Virtualization Technologies

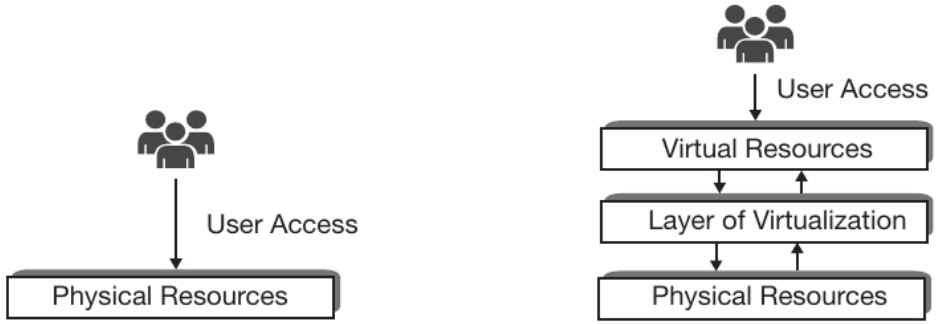
Introduction to virtualization and basic concepts

Reference:

- [cam-san] Chapter 7

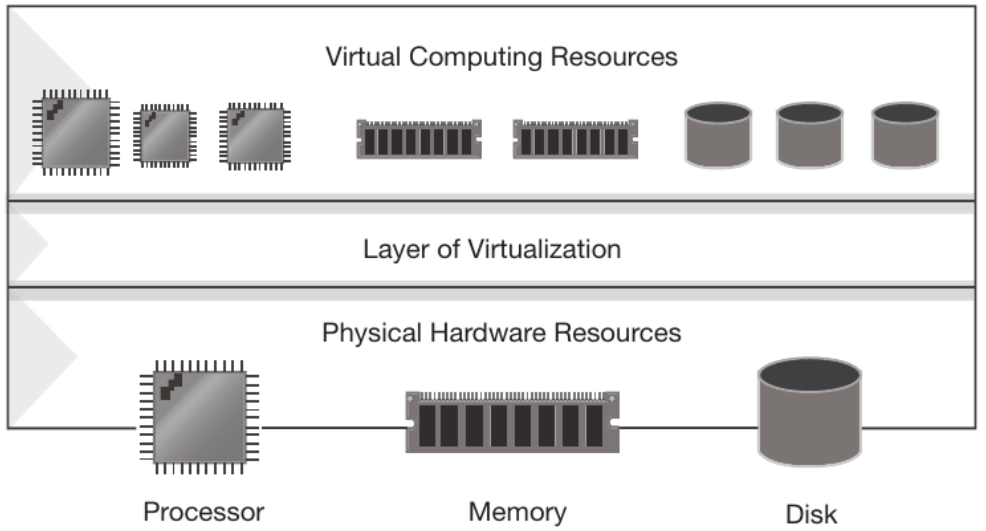
Virtualization

- Virtualization refers to the representation of physical computing resources in simulated form through an additional software layer
- This software, referred as virtualization layer, is installed over the physical machine to provide a virtual form of the hardware
- This virtual resources are used to satisfy users' computing needs
- Virtualization decouples physical computing resources from direct access of users



