Cloud Computing for Beginners

Infrastructure as a Service (IaaS)

By Idan Gabrieli

Cloud Computing

Virtualization Technologies

UNDER-UTILIZED PHYSICAL SERVERS

Physical Bare-metal Rack Server

Rack Server like a "Pizza Box"

CPU | Mem. | Storage | NICs

Application

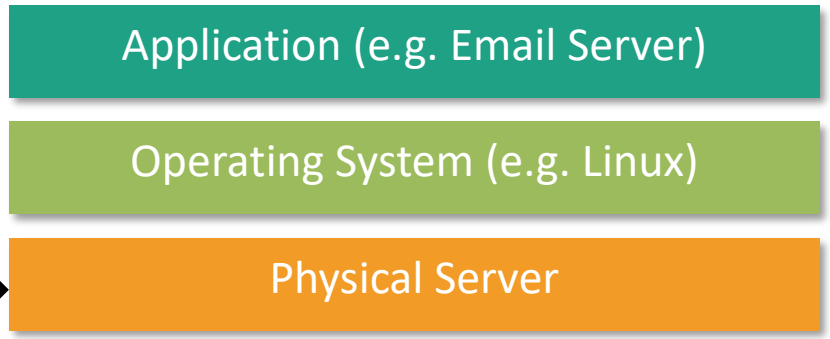
Operating System (e.g. Linux)

Physical Server

UNDER-UTILIZED PHYSICAL SERVERS

One App in One Server?

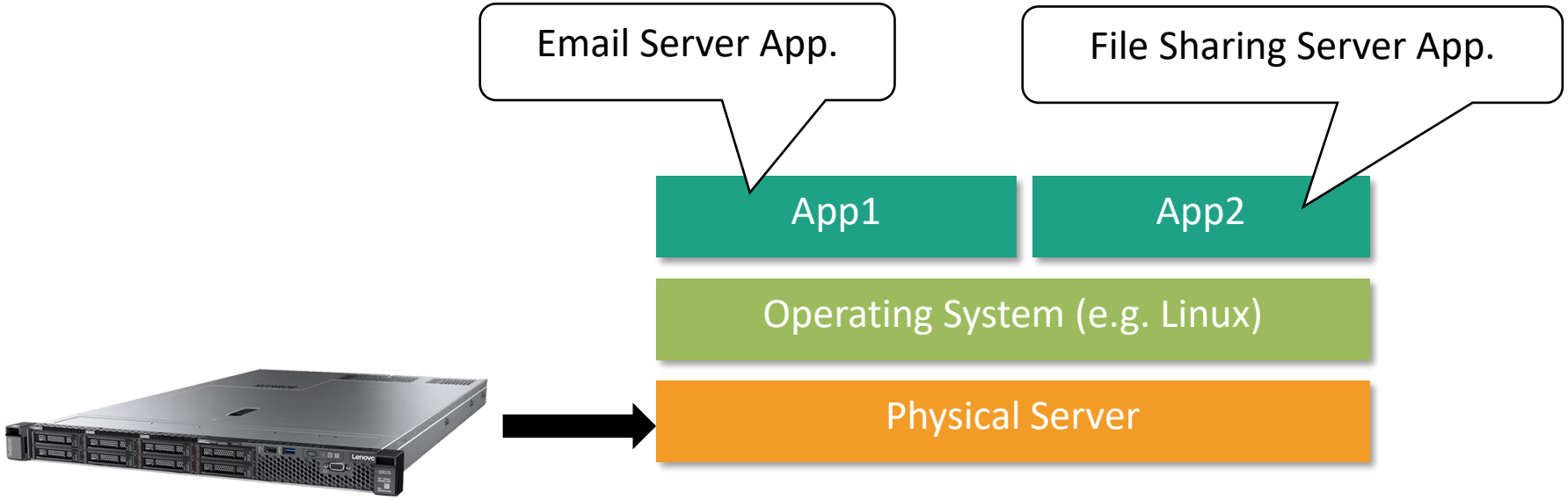
CPU – 15%, Memory – 10%



Under-utilized Server ☹

UNDER-UTILIZED PHYSICAL SERVERS

More than one apps in one server?



Mixing Server-side app is **not recommended**

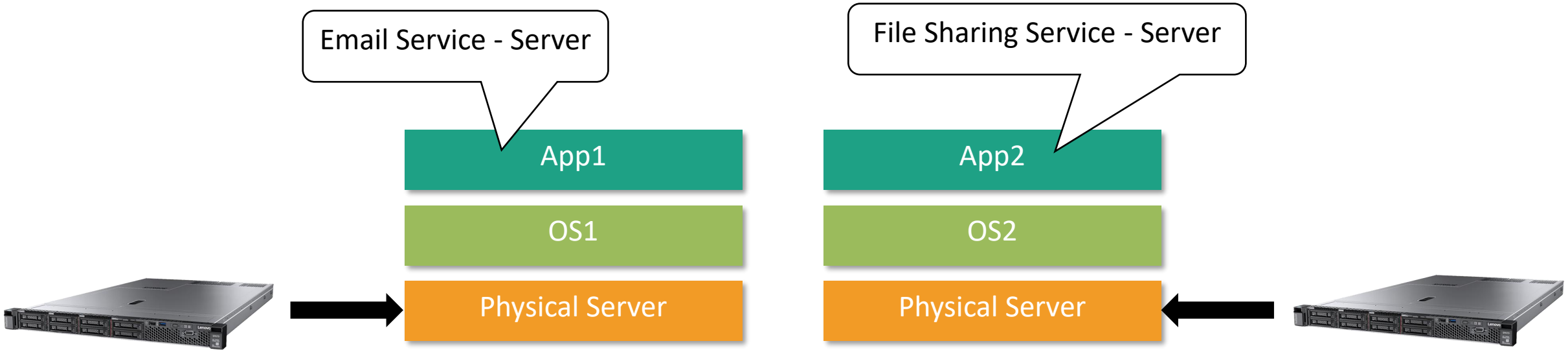
Server-side app is optimized to work with a **specific OS**

Server-side app will require specific **computing power** and **memory capacity**

