Introduction to Virtualization

Unit-1

What is IT infrastructure?

- Information technology infrastructure, or IT infrastructure, refers to the combined components needed for the operation and management of enterprise IT services and IT environments.

Why is IT infrastructure important?

With an IT infrastructure, a company can:

- Provide a positive customer experience by providing uninterrupted access to its website and online store.
- Develop and start solutions to market with speed.
- Collect data in real time to make quick decisions.
- Improve employee productivity.

How do the components of IT infrastructure work?

Two core groups of components are:

hardware and **software**.

Hardware uses software—like an **operating system**—to work. Likewise, an operating system manages system resources and hardware.

Operating systems also connect software applications and physical resources using **networking components**.

Hardware

Hardware components can include:

- Desktop computers
- Servers
- Data centers
- Hubs
- Routers
- Switches
- Facilities

Facilities

Facilities or **physical plants** provide space for networking hardware, servers and data centers. It also includes the network cabling in office buildings to connect components of an IT infrastructure together.

Network

Networks are composed of **switches, routers, hubs and servers**. Switches connect network devices like routers, servers and others on local area networks (LAN). Routers allow devices on different LANs to communicate and move packets between networks. Hubs connect multiple networking devices to act as a single component.

Server

A core hardware component needed for an enterprise IT infrastructure is a server. **Servers** are essentially computers that allow multiple users to access and share resources.

Server room/data center

Organizations house multiple servers in rooms called server rooms or data centers. Data centers are the core of most networks.

Software

Software components can include:

- Content management systems (CMS)
- Customer relationship management (CRM)
- Enterprise resource planning (ERP)
- Operating systems
- Web servers

Types of infrastructure

The two primary types of IT infrastructure are :

- ***Traditional IT infrastructure, and***

- ***Cloud infrastructure.***

Traditional IT Infrastructure

- A traditional IT infrastructure is made up of the usual hardware and software components: facilities, data centers, servers, networking hardware desktop computers and enterprise application software solutions.

- Typically, this infrastructure setup requires more power, physical space and money than other infrastructure types. A traditional infrastructure is typically installed on-premises for company-only or private use.

Cloud infrastructure

- A cloud computing IT infrastructure is similar to traditional infrastructure.
- However, end users can access the infrastructure via the internet, with the ability to use computing resources without installing on-premises through virtualization.
- Virtualization connects physical servers maintained by a service provider at any or many geographical locations.
- Then, it divides and abstracts resources, like storage, to make them accessible to users almost anywhere an internet connection can be made.

An optimal IT infrastructure

- IT infrastructure setups vary by business needs and goals, but some goals are universal for every enterprise.

- The optimal infrastructure provides a business **high-performance storage**, a **low-latency network**, **security,** an **optimized wide area network (WAN)**, **virtualization** and **zero downtime**.

An optimal IT infrastructure

- **High-performance storage** systems store and back up data and include a data recovery system in case of disasters.
- **Low-latency networks** use enterprise-level infrastructure components to reduce the delay of data flow.
- **Secure infrastructures** include systems that control information access and data availability. It can also safeguard a business against breaches and cyberattacks wherever the data resides, maintaining the customers' trust.
- **WANs** manage the network by prioritizing traffic and giving certain applications more or less bandwidth as needed.
- **Virtualization** provides faster server provisioning, increases uptime, improves disaster recovery and saves energy.
- **Zero downtime** aims to reduce disruptions to business operations and eliminates system downtime to keep costs down and profits up.

Virtualization

- Virtualization technology is one of the fundamental components of cloud computing, especially in regard to infrastructure-based services.

- Virtualization allows the creation of a secure, customizable, and isolated execution environment for running applications.

- The basis of this technology is the ability of a computer program—or a combination of software and hardware—to emulate an executing environment separate from the one that hosts such programs.

- For example, we can run Windows OS on top of a virtual machine, which itself is running on Linux OS.

Virtualization

- The term virtualization is often synonymous with **hardware virtualization**, which plays a fundamental role in efficiently delivering **Infrastructure-as-a-Service (IaaS)** solutions for cloud computing.

- Virtualization technologies provide a virtual environment for not only executing applications but also for storage, memory, and networking.

