/2480/9
  - Kalkman, Intel/AMD virtualization isolation and containment (2020)
    https://michielkalkman.com/posts/virtualization-cpu-io-isolation-modeling/
  - Deierling, Achieving a Cloud-Scale Architecture with DPUs (2021), Nvidia Blog
    https://developer.nvidia.com/blog/achieving-a-cloud-scale-architecture-with-dpus/

# Agenda

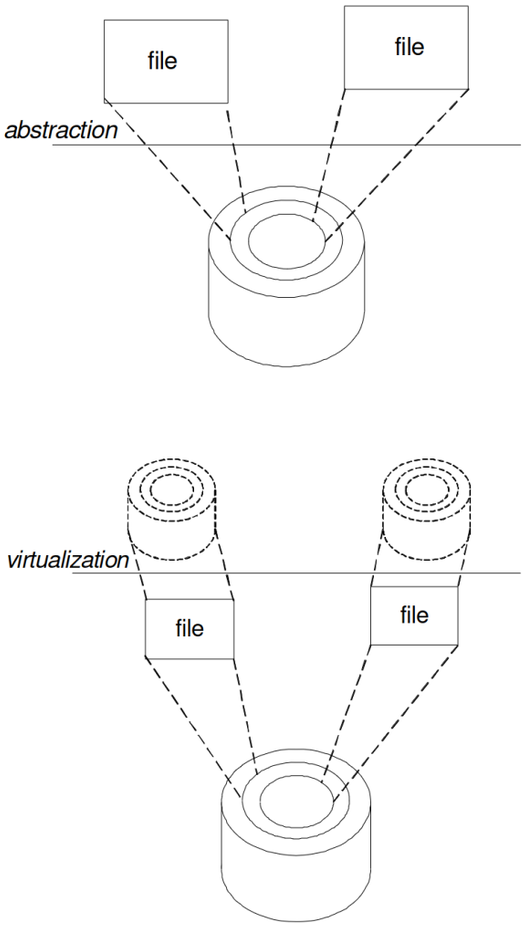
1. Introduction to Virtualization
2. Virtualization Approaches
   - Type I Hypervisors (native)
   - Type II Hypervisors (hosted)
3. Virtualization Technology
   - Intel VT-x
   - Examples (AWS EC2, XEN, KVM)
4. Virtual Resources
   - CPU / Memory
   - Storage
   - Networking

# **Abstraction** vs **Virtualization**

– **Abstraction**: systems are built on multiple abstraction levels
  - Higher levels hide details of lower levels
  - Example: files are an abstraction of the disk

– **Virtualization**: creating a virtual representation of a resource (on a physical machine)
  - Does not necessarily hide any low-level details
  - Example: virtual memory

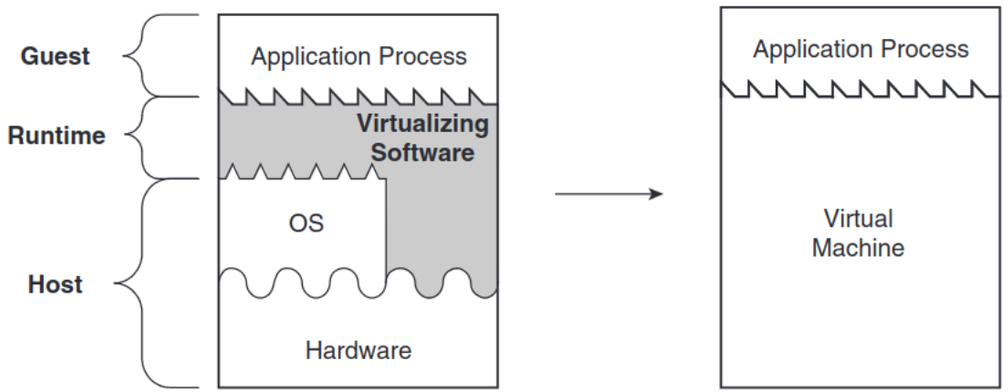
» A cloud (service) is typically created by adding management and automation layers (abstraction) to multiple layers of virtualization.



# **Virtual Machines** (VM)

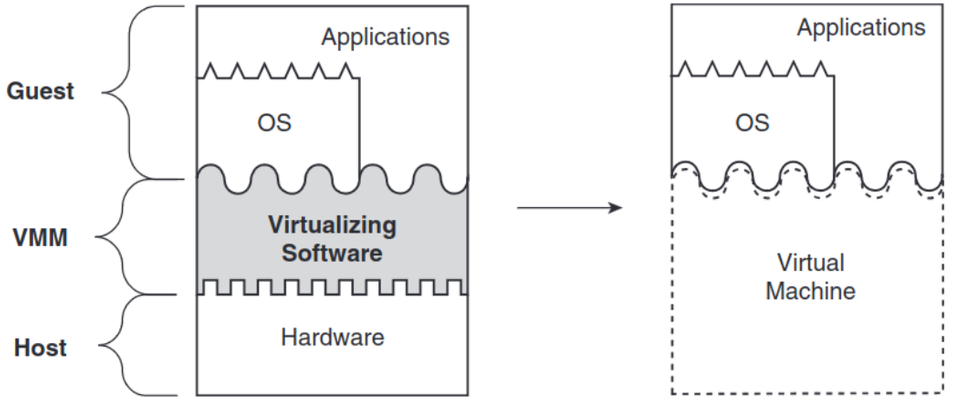
## **Process** Virtual Machines

- Typically using intermediate language (Java bytecode) and make their execution platform agnostic
- Example: Java Virtual Machine (JVM)




## **System** Virtual Machines

- Try to mimic the complete hardware system (CPU, memory, …)
- Example: Xen, KVM, VMware

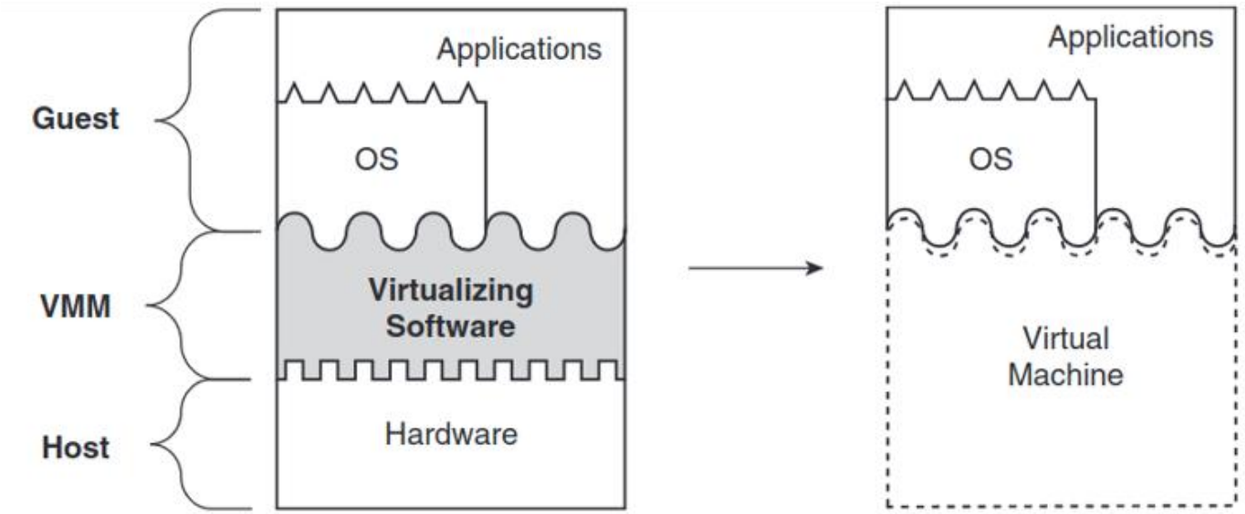



**VMM** = Virtual Machine Monitor or Hypervisor
(thin layer of software that supports virtualization)

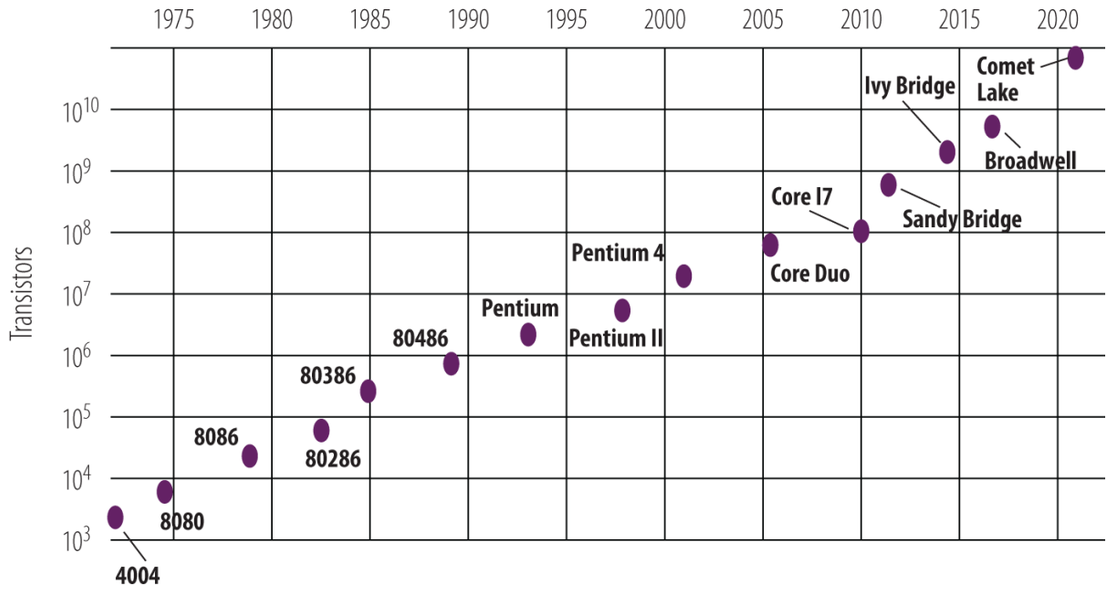
# Requirements of Virtualization

According to Popek and Goldberg (1974) there are three properties of interest when any arbitrary program is run on a virtual machine:

