

# Virtualization and Containerization

# Virtualization

Virtualization and Containerization

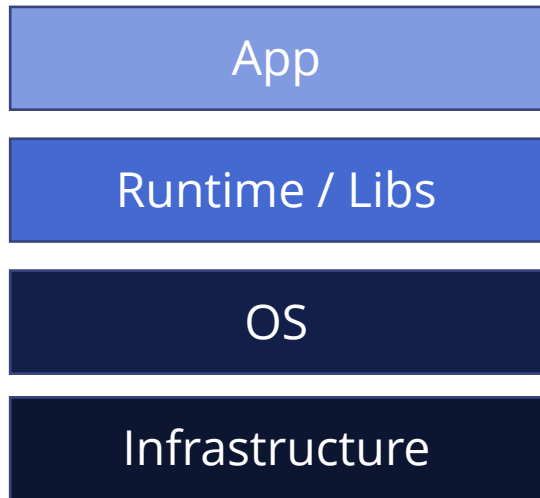
softserve

# What is Virtualization?

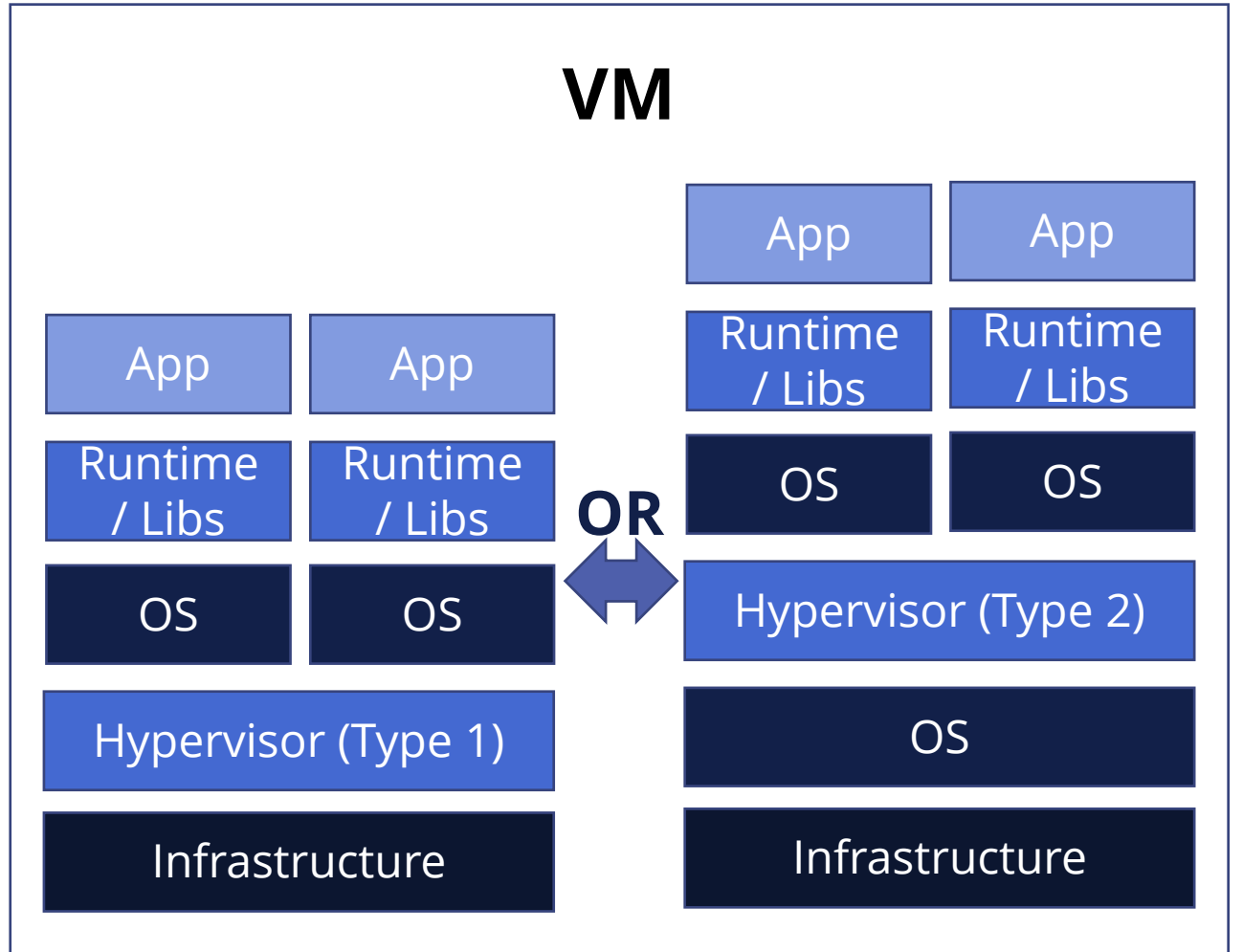
- **Virtualization** is using software to create an abstraction layer over the computer hardware that allows the hardware elements of a single computer to be divided into multiple virtual computers.
- A **virtual machine (VM)** is one of such virtual computers.
- Each VM behaves as an **independent computer**, running its own operative system (OS).
- Virtualization aims at a more **efficient** utilization of the physical computing resources of the actual computer.