Virtualization technologies have gained renewed interested recently due to the confluence of several phenomena:

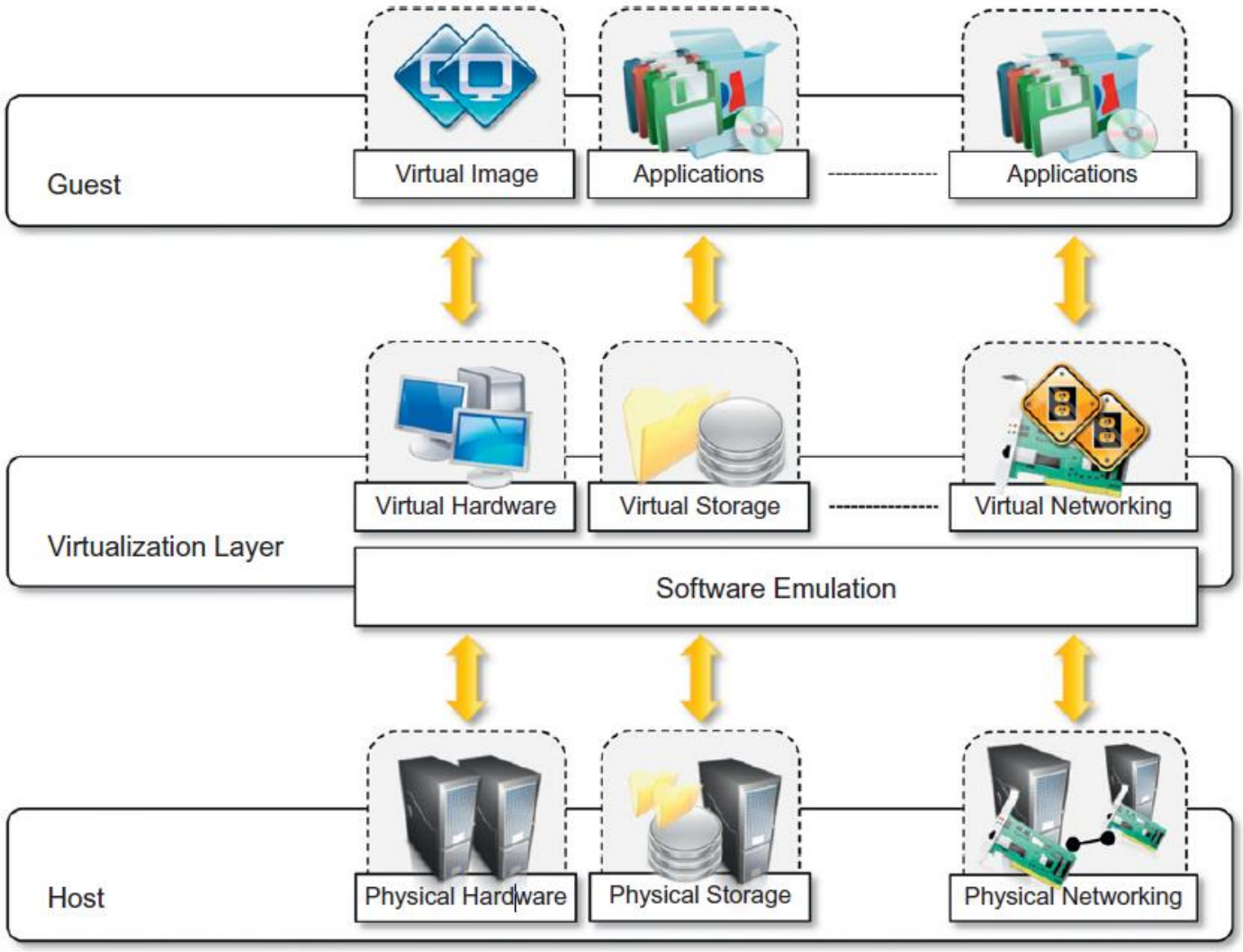
- Increased performance and computing capacity.
- Underutilized hardware and software resources.
- Lack of space.
- Greening initiatives.
- Rise of administrative costs.

Characteristics of virtualized environments

Three major components of a virtualized environment:

- *guest,*
- *host, and*
- *Virtualization layer (virtual machine manager)*

The virtualization reference model



Characteristics of virtualized environments

1. Increased security (*controls and filters, sandboxed environment*)
2. Managed execution (*sharing, aggregation, emulation, and isolation*)
3. Performance tuning
4. Virtual machine migration
5. Portability

Advantages of Virtualization in Cloud Computing

- 1. Cost Savings
- 2. Resource Utilization
- 3. Scalability
- 4. Isolation
- 5. Improved Management
- 6. Flexibility
- 7. Disaster Recovery

Challenges of Virtualization in Cloud Computing

- Performance Overhead
- Security Concerns
- Complexity
- Resource Contention

How Does Virtualization Work?

- The core of virtualization is using hypervisors or virtual machine monitors (VMMs).
- These vital software components serve as bridges between actual hardware and virtual computers.
- Hypervisors give room to the construction, administration, and execution of virtual machines (VMs), assuring their isolation and effective use of computer resources.

Types of Hypervisors

- **Type 1 Hypervisor (Bare-Metal)**

This hypervisor operates directly on physical hardware without needing an underlying operating system. **VMware ESXi** and **Microsoft Hyper-V** are two examples.