- ***Efficiency*** - minimal overhead, good performance
- ***Resource control (safety)*** - isolation between guest and VMM / host, isolation between guests
- ***Equivalence*** - same results with and without VMM

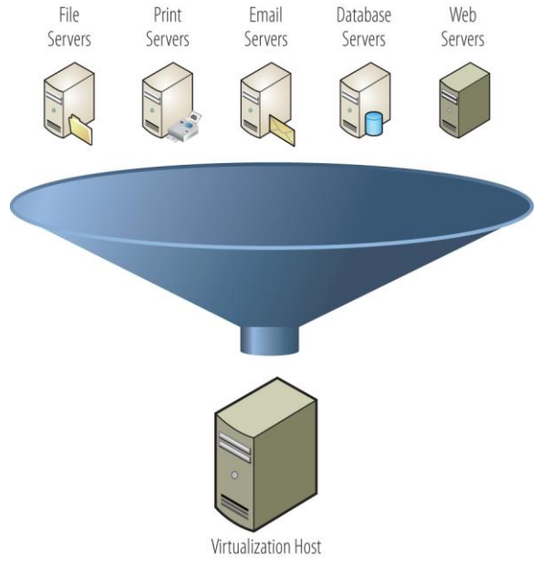


# Understanding the Importance of Virtualization



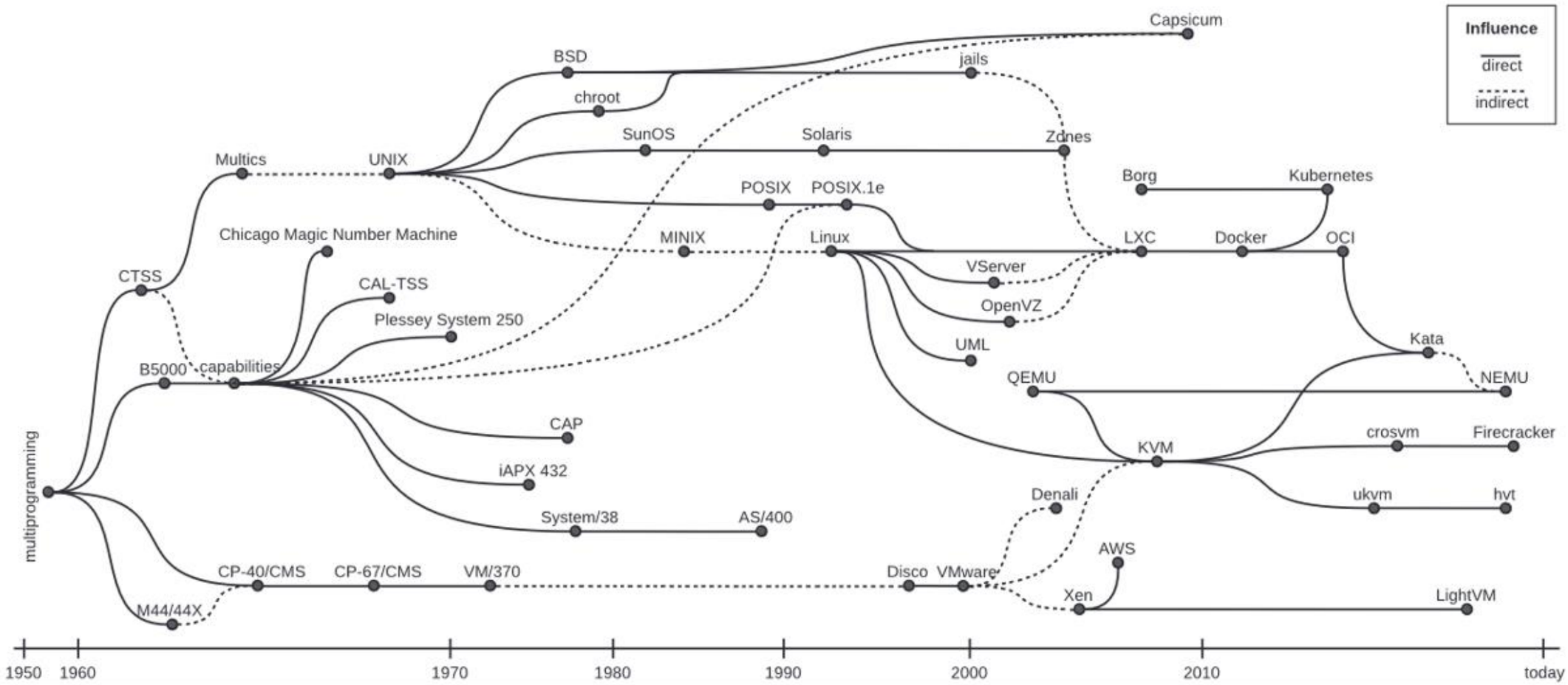

**Moore's law** says that processing power (the speed and capacity of computer chips) roughly doubles every 18 months.

Combination of the effect of Moore's law and the "*one server, one application*" model → servers do less and less work

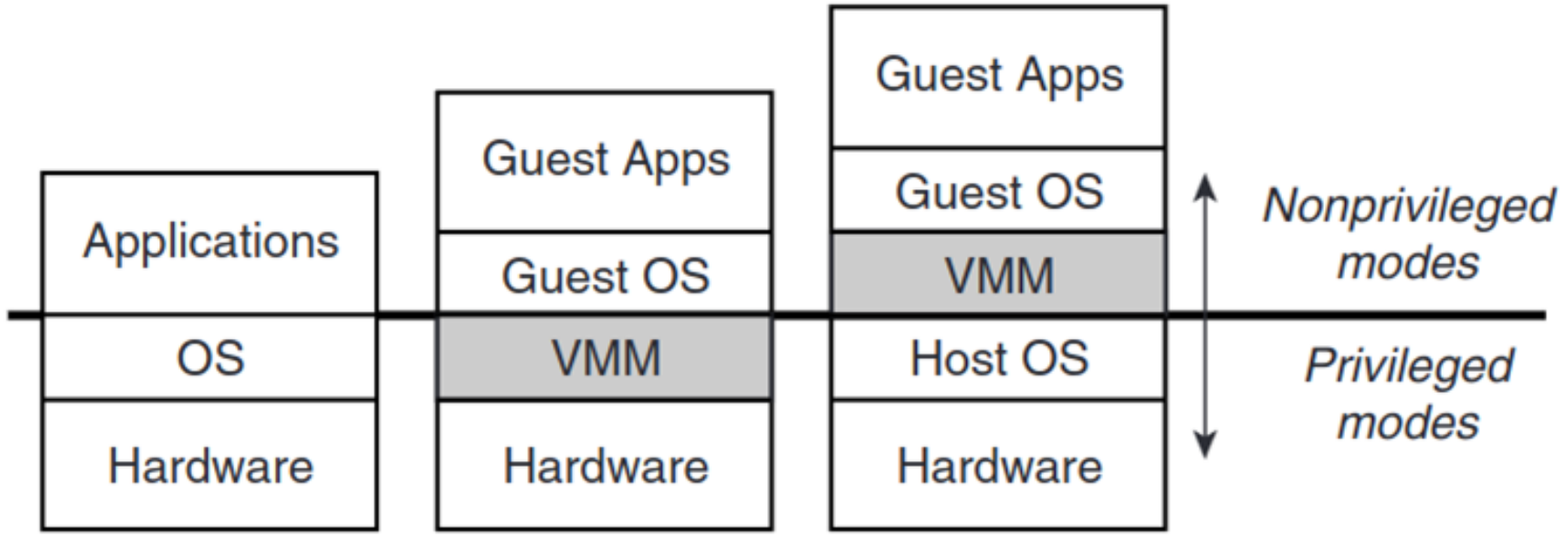



» Virtualization helped to overcome this by reducing footprint and total cost of ownership per server: TCO is 3-10 times the costs of the server itself in 3 year

# Evolution of Compute Virtualization



# Different Types of Virtualization




**Type I Hypervisor**
(native, bare metal)

Examples:
- VMWare ESX
- Citrix Xen Server
- Microsoft Hyper-V

**Type II Hypervisor**
(hosted)

Examples:
- Oracle VirtualBox
- VMWare Workstation

# **Type II Hypervisor** (Hosted Virtualization)

– Hypervisor runs a regular application on top of the host operating system.

– Software-level representation (emulation) of physical resources.

**Advantages**

- Guest applications run within process boundaries (instructions aren't executed directly on host's hardware) → **complete isolation**
- Easy to handle privileged instructions.

**Disadvantages**

- Emulating hardware resources, e.g., modern processors, is difficult.
- Execution overhead (~ 100x slower than direct execution)

» This virtualization approach is not suitable for offering compute (VM) at scale in the cloud because it results in less efficient resource utilization and higher latency.