Sharing maintenance **down-time**

UNDER-UTILIZED PHYSICAL SERVERS

Back to the same problem - Two Apps using Two Servers



UNDER-UTILIZED PHYSICAL SERVERS

Magnitude is Growing in a Data Center

10% CPU

Email

5% CPU

File Sharing

15% CPU

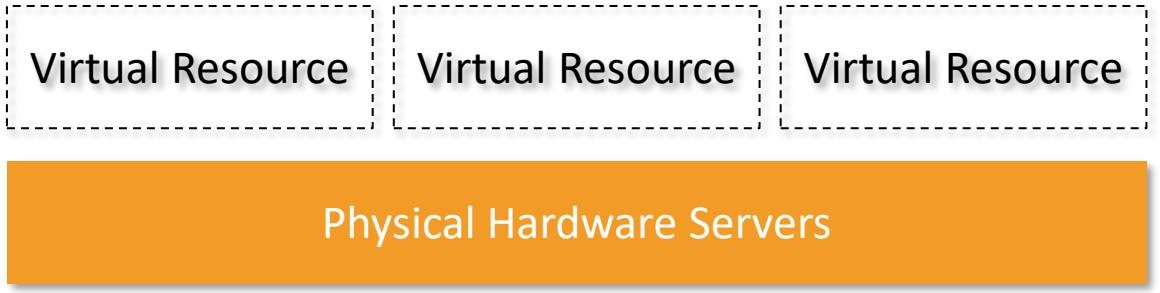
Web Server

A HUGE waste of IT resources

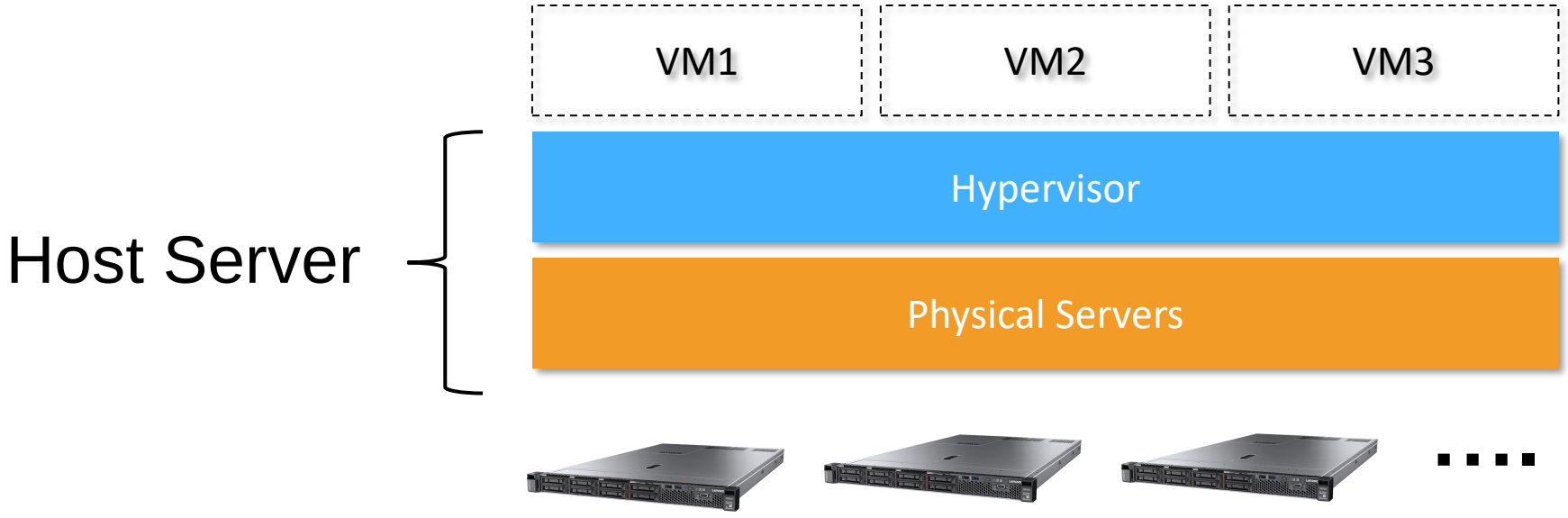
Virtualization Technologies

let’s make it virtual

VIRTUALIZATION WITH VIRTUAL MACHINES



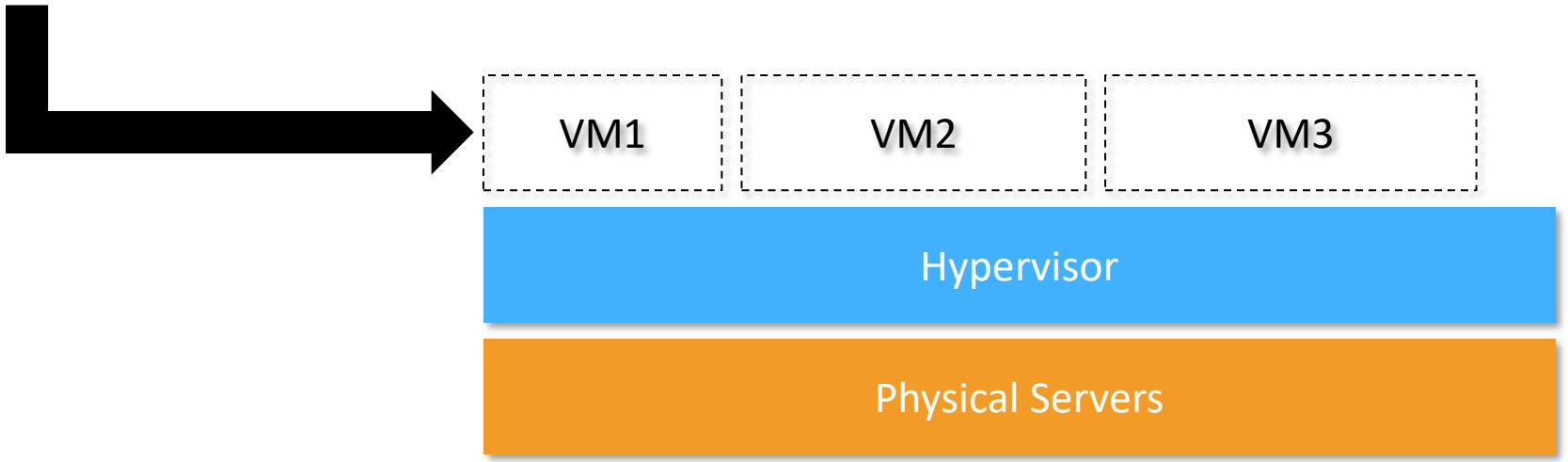
VIRTUALIZATION WITH VIRTUAL MACHINES



