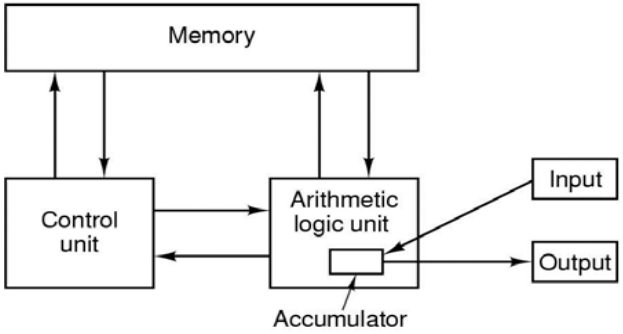
Distributed Systems

Cloud Virtualisation

The von Neumann Architecture

- All general purpose computers are now based on the key concepts of the von Neumann architecture:
  - A single read-write memory for data and instructions
  - The memory is addressable by location in a way which does not depend on the contents of the location
  - Execution proceeds using instructions from consecutive locations unless an instruction modifies this sequentiality explicitly
- The von Neumann bottleneck: a single channel for both instructions and data
- Harvard Architecture: separate memories - double the bandwidth of the simple von Neumann architecture

The von Neumann and Harvard Architectures



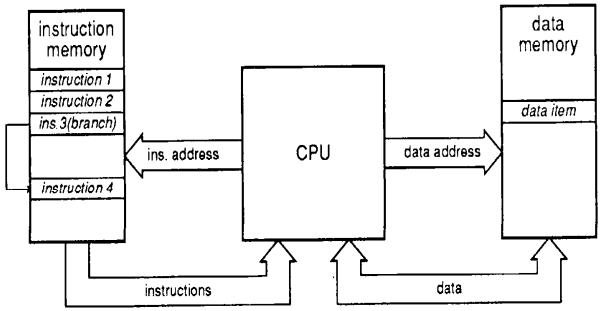


Figure 3: The Harvard architecture

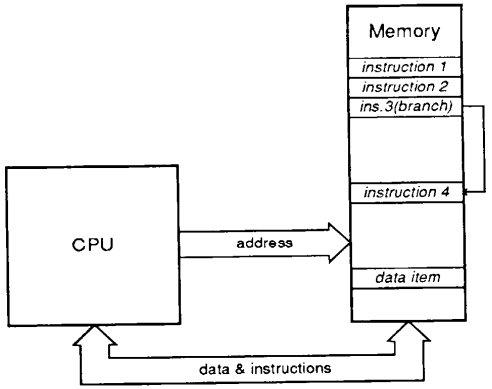


Figure 2: The von Neumann architecture

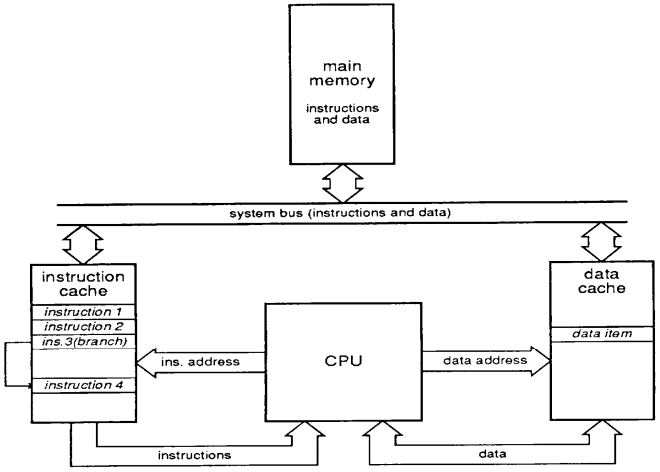


Figure 4: A modified Harvard architecture

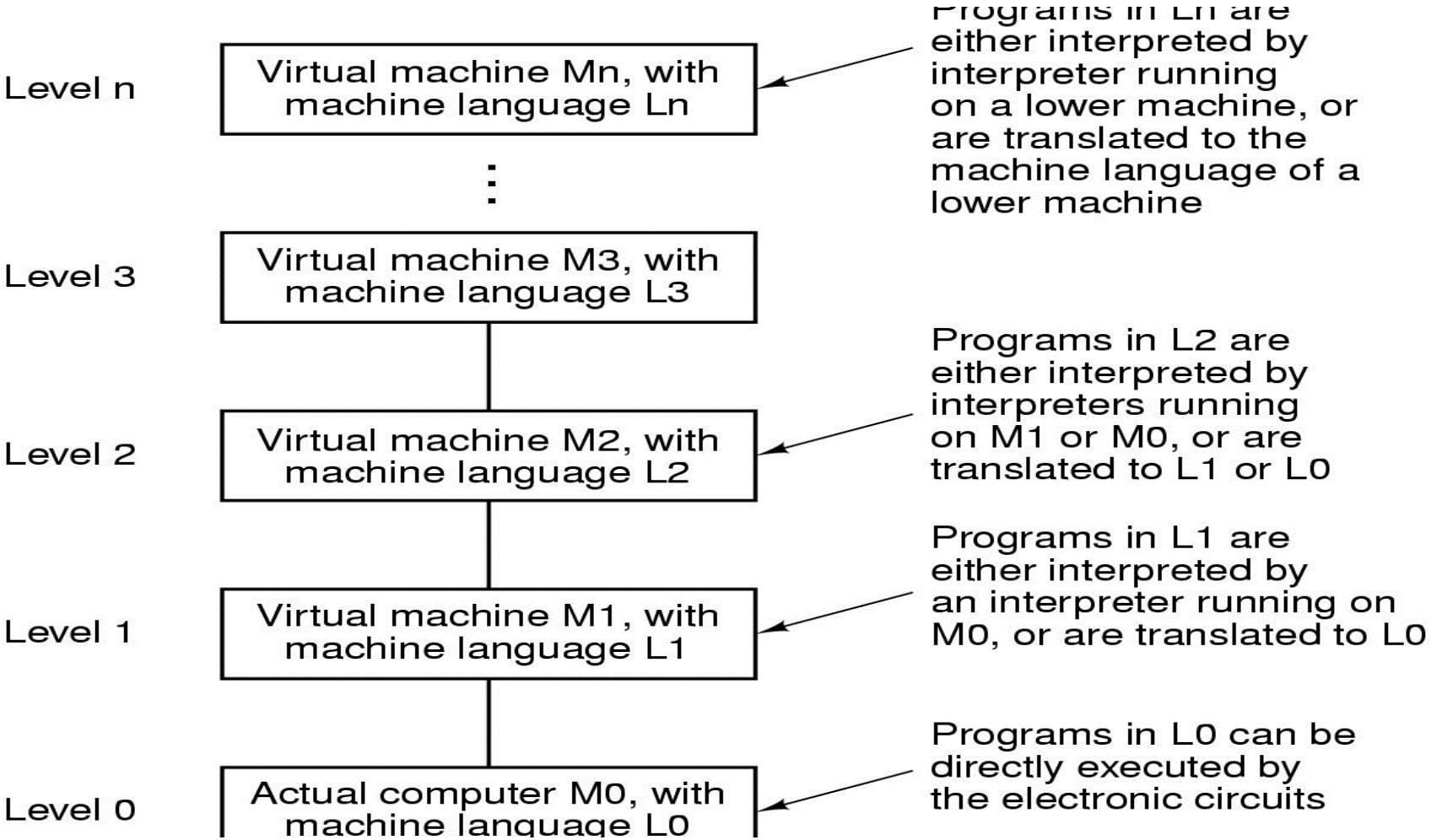
Computer Components

- CPU
  - Datapath
  - Control
- Memory (hierarchy)
  - Main Memory
  - Secondary memory
  - Cache
- I/O devices
- Buses (external and internal to CPU)

Instruction Execution: The Fetch-Decode-Execute Cycles

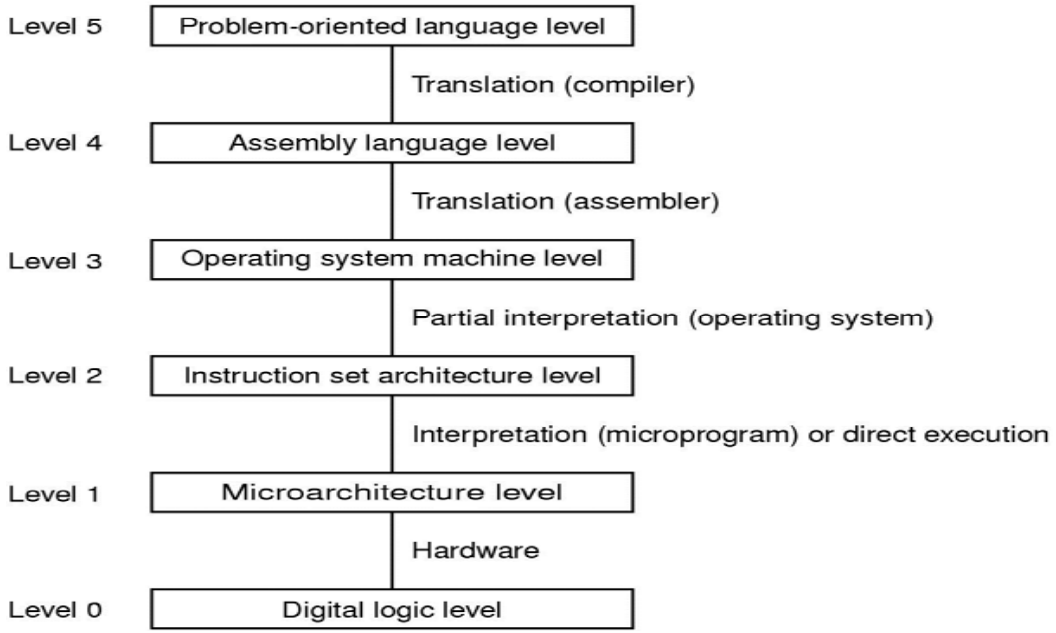
- The CPU executes instructions in a series of small steps:
  - **Fetch** the next instruction from memory in the *Instruction Register (IR)*
  - Change the *Program Counter (PC)* to point to the following instruction
  - **Determine the type** of fetched instruction
  - If the instruction uses a word in memory, determine where it is
  - Fetch the word, if needed, into the CPU
  - **Execute** the instruction
  - Go to step 1