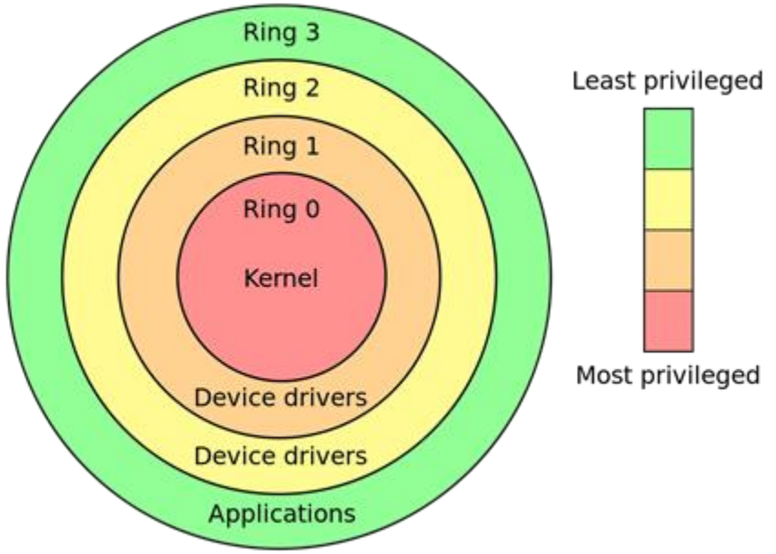
# Type I Hypervisor

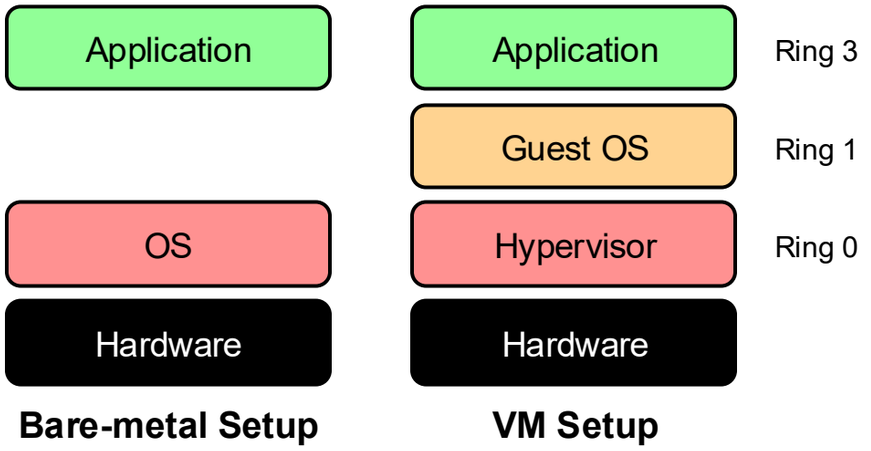
- Full virtualization using binary translation
- Paravirtualization (OS assisted virtualization)
- Hardware assisted virtualization

# x86 Privilege rings

- x86 CPU hardware provides four protection rings.
- Only code in more privileged rings can read and write memory of lower rings.

- **Ring 0** is the level with the most privileges.
  - Interaction with physical hardware (e.g., reset processor, perform I/O operations )



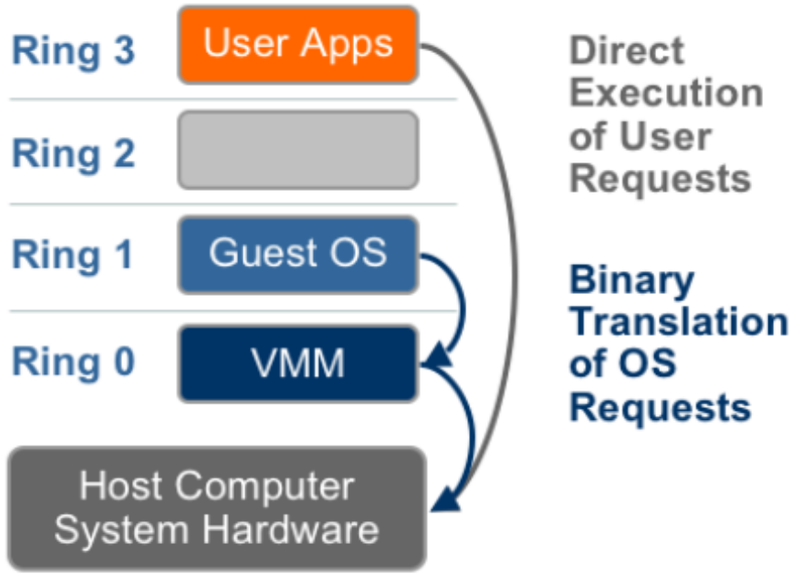



# Full Virtualization using Binary Translation (Type I)

Translates kernel code to replace non-virtualizable instructions with new sequences of instructions that have the intended effect on the virtual hardware. User level code is directly executed on the processor.

**Advantages**
- Allowing the guest OS and apps to run without modification or awareness of the virtualization.
- Direct execution for user-level code
  → high performance

**Disadvantages**
- Difficult to implement VMM
- Translation impacts performance
  (e.g. virtual memory → physical memory → machine memory)

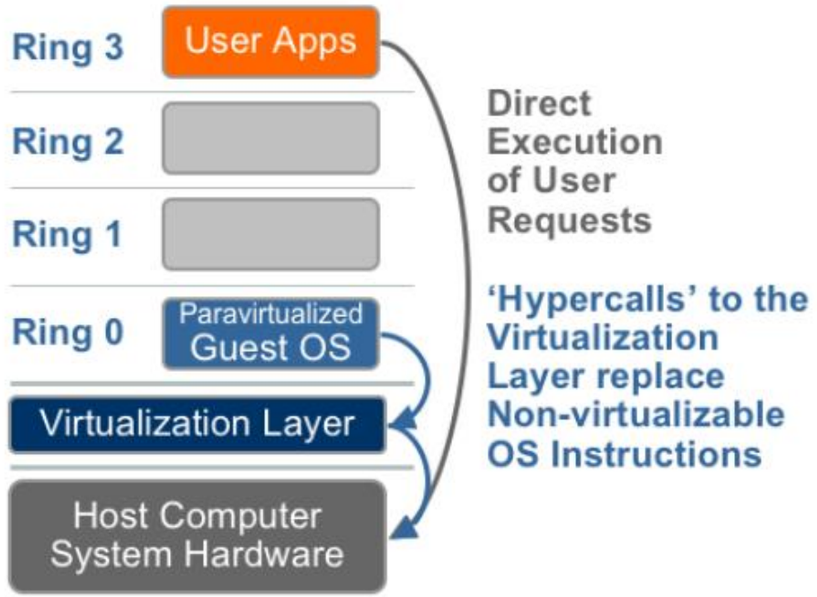


# Paravirtualization (Type I)

Modifying the guest OS kernel to replace non-virtualizable instructions with hypercalls that communicate directly with the virtualization layer hypervisor. It is therefore also referred to as OS Assisted Virtualization.

**Advantages**
- Reduces the virtualization overhead due to fewer context switches and less bookkeeping

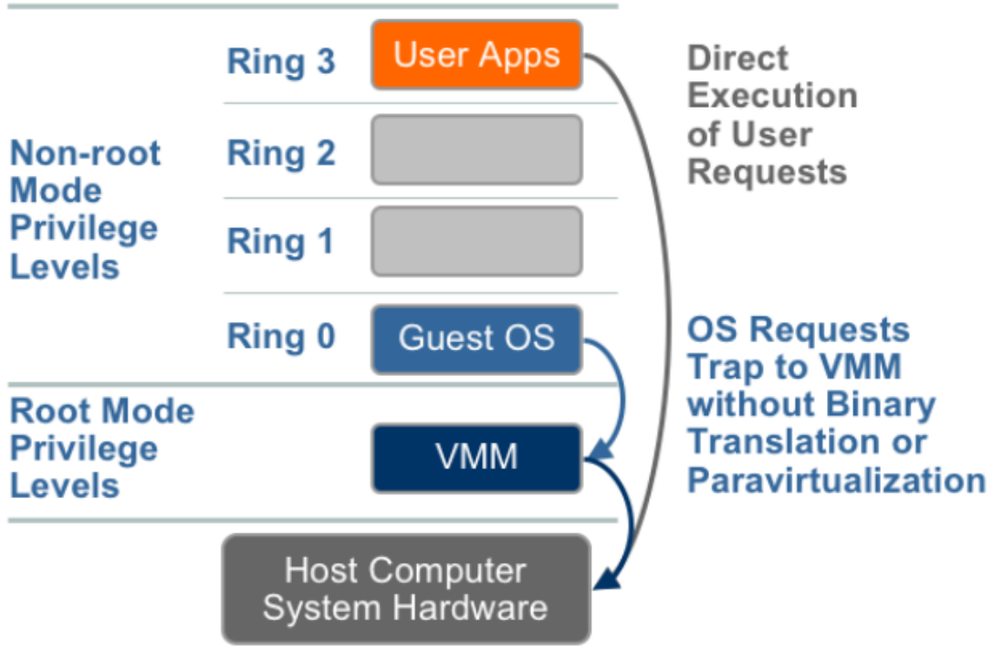
**Disadvantages**
- Poor compatibility and portability
  → cannot support unmodified OS
- Requires substantial modification of the OS




» Typically used in virtual device drivers (e.g., Vmxnet) because CPU and memory paravirtualization is much more difficult.

# Hardware-assisted Virtualization (Type I)

- Hardware extension for virtualization added to the CPU (e.g., AMD-V, Intel VT-x)
- CPU execution mode feature that allows the VMM to run in a new root mode (e.g., VT root mode) below ring 0.
- Allows direct execution of VM on processor until a privileged or sensitive instruction is executed, which are set to automatically trap to the hypervisor.