- **Type 2 Hypervisor (Hosted)**

These hypervisors run on top of a host operating system and use its resources. **VMware Workstation** and **Oracle VirtualBox** are two well-known examples.

Taxonomy of virtualization techniques

- Virtualization is mainly used to emulate **execution environments**, **storage**, and **networks**.

- **Execution virtualization** constitutes the **oldest**, **most popular**, and **most developed** area.

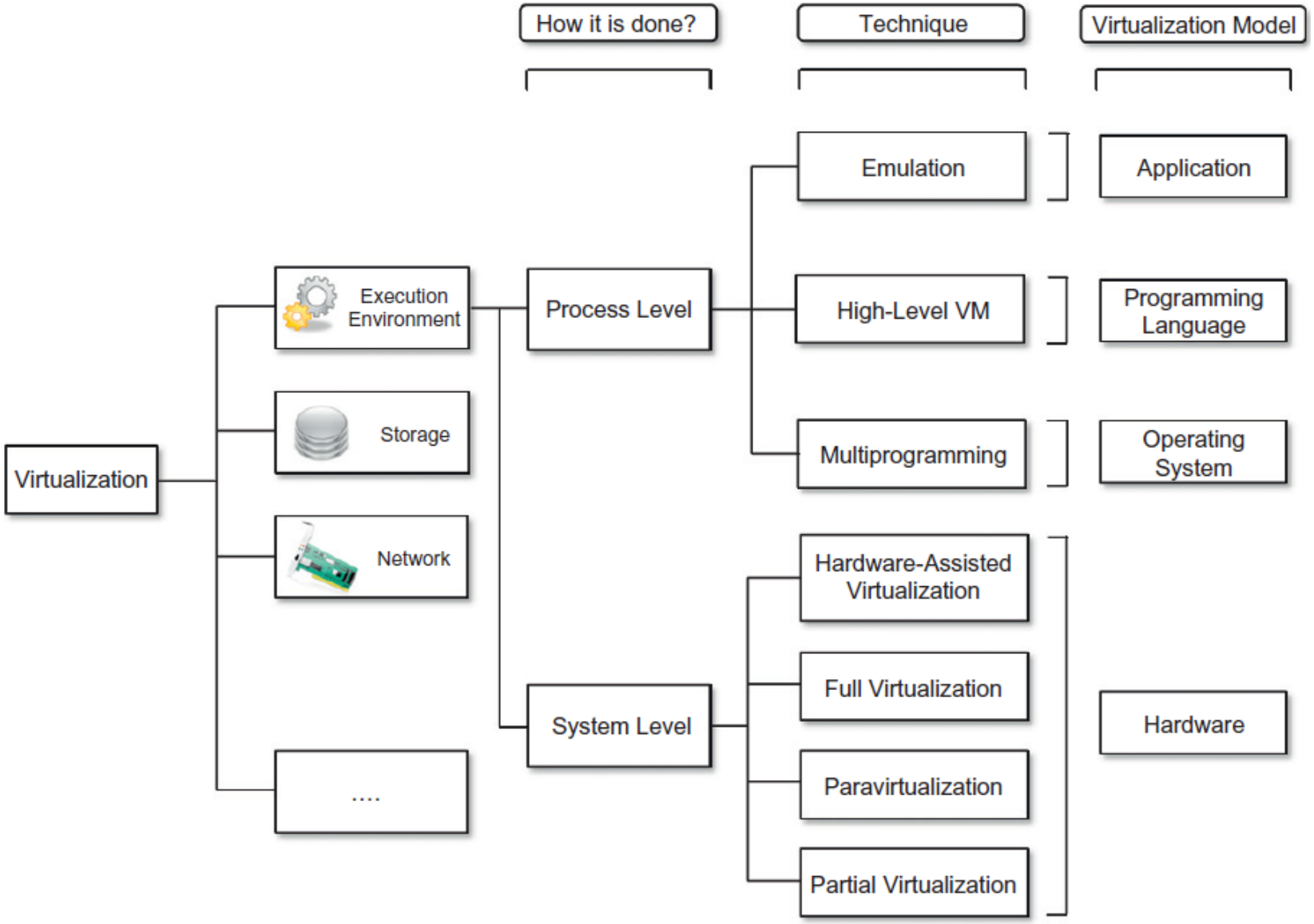
- Execution virtualization techniques can be **divided** into **two** major **categories** by considering the **type of host** they require:
  - *Process-level, and*
  - *System-level*

Taxonomy of virtualization techniques

- ***Process-level*** techniques are implemented on top of an existing operating system, which has full control of the hardware.

- ***System-level*** techniques are implemented directly on hardware and do not require or require a minimum of support from an existing operating system.