Multi-Level Approach

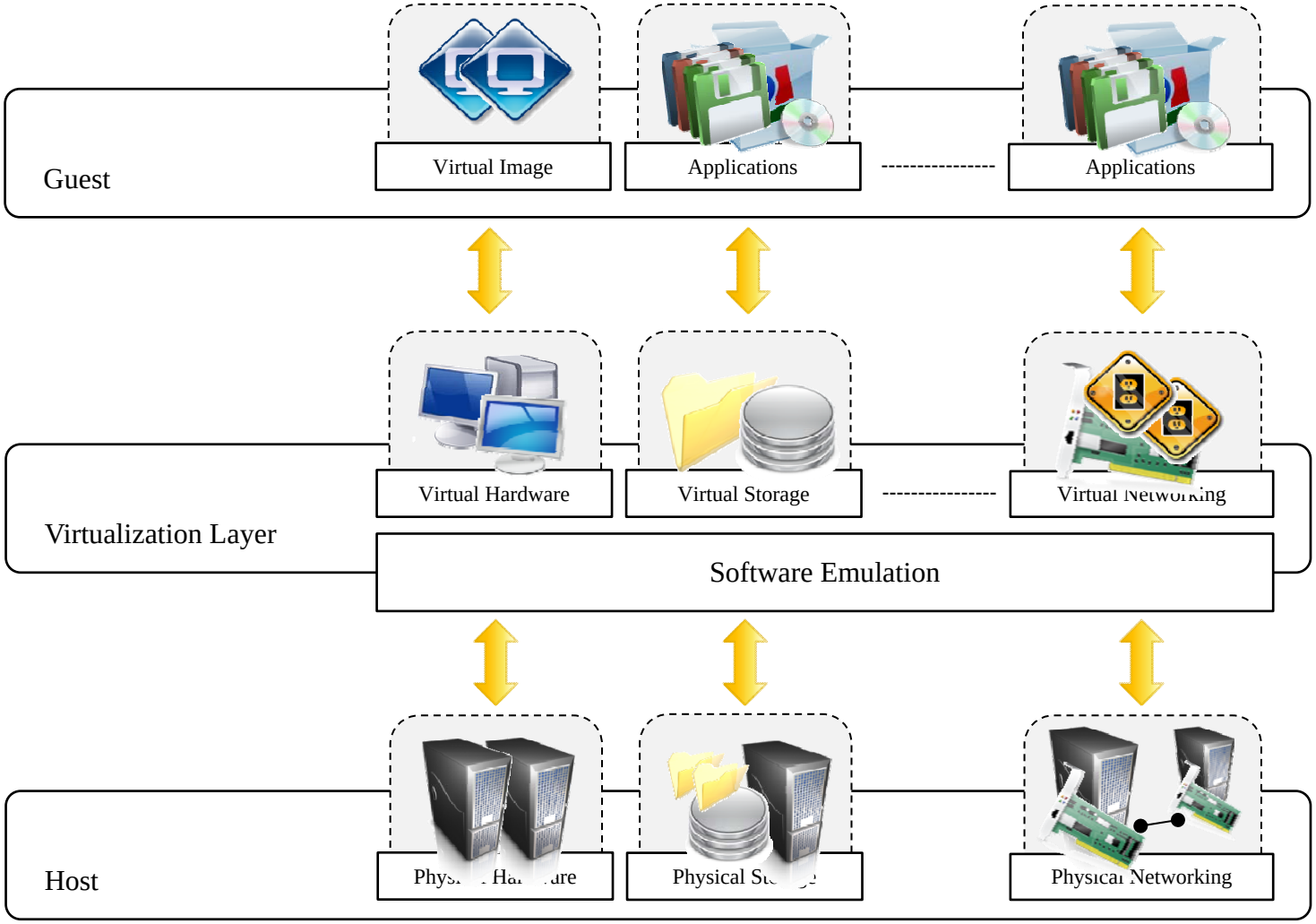


Contemporary Typical Multilevel Machines

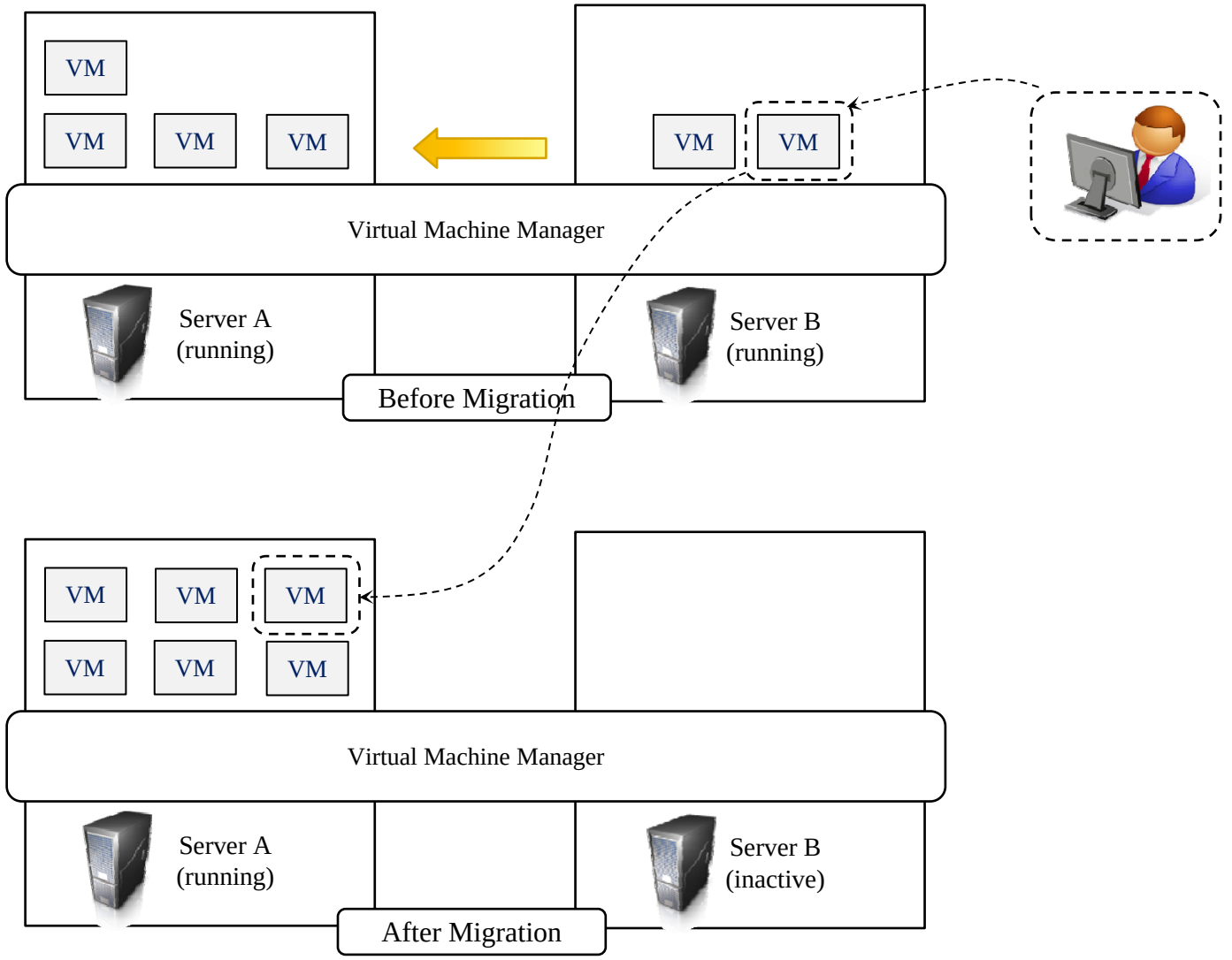
- Digital Logic design. Gate level
- Micro-architecture Level
  - Programming Model (Registers)
  - Datapath & Control
- Instruction Set Architecture Level-ISA
  - Instruction types and formats
  - Addressing



