Docker Introductions

Rajesh Kumar

DevOps Architect

@RajeshKumarIN | www.RajeshKumar.xyz

What We'll Learn

Linux Containers

- Containers vs Virtual Machines <<
- **FIGHT!!**

Kernel namespaces, cgroups, Capabilities…

Docker Engine

- Execution Driver: libcontainer vs
- LXC

AUFS, OverlayFS, Device Mapper…

Docker Images

- `docker build | docker images |`
- `docker inspect…`

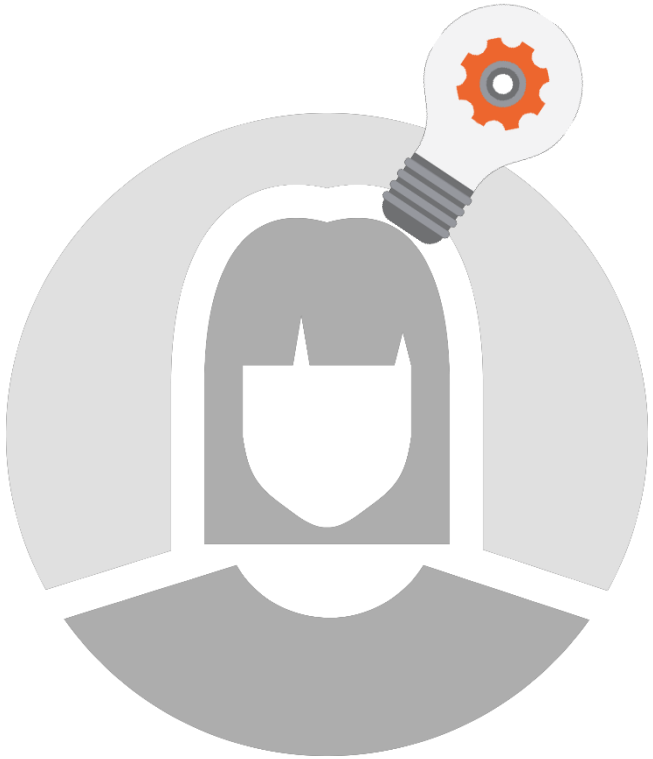
Union mounts, Layering, Dockerfile

Docker Containers

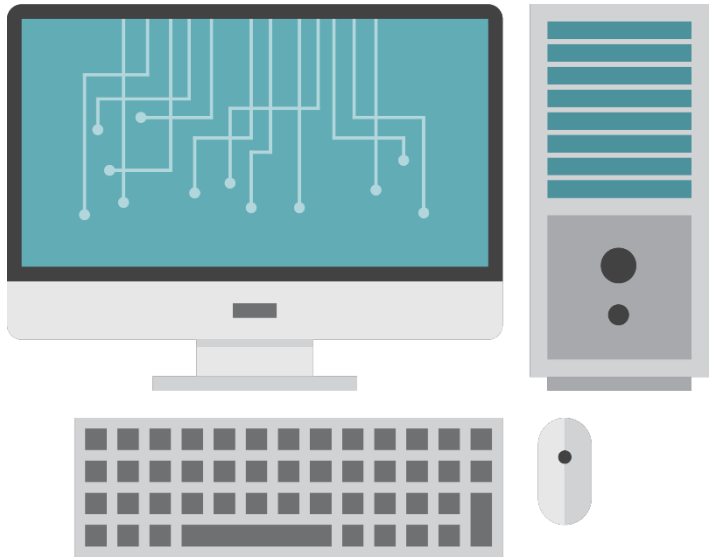
- `docker start|stop|restart`

Registries, Volumes, Networking….

Prerequisites



- Basic computer knowledge
- Do **not** need to be a Linux expert!



- 1 – 2 Linux machines (can be VMs)