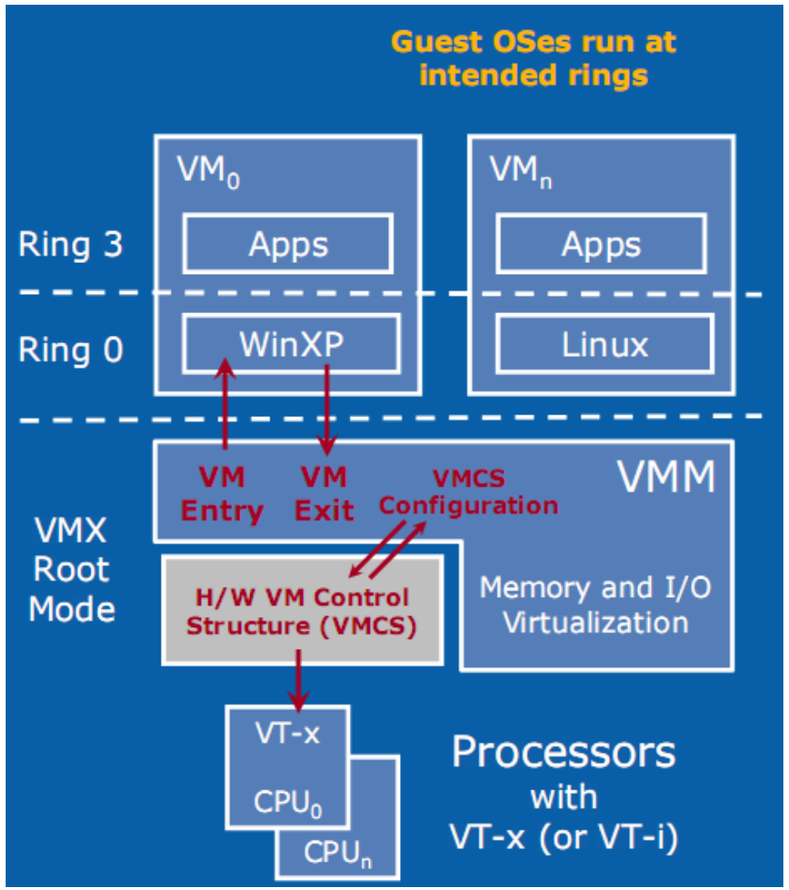

**Advantages**

- Fast and supported on most CPUs today.

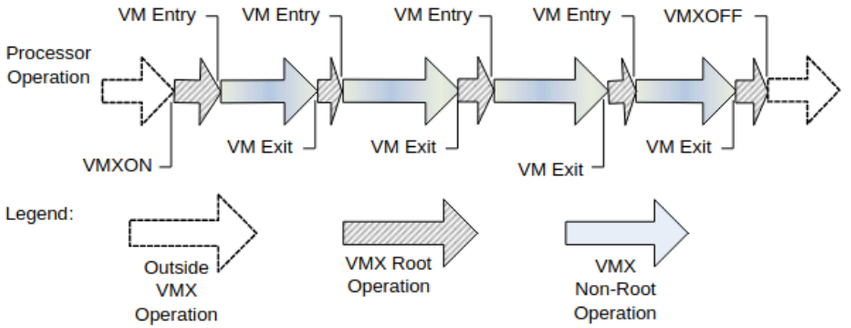
**Disadvantages**

- Focus is mainly on the CPU and memory, but IO virtualization still requires translation.

# Intel Virtualization Technology (VT-x)



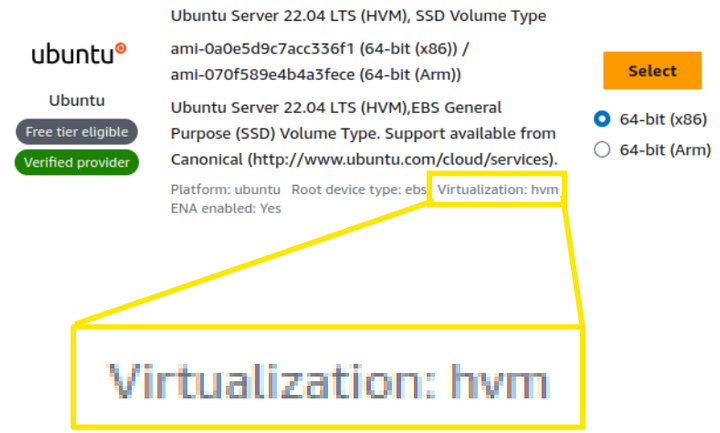

- Virtual Machine Monitor (VMM) configures on the **Virtual Machine Control Structure** (VMCS) which privileged instructions (e.g., exceptions, interrupt, IO) trigger a `vmexit`.
- VMCS stores context info which contains e.g., CPU register values to save and restore during transitions between root and non-root.
- Using `vmentry` the VMM can transfer control to VM (one at a time) until `vmexit`.



# Example: Amazon EC2

Amazon Machine Images (AMI) use one of two types of virtualization: **paravirtual** (PV) or **hardware virtual machine** (HVM).

| Characteristic | HVM | PV |
|---|---|---|
| Description | HVM AMIs are presented with a fully virtualized set of hardware and boot by executing the master boot record of the root block device of your image. This virtualization type provides the ability to run an operating system directly on top of a virtual machine without any modification, as if it were run on the bare-metal hardware. The Amazon EC2 host system emulates some or all of the underlying hardware that is presented to the guest. | PV AMIs boot with a special boot loader called PV-GRUB, which starts the boot cycle and then chain loads the kernel specified in the `menu.lst` file on your image. Paravirtual guests can run on host hardware that does not have explicit support for virtualization. For more information about PV-GRUB and its use in Amazon EC2, see User provided kernels. |
| Supported instance types | All current generation instance types support HVM AMIs. | The following previous generation instance types support PV AMIs: C1, C3, M1, M3, M2, and T1. Current generation instance types do not support PV AMIs. |




**PV on HVM**

PV drivers are available for HVM guests (performance advantages in storage and network I/O)

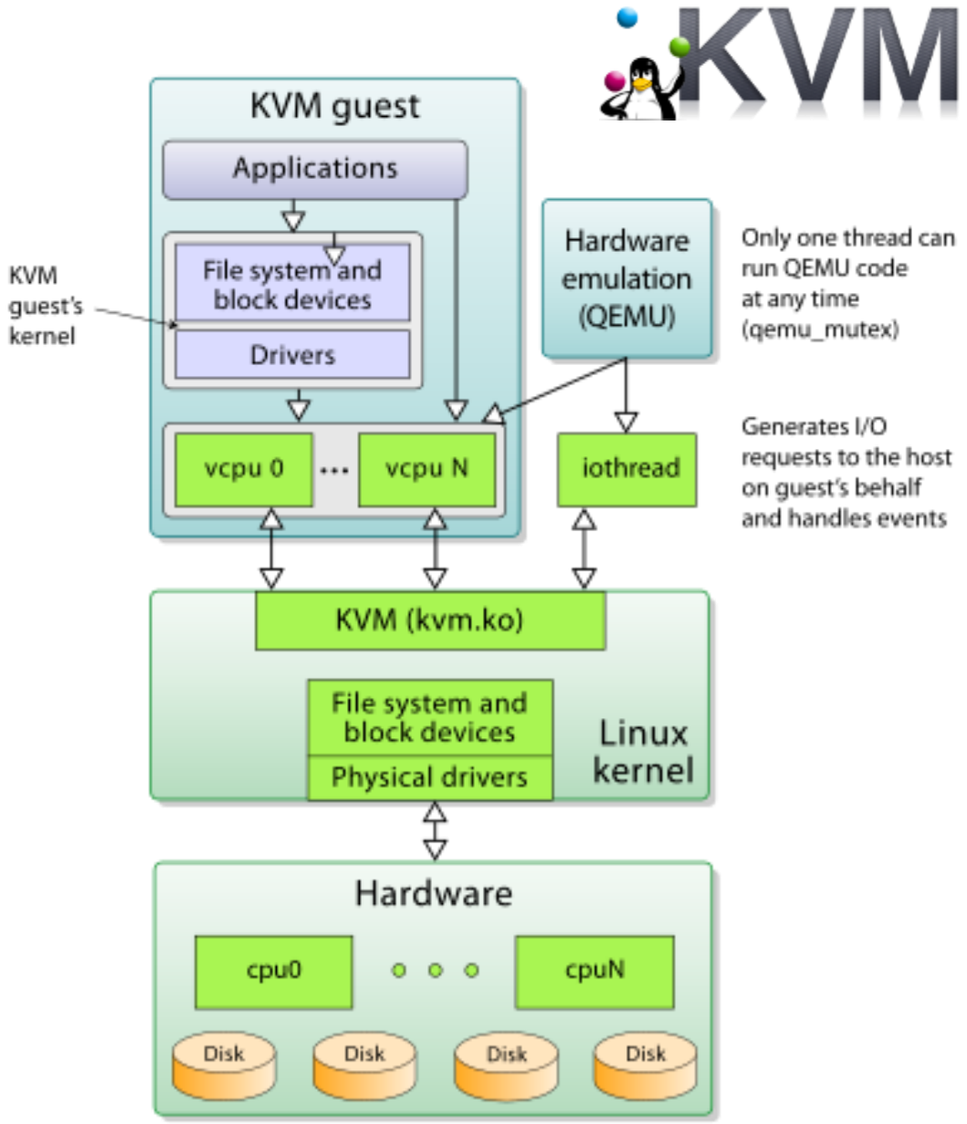
» HVM guests can get the same, or better, performance than PV guests.

# Example: KVM (Kernel Virtual Machine)

- **KVM** is a virtualization technology which uses Linux as a hypervisor (kernel module).
- Part of the kernel since version 2.6.20 (2007) → *Open Source*
- Supports hardware-assisted (Intel VT and AMD-V) and in addition paravirtualization for IO (VirtIO API).

- **QEMU** (Quick EMUlator) is a generic machine emulator and virtualizer.
- Emulates machine's processor including the devices using dynamic binary translation.
- Often used in combination with KVM.