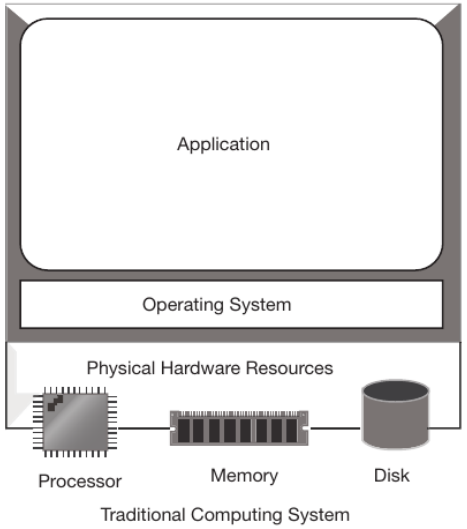
Resource Virtualization

- Any kind of IT resources can be virtualized: from the basic computing devices (e.g. processor, RAM, etc) to other resources like storage, network devices or peripheral (keyboard, mouse, printer, etc.)
- In case of basic computing devices, a virtualized representation can function only when a physical resource empowers the virtual representation (a virtual processor cannot be created without a physical one)
- The virtualization layer transforms the physical resources: the simulated devices produced through virtualization may or may not be equal to the actual physical component, in architecture, quantity or quality, for example:
  - Three virtual processor might be produced by using one physical processor
  - One 32-bit virtual processor can be produced from a 64-bit physical processor

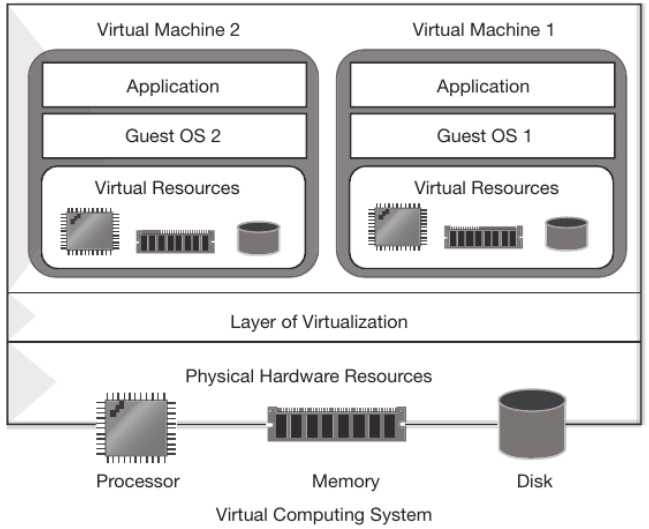


Machine/Server Virtualization

- Machine/server virtualization is the concept of creating a virtual machine (or virtual server) on actual physical machine.
- The physical machine is called *host system*, the virtual machine is the *guest system*
- In conventional computing system, there has always been a one-to-one relationship between physical computer and operating system, through virtualization we can have multiple systems running over a single host
- Each guest systems remains independent of the others and (indirectly) accesses the hardware of the host system
- It runs applications within its own operating environment