# The Road to Virtualization

## Bare Metal

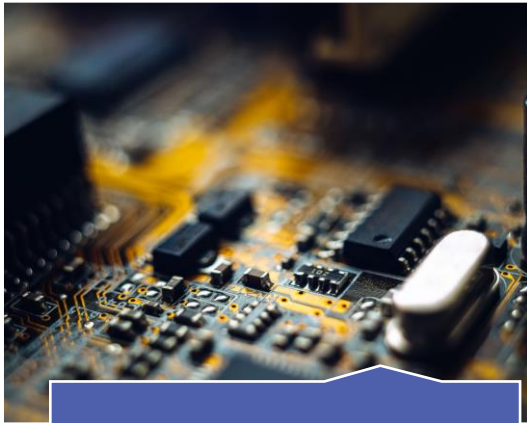


## VM



softserve

# Pros of Virtualization (Related to Bare Metal Servers)



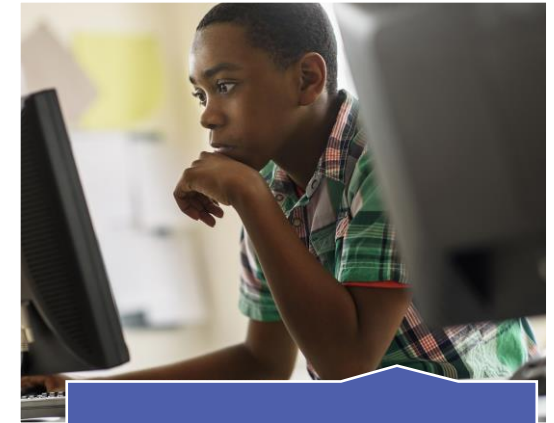
Resource efficiency



Easier  
management



Reduced downtime



Faster provisioning

# Types of Virtualization

Desktop  
virtualization

Network  
virtualization

Storage  
virtualization

Application  
virtualization

Data center  
virtualization

CPU  
virtualization

GPU  
virtualization

# Virtualization Tools

## Type 1 Hypervisors

VMware vSphere / ESXi

Microsoft Hyper-V

Xen / Citrix XenServer

Red Hat Enterprise  
Virtualization (RHEV)

KVM

## Type 2 Hypervisors

Oracle VirtualBox

VMware Workstation

VMware Fusion

Oracle VM Server for x86

CentOS Virtualization

# Containerization



# What is Containerization?

- **Containerization** is the packaging of software code with just the OS libraries and dependencies required to run the code, creating a single lightweight executable that runs consistently on any infrastructure.
- Such executable is called the **container**.
- Containers have become the **de facto compute units** of modern cloud-native applications.
- The container is **abstracted away from the host OS**, and hence, it stands alone and becomes portable.