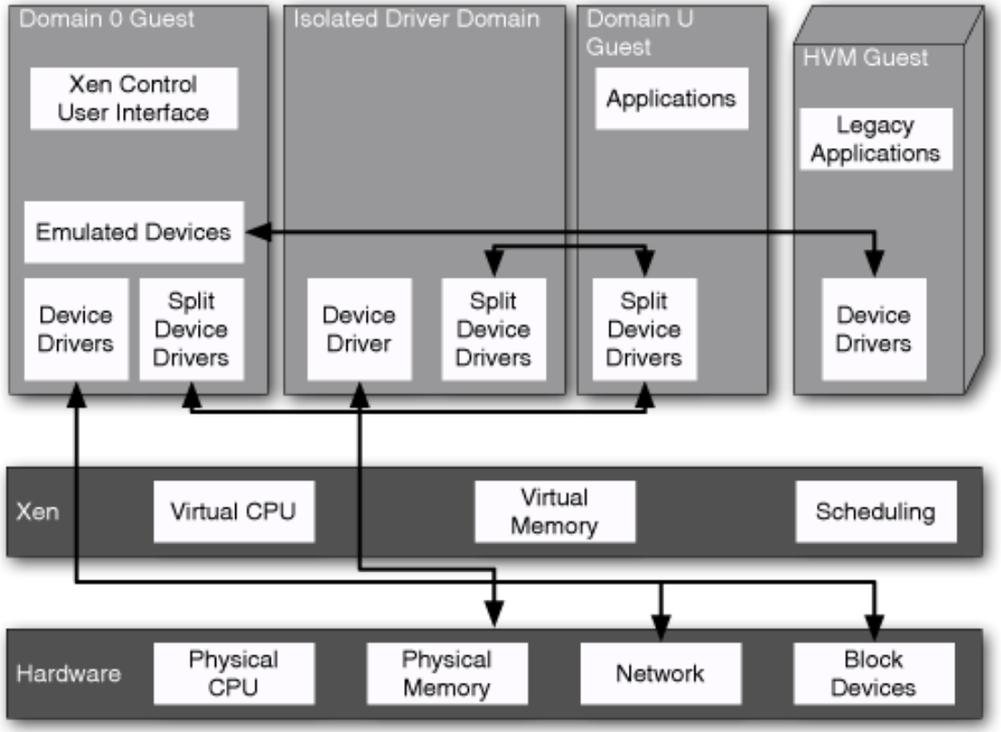
*Which hypervisor type is KVM / QEMU?*



# Example: Xen

- **DomU** (*unprivileged* domain): Guest systems running on Xen hypervisor
- **Dom0** (*privileged* guest): Management System governing access to hardware for DomU
  - User Interface of the XEN system
  - Handles all I/O, driver support, etc. for individual guests
  - If Dom0 is unavailable, all DomU are unavailable

- Different virtualization techniques:
  - *PV*: Paravirtualization, specialized Kernel necessary, very lightweight
  - *HVM*: Hardware Virtual Machine, full virtualization



# Virtual Resources: CPU

- Host CPUs are virtualized by granted slices of CPU-time to guests.
- There is no fixed mapping between virtual CPU of the guest (vCPU) and CPU on the host
  - Work might be schedule on any core of the host
  - Gang Scheduling (all-or-nothing) might apply
- Hyperthreading, multiple Cores per CPU must be taken into account
  - Not doubling resources but might add 30% more performance (remember Conway's Law)
  - Makes work not faster but easier to schedule

- How many vCPUs can be provision?

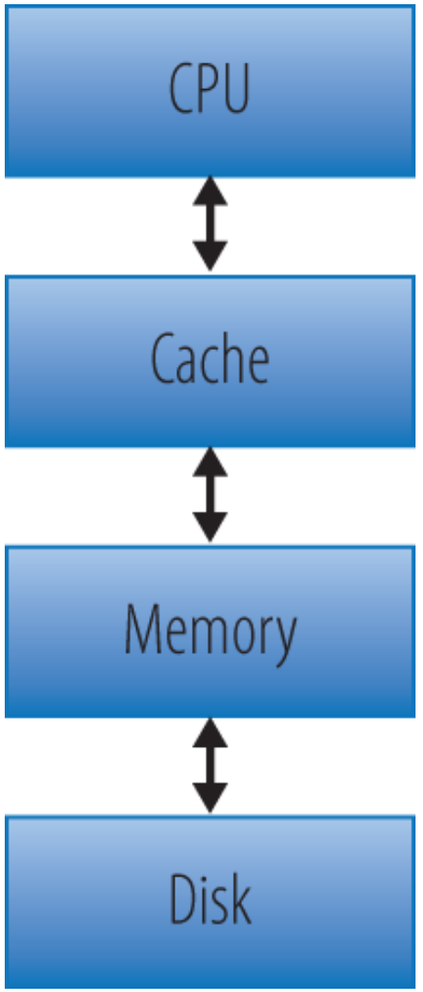
- `(Threads x Cores) x Physical CPU = Number vCPU`
- Ratios vCPU:CPU between 1:1-3:1 (computing intensive) to 6:1 (non computing)

# Virtual Resources: Memory

Recap *sysad*, *bsys*:

- Primary Storage / Secondary Storage
- If too much RAM is consumed, memory is paged to disk
  - Paging can harm the entire host since paging is expensive (swap-partition anyone?)
- If too less RAM is consumed, memory is “blocked” normally by the process

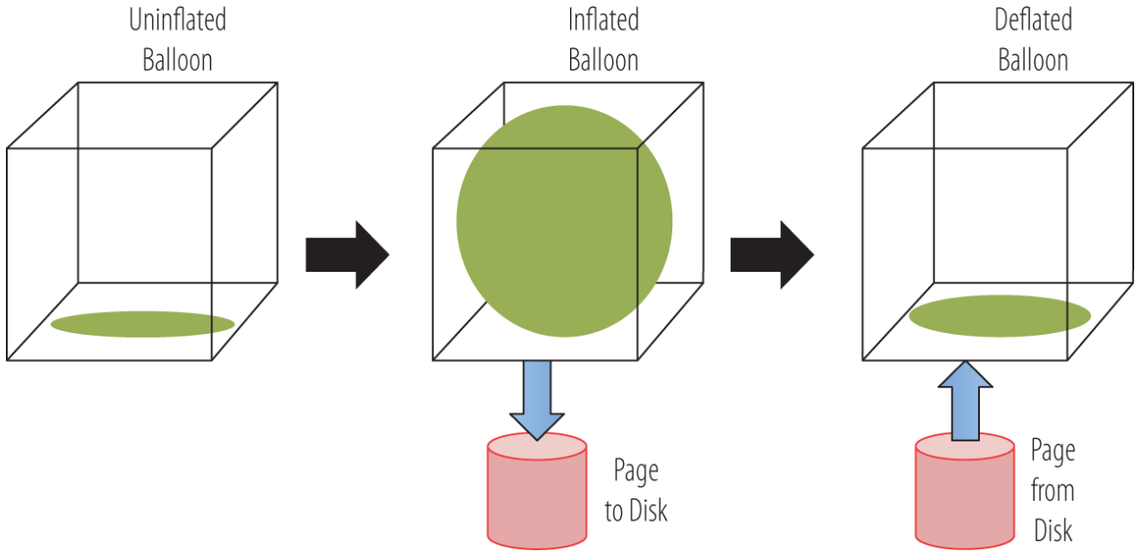
– Things get more complicated as host sees the guest as blackbox.
→ **sophisticated methods** (ballooning, overcommitment, memory sharing) need to be applied to get most out of the physical memory.



# Virtual Memory: Ballooning

Pattern to be applied when there is discussion about memory.

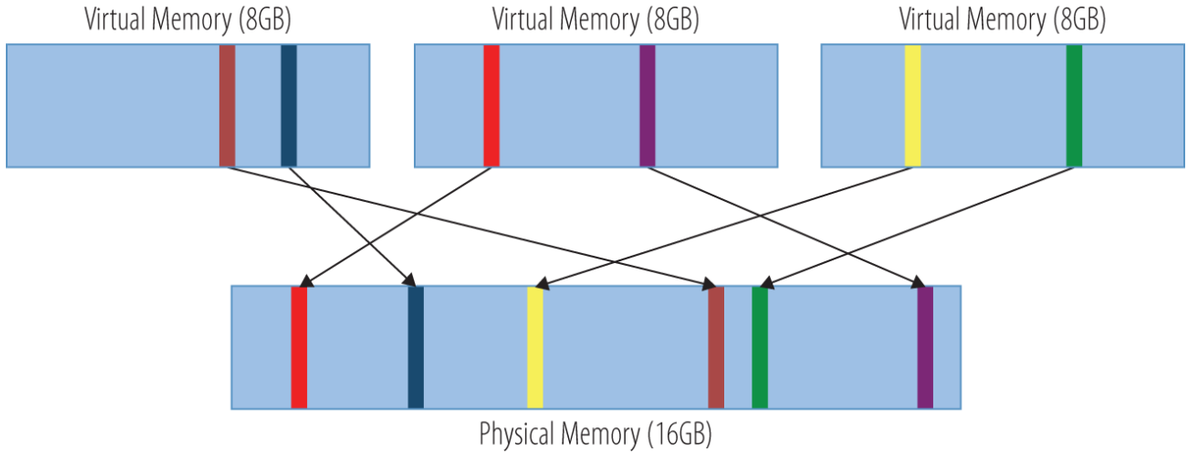
1. Memory is needed → inflating the balloon
2. OS of Host analyses slices of memory to be paged and pages them to disk
3. Memory gets free and can be provided to the demanding guest
4. If memory need decreases → deflate the balloon. The stale slices are loaded from disk again.