VIRTUALIZATION WITH VIRTUAL MACHINES

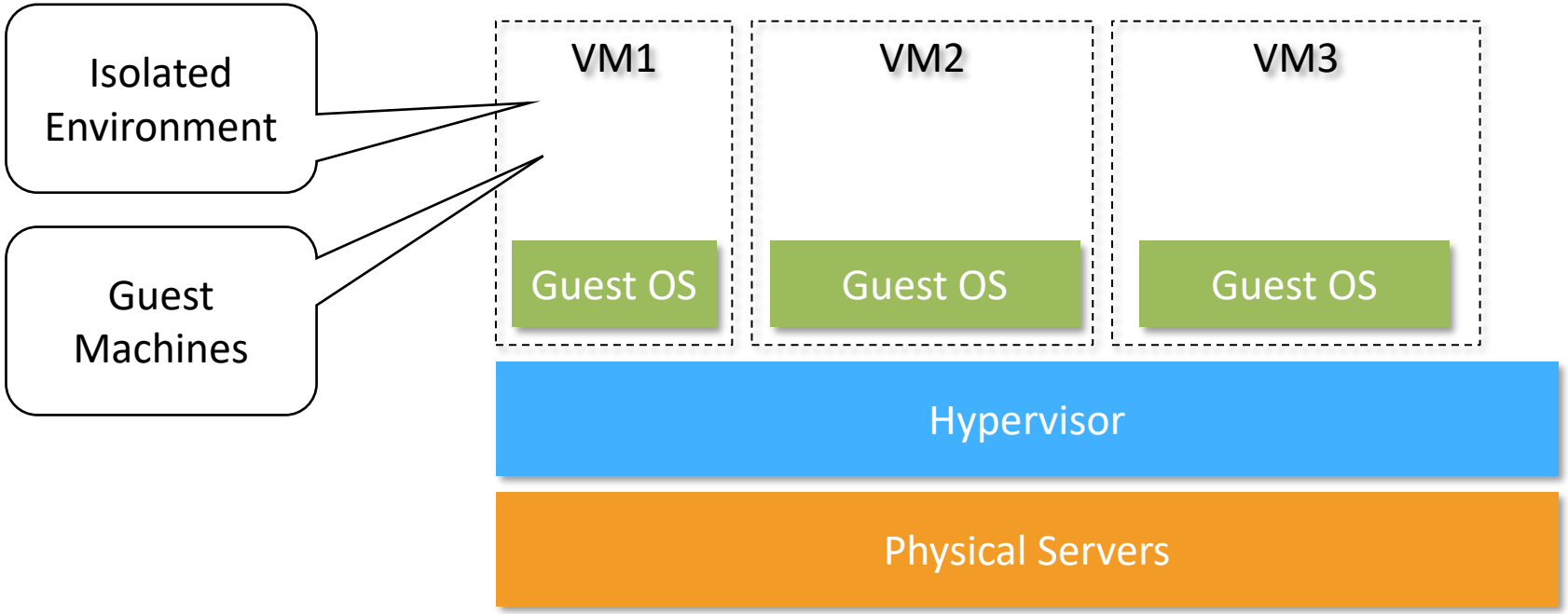
Dynamically allocate/deallocate VMs

VM Profile – CPU, Memory, Storage, Network Interfaces



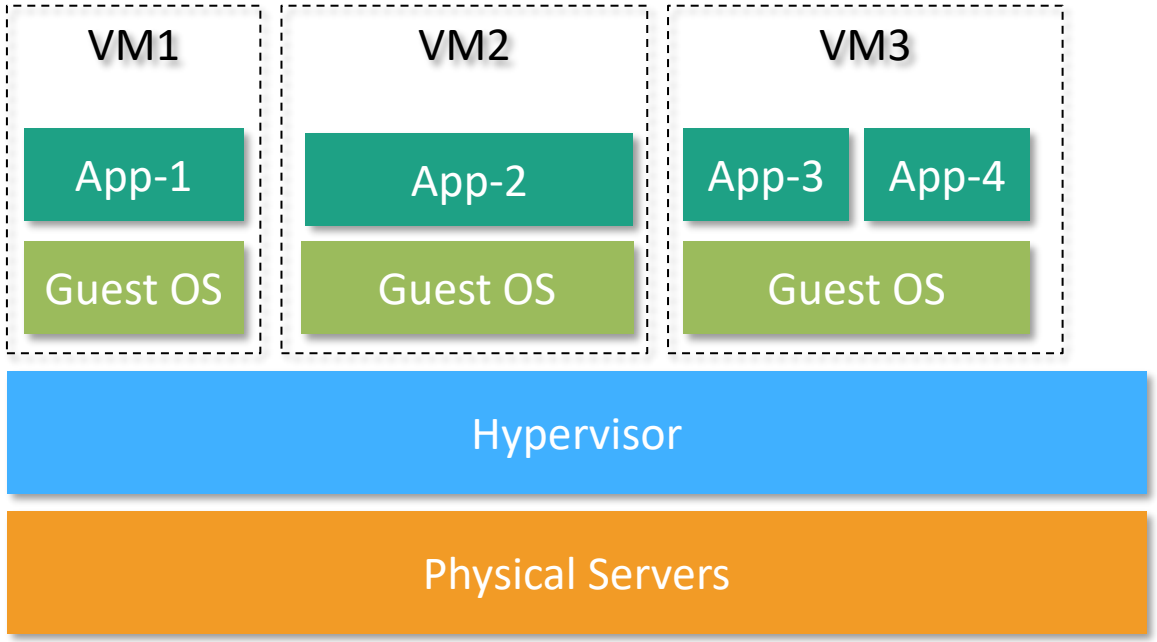
VIRTUALIZATION WITH VIRTUAL MACHINES

Each VM is an isolated environment



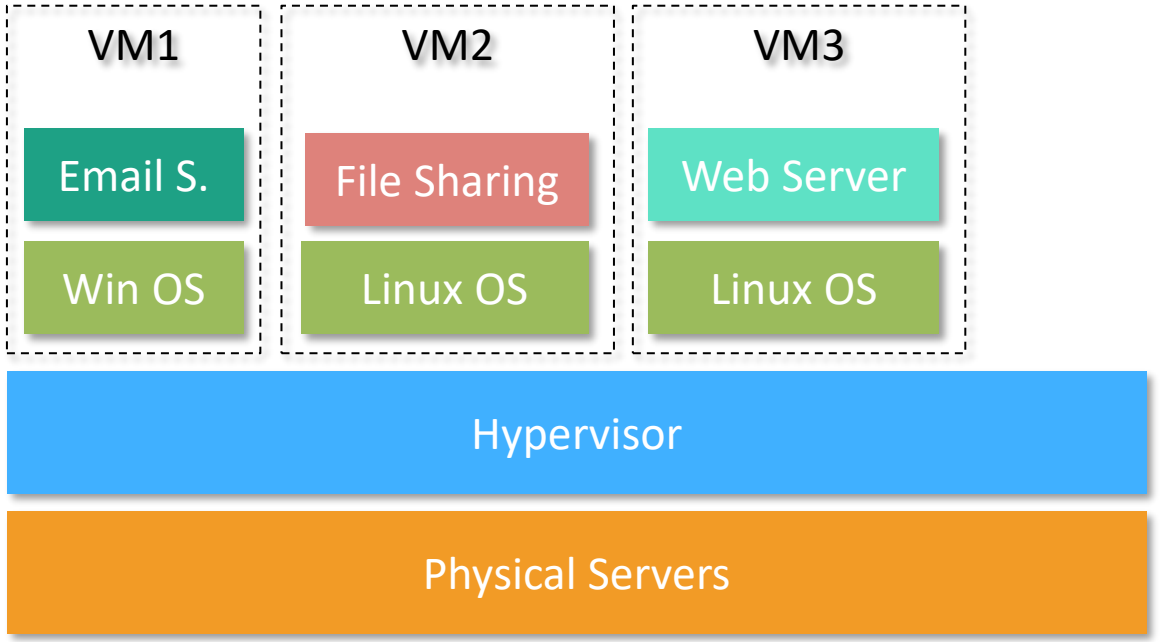
VIRTUALIZATION WITH VIRTUAL MACHINES

Apps in Each VM



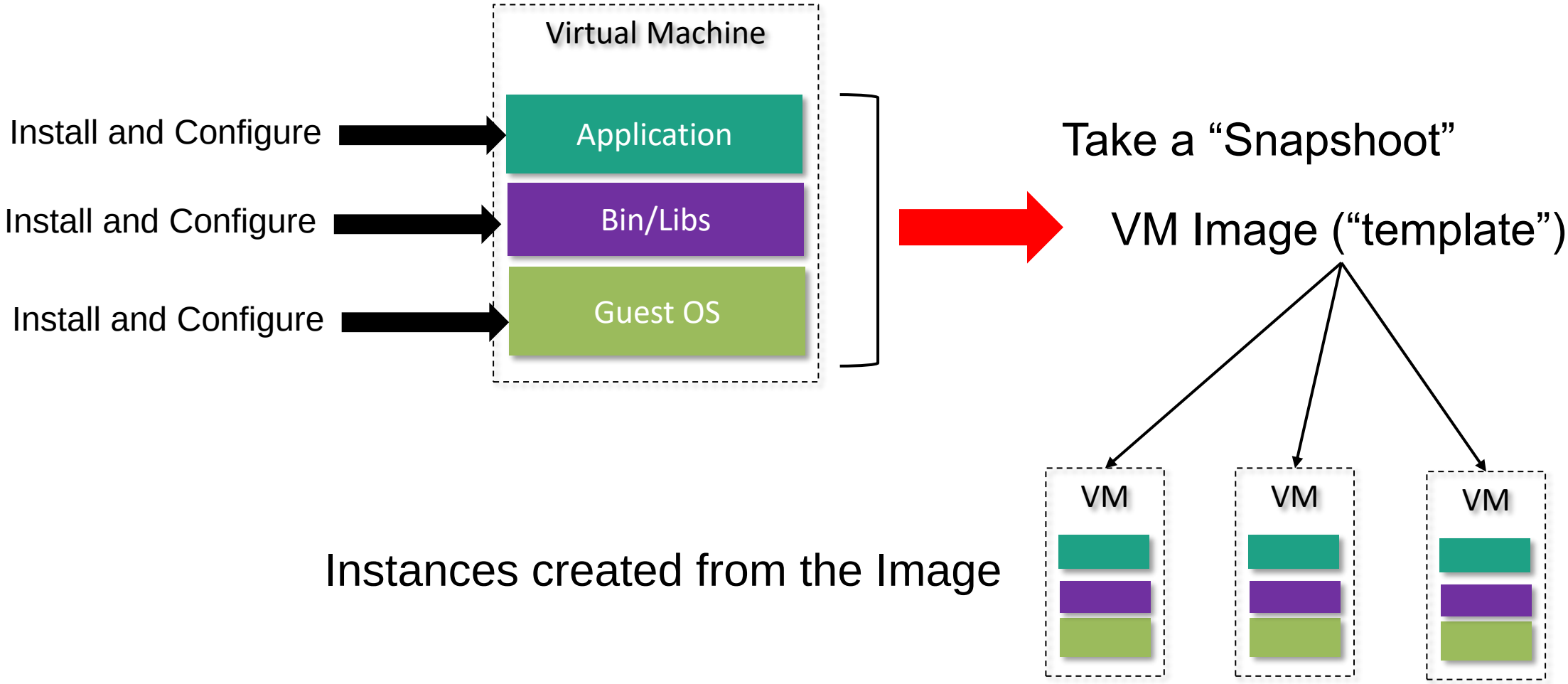
VIRTUALIZATION WITH VIRTUAL MACHINES

Back to Our Example



VIRTUALIZATION WITH VIRTUAL MACHINES