Traditional Computing System



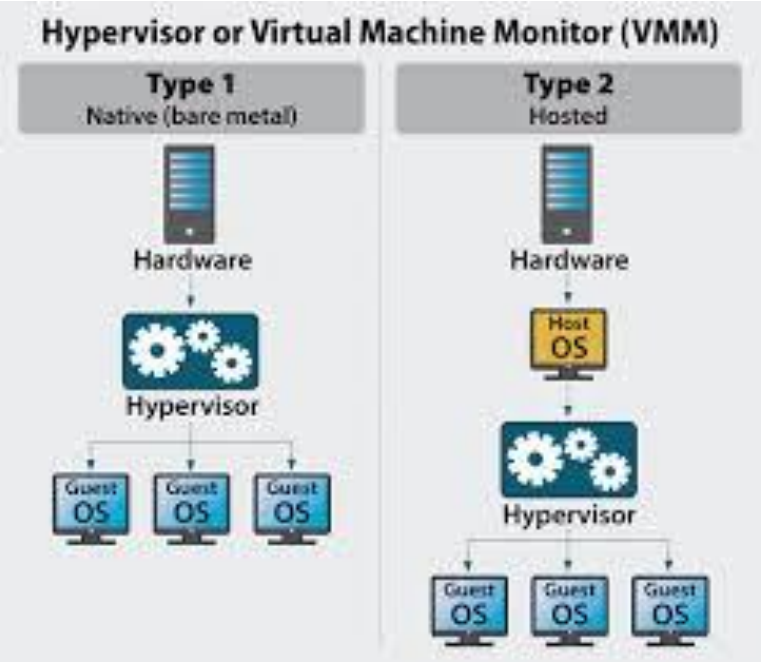
Virtual Computing System

Hypervisor

- The virtualization layer is a set of control programs that creates the environment for the virtual machines to run on
- This layer provides the access to system resources (CPU, RAM, Disk, etc) to the virtual machines and controls and monitors the execution of the virtual machines over it
- This software layer is referred as the *Hypervisor* or *Virtual Machine Monitor* (VMM)
- The hypervisor abstracts the underlying software and/or hardware environment and offers virtual resources to the guest systems
- *Hypervisor-based virtualization techniques can be divided into different categories as full virtualization, para-virtualization, hardware-assisted virtualization.*
- *In addition, other lightweight virtualization techniques are available, e.g. operating system virtualization, etc.*

Hypervisor types

- There are two different techniques of server virtualization: hosted approach and bare metal approach
- For each technique we have a different type of hypervisor, type 1 and type 2
- **Hosted approach**: an operating system (host OS) is first installed on the physical machine. Over the OS a hypervisor (a software program) is installed to run VMs. Software like VMWare or Microsoft Virtual PC are example of type 2 hypervisor software
- **Bare Metal Approach**: the hypervisor is directly installed over the physical machine. This type 1 hypervisor accesses directly the hardware of the host. VMWare ESX or Microsoft Hyper-V are example of bare-metal hypervisors



Pros/Cons – Hypervisor types

- **Hosted approach**
  - Benefits: The host OS supplies the hardware drivers for the underlying physical resources. This eases the installation and configuration of the hypervisor. It makes the type-2 hypervisors compatible for a wide variety of hardware platform.
  - Drawbacks: A hosted hypervisor does not have direct access to the hardware resources and hence, all the requests from virtual machines must go through the host OS. This may degrade the performance of the virtual machines.
- **Bare Metal Approach**
  - Benefits: Since the bare metal hypervisor can directly access the hardware resources in most of the cases it provides better performance in comparison to the hosted hypervisor. For bigger application like enterprise data centers, bare-metal virtualization is more suitable because usually it provides advanced features for resource and security management and administrators get more control over the host environment.
  - Drawbacks: As any hypervisor usually have limited set of device drivers built into it, so the bare metal hypervisors have limited hardware support and cannot run on a wide variety of hardware platform.

Full Virtualization

- In full virtualization (also called as *native virtualization*), the hypervisor fully simulates or emulates the underlying hardware
- The guest operating system assumes that they are running on actual physical resources and thus remain unaware that they have been virtualized
- This enables the **unmodified** versions of available operating systems (like Windows, Linux and else) to run as guest OS over the hypervisor
- It is the responsibility of the hypervisor to handle all OS-to-hardware (i.e. guest OS to physical hardware) requests during the execution of guest machines
- The guest OS remains completely isolated from physical resources by the hypervisor

Para-Virtualization

- In full virtualization the guest OS running in the VM does not have any knowledge that it runs over a virtualized platform, and it does not require any special modification
- All the virtualization activities are managed by the hypervisor, e.g. instruction emulation, establishing communication with the underlying hardware, etc.
- In para-virtualization, a portion of the virtualization management task is transferred (from the hypervisor) to the guest operating systems
- Normal versions of available operating systems are not capable of doing this, so they cannot be used in para-virtualization
- Each guest OS requires special modifications (porting), and it is explicitly ported for the para-application program interface (API) of the platform
- Thus, in para-virtualization, each guest OS needs to have prior knowledge that it runs over the virtualized platform. Moreover, it also has to know on which specific hypervisor they will run. Depending on the hypervisor, the guest OS is modified as required to participate in the virtualization management task.