Virtualisation Reference Model



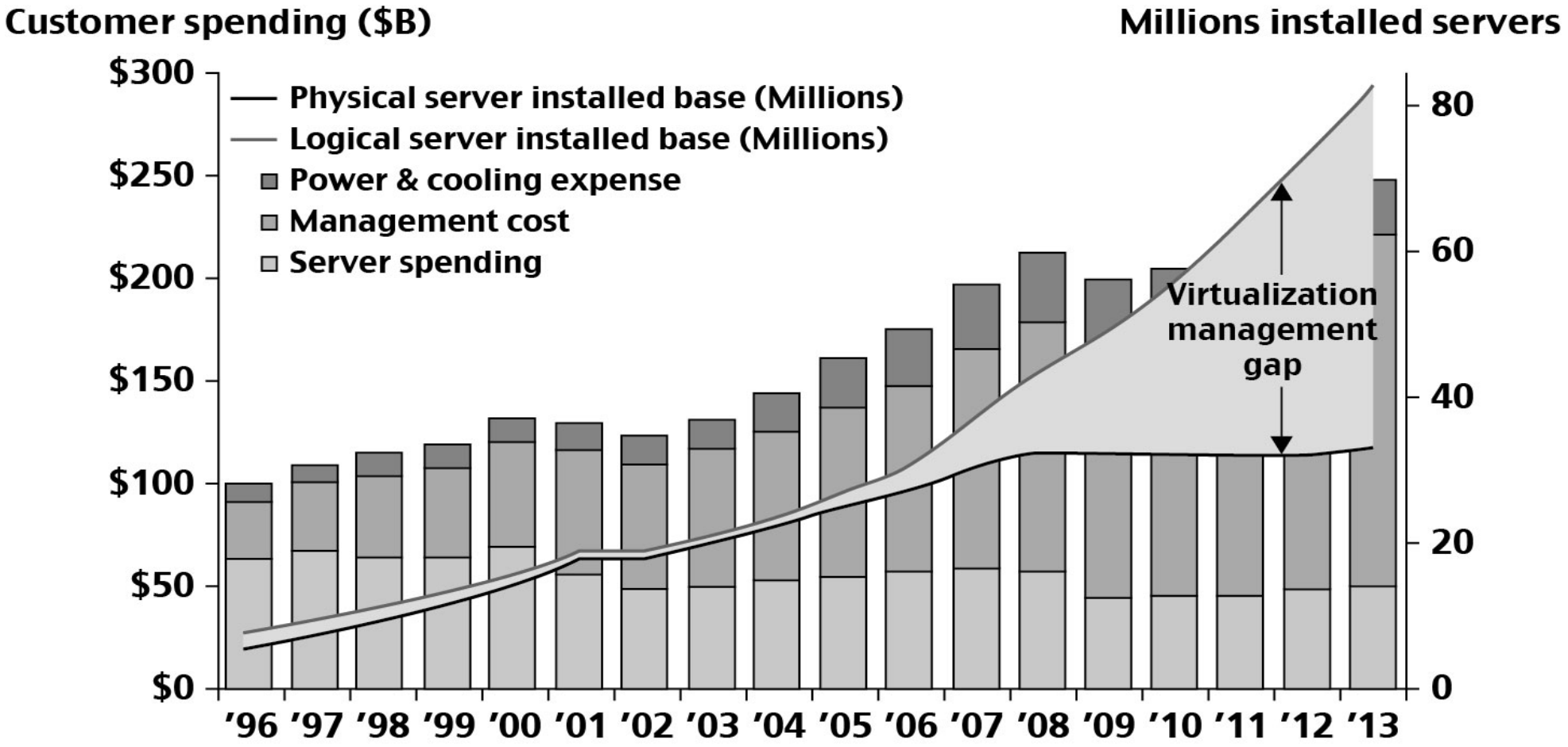
Virtualisation and Cloud



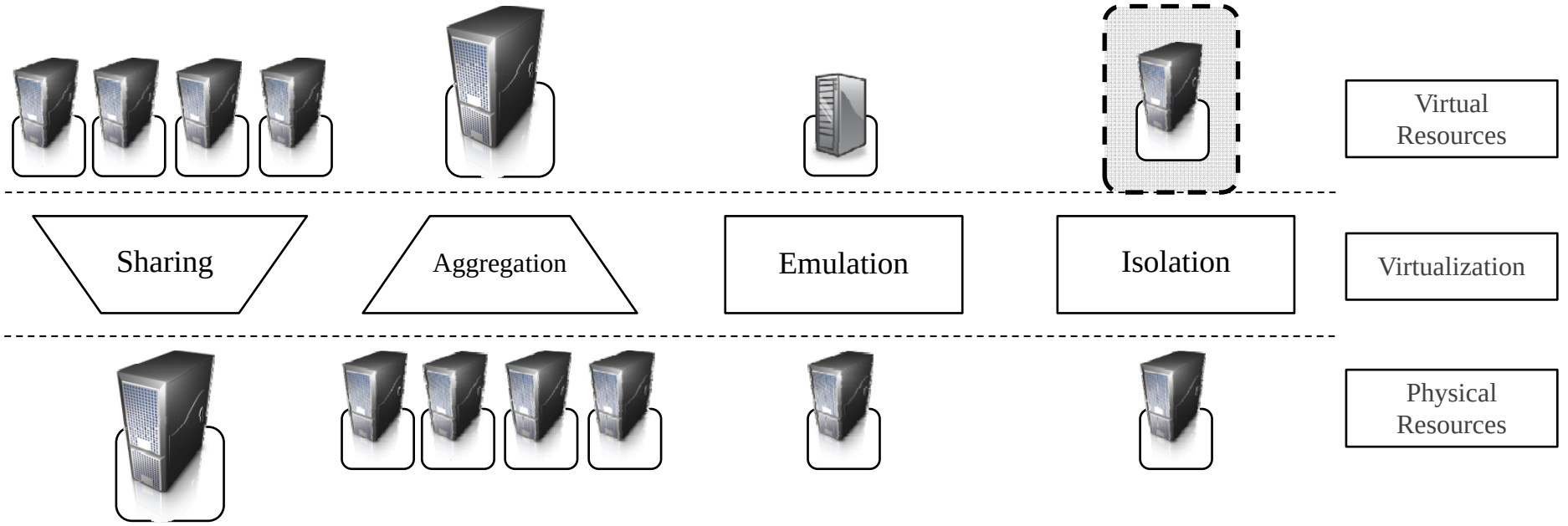
Motivation for virtualisation

- Originated from hardware virtualisation
- Performance
- Computing capacity
- Resource utilisation
- Lack of space - server consolidation
- Greening initiatives
- Admin costs

