Cloud computing, virtualization and containerization

Massimo Mecella

(based on Italian slides by Luca Cabibbo)

Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti

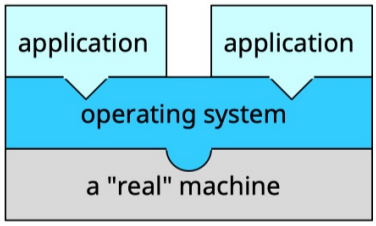
- Goals and topics

- Goals
  - introduce system virtualization and virtual machines
  - describe some virtualization techniques and options
  - introduce some virtualization systems
  - discuss virtual machines as an option for software release

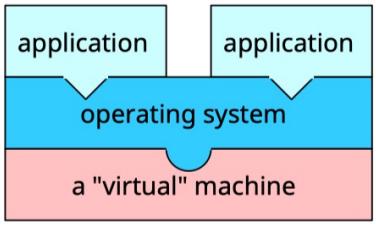
- Subjects
  - system virtualization and virtual machines
  - techniques for system virtualization
  - virtualization systems
  - applications and benefits of system virtualization
  - virtual machines and software release
  - discussion

* System and virtual machine virtualization

- There *system virtualization* allows a "real" computer to host one or more "virtual" computers - called *virtual machines*



a "real" computer - with its OS and applications



a "virtual" computer - with its OS and applications

Virtualization (in general)

- In general, the *virtualization* it is intended to provide access to a set of computational resources *virtual* starting from a set of computational resources *real*
  - think, for example, of a file system or a Virtual Private Network

how
  - resource virtualization takes place on the basis of a layer *virtualization software* - between real resources and consumers of virtual resources

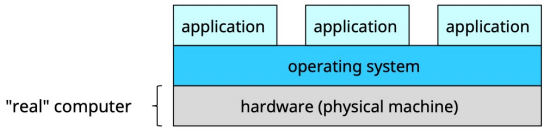
System virtualization

- There **system virtualization** (*system virtualization* or *hardware virtualization*) virtualizes the hardware of an entire real physical computer ("system") to provide one or more *virtual machines*
  - virtualization software is called **hypervisor** or **virtual machine monitor** (**VMM**)
  - it is often called simply "virtualization"
- System virtualization is an important technology especially in distributed systems and in the cloud
  - is an enabling technology for flexible management of (virtual) execution environments - to flexibly run a set of applications and services
  - is a key enabling technology of cloud computing
  - it can sustain qualities such as availability and scalability

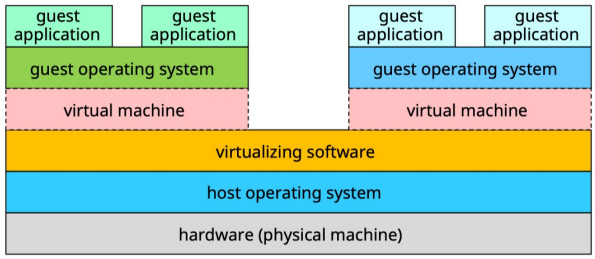
Virtual machines

- A **virtual machine** (**virtual machine** or **VM**) is the emulation of a real machine
  - "machine" means the hardware of a computer
  - a virtual machine exposes the same interface as a real computer - that is, a set of (virtual) hardware resources, such as one or more processors, memory, storage and network devices, ...
  - a virtual machine therefore provides thevirtual hardware of a complete computer - where you can install and run an OS and a set of services and applications

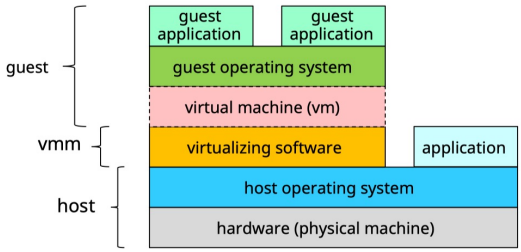
- An example of a non-virtualized computer



- Another example, where the host hosts multiple VMs

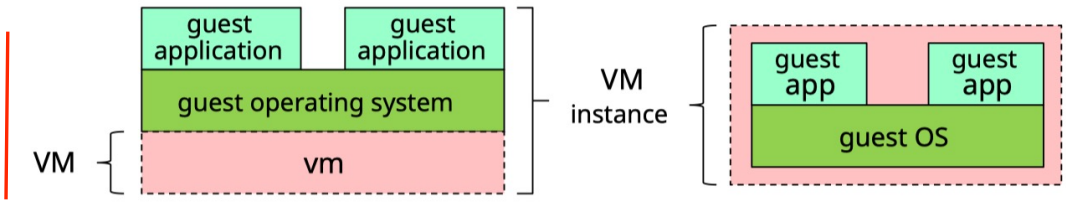


- An example of a virtualized computer
  - the physical computer is the **host**
  - virtualization software is a *hypervisor* or *virtual machine monitor* (*VMM*)
  - the virtual machine is the **guest**