VM Image



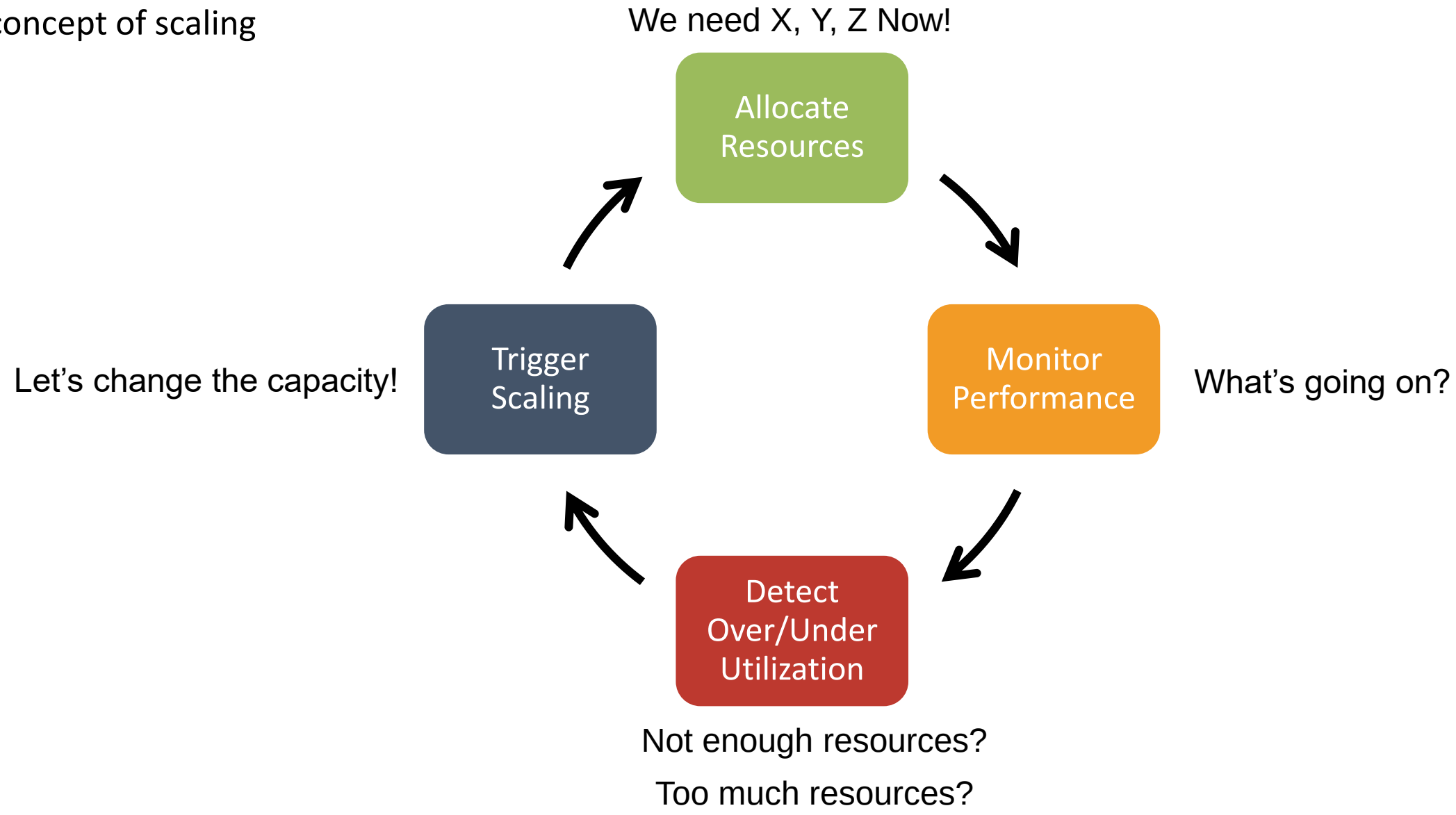
VERTICAL AND HORIZONTAL SCALING

The concept of scaling

Scaling is the process of managing our **cloud resources' capacity** to help our application meet a set of **performance requirements**

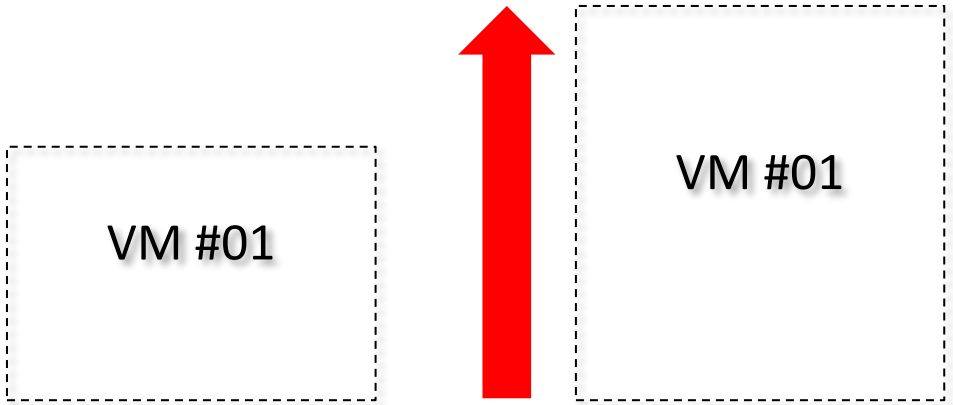
VERTICAL AND HORIZONTAL SCALING

The concept of scaling



VERTICAL AND HORIZONTAL SCALING

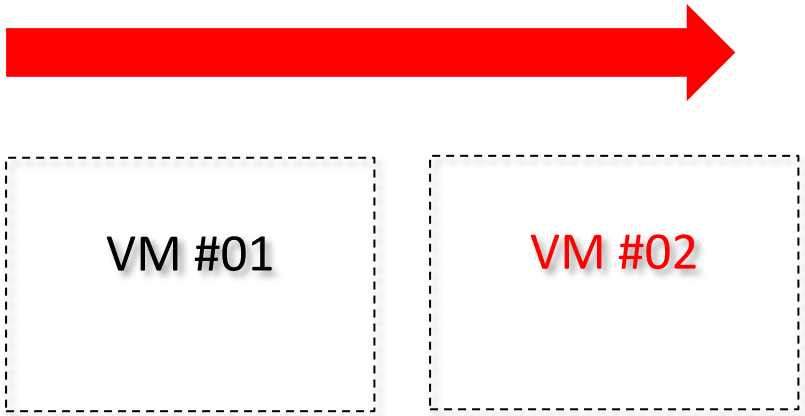
Vertical Scaling (up/down)



2 x vCPU
1 x GB RAM

4 x vCPU
2 x GB RAM

Horizontal Scaling (out/in)