Datacenter and Server Cost Distribution



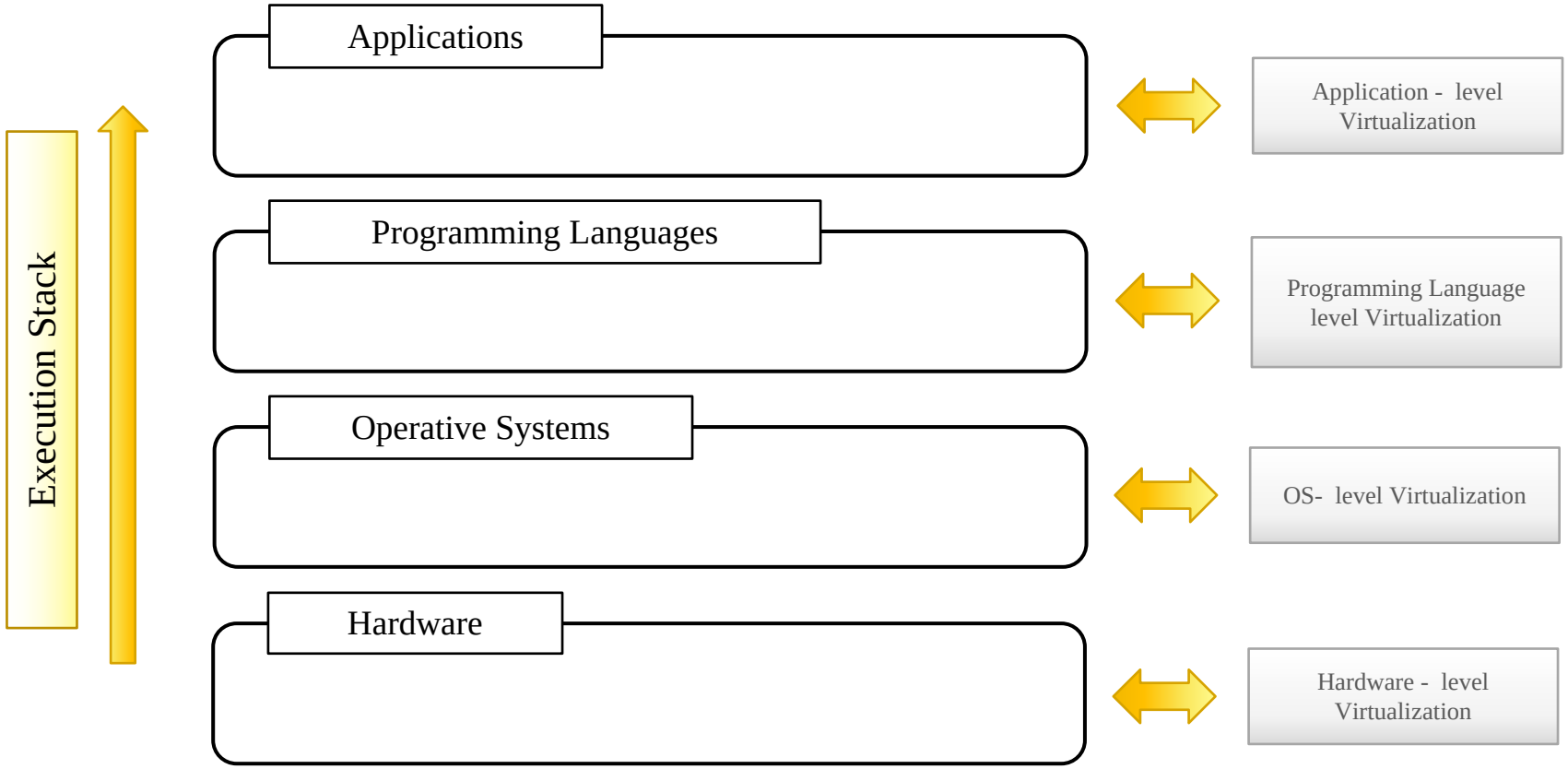
Managed Execution

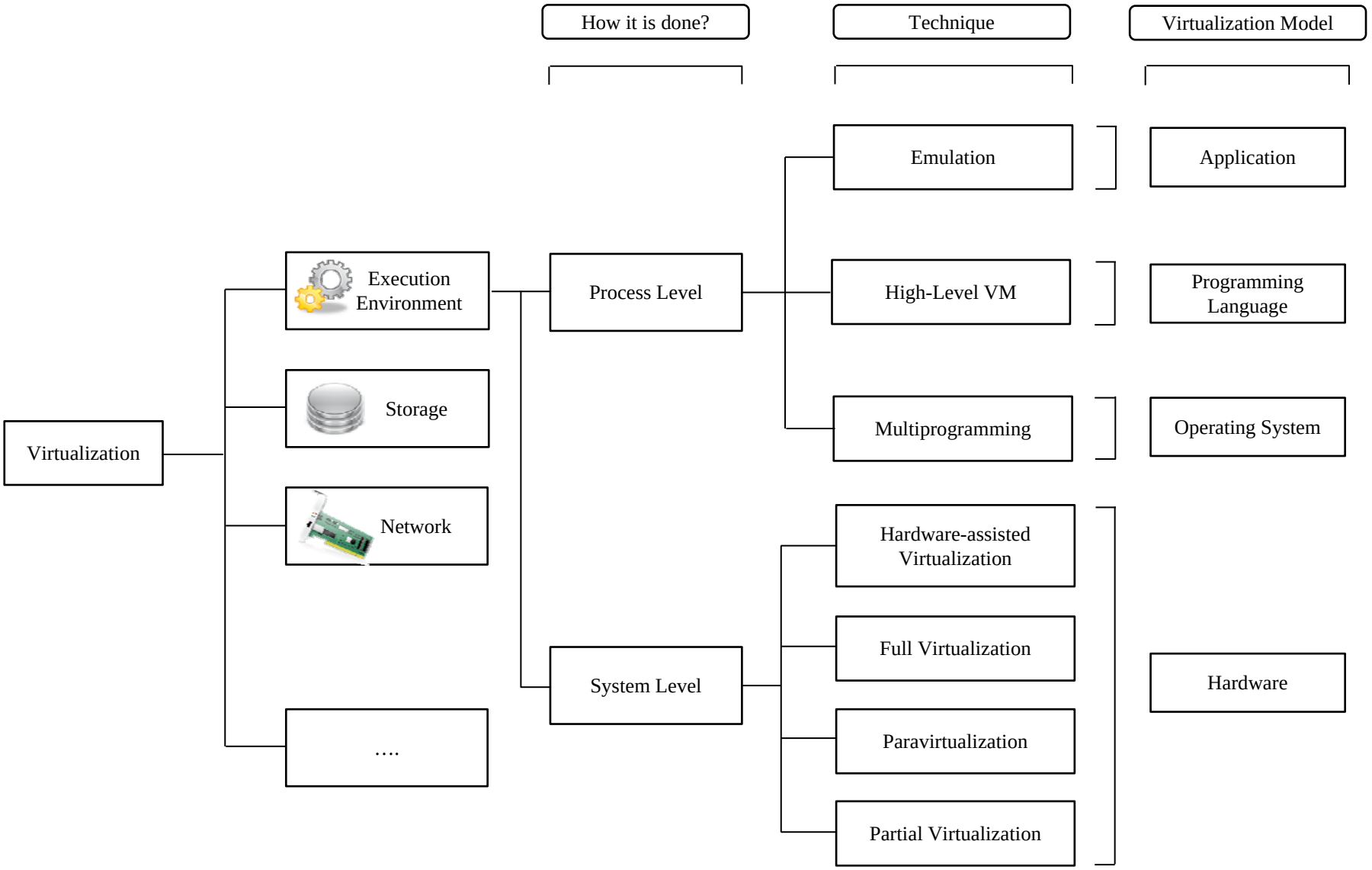


VM-driven Infrastructure vs Cloud-driven Infrastructure

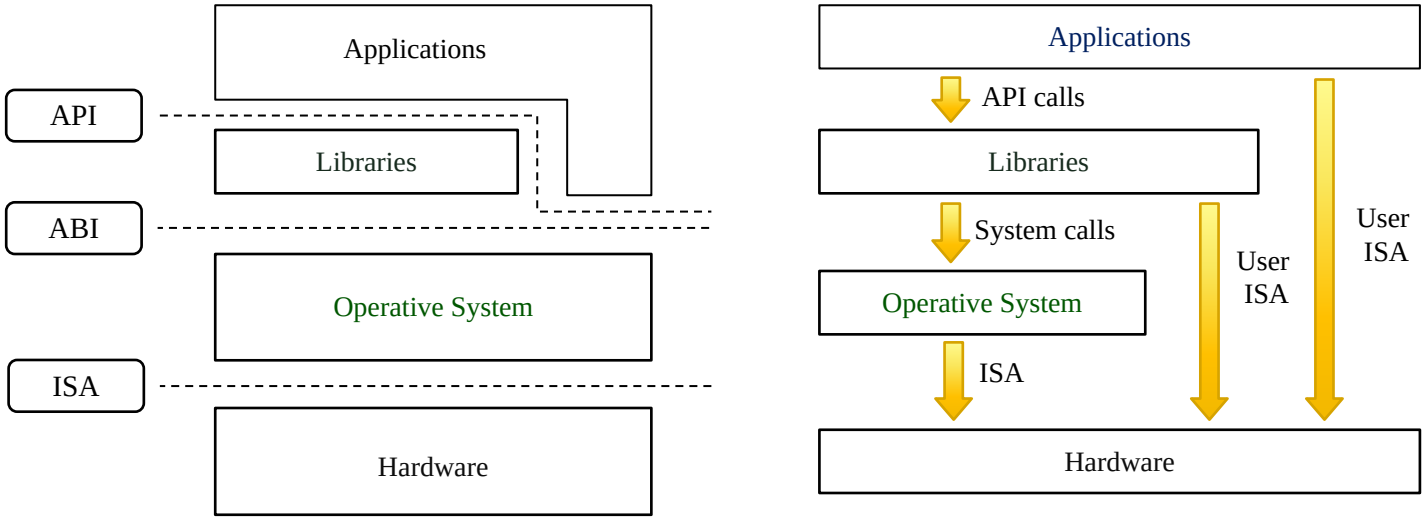
- User driven Provisioning, instead of admin driven
- Public Scalability – Can not scale beyond organizational hardware
- Pay for what you use (unless private cloud), invest in infrastructure
- Resilience and uptime guarantees (unless private cloud), VM is as resilient as internal infrastructure
- If organization loses network connectivity cloud infrastructure is lost – internal VM driven infrastructure stay online for use.