It’s all about **applications**

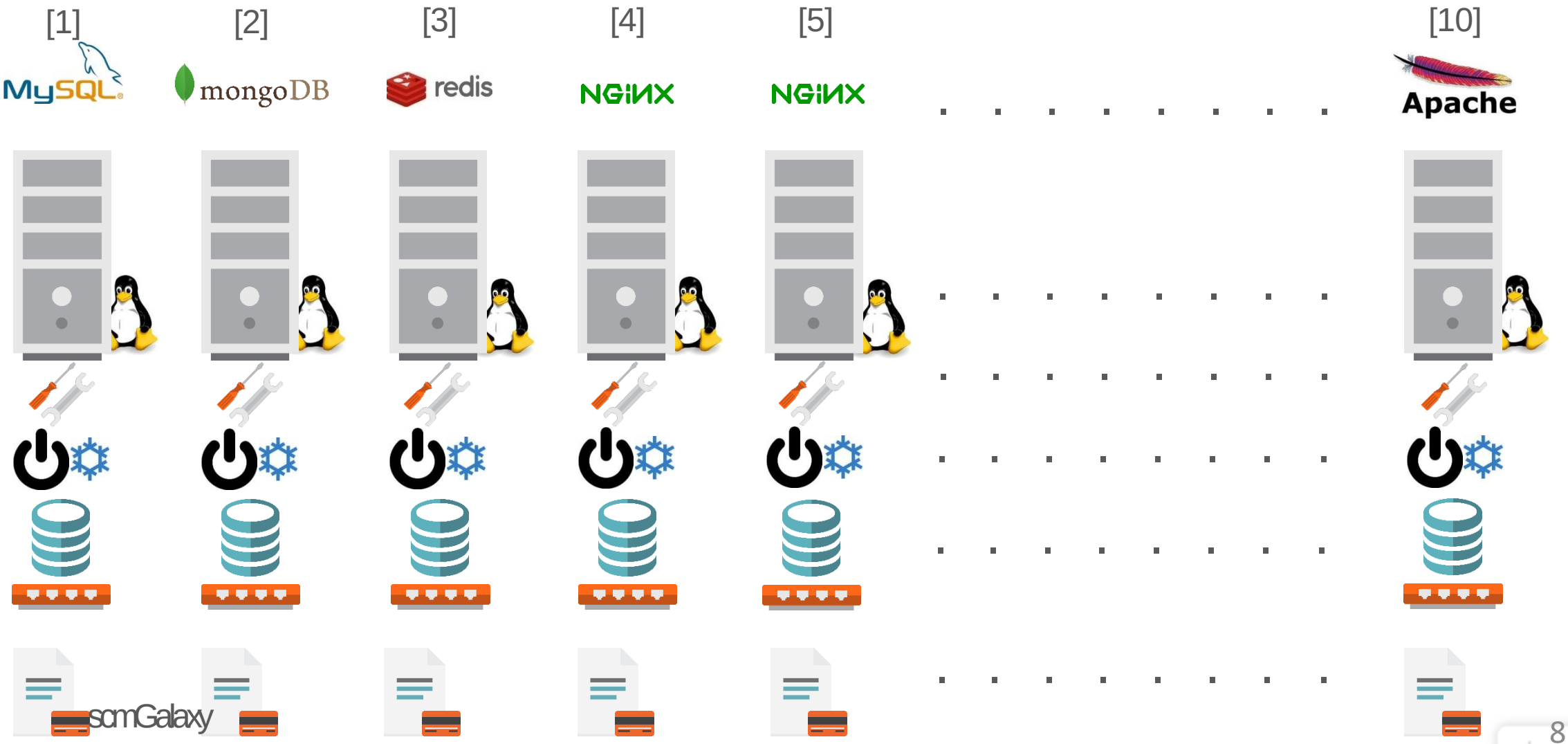
server : application

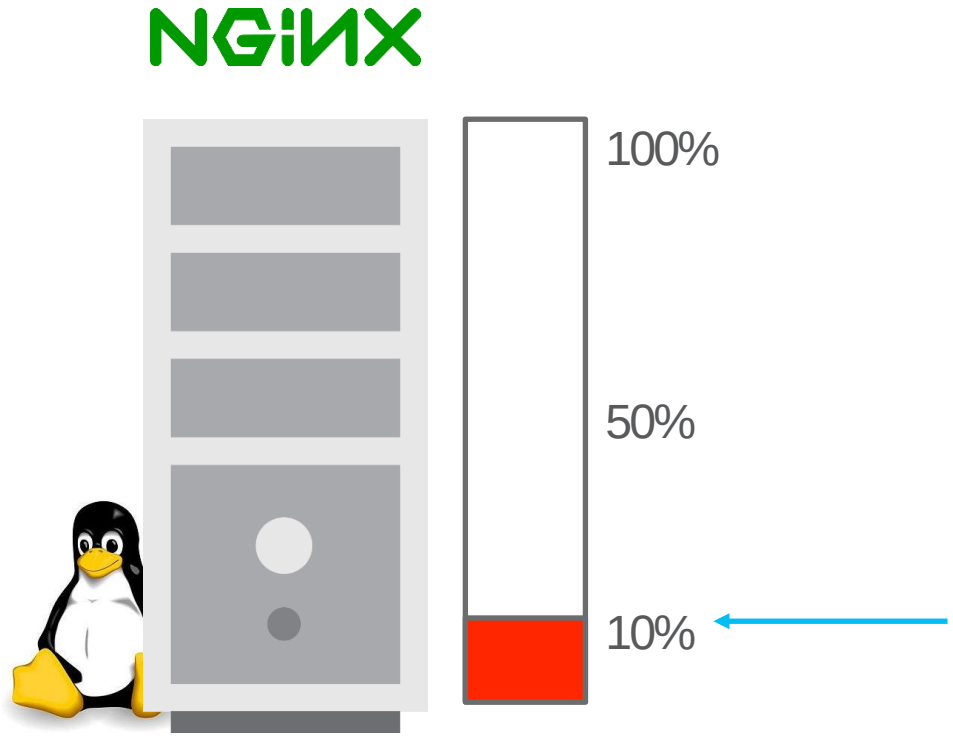
1 : 1

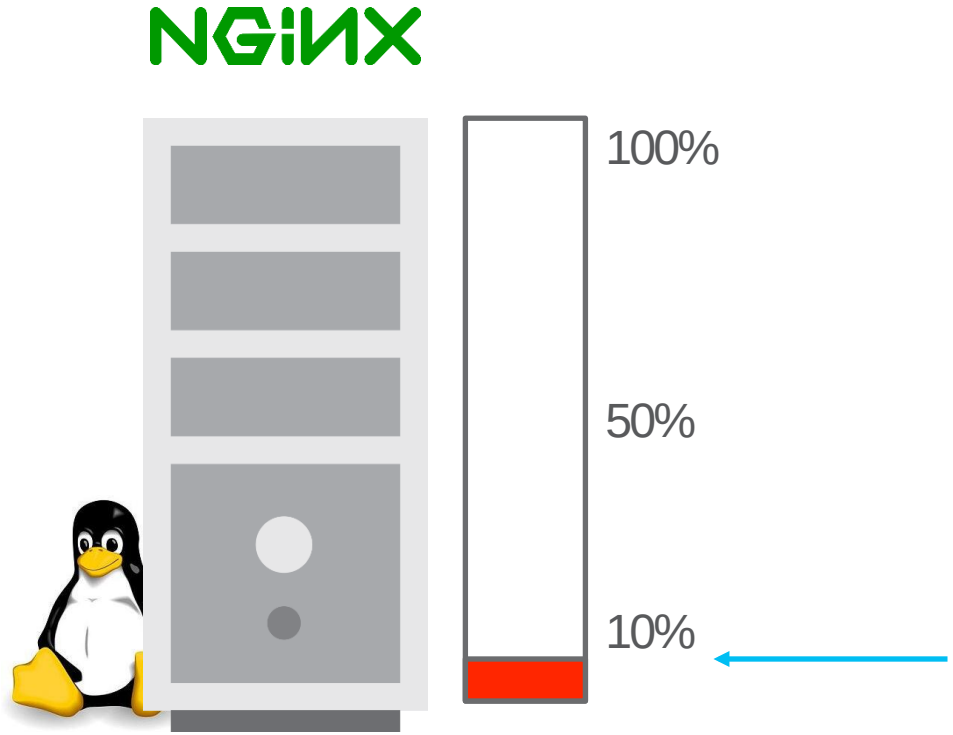
server : application

1 : 1

server : application
1 : 1