VMs and VM instances

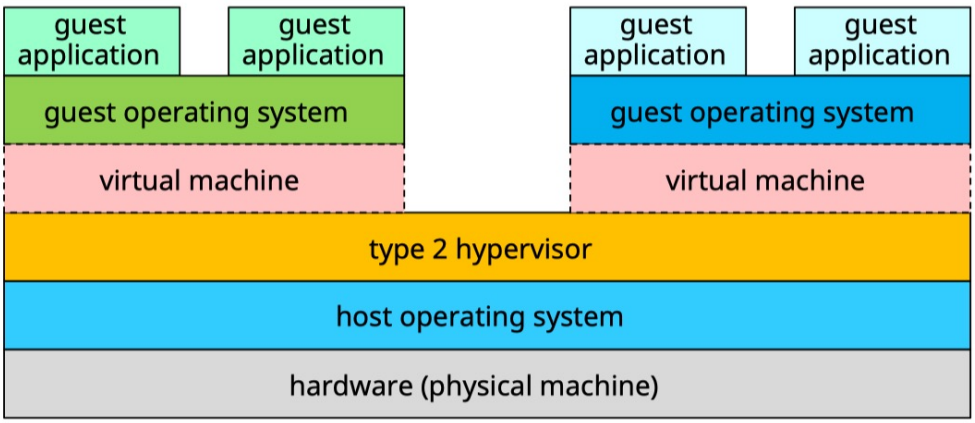
- A clarification on the term "virtual machine"
  - a *virtual machine* (*VM*) is a virtual entity that emulates the hardware of a real computer
  - a *virtual machine instance* (*VM instance*) is a VM along with its OS, applications, and state



  - however, the term virtual machine is also often used to indicate a virtual machine instance

Hypervisor type 1 and 2

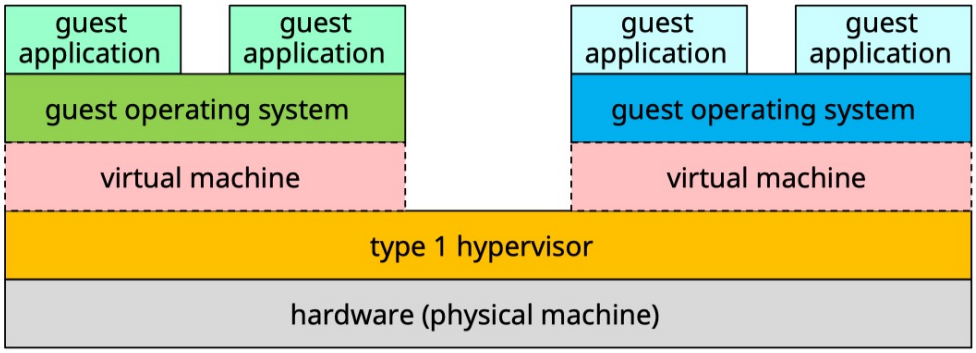
- Two main types of hypervisors
  - *type 2* (*hosted VMM* or *hosted hypervisor*)
    - VMware Workstation, Oracle VM Virtualbox, ...



Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti

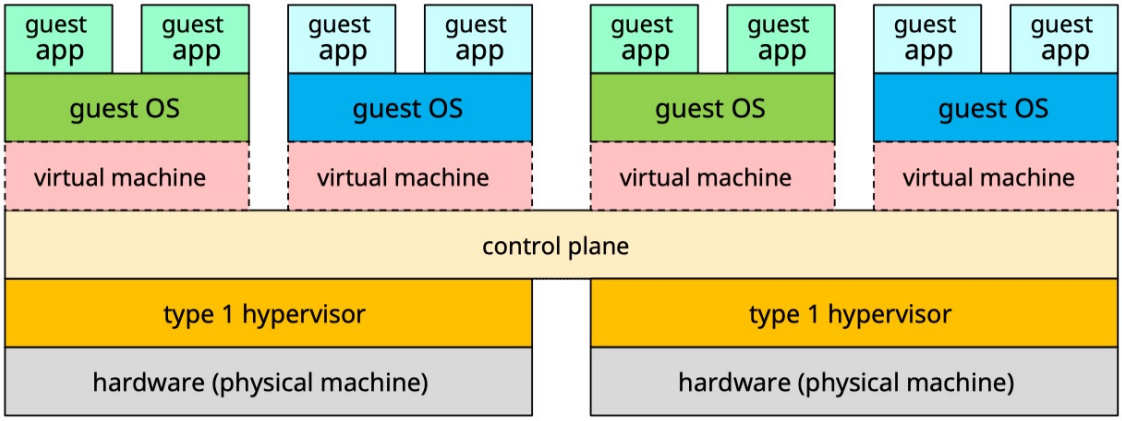
Hypervisor type 1 and 2

- Two main types of hypervisors
  - *type 1* (*native VMM* or *bare-metal hypervisor*)
    - VMware vSphere, Xen, ...



Hypervisor type 1 and 2

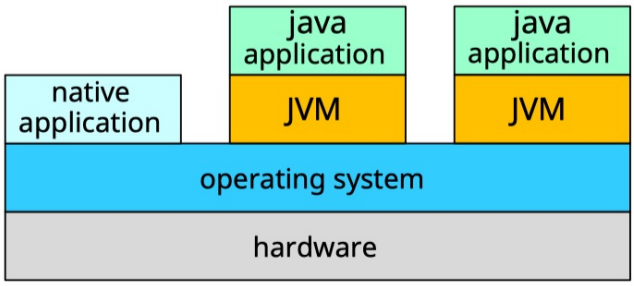
- Two main types of hypervisors
  - Type 1 hypervisors can support virtualization of a cluster of physical hosts - using a distributed software layer (*control plane*)



Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti

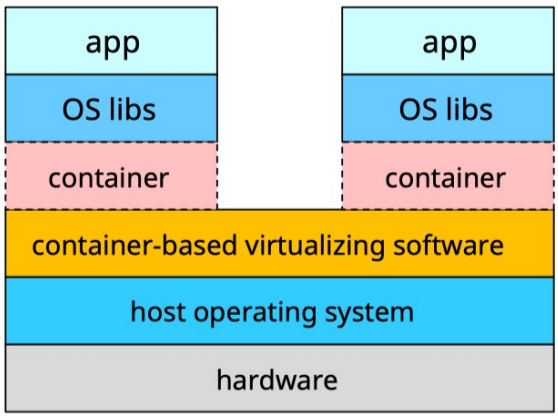
Process virtualization

- Another form of virtualization is the *process virtualization* (*process virtualization*)
  - e.g., the Java Virtual Machine (JVM)



Container-based virtualization

- Another form of virtualization is the *virtualization based on container* (*container-based virtualization* or *OS-level virtualization*)
  - every *container* is a computer with a virtual OS kernel (the host OS kernel)



* System virtualization techniques

- The hardware of a computer (real or virtual) is composed of a set of resources (real or virtual) - CPU, memory, I / O, networks, storage, ...
  - the hypervisor must provide and manage the virtual resources of the VMs in terms of the underlying physical resources
  - there are multiple virtualization techniques for each type of resource
  - We now describe some virtualization techniques and options used in system virtualization

- General requirements for virtualization

- Virtualization requirements
  - a hypervisor must provide the illusion that each VM acts like a real computer
  - a hypervisor should deliver this illusion effectively and efficiently with these characteristics
    - *fidelity* - the behavior of a program in a VM should match that of a real computer
    - *safety* - the hypervisor should have complete control of the virtualized resources
    - *efficiency* - most of the VM code should be executed directly from the host computer, without hypervisor intervention

Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti

- Processor virtualization

- A processor is characterized by its ISA (Instruction Set Architecture)
  - the ISA defines the processor's instructions and its state (registers and memory)
  - two main approaches to processor virtualization
    - processor emulation
    - processor virtualization

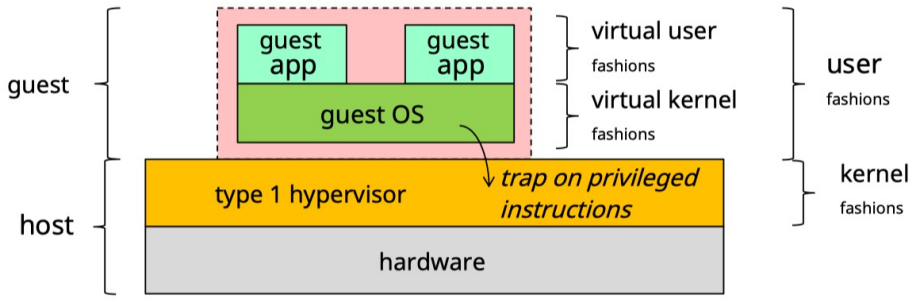
- L'*processor emulation* takes place through the *virtualization of the ISA* (CPU emulation)
  - is based on the *binary translation* of the instructions
  - today it is an unimportant technique, thanks to the widespread use of x86 processors

