Virtualization

Virtualization (Definition)

Virtual (not actual / not physical) version of

- Hardware ,
- Operating system ,
- Storage device,
- Network resources
- etc

Types

- Hardware virtualization (platform*)*
  - Virtual machine that acts like a real computer with an operating system
- OS-level virtualization (Software)
  - Multiple virtualized environments within a single OS instance
- Memory virtualization
  - Aggregating RAM resources from networked systems into a single memory pool
- Storage virtualization
  - Abstracting logical storage from physical storage

A **virtual machine** (**VM**) is an emulation of a particular computer system

VM is a software implementation of a machine that executes programs like a physical machine.

System VM

- Platform which supports the execution of a complete OS.
- Usually emulate an existing architecture
- Built with the purpose of
  - Providing a platform to run programs where the real hardware is not available for use
  - Having multiple instances of virtual machines leading to more efficient use of computing resources, both in terms of energy consumption and cost effectiveness
    - hardware virtualization, the key to a cloud computing environment)

Process VM (language VM)

- Designed to run a single program
- Supports a single process
- Closely suited to one or more programming languages
- Provides program portability and flexibility
- Software running inside is limited to the resources and abstractions provided by the virtual machine

Hypervisor or **VM Monitor** (**VMM**)

A piece of computer software, firmware or hardware that creates, runs & manages VM.

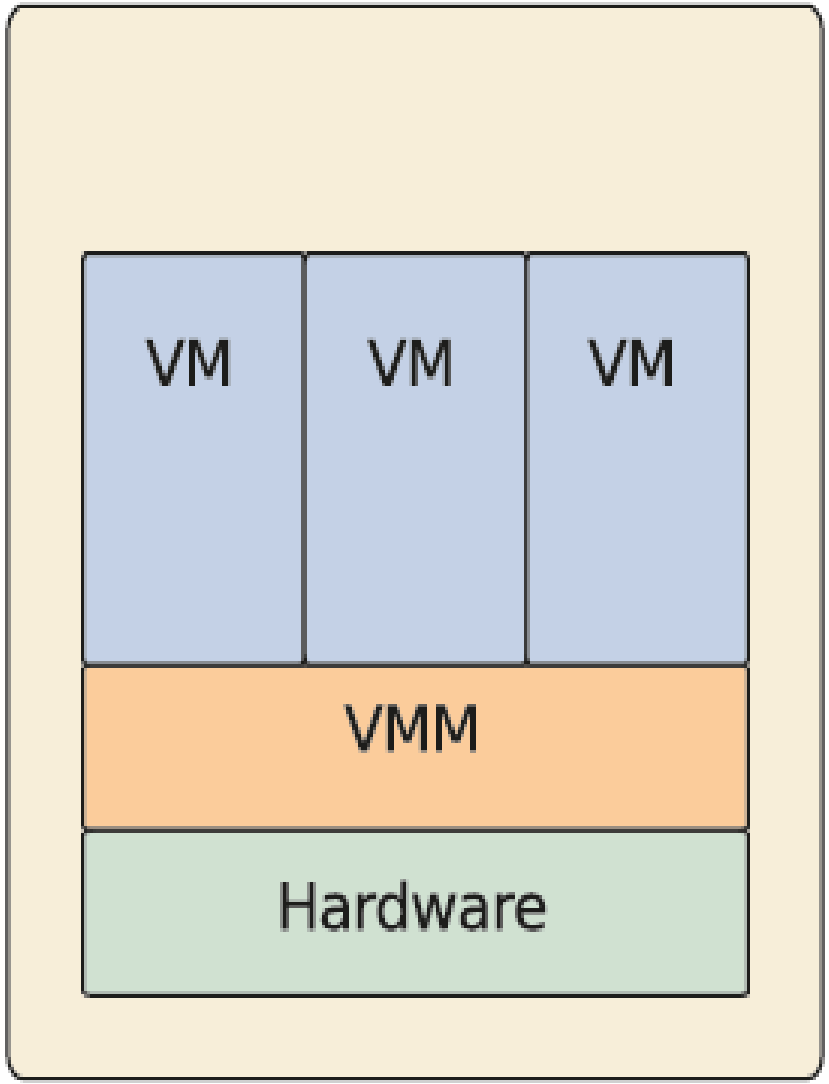
Software layer that virtualizes all of the resources of a physical machine.