Pros/Cons – Para-Virtualization

- Advantages:
  - Para-virtualization allows calls from guest OS to directly communicate with hypervisor (without any emulation of instructions). The use of modified OS reduces the virtualization overhead of the hypervisor as compared to the full virtualization
  - In para-virtualization, the system is not restricted by the device drivers provided by the virtualization software layer. The virtualization layer (hypervisor) can exploit the guest operating systems device drivers
- Limitations:
  - Unmodified versions of available operating systems (like Windows or Linux) are not compatible with para-virtualization hypervisors. Modifications are possible in Open-source operating systems (like Linux). But for proprietary operating systems (like Windows), it depends upon the owner. If the owner agrees to supply the required modified version of the OS for a hypervisor, then only that OS becomes compliant with a specific para-virtualization system.
  - Security is compromised in this approach as the guest OS has a comparatively more control of the underlying hardware

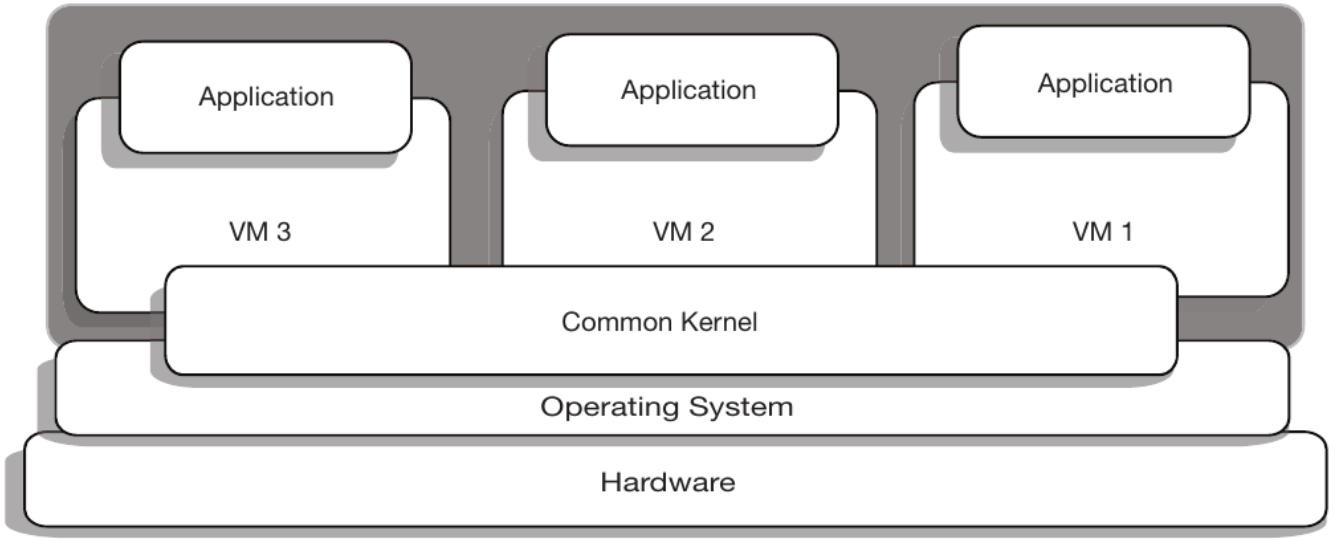
Hardware assisted virtualization

- Since 2006, hardware vendors started manufacturing devices tailored to support virtualization, natively on hardware
- Intel and AMD started this by including new virtualization features in their processors
- AMD-Virtualization (AMD-V) and Intel Virtualization Technology (Intel-VT) allow some privileged CPU calls from the guest OS to be directly handled by the CPU
- These calls do not require to be translated/emulated by the hypervisor; this eliminates the need for para-virtualization
- This kind of virtualization is only possible when specific combinations of hardware components are used

We will see that extensively, don't worry!

Operating system level virtualization

- Operating system level virtualization (also called as system level virtualization) works in totally different way than the virtualization techniques as discussed
- In this kind of virtualization technique, no hypervisor is used, and the virtual servers are enabled by the kernel of the operating system of physical machine
- *The kernel of the operating system installed over physical system is shared among all the virtual servers running over it*



Operating system level virtualization

- The kernel and the host OS takes care of creating multiple logically-distinct user-space instances (virtual servers) over a single instance of an OS kernel
- **Advantages**: The advantages of OS level virtualization is that it is more lightweight since all the virtual servers share a single instance of an OS kernel. This enables a single physical system to support many others virtual servers than the number of complete virtual machines it could support
- **Limitations**: All virtual machines have to use the same operating system (due to sharing of OS kernel). Although different distributions (like Linux distribution) of the same system kernel are allowed. The virtualized environment is not a complete system, some limitations are introduced.