Taxonomy of virtualization techniques



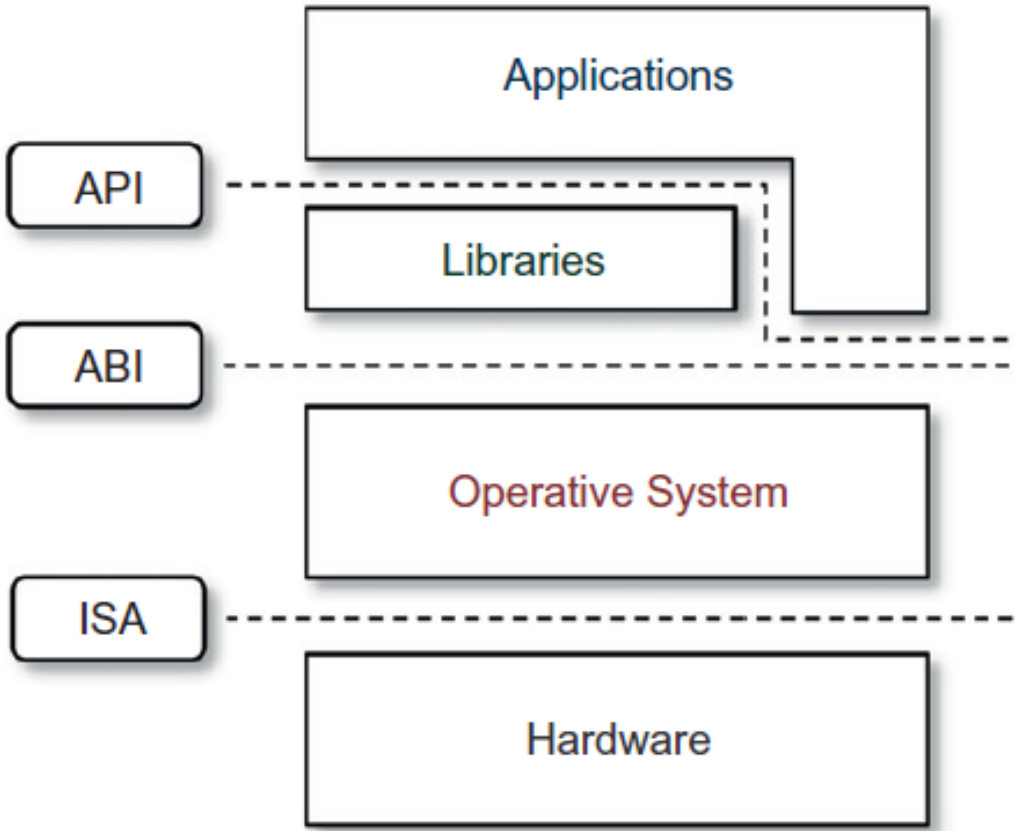
Execution virtualization

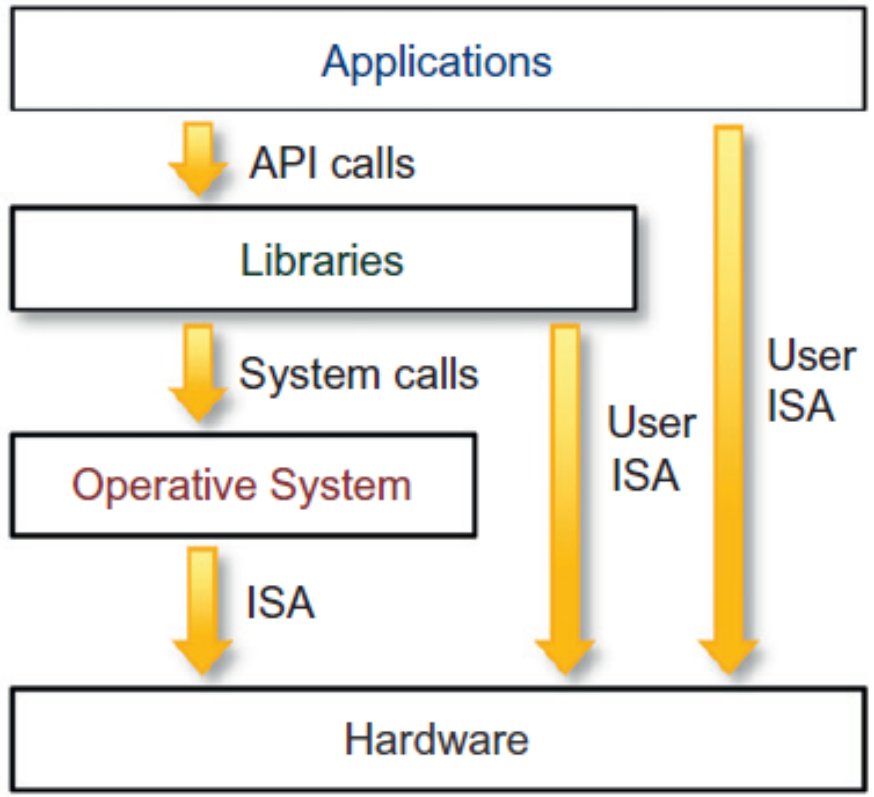
- Execution virtualization includes all techniques that aim to emulate an execution environment that is separate from the one hosting the virtualization layer.