Type I & Type II Hypervisor

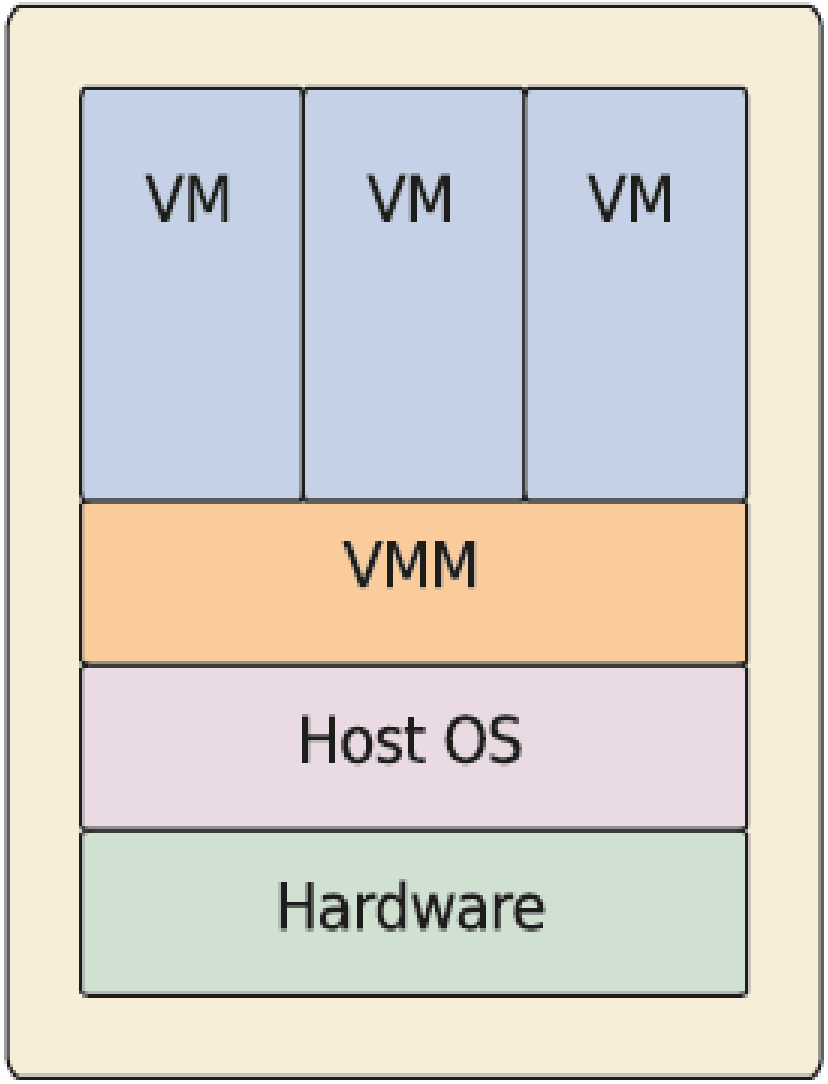
- Type-I:
  - Native or bare-metal hypervisors
  - Run directly on the host's hardware to control the hardware and to manage guest operating systems.
  - A guest operating system runs as a process on the host
  - Have resource management components such as memory, CPU, and I/O. For instance, there is a VM scheduler which is used for CPU resource management
  - Oracle VM Server for SPARC, Oracle VM Server for x86, the Citrix XenServer, VMware ESX/ESXi and Microsoft Hyper-V 2008/2012.

- Type-II:
  - Hosted hypervisors
  - Run on a conventional operating system
  - Abstract guest operating systems from the host operating system.
  - Rely on the OS process scheduler, i.e. each running VM is another OS process
  - VMware Workstation and VirtualBox are examples

- KVM (Kernel-based VM)
  - A type II hypervisor that converts host OS to Type I hypervisor.



Hypervisor Type I



Hypervisor Type II

Virtualization Techniques Classification from the OS view

- Execution of modified guest Oss
  - Operating System-level virtualization..
  - Para-virtualization
- Execution of unmodified guest Oss
  - Binary translation
  - Hardware assisted

Operating System-level virtualization

- Virtualizes the physical server at the operating system Level
- Host OS is a modified kernel that allows the execution of multiple isolated Containers (Virtual Private Server (VPS), jail, or virtualized server)
- Each Container is an instance that shares the same kernel of the host OS.
- Examples: Linux-VServer, Solaris Zones, and OpenVZ
- Adv:
  - Low overhead and their implementations are widely used.