Processor virtualization

- If the real and virtual processor are of the same type, then many instructions of the software running in a VM can be executed directly by the host processor
  - however, there are "problematic" instructions - which do not need to be executed directly by the host processor
  - in particular, some of the guest OS kernel instructions are "problematic"
    - e.g., instructions for enabling and disabling interrupts and instructions for managing the Memory Management Unit (MMU)

Hardware-assisted virtualization

- The main technique for processor virtualization is the *hardware-assisted virtualization* (also called *trap-andemulate*)
  - VM instructions are executed by the host's real processor - which normally executes them directly - but which captures the VM's non-virtualizable instructions ("trap") and handles them as hypervisor calls ("emulated")
    - relies on the use of hardware virtualization technologies - e.g., Intel VT-x and AMD SVM in x86 processors (since 2005)



Processor virtualization

- Additional Processor Virtualization Techniques
  - *full virtualization*
    - combines direct execution (of most instructions) with binary translation (of "problematic" instructions)
  - *paravirtualization*
    - the guest OS is modified to eliminate all "problematic" instructions from its kernel, replacing them with hypervisor calls
  - are techniques used mostly before hardware processor virtualization technologies - x86 processor virtualization began in the late 1990s

Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti
Sapienza
Università di Roma

Virtualization of multiple processors

- Processor virtualization is particularly effective when applied to multi-core processors or multiprocessor computers
  - the host's physical processors (with their cores) are virtualized into virtual CPUs (*virtual CPU* or *vCPU*) and assigned to VMs
    - each vCPU has only one core
    - one or more vCPUs can be assigned to each VM
    - it is also possible to specify quotas
  - in this way, a multi-processor system is virtualized into a multi-computer system

- Memory virtualization

- In system virtualization, you need to manage a double layer of physical memory virtualization - because each VM has its own virtual memory
  - today processors provide hardware support for nested memory virtualization - e.g. Intel EPT and AMD NPT technologies (since 2008)
- Hypervisors also offer additional techniques for efficient memory management
  - eg, page deduplication and ballooning

Dipartimento di Ingegneria informatica
automatica e gestionale Antonio Ruberti

- I / O virtualization

- Usually the hypervisor Not assigns the hardware I / O devices (such as disks and network cards) physically present on the host to the VMs
  - the hypervisor assigns virtual devices to each VM, which may also be different from those physically present on the host and which can be configured separately for each VM
  - the hypervisor can also assign VMs to virtual devices that do not have an equivalent physical counterpart - e.g., a virtual switch
  - the guest OS accesses these virtual devices through its own drivers, as if they were real devices
    - the I / O operations for these virtual devices are then captured by the hypervisor, which manages them appropriately
    - Physical device drivers often need to be used as well

- Storage virtualization

- Storage virtualization aims to abstract the physical storage layer from that of the VMs
  - some possible options
    - a virtual disk that corresponds directly to a physical disk or to one of its partitions
      - performance is better
    - a virtual disk implemented as a set of host files - *image file* (*disk image files*)
      - flexibility is greater
  - in any case, the VMs see virtual disks

Storage virtualization

- Storage virtualization can be applied
  - to drives directly connected to the host computer
    - DAS unit, Direct Attached Storage
  - to networked units
    - Storage Area Network (SAN) units - provide block access to remote units
    - Network Attached Storage (NAS) drives - provide access to files on remote drives
  - SAN / NAS drive-based virtualization increases flexibility
- The *shared folders* (*shared folder*) are a typical feature of type 2 hypervisors
  - one or more shared folders physically reside on the host's file system - and are shared between the host OS and one or more guest VMs

- Virtualization of the network

- A physical network
  - a set of physical hosts - each with one or more physical network cards (pNICs)
  - one or more physical switches (pSwitches), to connect multiple physical network segments
- A virtual network (as a first approximation)
  - a set of VMs - each with one or more virtual network adapters (vNICs)
  - one or more virtual switches (vSwitches), to connect the virtual network cards (vNICs) to each other and to the physical network cards (pNICs)
  - to connect the VMs to each other and to the network to which the host belongs, in an appropriate way