- Virtualizing an execution environment at different levels of the computing stack requires a reference model that defines the interfaces between the levels of abstractions, which hide implementation details.

- From this perspective, virtualization techniques actually replace one of the layers and intercept the calls that are directed toward it.

Machine reference model





Machine reference model

- **Bottom layer:** ***Instruction Set Architecture-*** defines the instruction set for the processor, registers, memory, and interrupt management.
- **ISA:** interface between H/w and S/w. Important for OS developers and developers of applications that directly manage the underlying hardware.

- ***Application binary interface (ABI):*** separates the operating system layer from the applications and libraries, which are managed by the OS.
- System calls are defined at this level.
- This interface allows portability of applications and libraries across operating systems that implement the same ABI.

Machine reference model

- ***Application programming interface(API):*** it represents the highest level of abstraction which interfaces applications to libraries and/or the underlying operating system.

- For any operation to be performed in the application level API, ABI and ISA are responsible for making it happen.

- The high-level abstraction is converted into machine-level instructions to perform the actual operations supported by the processor.

- The machine-level resources, such as processor registers and main memory, are used to perform the operation at the hardware level of the central processing unit (CPU)

Priviliged/Non-privileged Instructions

- The **instruction set** exposed by the hardware has been divided into **different security classes** that define who can operate with them.
- The first distinction can be made between **privileged** and **non-privileged** instructions.
- **Non-privileged** instructions are those instructions that can be used without interfering with other tasks because they do not access shared resources.
- This category contains, for example, all the floating, fixed-point, and arithmetic instructions.

Priviliged/Non-privileged Instructions

- **Privileged** instructions are those that are executed under specific **restrictions** and are mostly used for **sensitive operations**, which **expose** (behavior-sensitive) or **modify**(control-sensitive) the privileged state.

- For instance, **behavior-sensitive** instructions are those that operate on the **I/O**, whereas **control-sensitive** instructions **alter** the **state** of the **CPU registers.**

Priviliged/Non-privileged Instructions

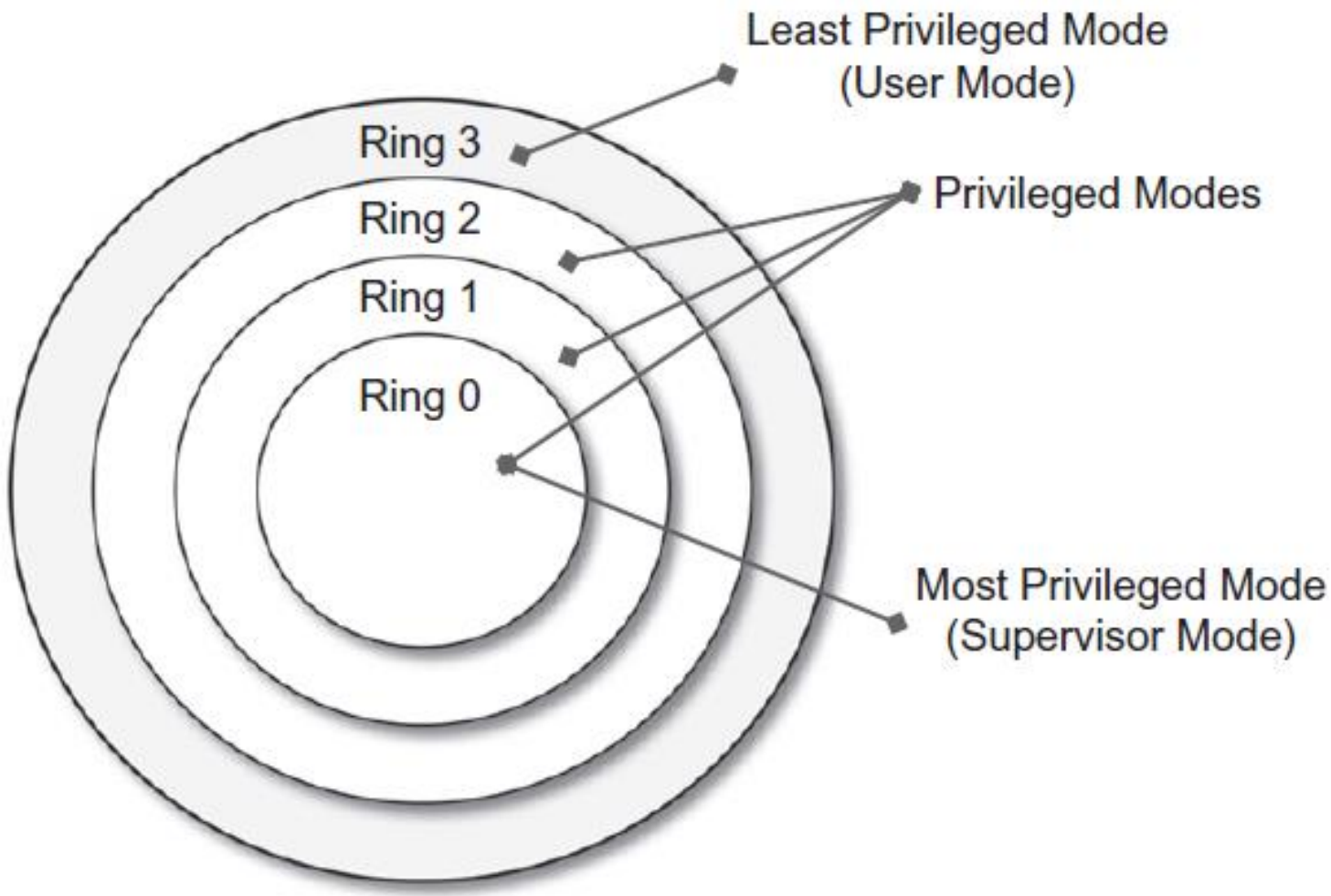
- Some types of architecture feature **more than one class** of privileged instructions and implement a finer control of how these instructions can be accessed.