- Drawback:
  - Does not support multiple Kernels

Para-Virtualization

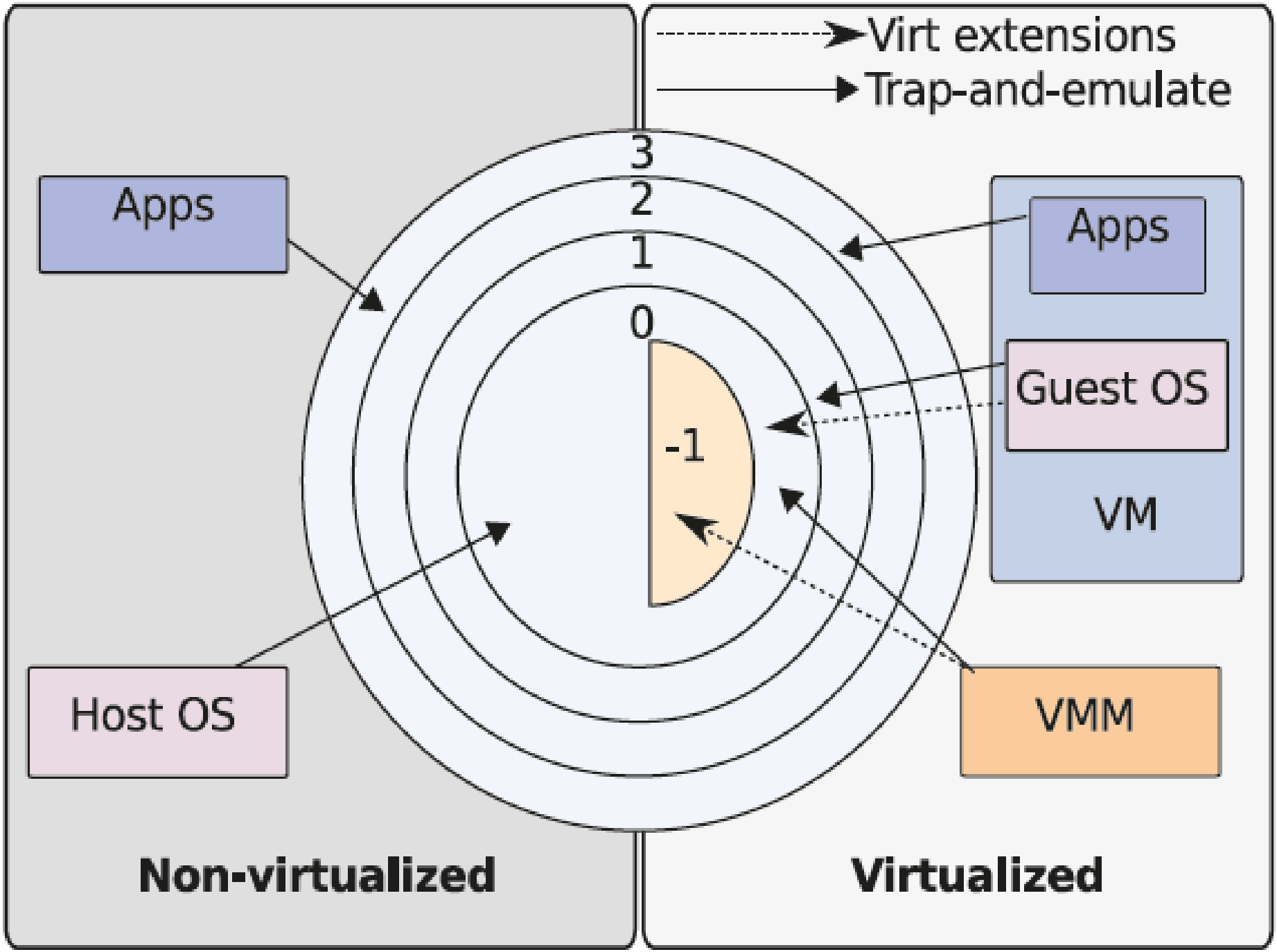
- Adds a special set of instructions (named Hypercalls)
- Replaces instructions of the real machine's instruction set architecture
- Adv:
  - Low virtualization overhead
- Examples :Denali , Xen , and Hyper-V
- In the x86 architecture
  - VMM(or Hypervisor) runs just above the physical hardware (Ring 0)
  - Guest OSs run in higher levels
- Adv:
  - Supports multiple Kernels
- Drawback:
  - To take full advantage of this technique the kernel of the guest OSs needs to be modified in order to make use of the Hypercalls.

Binary Translation (BT)

- *Emulation of a processor architecture over another processor* architecture.
- Allows executing unmodified guest OSs by emulating one instruction set by another through translation of code
- Example: QEMU which is a processor emulator developed by Fabrice Bellard. Currently, QEMU supports full CPU emulation of x86, x86-64, ARM
- BT is also used for virtualization
  - Using BT in conjunction with direct execution the purpose of virtualization of having copies (VMs) of the physical machine is achieved

Hardware Assisted

- Processor vendors such as AMD and Intel introduced virtualization extensions to their line of products.
- Implements a Ring with a higher privileged mode in the processor architecture
- The CPU extensions for virtualization support allows executing unmodified guest OSs in Ring 0 (non-root mode) and the VMM or Hypervisor in Ring -1 (root mode).
- Hardware assisted virtualization enhances CPUs to support virtualization without the need of binary translation or Para-virtualization.
- The
- Examples :Kernel-based Virtual Machine (KVM), VirtualBox, Xen, Hyper-V, and VMware products.


Virt extensions
Trap-and-emulate
3
2
1
0
-1
Apps
Host OS
Apps
Guest OS
VM
VMM
Non-virtualized
Virtualized