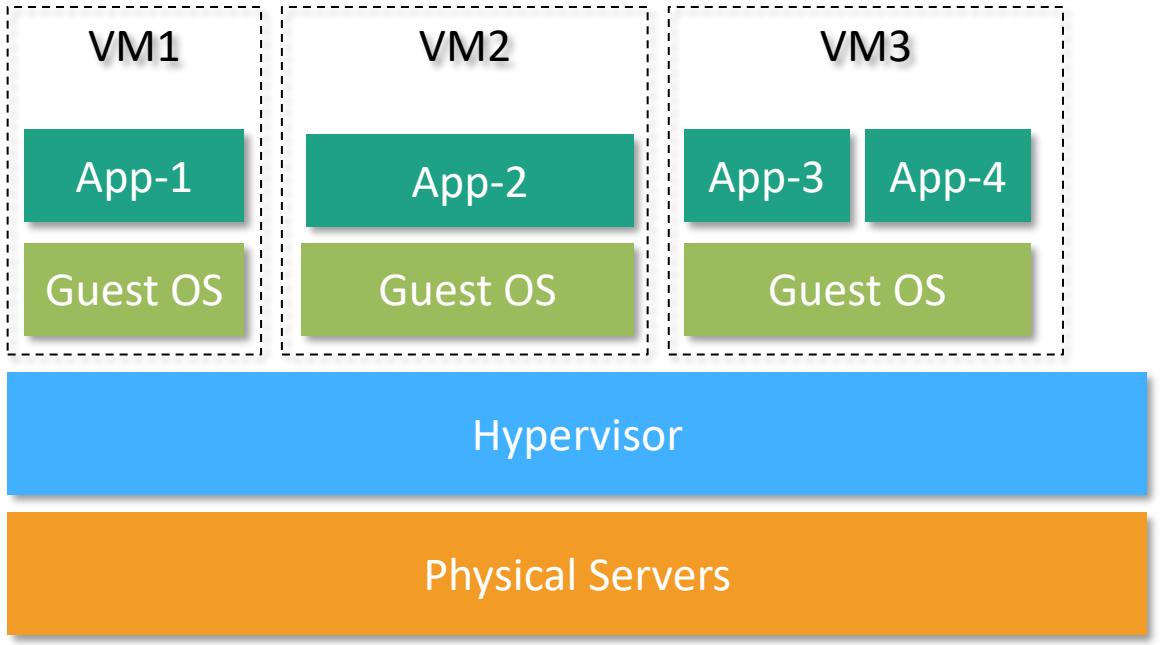
2 x vCPU
1 x GB RAM

2 x vCPU
1 x GB RAM

MICROSERVICES AND CLOUD-NATIVE APPS

No software change is needed

Legacy Applications



MICROSERVICES AND CLOUD-NATIVE APPS

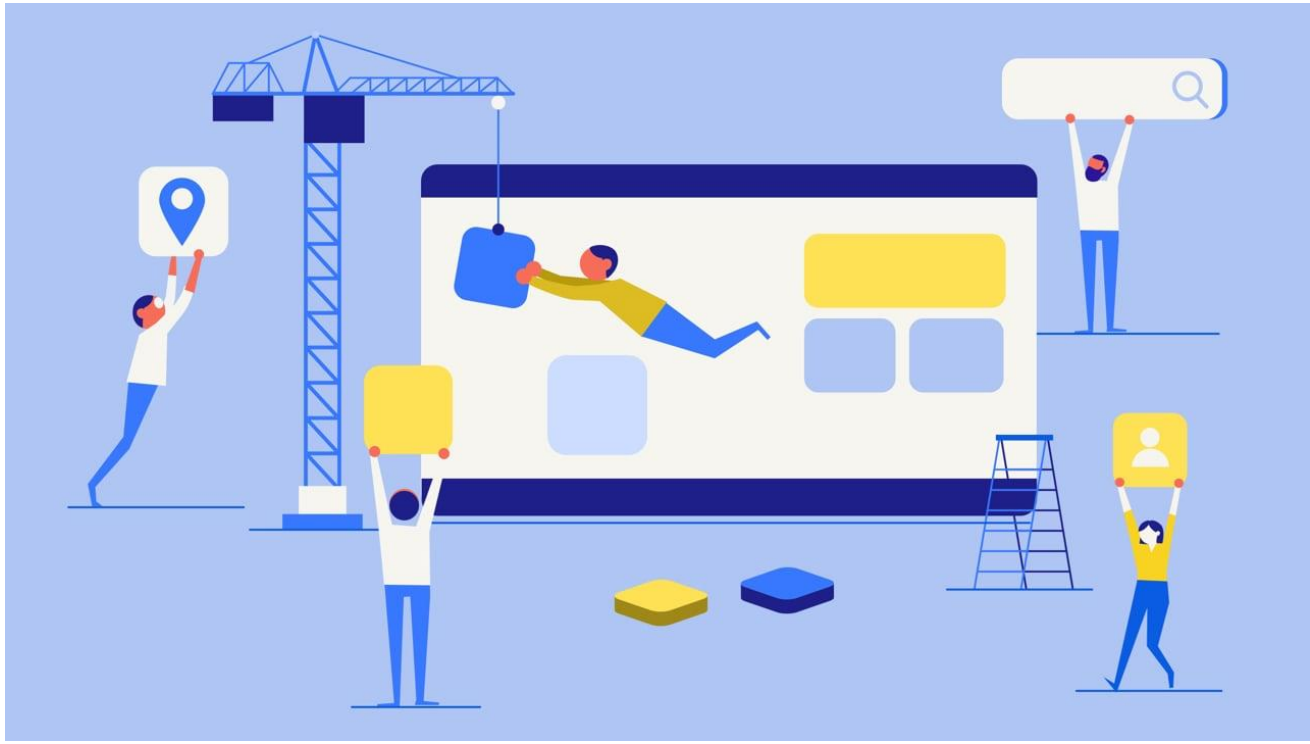
Re-design Applications

Cloud-Native Apps

↓

A new **software architecture**

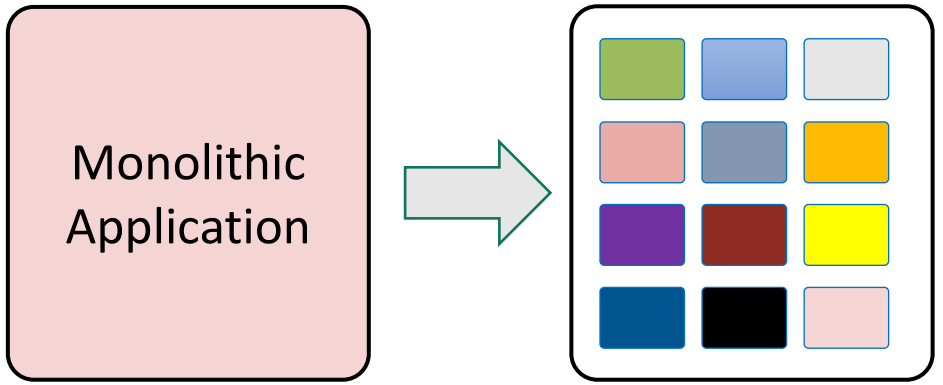
Microservices

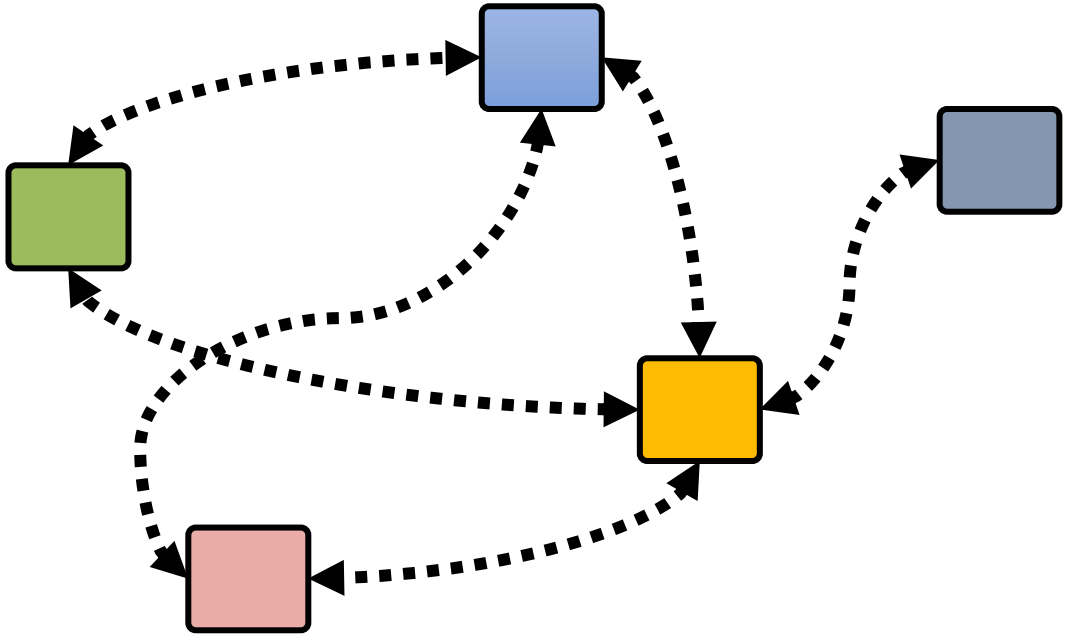


MICROSERVICES AND CLOUD-NATIVE APPS

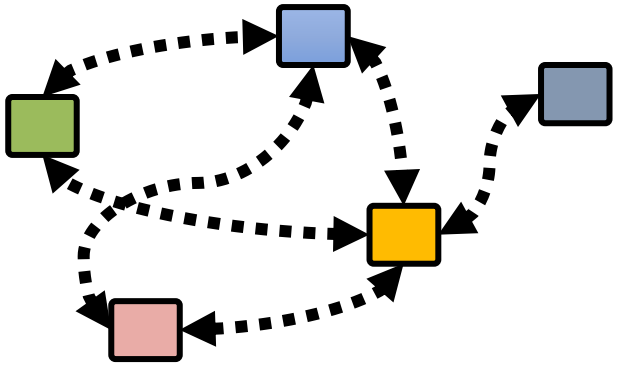
What are Micro-services?

- Software development architecture
  - Design, develop and maintain cloud-native applications
  - Small pieces called micro-services
- A micro-service
  - Single function module
  - Do one thing and do it well!
  - Each micro-service is like a mini-application
  - Developed independently and deployed independently