Other types of Virtualization

- Virtualization of computing infrastructure is not only about machine or server virtualization
- In order to create a complete infrastructure, we also need ***network*** and ***storage*** virtualization
  - *Network Virtualization*: network virtualization is the process of combining network resources and network functionalities into a single, (usually software-based) administrative entity called as a virtual network. It allows to create a virtual network for VMs to communicate. The virtual network usually exploits the real network infrastructure for data transmission
  - *Storage Virtualization*: In traditional computing system, the storages have always been directly linked with the physical servers. In a virtualized system storage must be virtualized. Like other computing resources, virtualization of storage also happens through layer of software which creates logical abstraction of the pooling of physical storage devices having linked together by network. Data stored in logical (virtualized) devices ultimately get stored in some physical storage disks.

Emulation

- Emulation in computing is *the act of making a system imitating another one*
- This means a system that has an architecture is enabled through emulation to support the instruction set of some other machine architecture
- Emulation software converts binary data written for execution on one machine to an equivalent binary form suitable to execute on another machine
- There are two ways for implementation of emulations:
  - In **binary translation** (also known as recompilation), a total conversion of the binary data (made for the emulated platform) is done. The conversion recompiles the whole instruction into another binary form suitable to run on the actual or targeted platform. There are two types of binary translation like static recompilation and dynamic recompilation
  - In **interpretation**, each instruction is interpreted by the emulator every time it is being encountered. This method is easier to implement but slower than binary translation process.

Emulation vs Virtualization

- Emulation and Virtualization are different
- Emulation can allow the simulation of the complete hardware in software, thus creating an environment for the execution of an operating system
- Emulation creates an environment to support the execution of a guest system on top of a host that has a different system architecture, thus it allows an operating system made for one computer architecture to run on the architecture supported by the emulator
- The latter is called *emulation-based virtualization* that is different from regular emulation
- *In regular virtualization, the instruction set used by the virtual system and the actual hardware system is same. Hence, the virtual machine code (OS and programs) can be executed directly by actual physical system*
- The translation of the instruction sets is not required (except for some exceptions). Without the translation layer, the performance of a virtual machine is much faster and nearly approaches the native speed
- **Virtualization therefore is significantly faster than emulation**
- It is important to highlight that Virtualization might still require emulation in some cases (emulate some hardware, some functions or some instructions)

Virtualization Advantages

- This process of running multiple VMs on the same hardware is called **server consolidation** and allows to increase hardware utilization. Virtualization helps to achieve a better utilization of existing resources.
- As a better usage of resources can be achieved, virtualization helps in ***reducing hardware costs*** and computing infrastructure costs
- Detaching hardware and software through a virtualization layer helps in ***simplifying the system administration*** by setting two clear realms, one the guest OS realm and one the host system realm
- Virtualization ***simplifies system installation***, as a new system can be created by cloning another VM, without requiring the installation and configuration of a full system
- It improves ***fault tolerance and*** enables ***zero downtime maintenance*** by allowing to migrate VMs among different hardware or to backup and recreate VMs
- It ***improves security*** as each VM is isolated from the others thanks to the additional isolation provided by the virtualization layer

Virtualization Downsides

- Each physical machine is a ***Single Point of Failure***. The major benefit of virtualization is resource sharing. Multiple virtual machines can run over one physical machine. But this has a downside. It increases the probability of failure for several virtual servers in cases of failure of single physical machine
- Virtualization results in ***Lower Performance***. There is a concern whether virtual environments have the capacity to accomplish the full performance of the actual physical system. It has been measured that virtual servers can achieve up to 85 percent to 90 percent of the performance of the actual physical server as VMs cannot get direct access to the hardware.