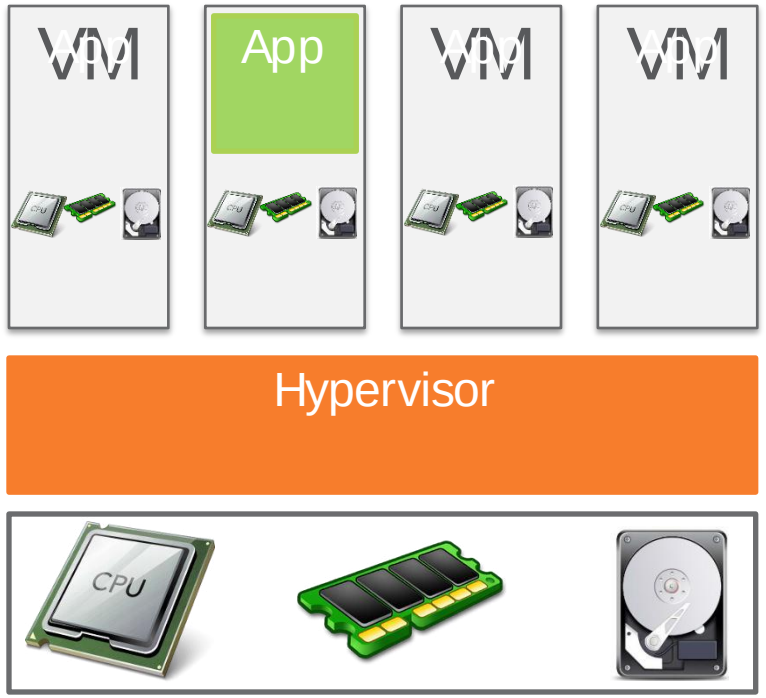





NGINX
100%
50%
10%

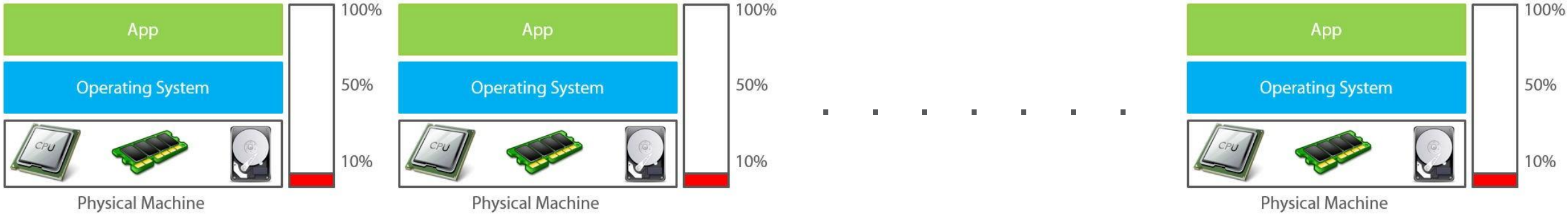
vmware®

vmware®



Physical Machine

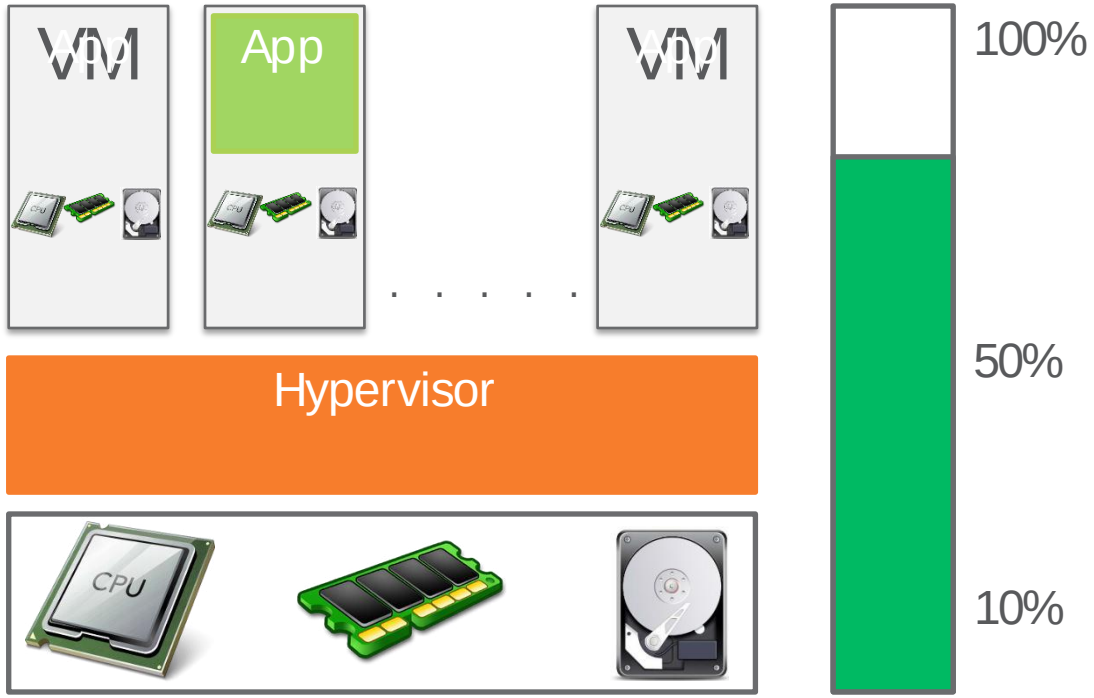
10 x Apps | 10 x Physical Machines | Less than 10% utilization