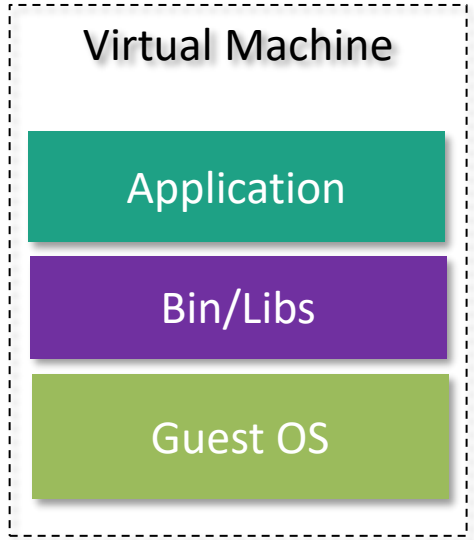
MICROSERVICES AND CLOUD-NATIVE APPS



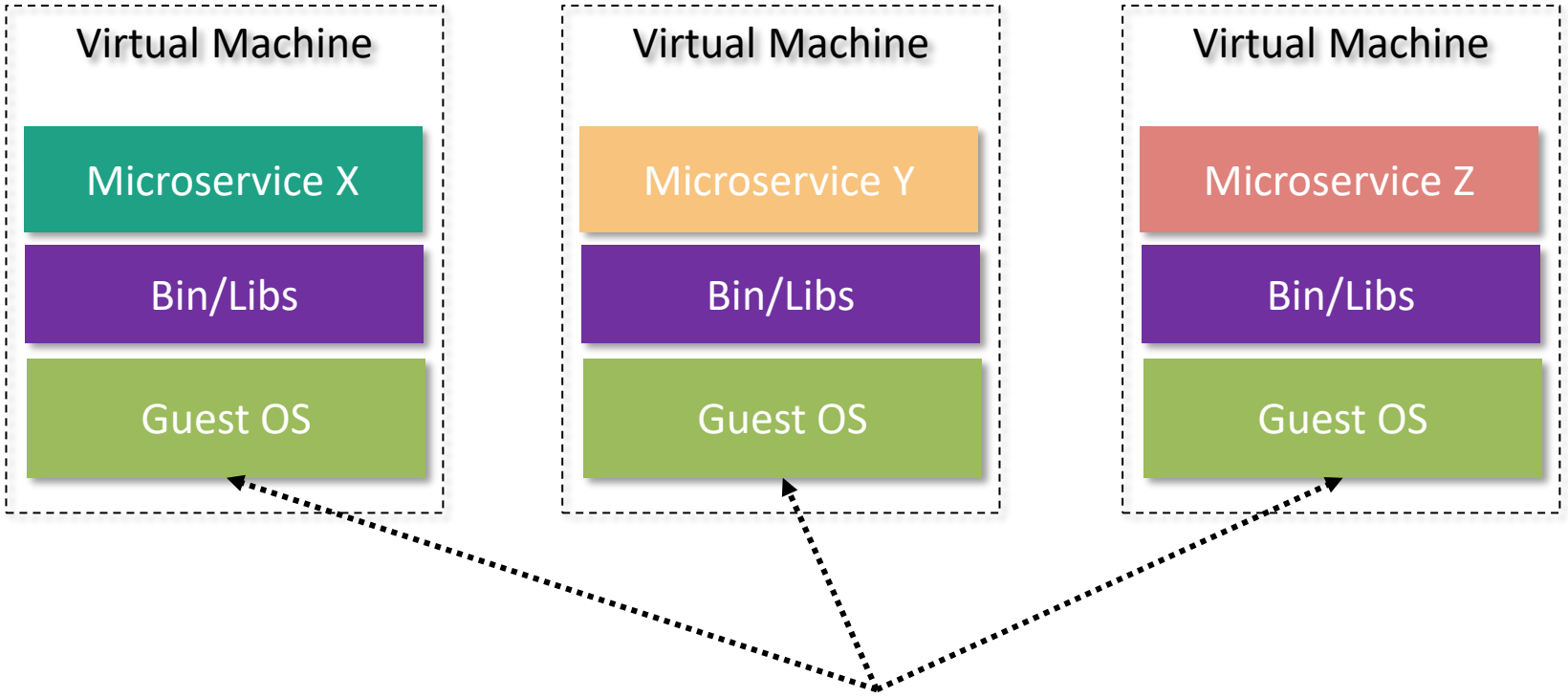
50 x Micro-Services

?

50 x VMs (with 50 x OS…)