# Virtual Memory: Overcommitment

Pattern to map only used / active memory to host memory

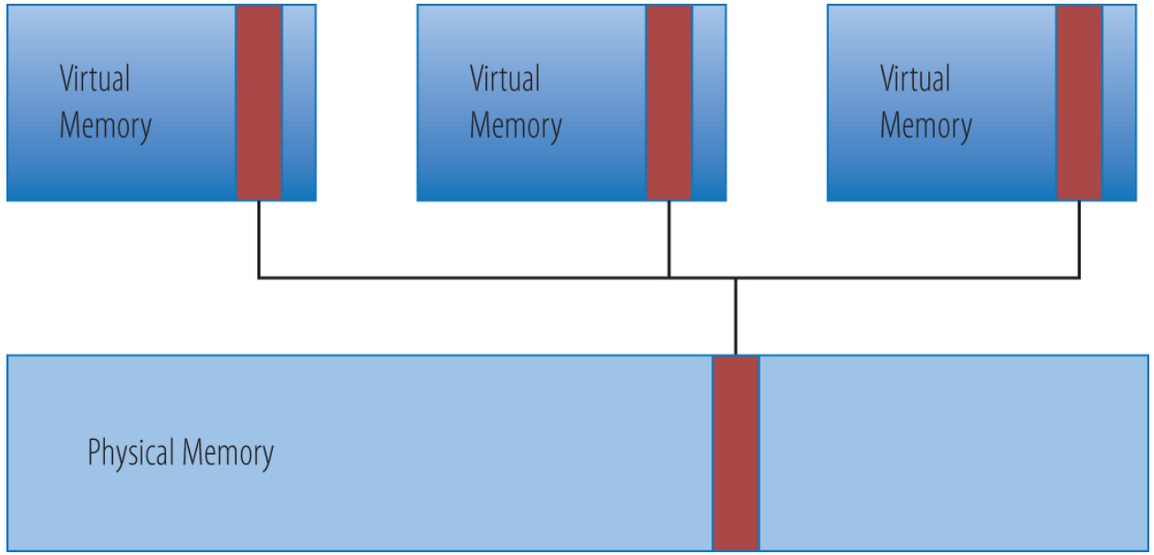
- Analyzing the memory consumption of the guests, the “inactive” parts are not reserved in the host
- Only actively used blocks are mapped to the host
- Memory patterns of guest must be known
- Ratios up to 1.5:1 or 2:1 are possible



# Virtual Memory: Sharing

Often, machines there are many similar guest on the same host

- If the system and the application is the same, the memory consumption might be (partially) the same
- In such a scenario, the memory of multiple guests is mapped once on the slice on the host
- 10%-40% less consumption on practice
- Works best with horizontal scaled infrastructure
- If memory is altered, copy-on-write is applied



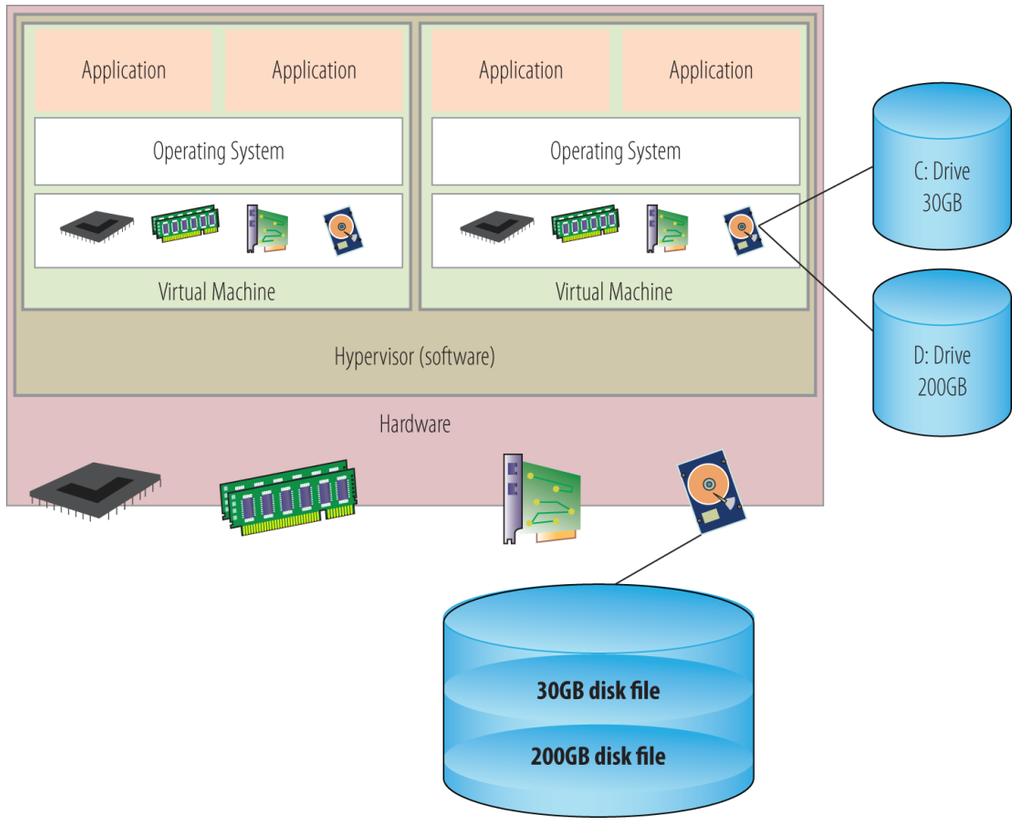
# Virtual Memory: Features used by virtualization technologies

| Techniques | VMware vSphere (vSphere 8.0) | Microsoft Hyper-V (Server 2022 R2) | KVM |
|---|---|---|---|
| Overcommit | Yes | Yes | Yes |
| Ballooning | Yes | Yes | Yes |
| Page sharing | Yes | No | Yes |
| Compression | Yes | No | No |

# Virtual Resources: Storage

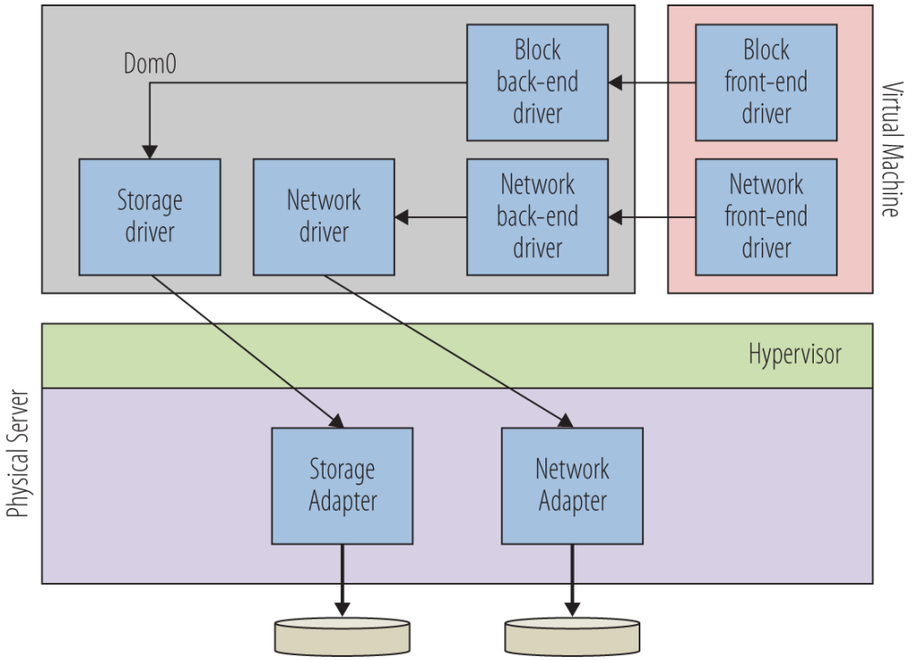
Disks are mapped as **Blockdevices** into the machine:

- Large Disk of any kind is bound to Hypervisor
- In the Machine, one or more disks are associated to a single guest
- Since devices are bound as blockdevices, filesystem is generated by the guest
- the Hypervisor / Host is unware about the filesystem