- For instance, a possible implementation features a hierarchy of privileges in the form of **ring-based security**: Ring 0, Ring 1, Ring 2, and Ring 3; **Ring 0** is in the **most privileged** level and **Ring 3** in the **least privileged** level.

- **Ring 0** is used by the **kernel** of the OS, **rings 1 and 2** are used by the **OS-level services**, and **Ring 3** is used by the **user**.

- **Recent systems** support only **two levels**, with **Ring 0** for **supervisor** mode and **Ring 3** for **user** mode.

Priviliged/Non-privileged Instructions



Supervisor Mode and User Mode

- All the current systems support at least two different execution modes: **supervisor mode** and **user mode**.

- The **first** mode denotes an **execution** mode in which all the instructions (**privileged** and **non-privileged**) can be executed **without** any **restriction**.

- This mode, also called **master mode** or **kernel mode**, is generally used by the operating system (or the hypervisor) to perform **sensitive operations** on **hardware- level resources**.

Supervisor Mode and User Mode

- In **user** mode, there are **restrictions** to control the machine-level resources.
- If code running in user mode invokes the privileged instructions, **hardware interrupts** occur and trap the potentially harmful execution of the instruction.
- The distinction between user and supervisor mode allows us to understand the role of the hypervisor and why it is called that.
- Conceptually, the **hypervisor runs above the supervisor mode**, and from here the prefix hyper is used.
- In reality, hypervisors are run in supervisor mode, and the division between privileged and non-privileged instructions has posed challenges in designing virtual machine managers.
- It is expected that all the sensitive instructions will be executed in privileged mode, which requires supervisor mode in order to avoid traps