Levels of Virtualisation




How it is done?
Technique
Virtualization Model
Virtualization
Execution Environment
Storage
Network
….
Process Level
System Level
Emulation
High-Level VM
Multiprogramming
Hardware-assisted Virtualization
Full Virtualization
Paravirtualization
Partial Virtualization
Application
Programming Language
Operating System
Hardware

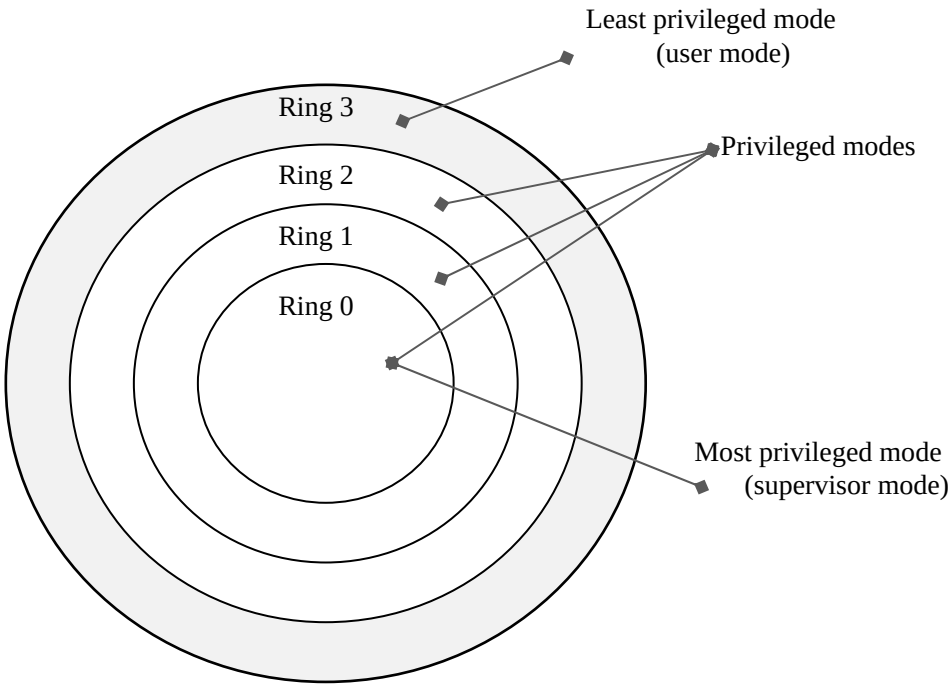
Execution Virtualisation: A machine reference model



- Virtualising 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 implementations details
- Virtualisation techniques replace one of the layers and intercept the calls that are directed toward it
- Required: emulation of interfaces and clear interaction with the underlying layer

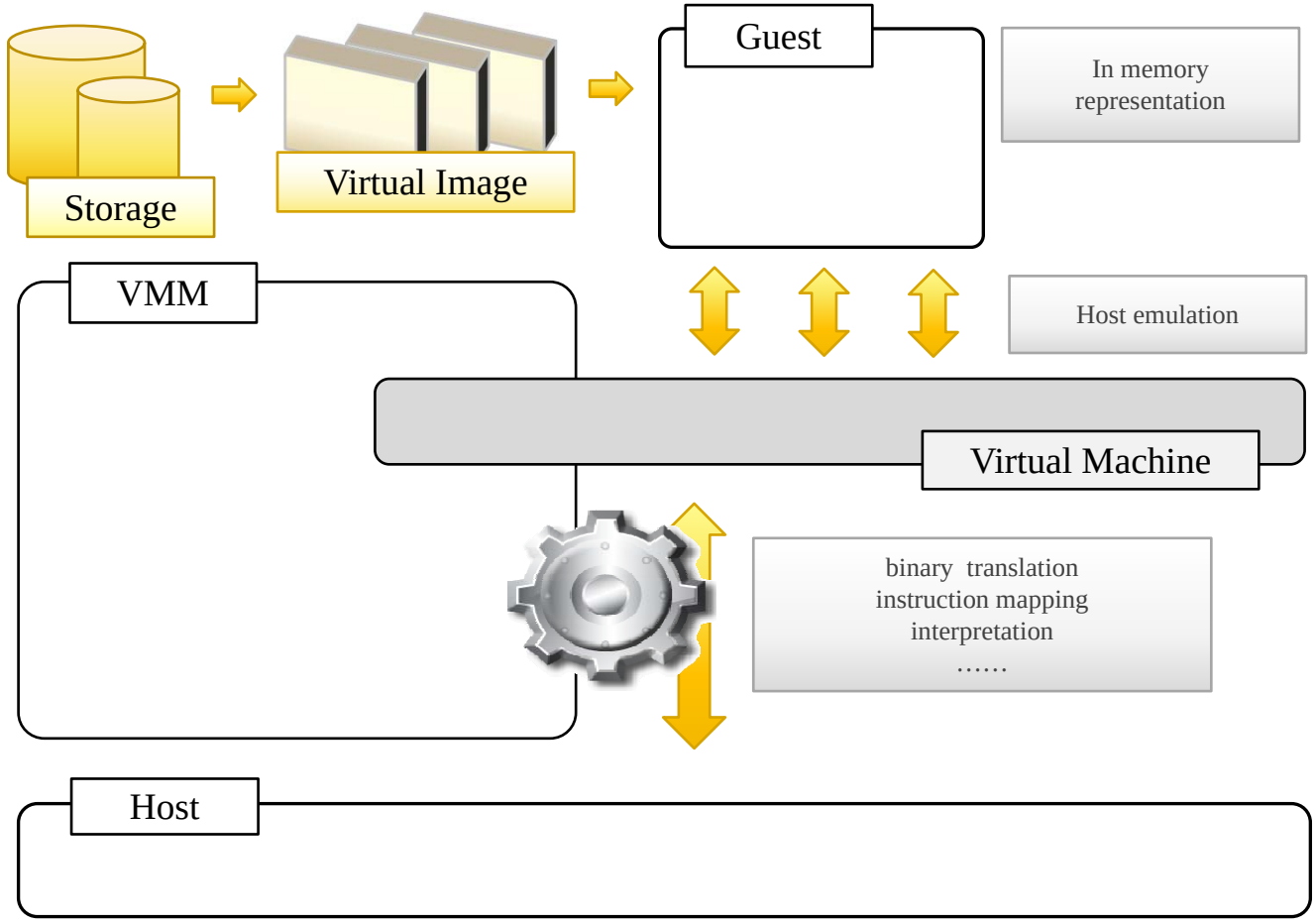
Execution Virtualisation

- Instruction set exposed by the hardware is divided into different security classes
  - non-privileged: not accessing shared resources
  - Privileged: used under specific restrictions, mostly used for sensitive operations
    - Behaviour-sensitive: expose privileged state
    - Control-sensitive: modify privileged state
- Execution can be in
  - Supervisor mode (kernel)
  - User mode (non privileged)



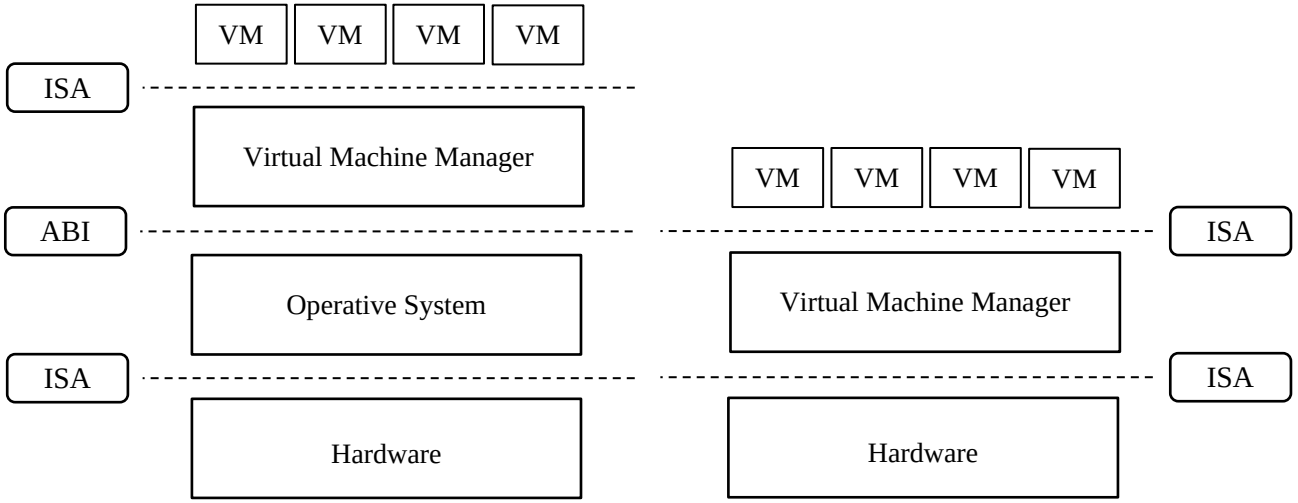
Hardware level Virtualisation – reference model

- Hardware virtualisation: provides an abstract execution environment on top of which a guest operating system can be run