10 x Apps

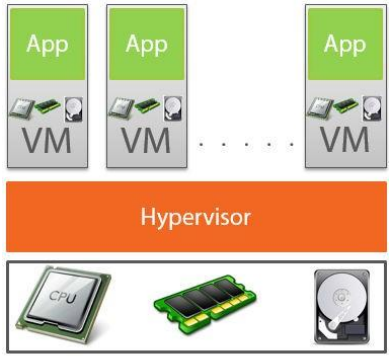
1 x Physical Machines

~80% utilization

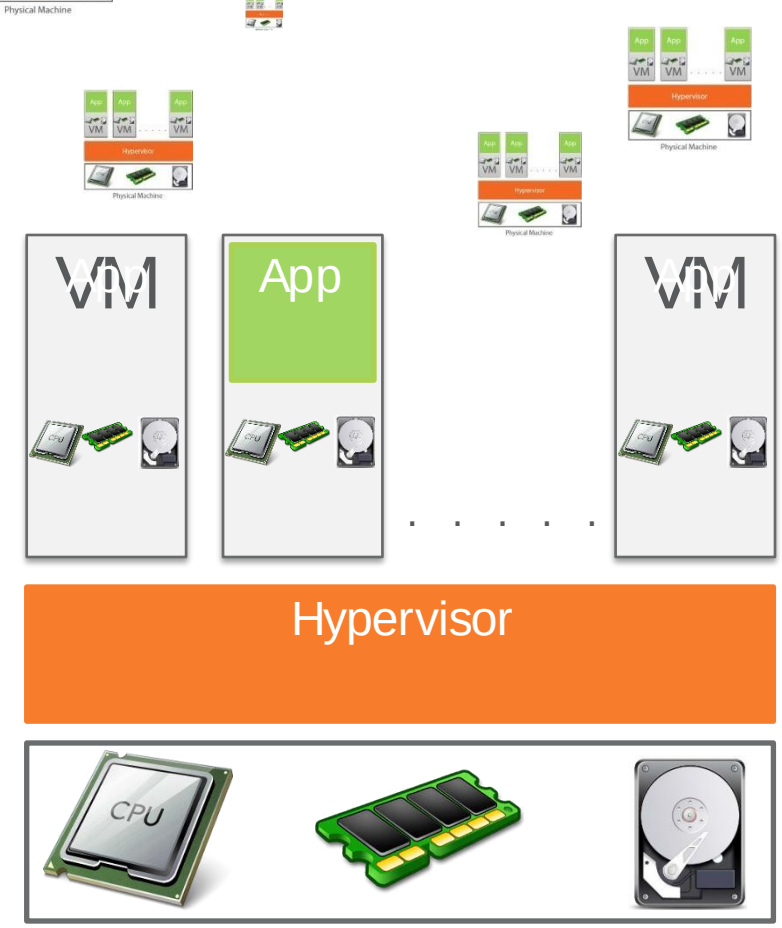


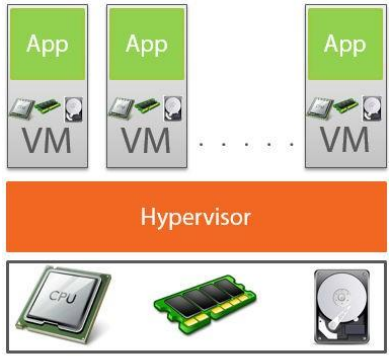
Physical Machine

scmGalaxy

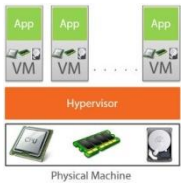

App
App
App
VM
VM
VM
Hypervisor
CPU
Physical Machine

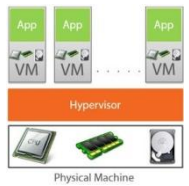
scmGalaxy

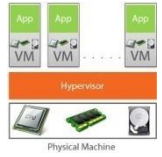

VM
App
VM
Hypervisor
CPU