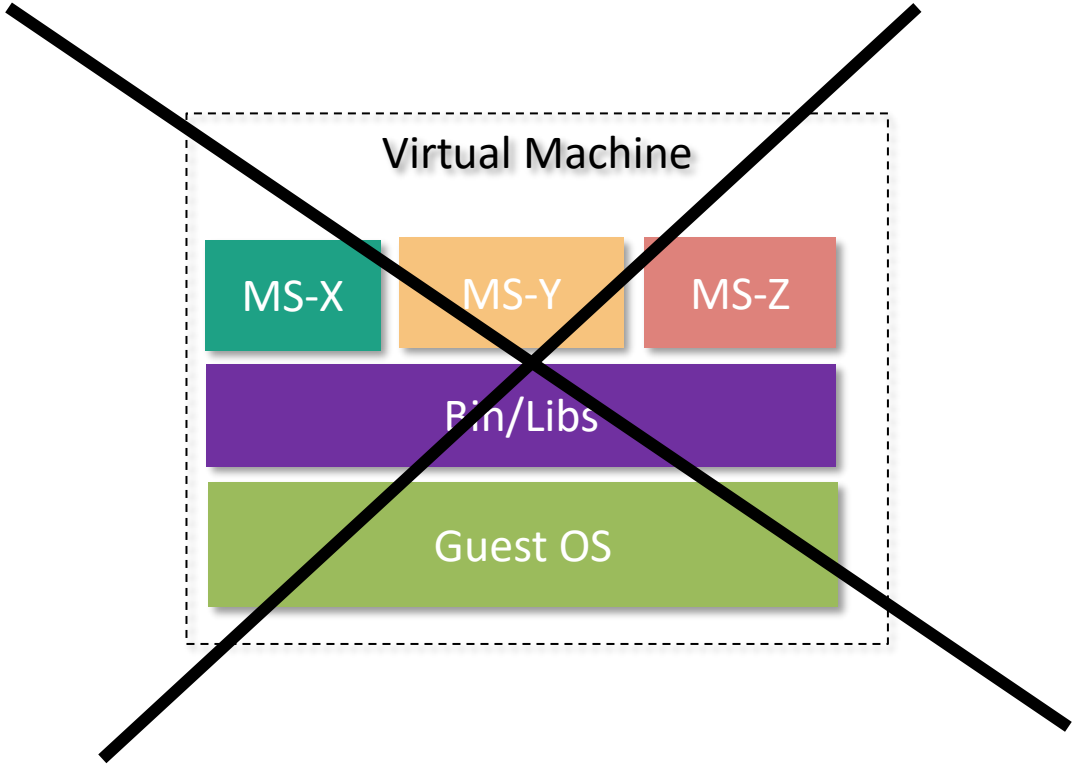


TRADITIONAL VIRTUALIZATION WITH VMs

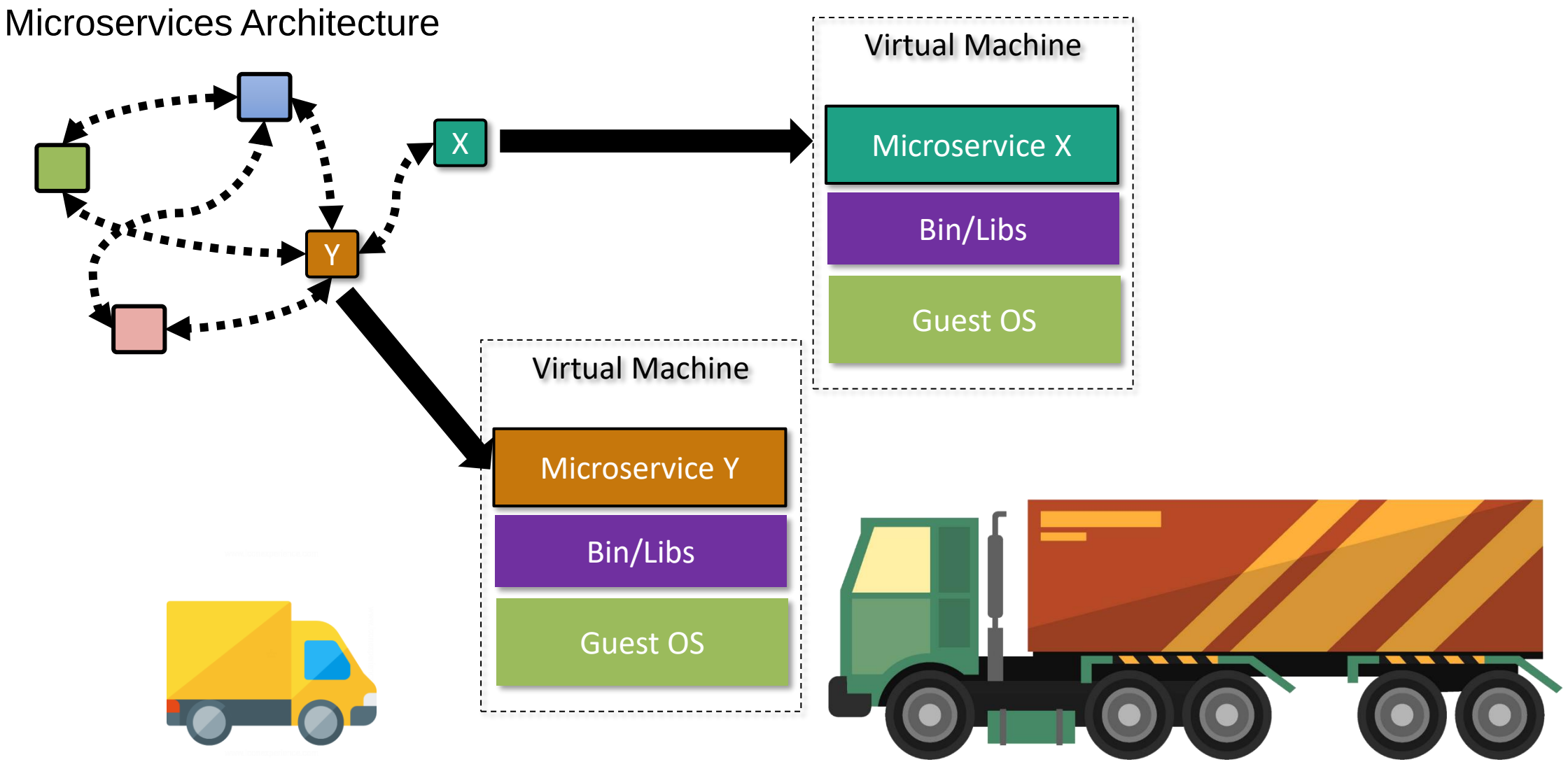


A HUGE **OS Overhead :-(**

TRADITIONAL VIRTUALIZATION WITH VMs

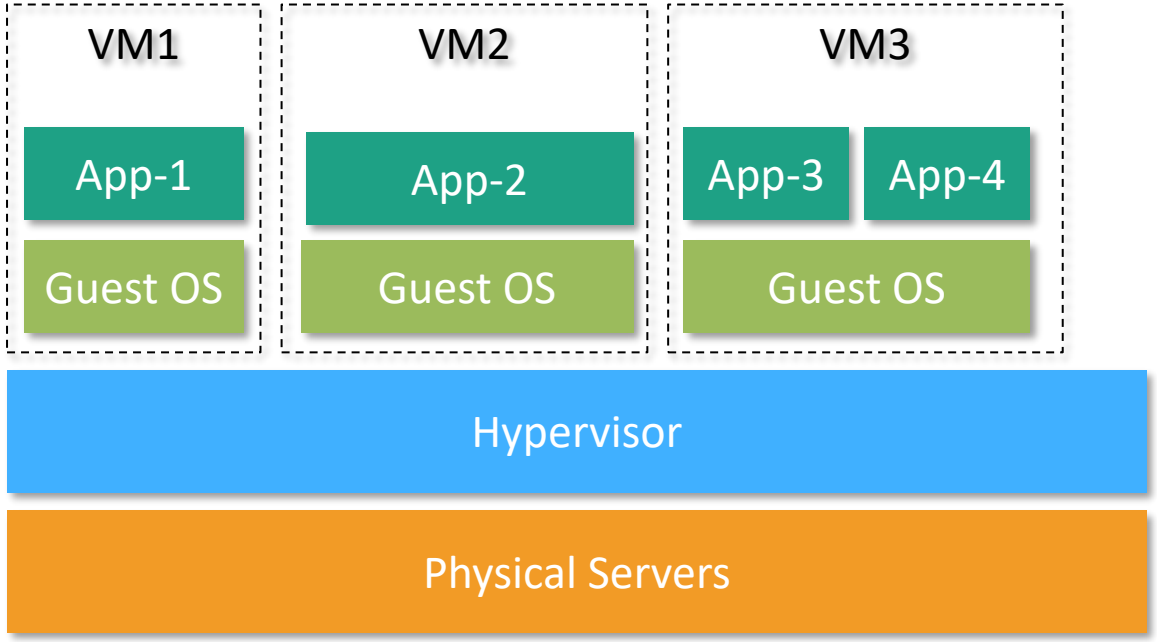


Microservices in VMs ?