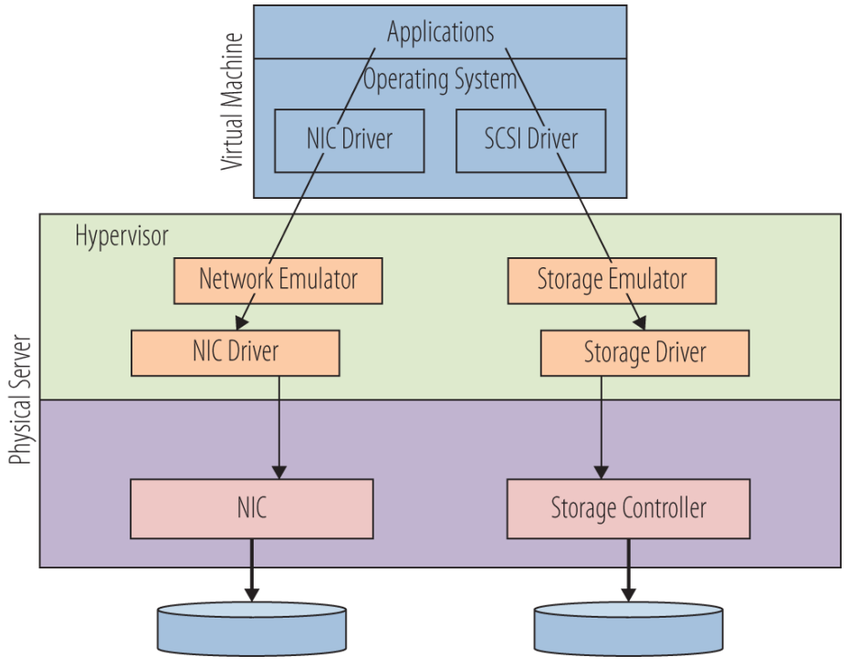


# Virtual Resources: Storage

## Xen



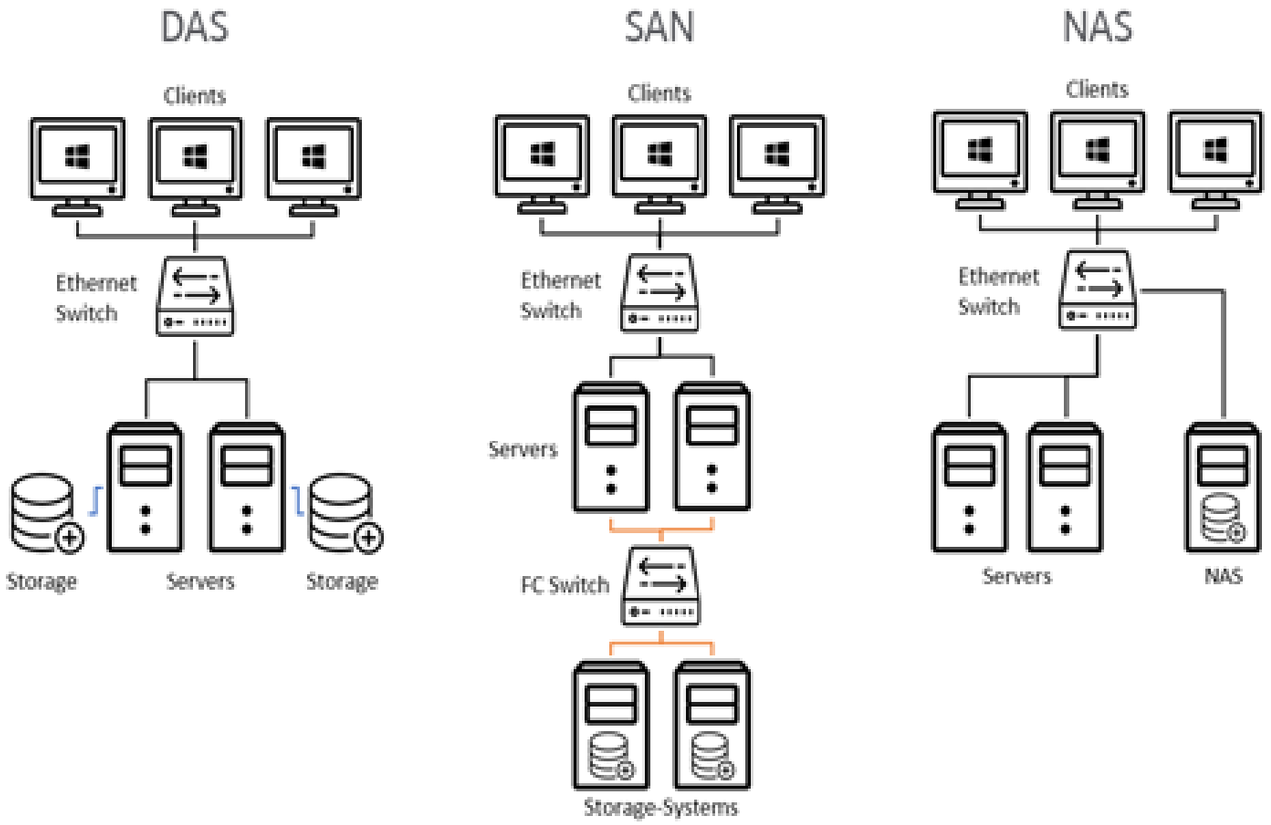

## VMware




**Storage Controller** → Direct attached Storage (DAS)

**NIC** (Network Interface Controller)→ Storage Area Network (SAN) or Networked Attached Storage (NAS)

# Storage Types: DAS, SAN, NAS




**DAS (Direct attached Storage)**

- Disk directly attached to server via SCSI / IDE/ SATA
- Cheap, easy, fast setup
- If server goes down, disk go down

**SAN (Storage Area Network)**

- Disk attached via FibreChannel to one/multiple servers
- Complex, second stack, more expensive
- Failover / Scaling possible

**NAS (Networked Attached Storage)**

- Disk attached directly as device to network
- Easier technical setup, cheaper, redudant
- Blockaccess needs workaround e.g. iSCSI

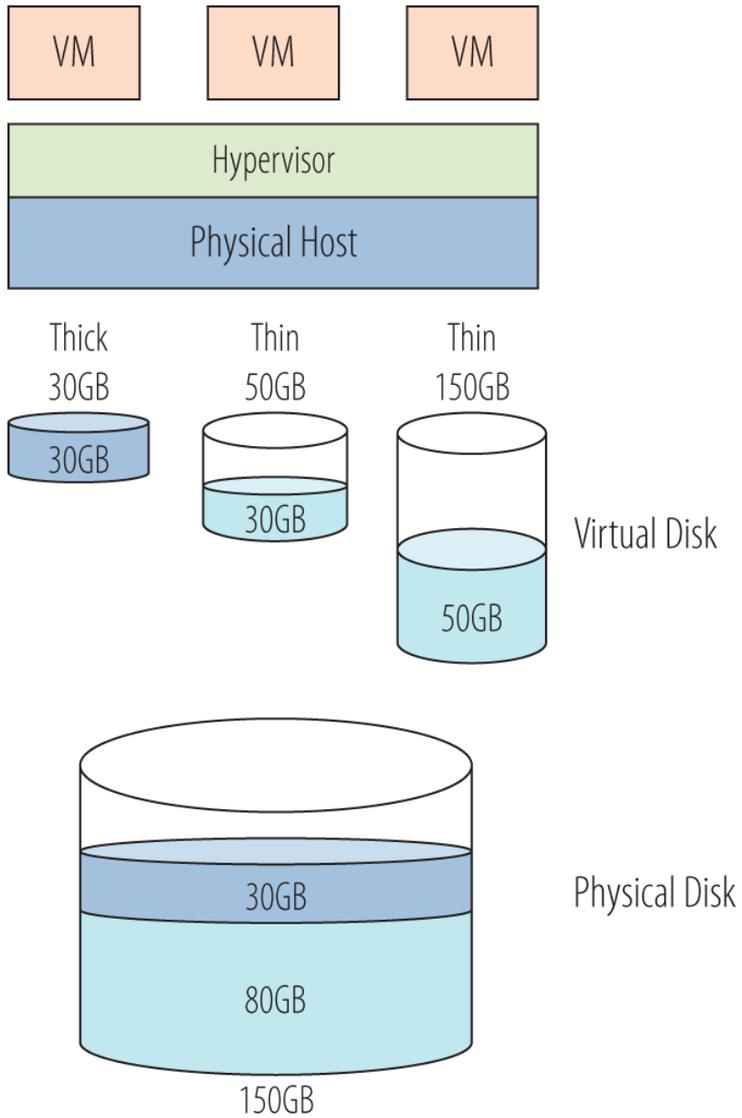
# Thin vs Thick Provisioning

**Thick Provisioning**

- Reserve the space for the guest, the guest allocate with its disk
- If the guest uses less, space is wasted

**Thin Provisioning**

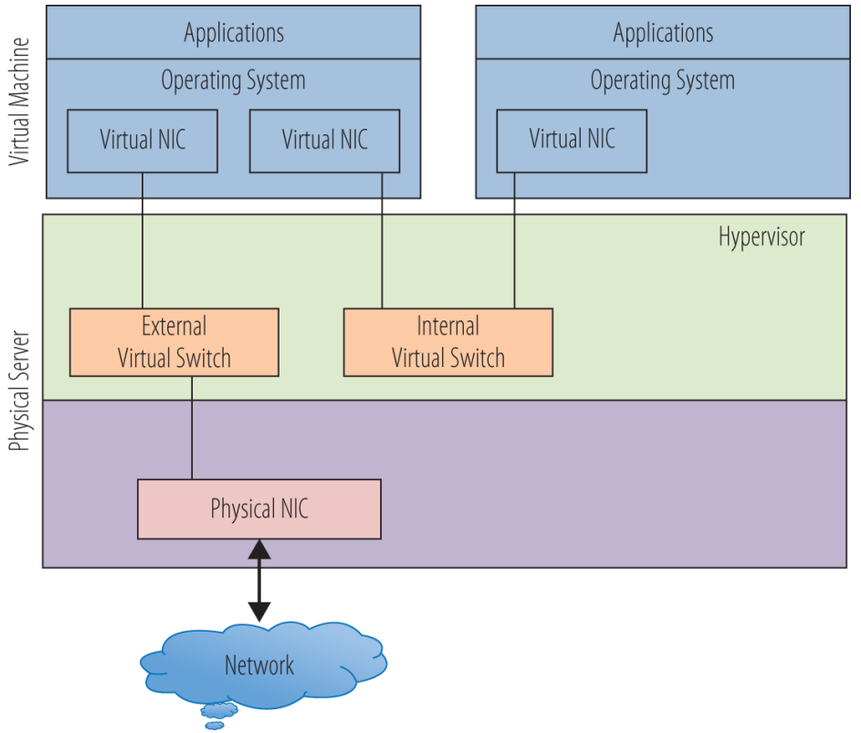
- Reserve only the space, the guest really uses and allocate the size the guest think it has dynamically
- If all guests consume all disks, problems occur.