Physical Machine

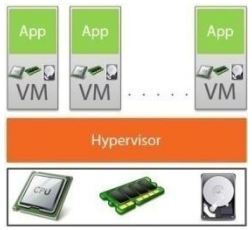


Physical Machine

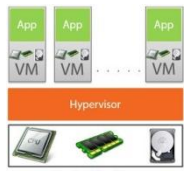


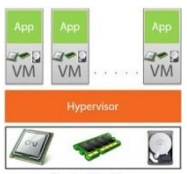




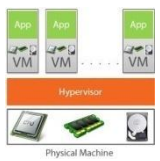


Physical Machine

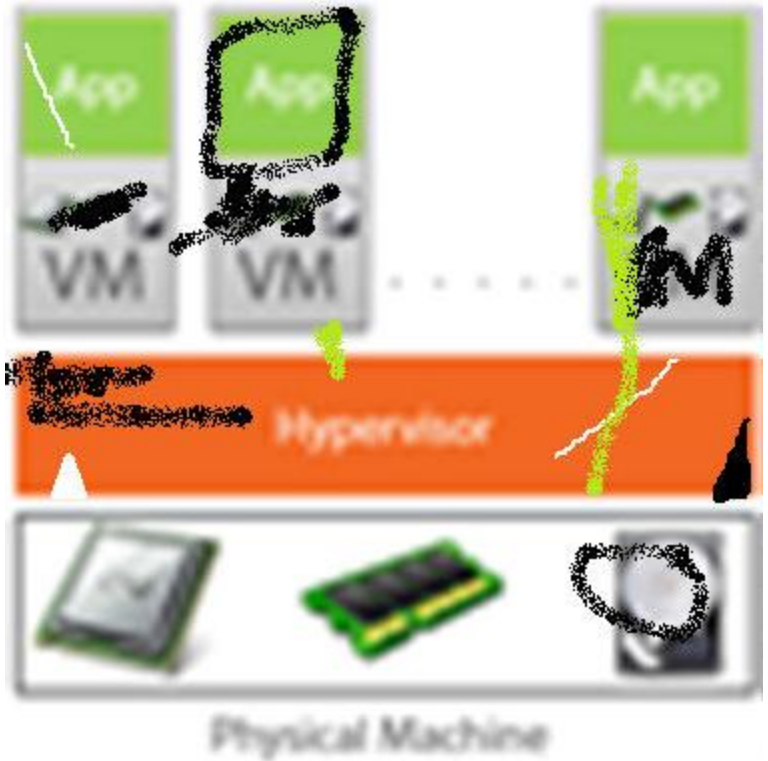




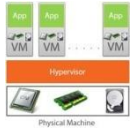
Physical Machine

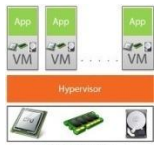


Physical Machine

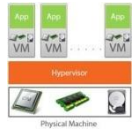