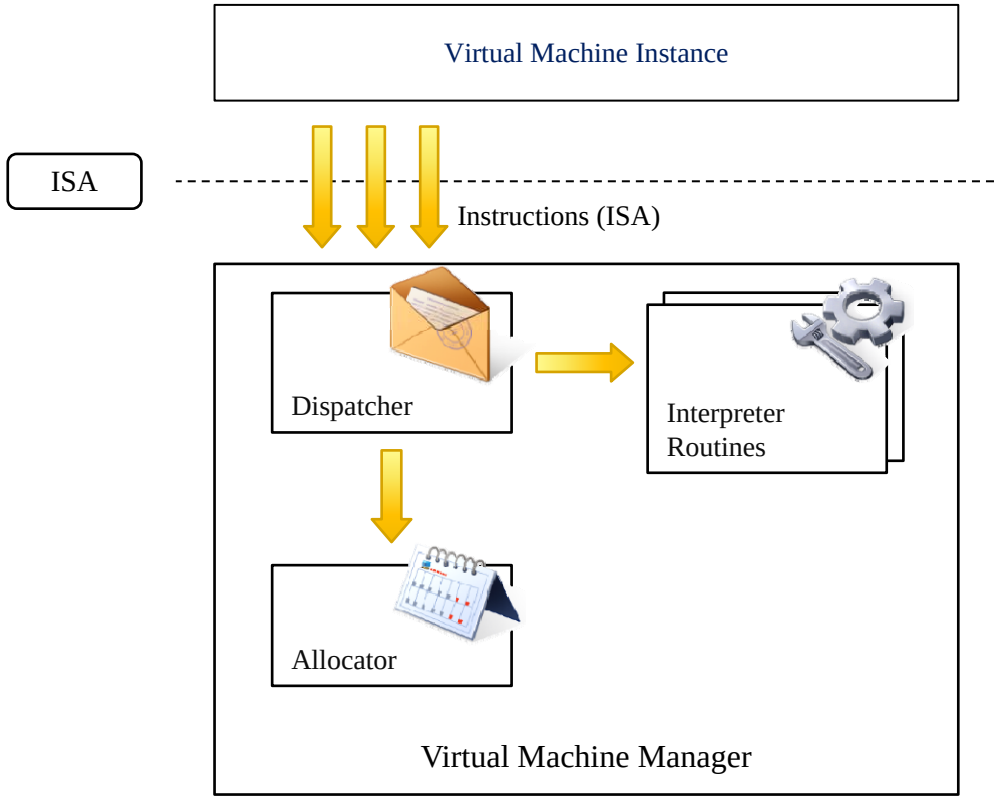
Hypervisors

- Type I (or native virtual machine) : run directly on top of the hardware – interact directly with ISA
- Type II (or hosted virtual machine)



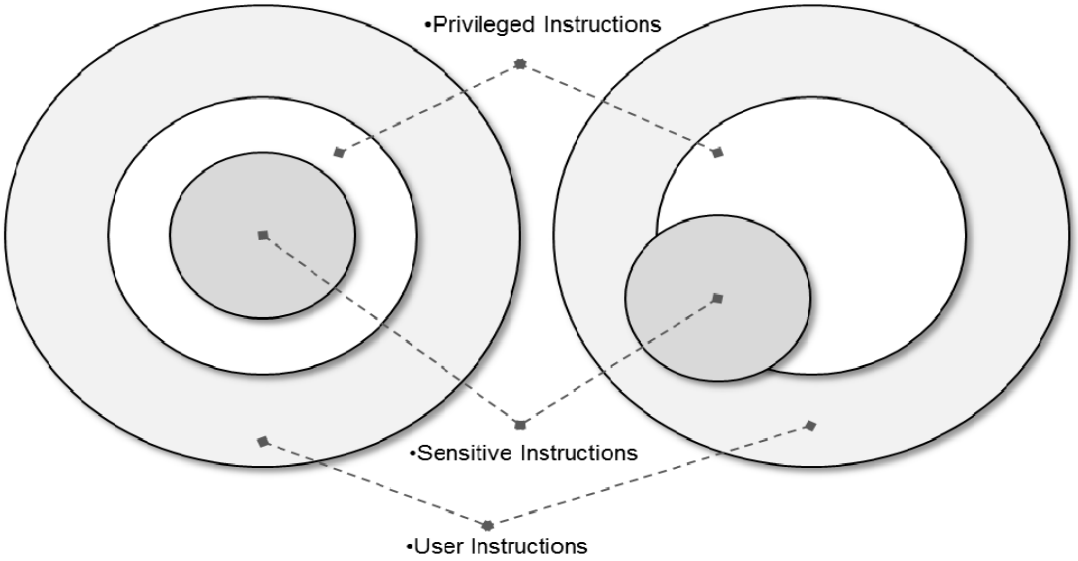
Hypervisors

- **Equivalence**
- **Resource control**: VMM should be in complete control
- **Efficiency**: a statistically dominant fraction of machine instructions should be executed without intervention from the VMM
- This is determined by the layout of the ISA of the host



Hypervisors

- **Theorem 1.** For Any conventional 3rd generation computer, a VMM may be constructed if the set of the sensitive instructions for that computer is a subset of the set of privileged instructions



- **Theorem 2**. A conventional 3rd generation computer is recursively virtualizable if:
  - It is virtualizable and
  - A VMM without any timing dependencies can be constructed for it
- **Theorem 3**. A hybrid VMM can be constructed for any conventional 3rd generation machine in which the set of user-sensitive instructions is a subset of the set of privileged instructions

Hardware Virtualisation techniques

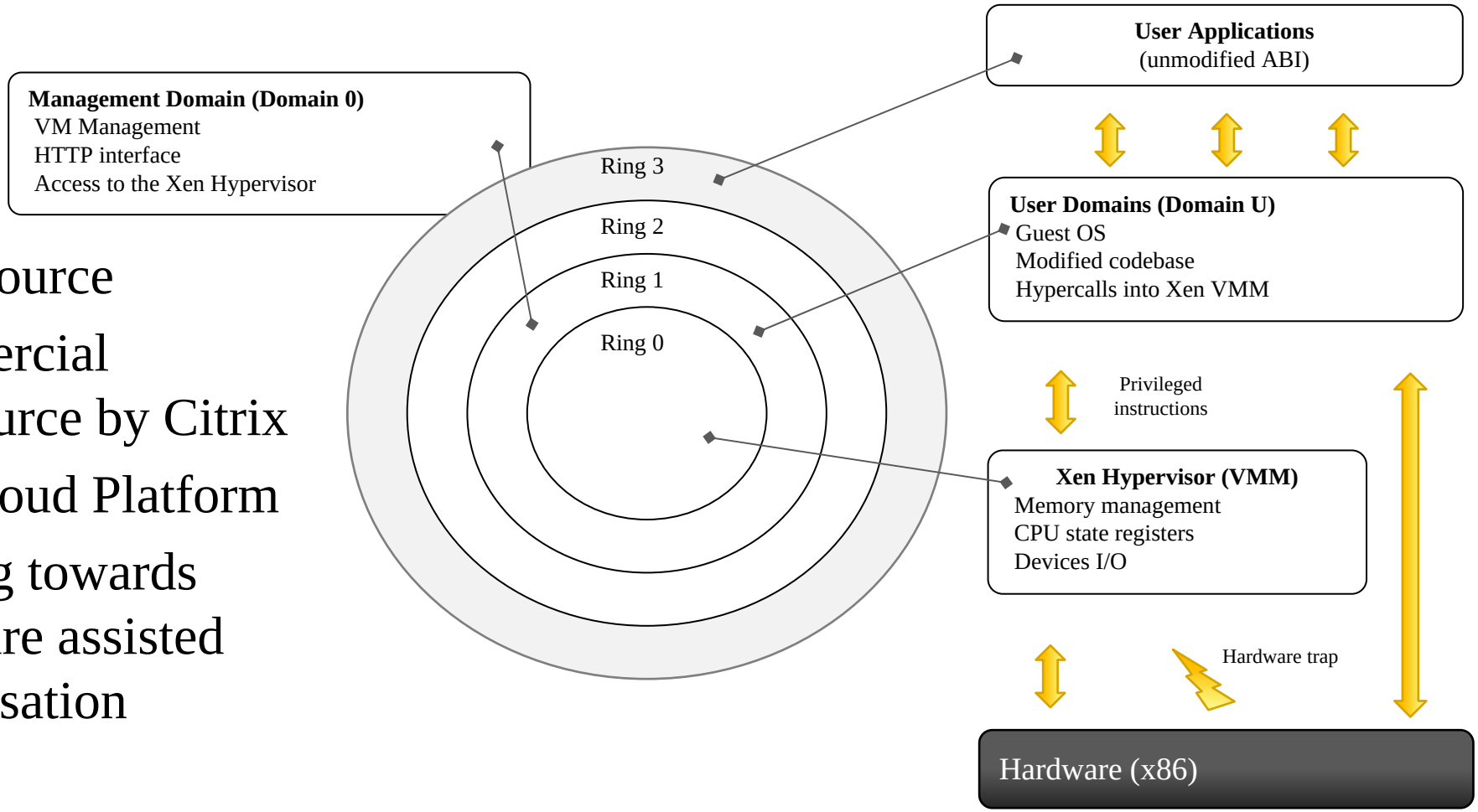
- Hardware-Assisted Virtualisation: run a guest operating system in complete isolation (Intel VT, AMD V)
- Full Virtualisation: run OS on top of a virtual machine
  - Key challenge: interception of privileged instructions
- Paravirtualisation: nontransparent, supports thin VMM. Guests need to be modified. Simplicity
- Partial Virtualisation: partial emulation of host, does not allow the execution of the guest OS in complete isolation. Not all features of OS are supported