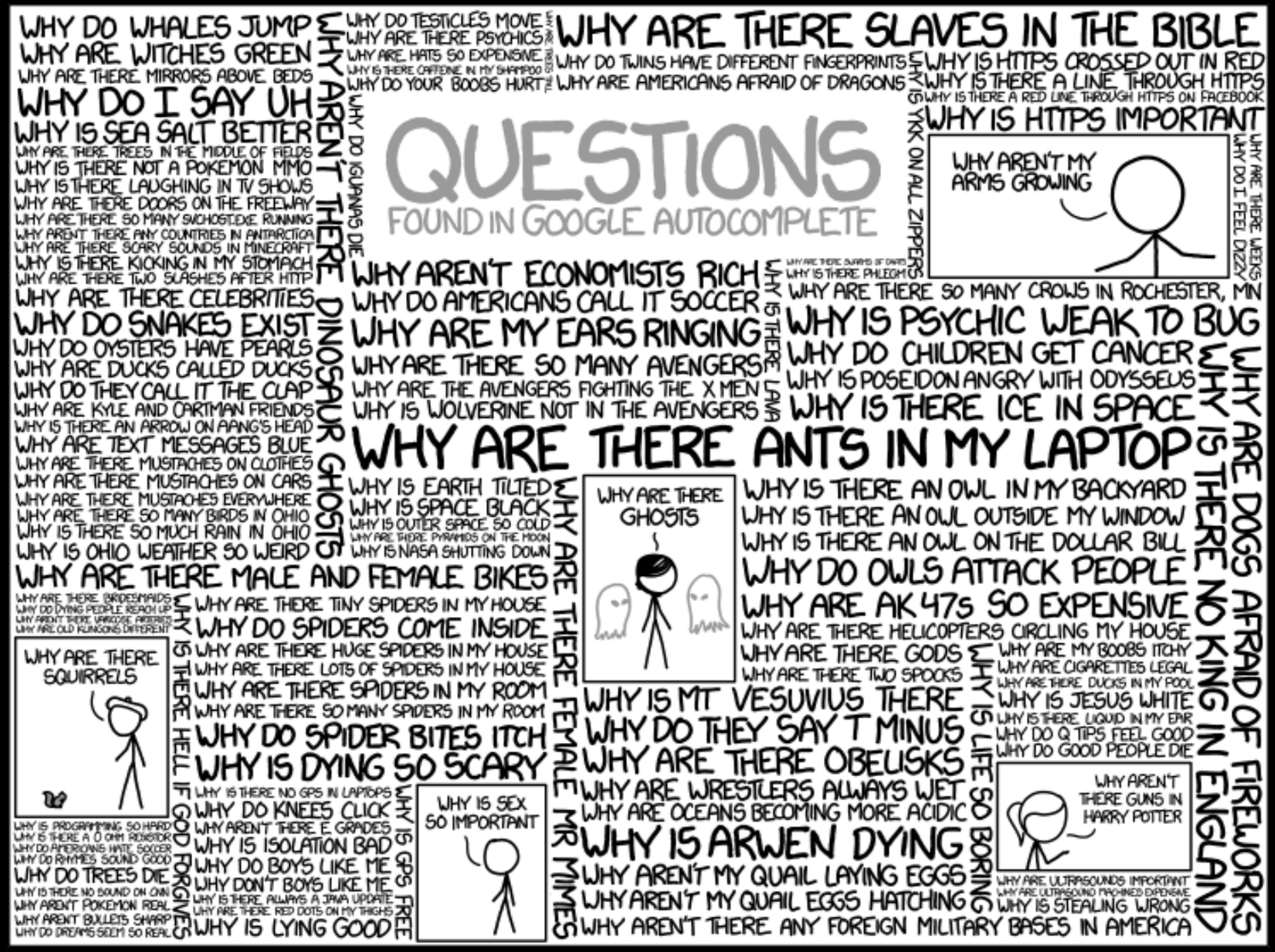


# Virtual Resources: Network

Network connection takes place via virtual network interface card (NIC):

- No direct connect to physical NIC of host (per default)
- Virtual Switches are applied to route traffic from VMs to NIC of host
- VM-to-VM traffic stays on same host




QUESTIONS
FOUND IN GOOGLE AUTOCOMPLETE
WHY DO WHALES JUMP
WHY ARE WITCHES GREEN
WHY ARE THERE MIRRORS ABOVE BEDS
WHY DO I SAY UH
WHY IS SEA SALT BETTER
WHY ARE THERE TREES IN THE MIDDLE OF FIELDS
WHY IS THERE NOT A POKEMON MMO
WHY IS THERE LAUGHING IN TV SHOWS
WHY ARE THERE DOORS ON THE FREEWAY
WHY ARE THERE SO MANY SVCHOSTS.EXE RUNNING
WHY AREN'T THERE ANY COUNTRIES IN ANTARCTICA
WHY ARE THERE SCARY SOUNDS IN MINECRAFT
WHY IS THERE KICKING IN MY STOMACH
WHY ARE THERE TWO SLASHES AFTER HTTP
WHY ARE THERE CELEBRITIES
WHY DO SNAKES EXIST
WHY DO OYSTERS HAVE PEARLS
WHY ARE DUCKS CALLED DUCKS
WHY DO THEY CALL IT THE CLAP
WHY ARE KYLE AND CARTMAN FRIENDS
WHY IS THERE AN ARROW ON AANG'S HEAD
WHY ARE TEXT MESSAGES BLUE
WHY ARE THERE MUSTACHES ON CLOTHES
WHY ARE THERE MUSTACHES ON CARS
WHY ARE THERE MUSTACHES EVERYWHERE
WHY ARE THERE SO MANY BIRDS IN OHIO
WHY IS THERE SO MUCH RAIN IN OHIO
WHY IS OHIO WEATHER SO WEIRD
WHY ARE THERE MALE AND FEMALE BIKES
WHY ARE THERE BRIDESMAIDS
WHY DO DYING PEOPLE REARM UP
WHY AREN'T THERE VARICOSE AIRBIKES
WHY ARE OLD KLINGONS DIFFERENT
WHY ARE THERE SQUIRRELS
WHY IS PROGRAMMING SO HARD
WHY IS THERE A 0 OHM RESISTOR
WHY DO AMERICANS HATE SOCCER
WHY DO RHYMES SOUND GOOD
WHY DO TREES DIE
WHY IS THERE NO SOUND ON CNN
WHY AREN'T POKEMON REAL
WHY AREN'T BULLETS SHARP
WHY DO DREAMS SEEM SO REAL
WHY ARE THERE TINY SPIDERS IN MY HOUSE
WHY DO SPIDERS COME INSIDE
WHY ARE THERE HUGE SPIDERS IN MY HOUSE
WHY ARE THERE LOTS OF SPIDERS IN MY HOUSE
WHY ARE THERE SPIDERS IN MY ROOM
WHY ARE THERE SO MANY SPIDERS IN MY ROOM
WHY DO SPIDER BITES ITCH
WHY IS DYING SO SCARY
WHY IS THERE NO GPS IN LAPTOPS
WHY DO KNEES CLICK
WHY AREN'T THERE E GRADES
WHY IS ISOLATION BAD
WHY DO BOYS LIKE ME
WHY DON'T BOYS LIKE ME
WHY IS THERE ALWAYS A JAVA UPDATE
WHY ARE THERE RED DOTS ON MY THIGHS
WHY IS LYING GOOD
WHY DO TESTICLES MOVE
WHY ARE THERE PSYCHICS
WHY ARE HATS SO EXPENSIVE
WHY IS THERE CAFFEINE IN MY SHAMPOO
WHY DO YOUR BOOBS HURT
WHY ARE THERE SLAVES IN THE BIBLE
WHY DO TWINS HAVE DIFFERENT FINGERPRINTS
WHY ARE AMERICANS AFRAID OF DRAGONS
WHY IS HTTPS CROSSED OUT IN RED
WHY IS THERE A LINE THROUGH HTTPS
WHY IS THERE A RED LINE THROUGH HTTPS ON FACEBOOK
WHY IS HTTPS IMPORTANT
WHY AREN'T MY ARMS GROWING
WHY ARE THERE SWARMS OF GNATS
WHY IS THERE PHLEGM
WHY ARE THERE SO MANY CROWS IN ROCHESTER, MN
WHY AREN'T ECONOMISTS RICH
WHY DO AMERICANS CALL IT SOCCER
WHY ARE MY EARS RINGING
WHY ARE THERE SO MANY AVENGERS
WHY ARE THE AVENGERS FIGHTING THE X MEN
WHY IS WOLVERINE NOT IN THE AVENGERS
WHY IS PSYCHIC WEAK TO BUG
WHY DO CHILDREN GET CANCER
WHY IS POSEIDON ANGRY WITH ODYSSEUS
WHY IS THERE ICE IN SPACE
WHY ARE THERE ANTS IN MY LAPTOP
WHY IS EARTH TILTED
WHY IS SPACE BLACK
WHY IS OUTER SPACE SO COLD
WHY ARE THERE PYRAMIDS ON THE MOON
WHY IS NASA SHUTTING DOWN
WHY ARE THERE GHOSTS
WHY IS THERE AN OWL IN MY BACKYARD
WHY IS THERE AN OWL OUTSIDE MY WINDOW
WHY IS THERE AN OWL ON THE DOLLAR BILL
WHY DO OWLS ATTACK PEOPLE
WHY ARE AK 47S SO EXPENSIVE
WHY ARE THERE HELICOPTERS CIRCLING MY HOUSE
WHY ARE THERE GODS
WHY ARE THERE TWO SPOOKS
WHY IS MT VESUVIUS THERE
WHY DO THEY SAY T MINUS
WHY ARE THERE OBELISKS
WHY ARE WRESTLERS ALWAYS WET
WHY ARE OCEANS BECOMING MORE ACIDIC
WHY IS ARWEN DYING
WHY AREN'T MY QUAIL LAYING EGGS
WHY AREN'T MY QUAIL EGGS HATCHING
WHY AREN'T THERE ANY FOREIGN MILITARY BASES IN AMERICA
WHY ARE MY BOOBS ITCHY
WHY ARE CIGARETTES LEGAL
WHY ARE THERE DUCKS IN MY POOL
WHY IS JESUS WHITE
WHY IS THERE LIQUID IN MY EAR
WHY DO Q TIPS FEEL GOOD
WHY DO GOOD PEOPLE DIE
WHY AREN'T THERE GUNS IN HARRY POTTER
WHY ARE ULTRASOUNDS IMPORTANT
WHY ARE ULTRASOUND MACHINES EXPENSIVE
WHY IS STEALING WRONG
WHY IS SEX SO IMPORTANT
WHY IS GPS FREE
WHY IS LIFE SO BORING
WHY ARE THERE FEMALE MR MIMES
WHY IS THERE NO KING IN ENGLAND
WHY ARE DOGS AFRAID OF FIREWORKS
WHY IS THERE HELL IF GOD FORGIVES
WHY DINOSAUR GHOSTS
WHY AREN'T THERE
WHY DO IGUANAS DIE
WHY IS YKK ON ALL ZIPPERS
WHY ARE THERE WEEKS
WHY DO I FEEL DIZZY
WHY IS THERE LAVA

# Summarizing Questions

- Explain why Moore's Law is important regarding virtualization
- Describe the difference between
  - Type1 and Type2 virtualization
  - Hardware-assisted, Full, and Paravirtualization
- Why are the properties and requirments for virtualization also important for cloud computing?
- Which virtualization approach provides the best performance?
- Compute the number of vCPUs for a machine with 4 Quadcores and hyperthreading enabled
- What can be a problem with Balooning?
- What is thick provision? When should you use it?
- Why is generating backups with Snapshots only not a good idea?
- What is the primary advantage of using Open vSwitch (OVS) with VxLAN in a cloud environment?