Physical Machine





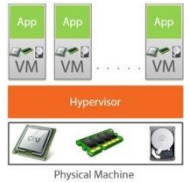
Physical Machine



Physical Machine

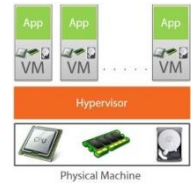
Physical Machine

Physical Machine

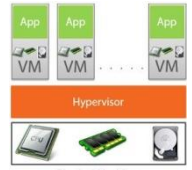


Physical Machine

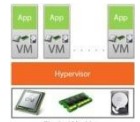
Physical Machine



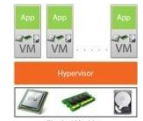
Physical Machine



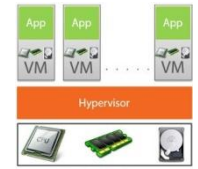
Physical Machine



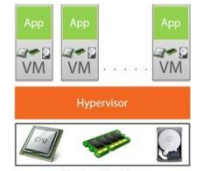
Physical Machine



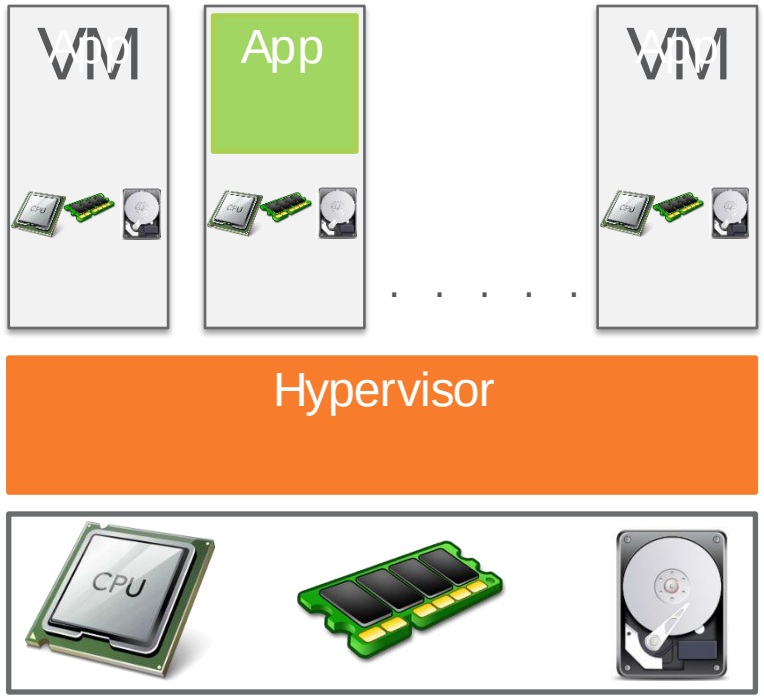
Physical Machine

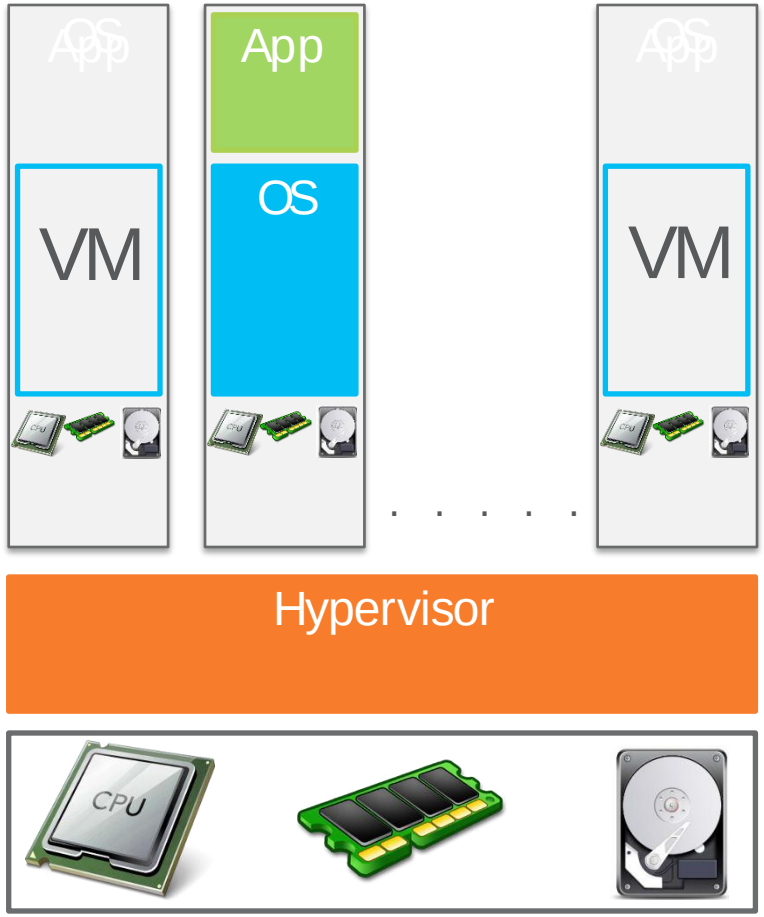


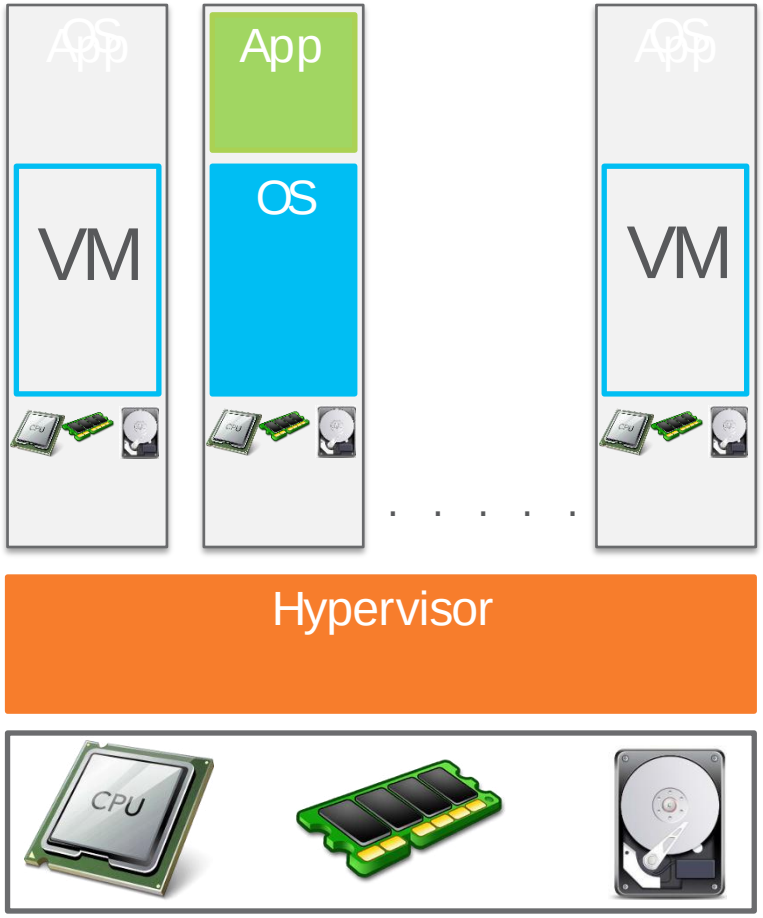
Physical Machine



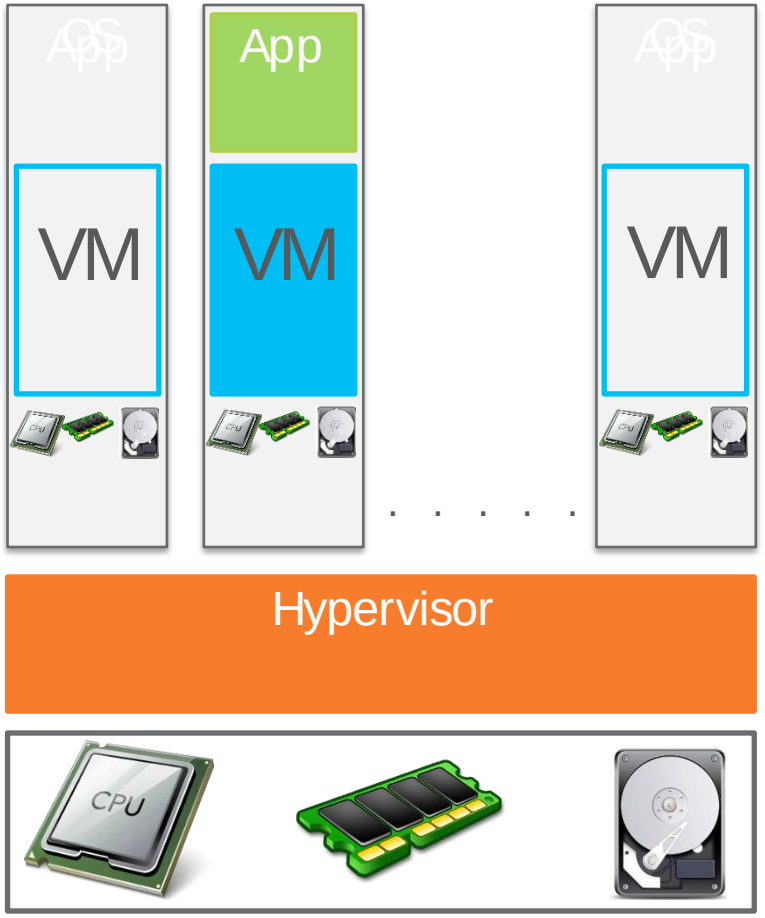
Physical Machine




App
OS
VM
App
OS
VM
Hypervisor
CPU

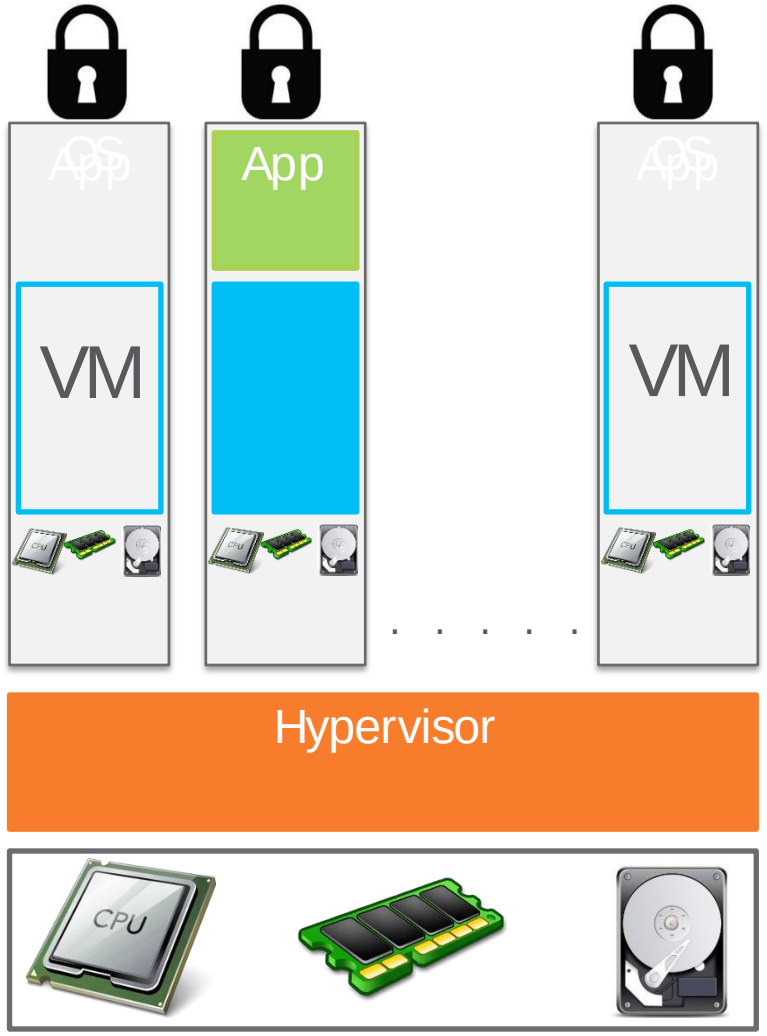


somGalaxy Physical Machine



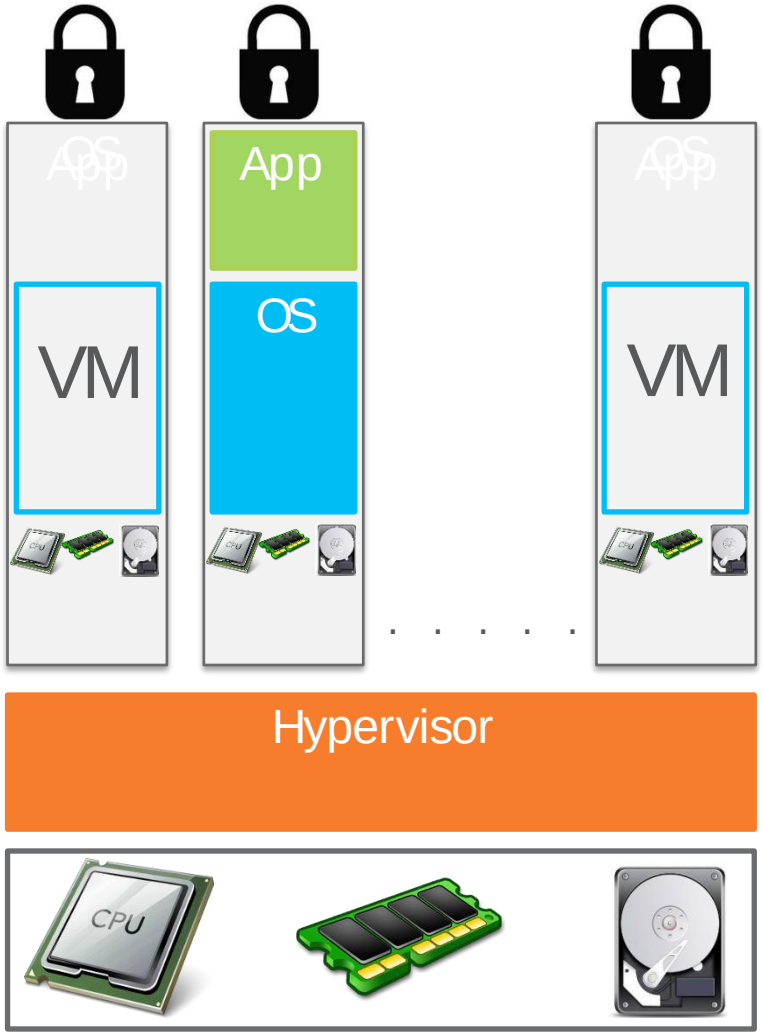