Disadvantages

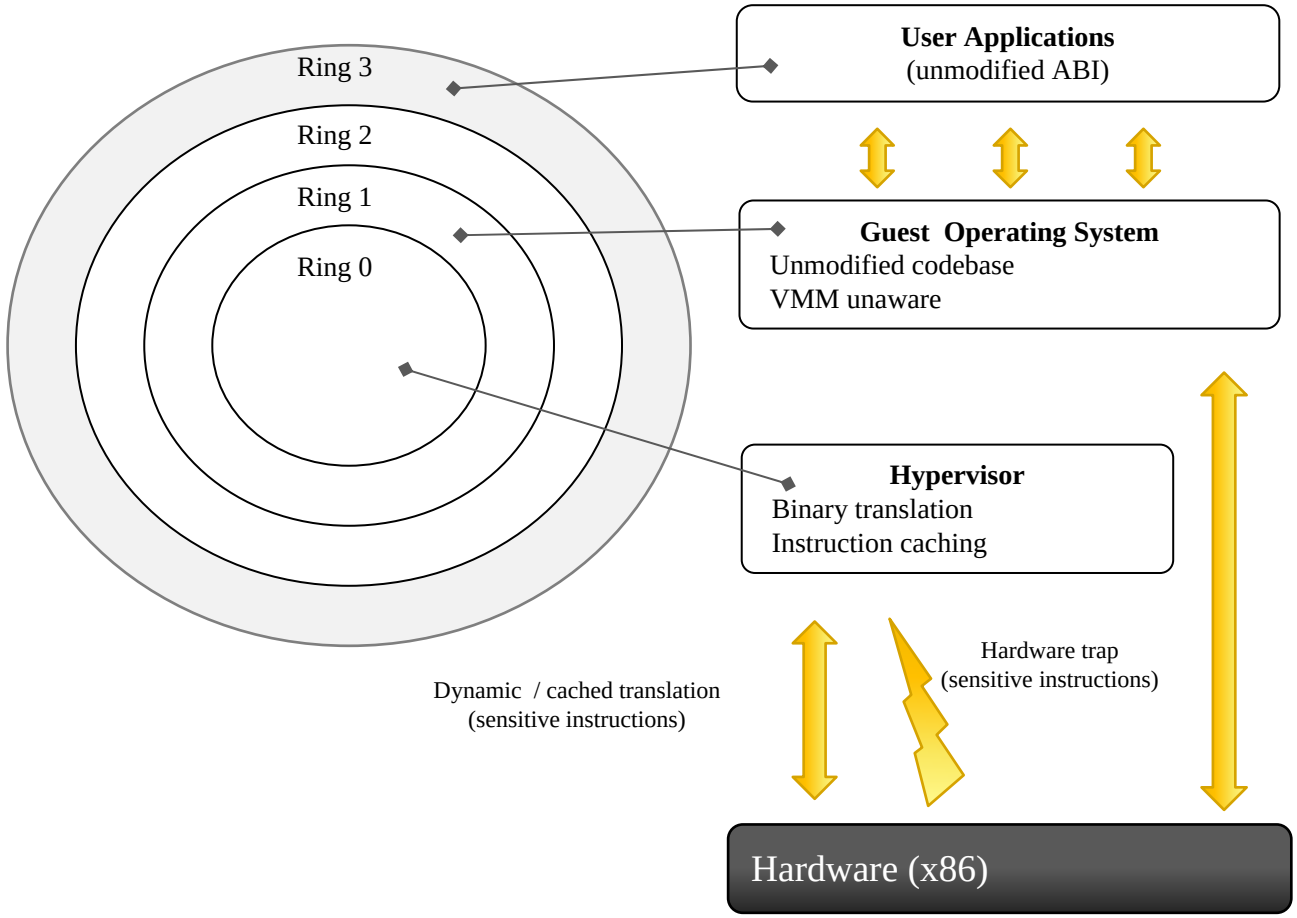
- Performance degradation
- Inefficiency and degraded user experience
- Security threats (e.g. phishing)

Xen: Paravirtualisation

- Open source
- Commercial XenSource by Citrix
- Xen Cloud Platform
- Moving towards hardware assisted virtualisation