Hardware-level virtualization

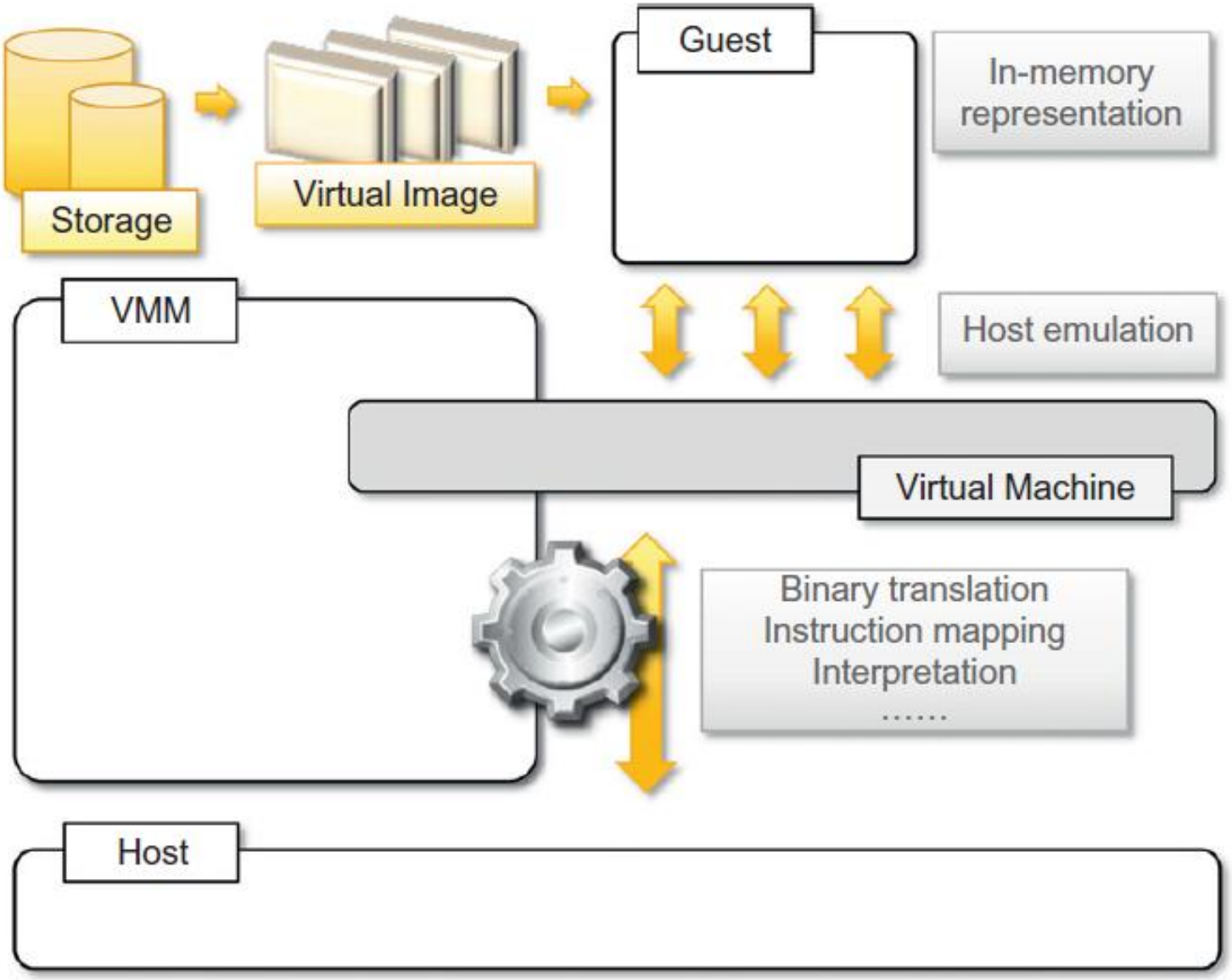
- Provides an abstract execution environment in terms of computer **hardware** on top of which a guest operating system can be run.

- In this model, the **guest** is represented by the **operating system**, the **host** by the **physical computer** hardware, the **virtual machine** by its **emulation**, and the **virtual machine manager** by the **hypervisor**.

- Hardware-level virtualization is also called **system virtualization**, since it provides ISA to virtual machines, which is the representation of the hardware interface of a system.

- This is to differentiate it from process virtual machines, which expose ABI to virtual machines

A hardware virtualization reference model



Hypervisors

- A fundamental element of hardware virtualization is the hypervisor, or virtual machine manager (VMM).

- It recreates a hardware environment in which guest operating systems are installed.

- There are two major types of hypervisor: Type I and Type II.

- **Type I** hypervisors run directly on top of the hardware. Therefore, they take the place of the operating systems and **interact directly with the ISA** interface exposed by the underlying hardware, and they emulate this interface in order to allow the management of guest operating systems.