# Deployment Architecture Evolution

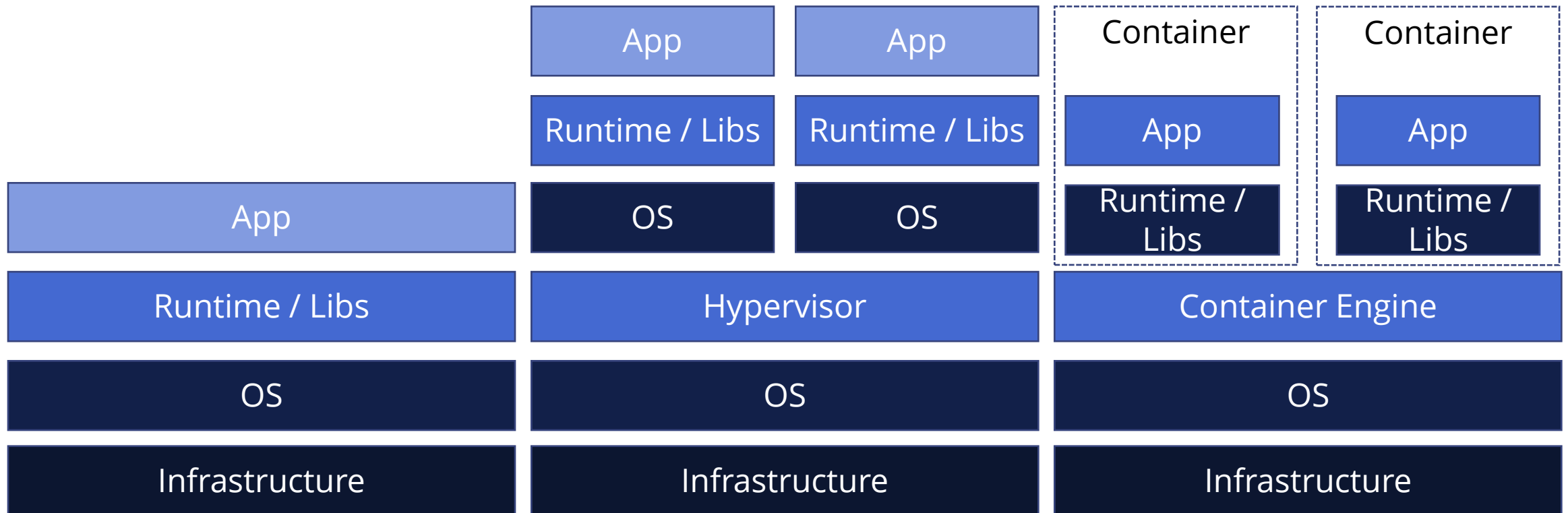
**Bare Metal**



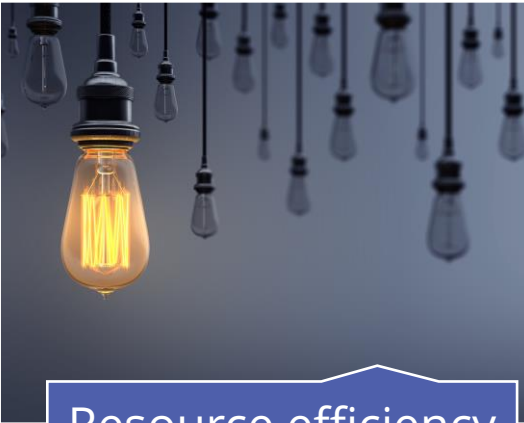
**VM**



**Containers**



# Pros of Containerization (Related to Virtual Machines)



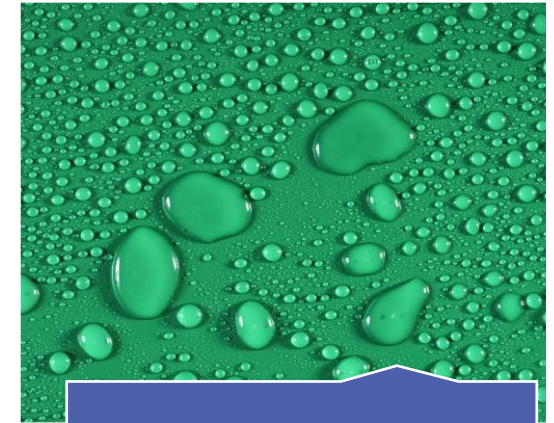
Resource efficiency  
(i.e., licensing)



Faster startup



Fault isolation



Scalability

# Containerization Tools

Docker: <https://www.docker.com/>

containerd: <https://containerd.io/>

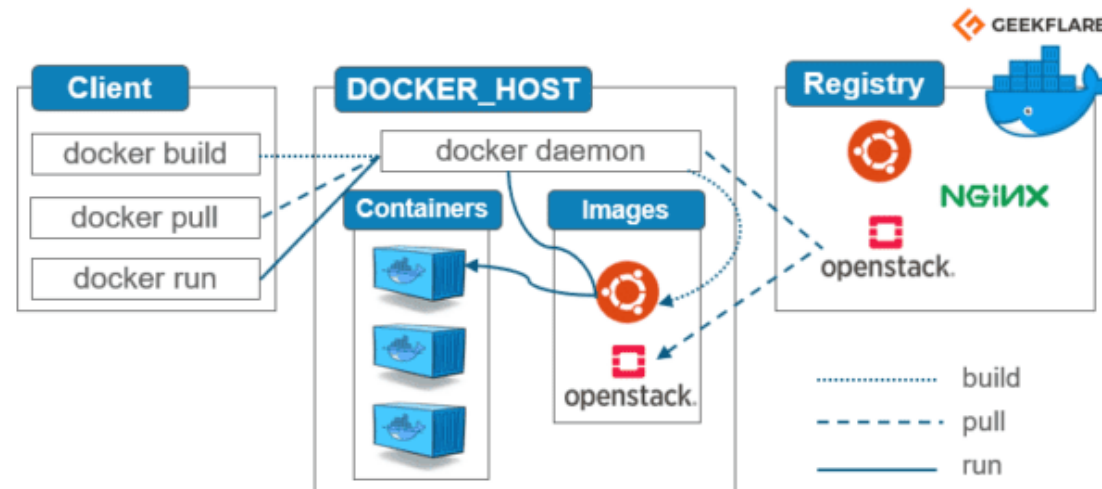
LXC: <https://linuxcontainers.org/>

CRI-O: <https://cri-o.io/>

Mirantis: <https://www.mirantis.com/>

# Docker

- **Docker** is an open-source platform for developing, shipping, and running applications.
- It is the industry **de facto standard** regarding containerization.
- It enables developers to build, deploy, run, update and manage **containers**.



## References and Useful Resources

- What is Virtualization?: <https://www.ibm.com/topics/virtualization>
- Containerization Explained:  
<https://www.ibm.com/topics/virtualization>
- Docker Overview: <https://docs.docker.com/get-started/overview/>
- Docker with NodeJS in 5 minutes:  
<https://www.youtube.com/watch?v=hXhI2ZLDgQM>