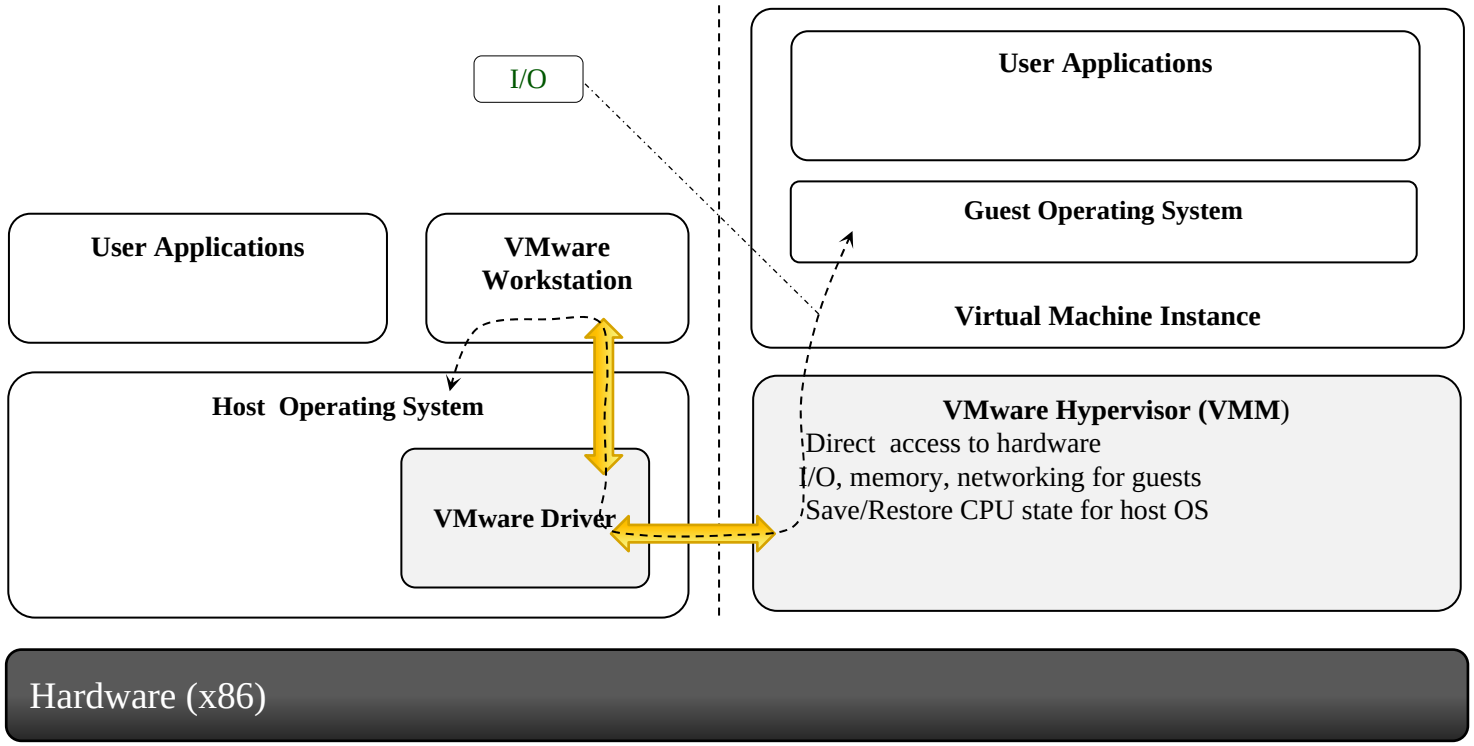


VMware: full virtualisation



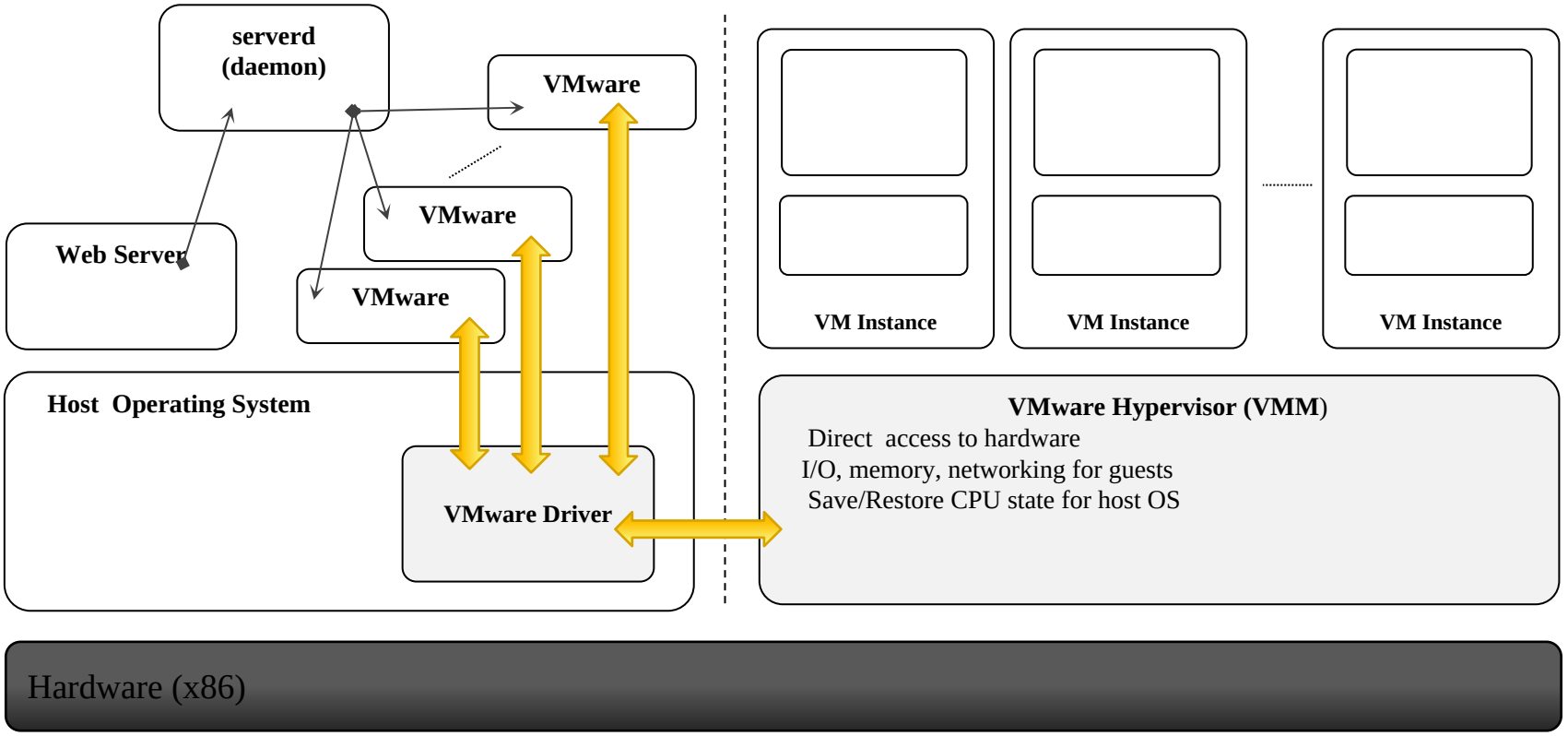
VMware: Desktop Environment

- Type II Hypervisor
- VMware Workstation (windows), VMware Fusion (Mac)



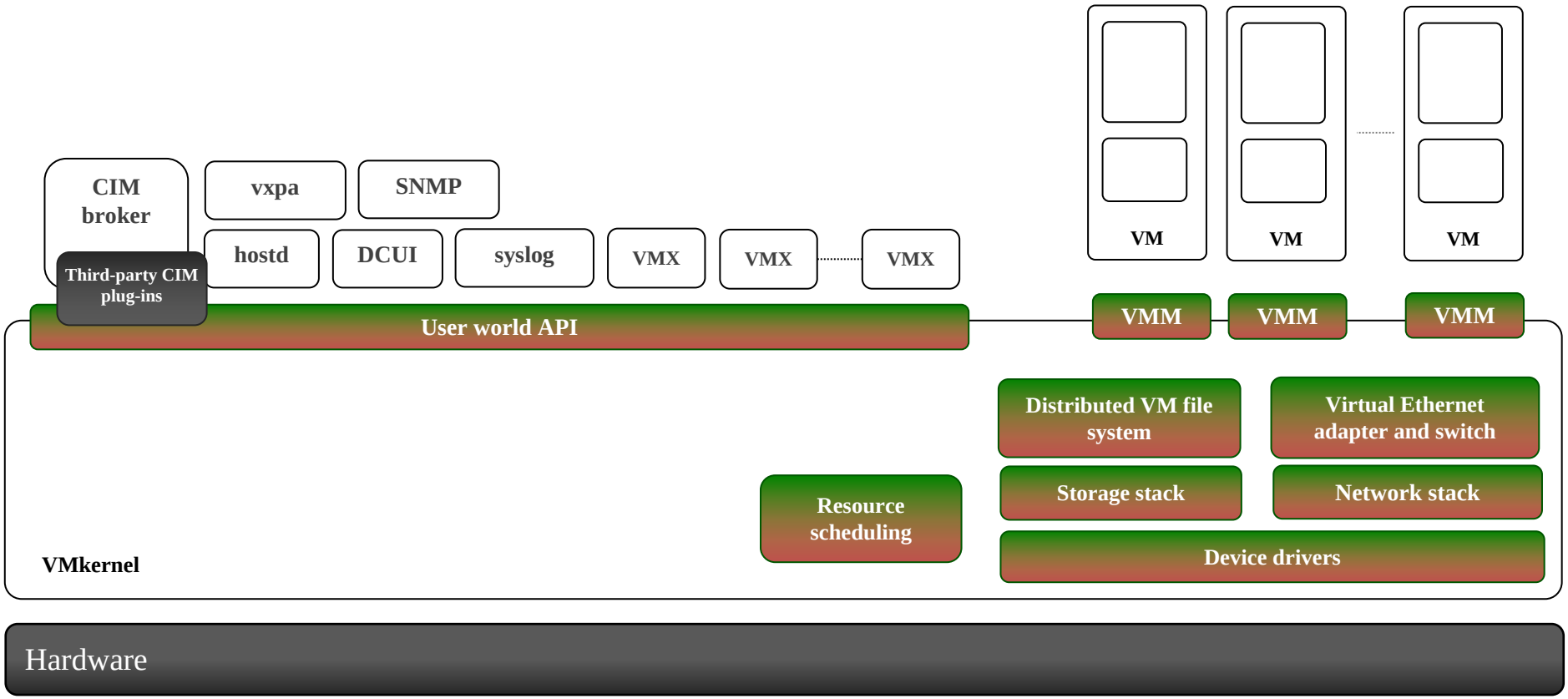
VMware: Server Environment

- Type I Hypervisor
- Vmware GSX Server

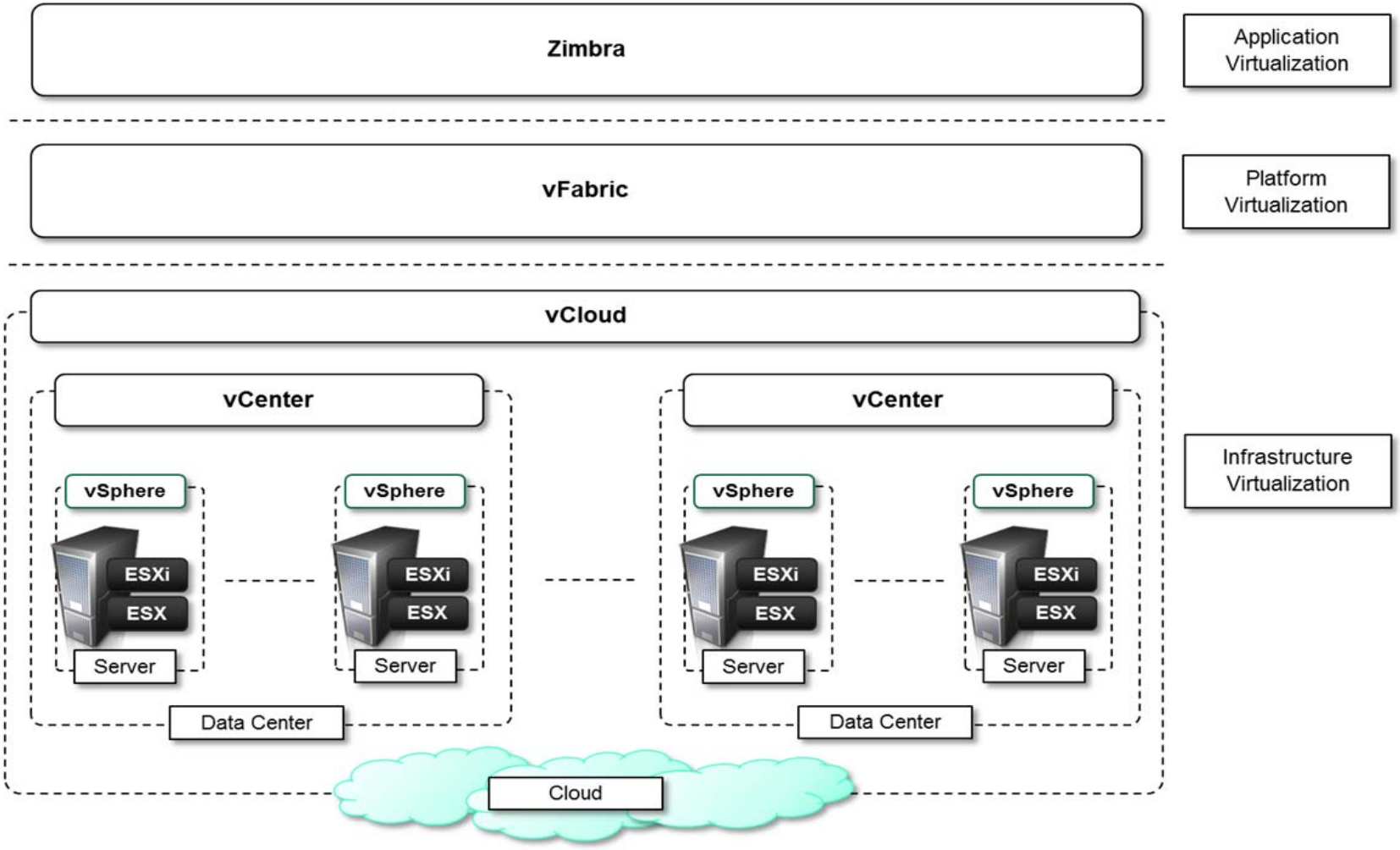


VMware: Server Environment

- Type I Hypervisor
- Vmware ESX Server



VMware: Cloud Solution stack



Microsoft Hyper V