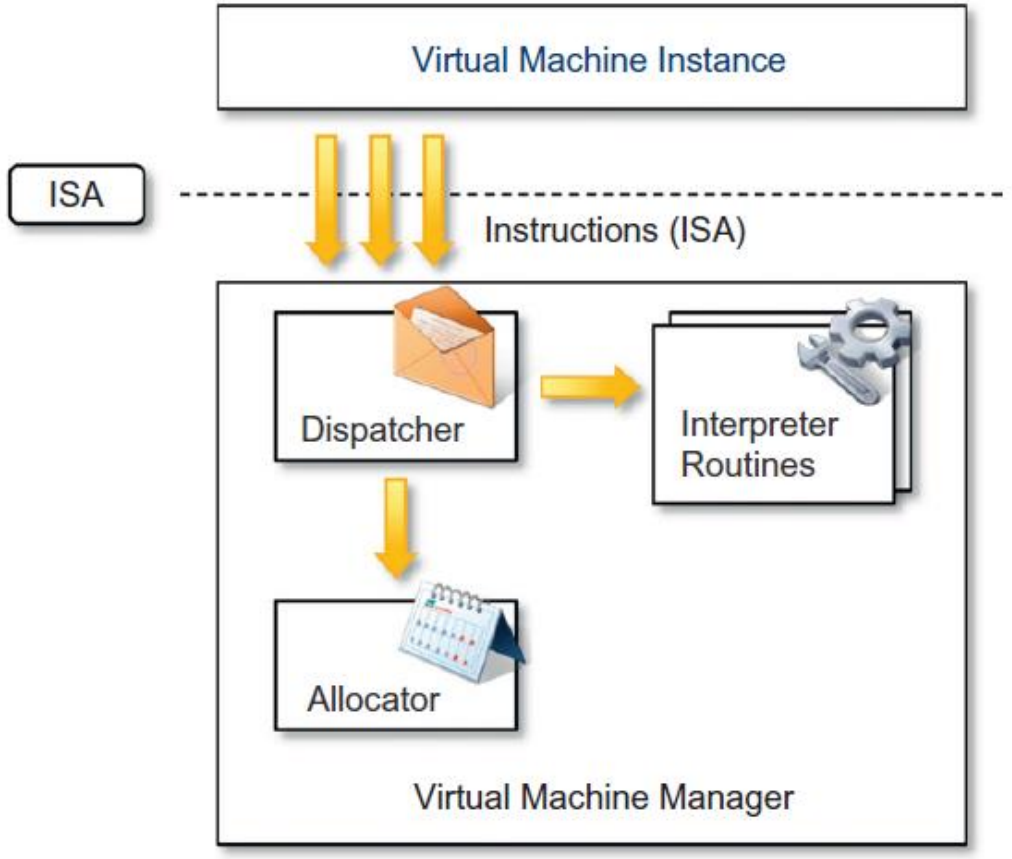
Hypervisors

- Type II hypervisors require the support of an operating system to provide virtualization services.

- This means that they are programs managed by the operating system, which interact with it through the ABI and emulate the ISA of virtual hardware for guest operating systems.

Hypervisors

- Conceptually, a virtual machine manager is internally organized as described.
- Three main modules, dispatcher, allocator, and interpreter, coordinate their activity in order to emulate the underlying hardware.



Types of Virtualization

- **Hardware Virtualization**

The term refers to a scenario in which the hardware provides architectural support for building a virtual machine manager able to run a guest operating system in complete isolation.

Types:

1. Full virtualization: VMM is installed directly on hardware (bare-metal/type-1)
2. Para-virtualization: VMM installed on top of a host OS (type-2)
3. Partial virtualization: Entire hardware is not virtualized, only a part of H/W can be virtualized.

Eg. Address space virtualization (most common). This allows **multiple applications and users** to run concurrently in a **separate memory space**, but they still share the **same hardware resources.**

Operating system virtualization

- It offers the opportunity to create different and **separated execution environments** for applications that are managed concurrently.
- There is **no virtual machine manager or hypervisor**, and the virtualization is done within a single operating system.
- The OS kernel allows for **multiple isolated user space** instances.
- The **kernel** is also **responsible** for **sharing** the system resources among instances and for limiting the impact of instances on each other.
- Operating system-level virtualization aims to provide separated and multiple execution containers for running applications.
- Operating system-level virtualization does not expose the same flexibility of hardware virtualization, since all the user space instances must share the same oper- ating system .

Other types of virtualization

- **Storage virtualization:**

It is a system administration practice that allows decoupling the physical organization of the hardware from its logical representation.

Using this technique, users do not have to be worried about the specific location of their data, which can be identified using a logical path.

Storage virtualization allows us to harness a wide range of storage facilities and represent them under a single logical file system.

There are different techniques for storage virtualization, one of the most popular being network-based virtualization by means of storage area networks (SANs).

Other types of virtualization

- **Network virtualization**

It combines hardware appliances and specific software for the creation and management of a virtual network.

Network virtualization can aggregate different physical networks into a single logical network **(external network virtualization)** or provide network-like functionality to an operating system partition **(internal network virtualization)**.

The result of external network virtualization is generally a **virtual LAN (VLAN).**

Other types of virtualization

- **Desktop virtualization**

Desktop virtualization abstracts the desktop environment available on a personal computer in order to provide access to it using a client/server approach.

Desktop virtualization provides the same out- come of hardware virtualization but serves a different purpose.

Other types of virtualization

- **Application server virtualization**
- Application server virtualization abstracts a collection of application servers that provide the same services as a single virtual application server by using load-balancing strategies and providing a high-availability infrastructure for the services hosted in the application server.

Virtualization vs cloud computing

	Virtualization	Cloud
Definition	Technology	Methodology
Purpose	Create multiple simulated environments from 1 physical hardware system	Pool and automate virtual resources for on-demand use
Use	Deliver packaged resources to specific users for a specific purpose	Deliver variable resources to groups of users for a variety of purposes
Configuration	Image-based	Template-based
Lifespan	Years (long-term)	Hours to months (short-term)
Cost	High capital expenditures (CAPEX), low operating expenses (OPEX)	Private cloud: High CAPEX, low OPEX Public cloud: Low CAPEX, high OPEX
Scalability	Scale up	Scale out
Workload	Stateful	Stateless
Tenancy	Single tenant	Multiple tenants