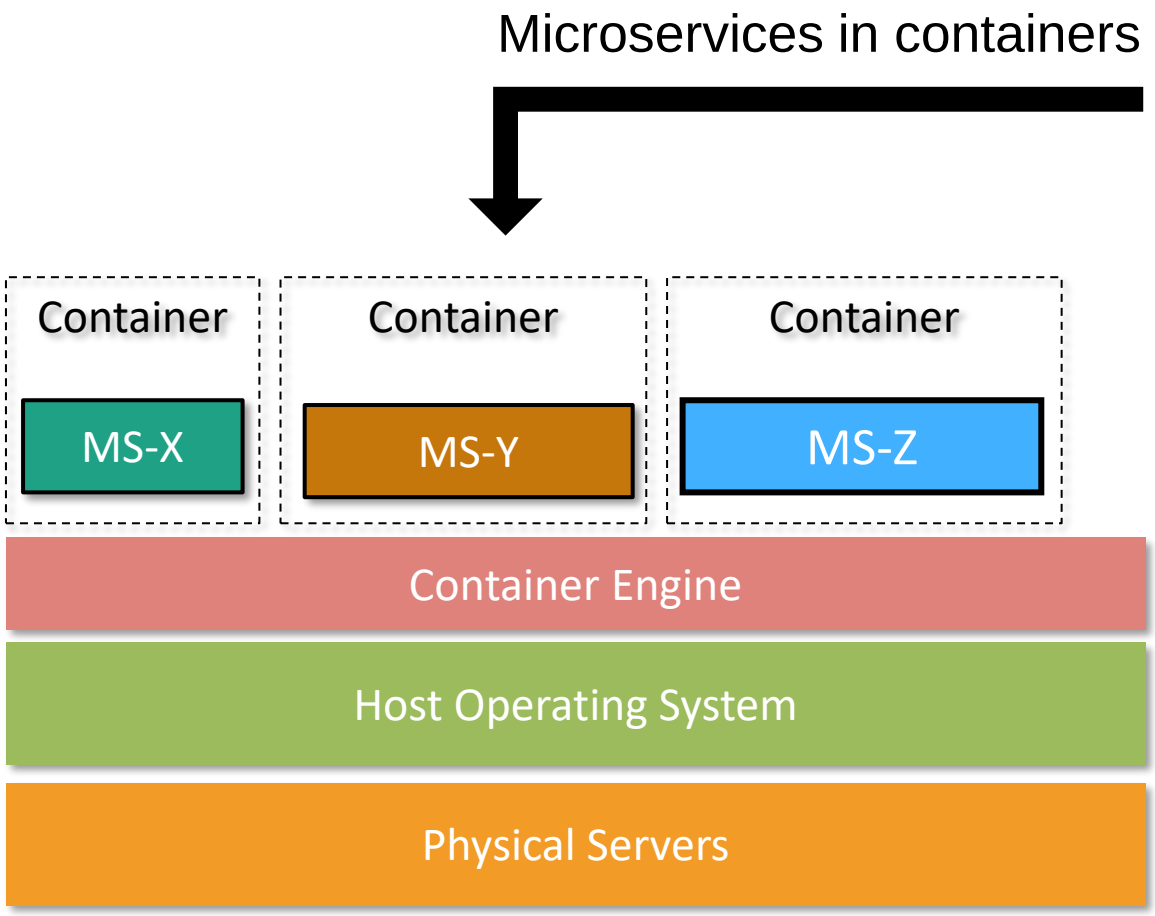
VIRTUALIZATION WITH CONTAINERS

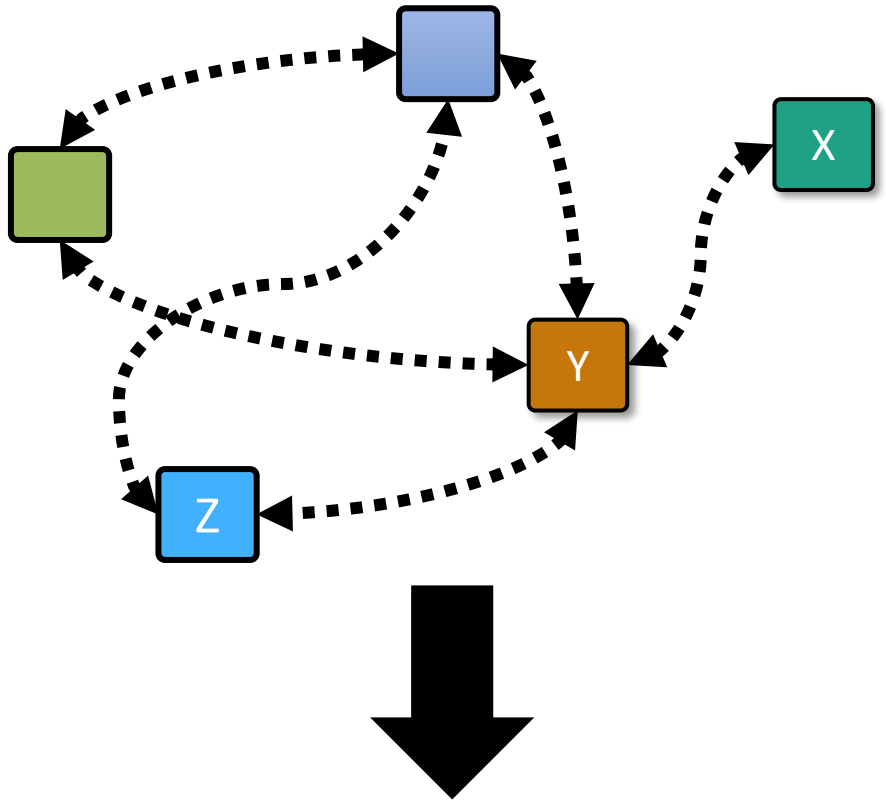
VIRTUALIZATION WITH CONTAINERS



VIRTUALIZATION WITH CONTAINERS

VIRTUALIZATION WITH CONTAINERS

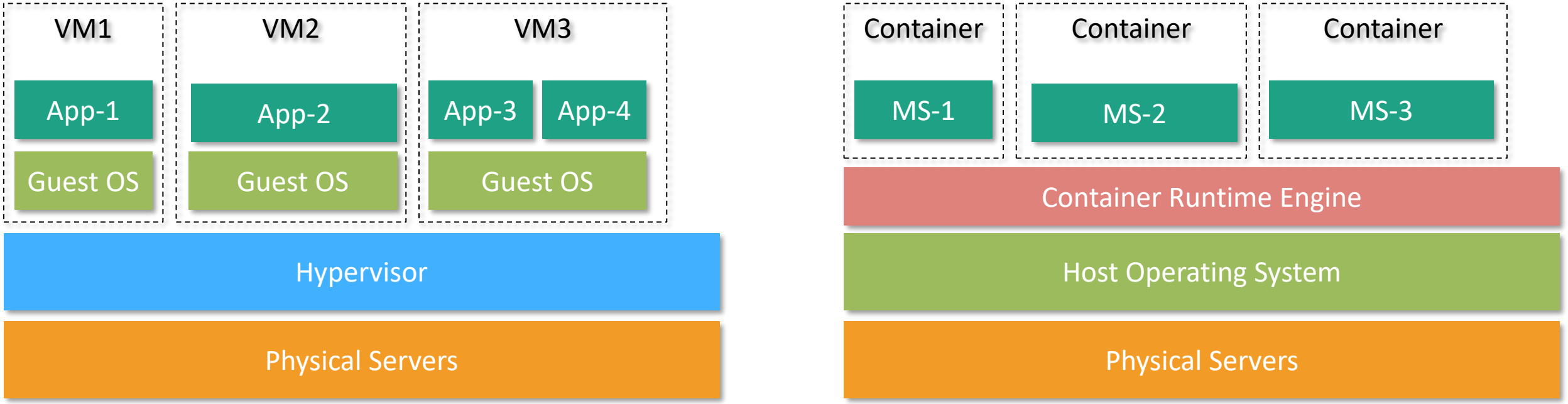




Containerized Application

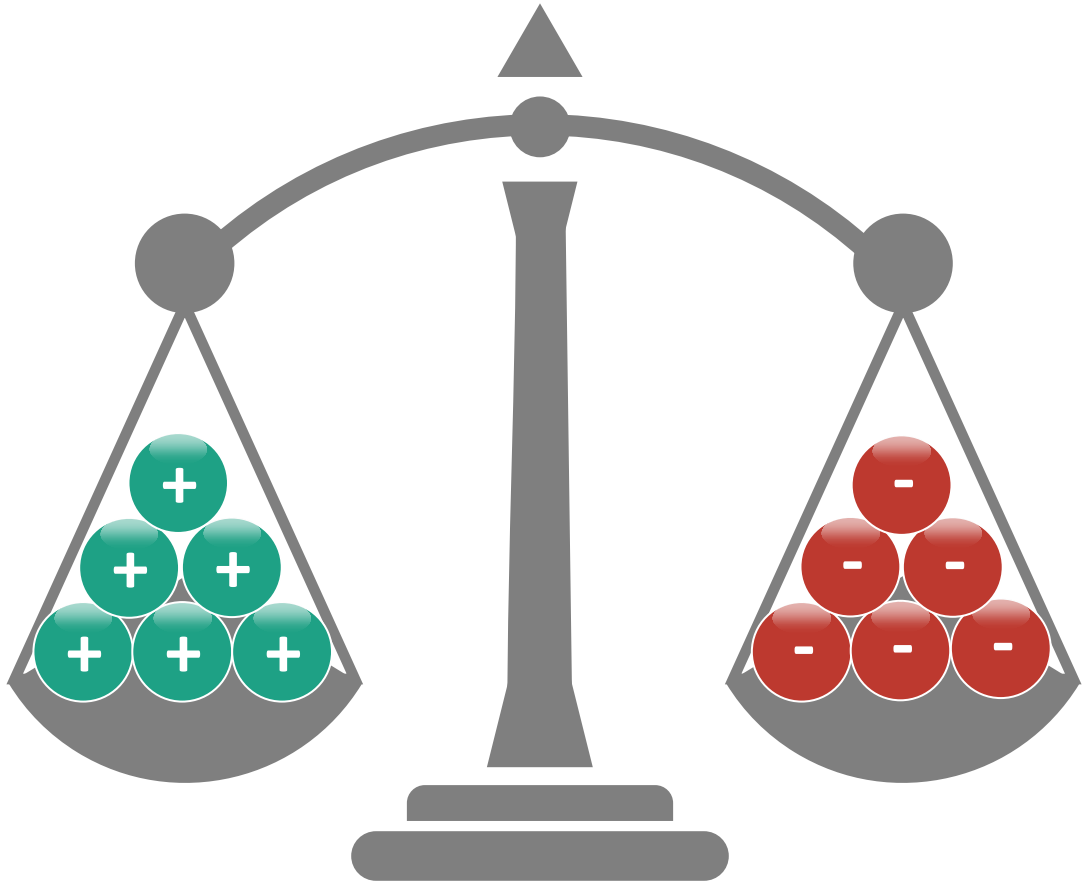
VIRTUALIZATION WITH CONTAINERS

VMs vs. Containers

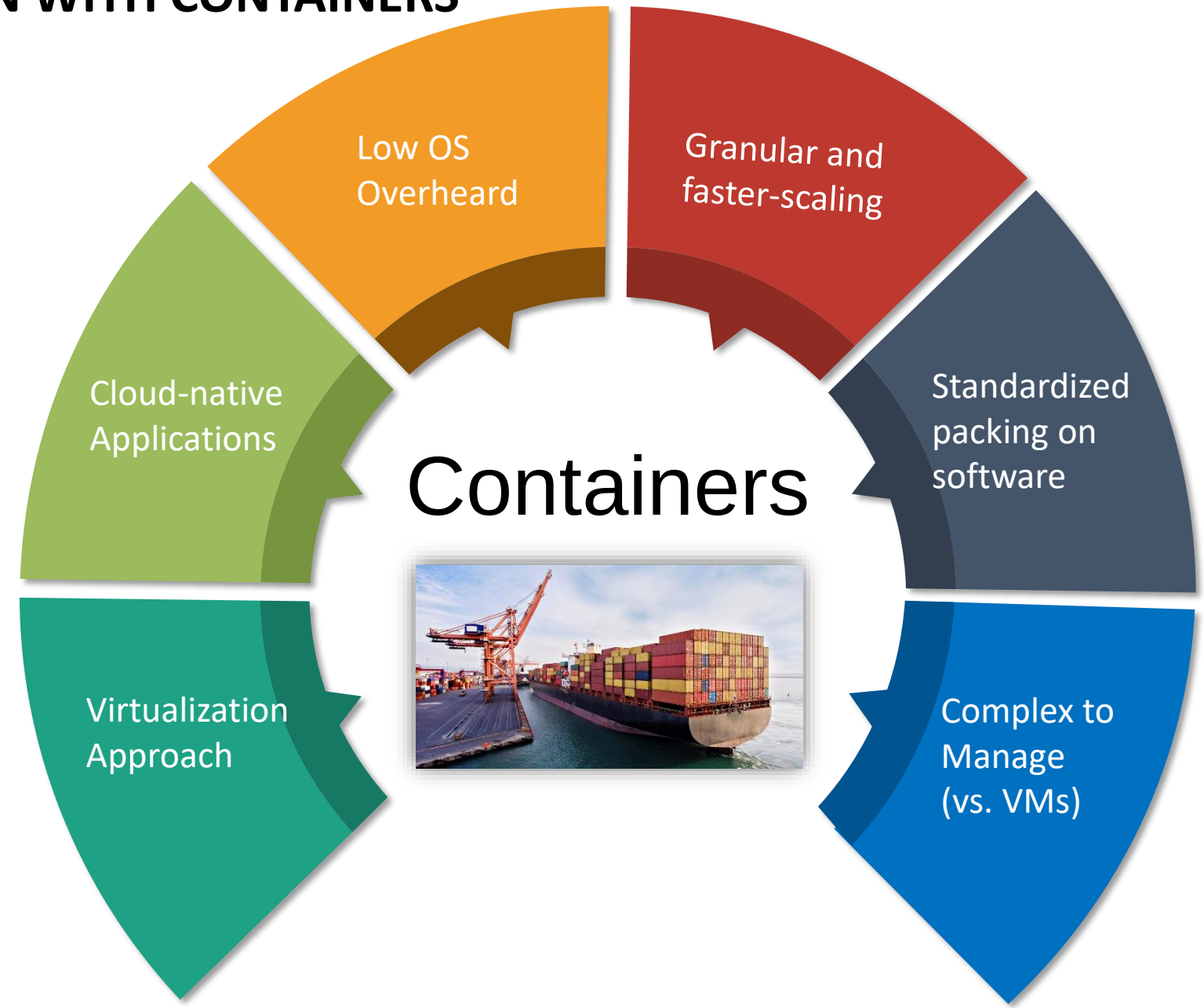


THE BENEFITS OF CONTAINERS

- Rapid Scalability
- Deployment in seconds….
- Deployed in multiple environments
- Streamline Faster Software Releases



VIRTUALIZATION WITH CONTAINERS



VIRTUALIZATION WITH CONTAINERS