scmGalaxy Physical Machine

> OS != Business Value

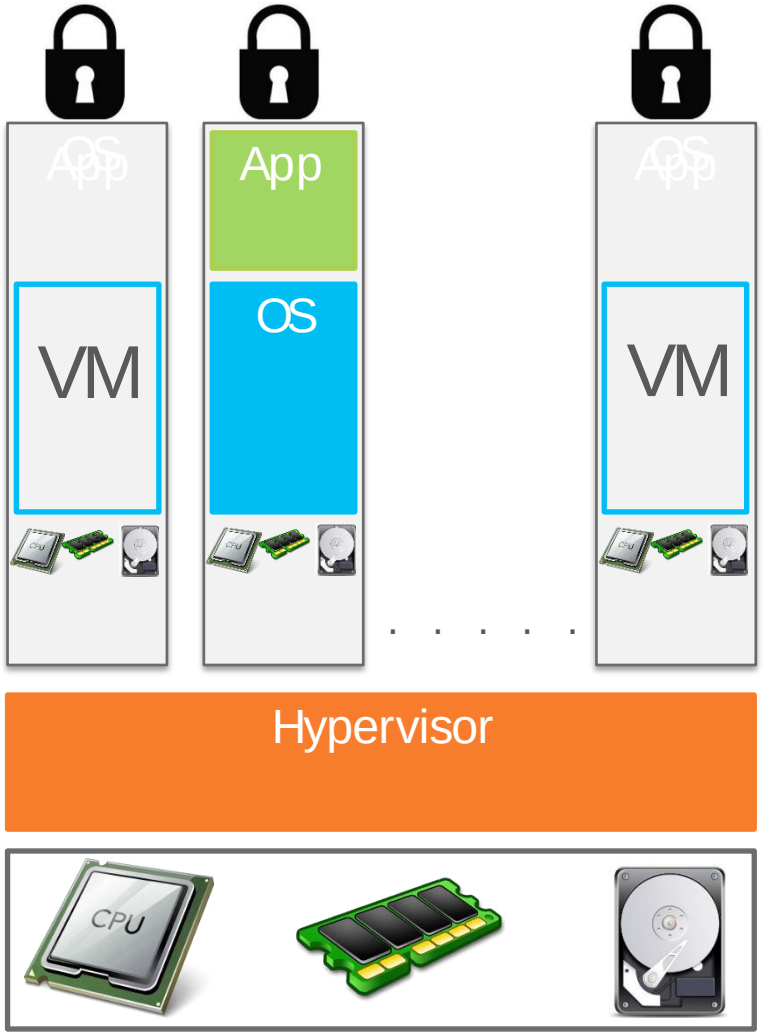


somGalaxy Physical Machine

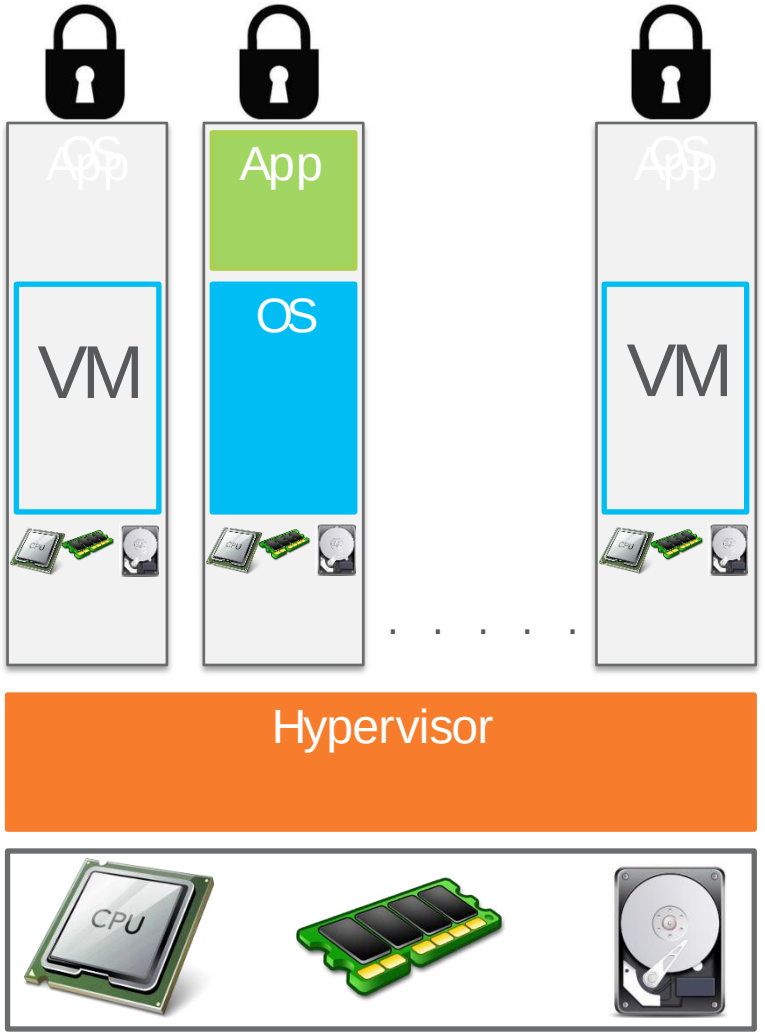
> OS != Business Value



somGalaxy Physical Machine

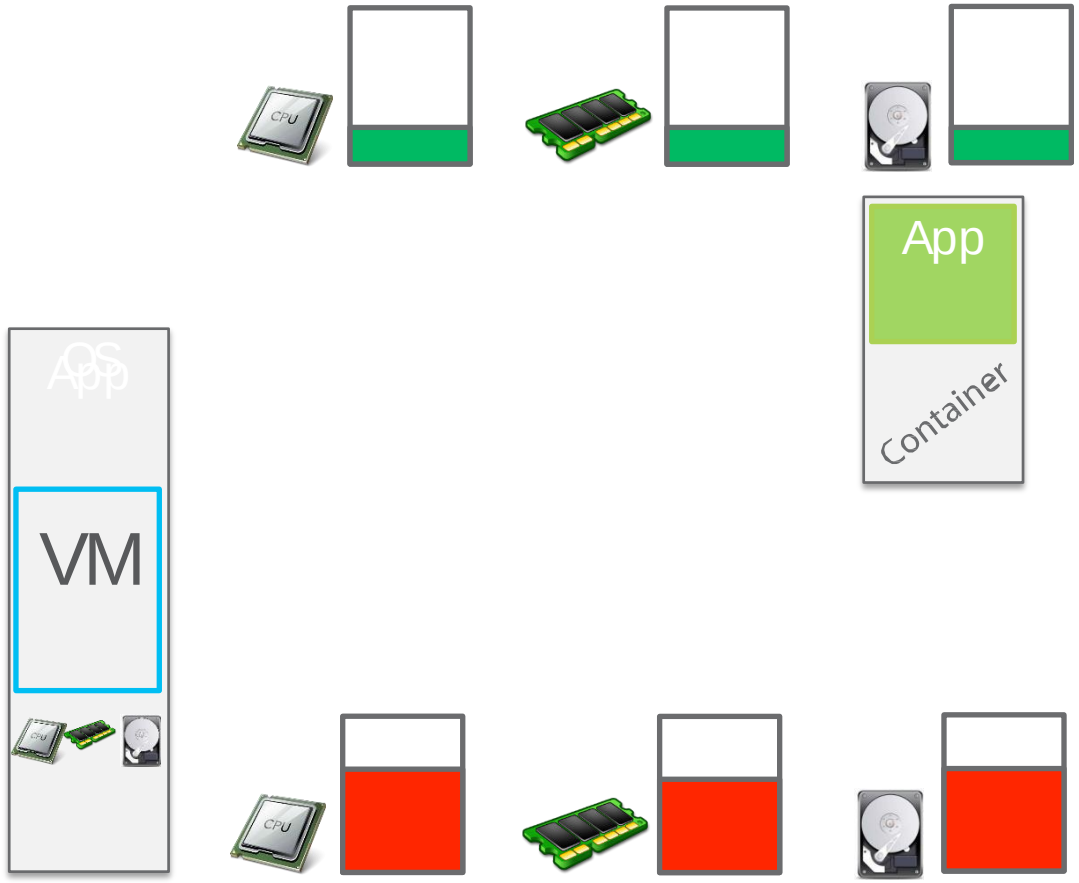


somGalaxy Physical Machine

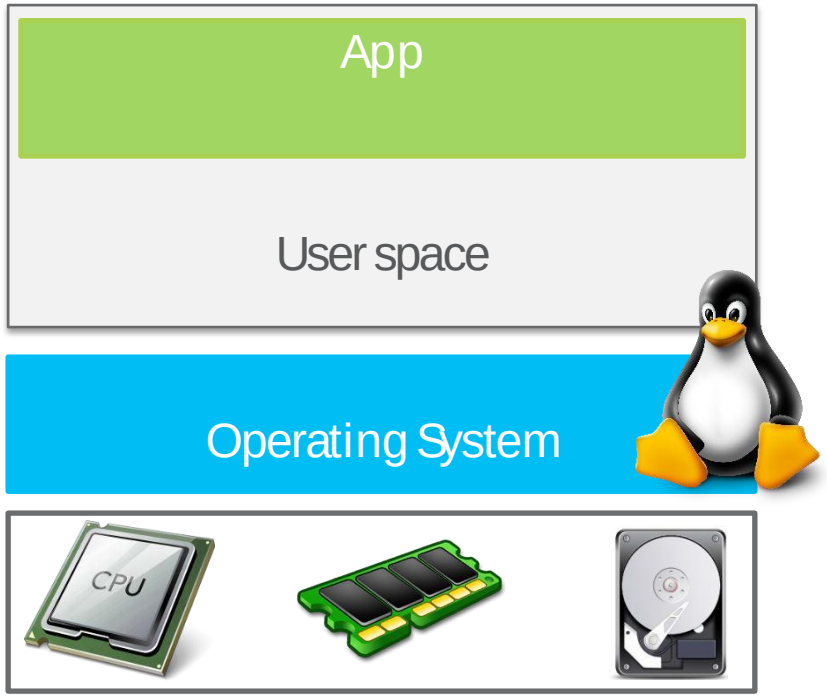


scmGalaxy Physical Machine

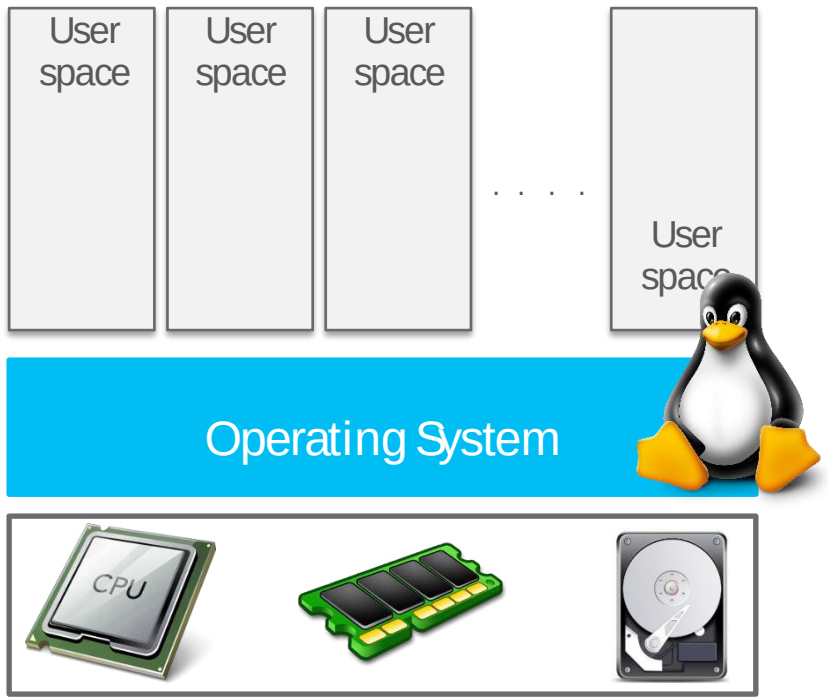

App
Container



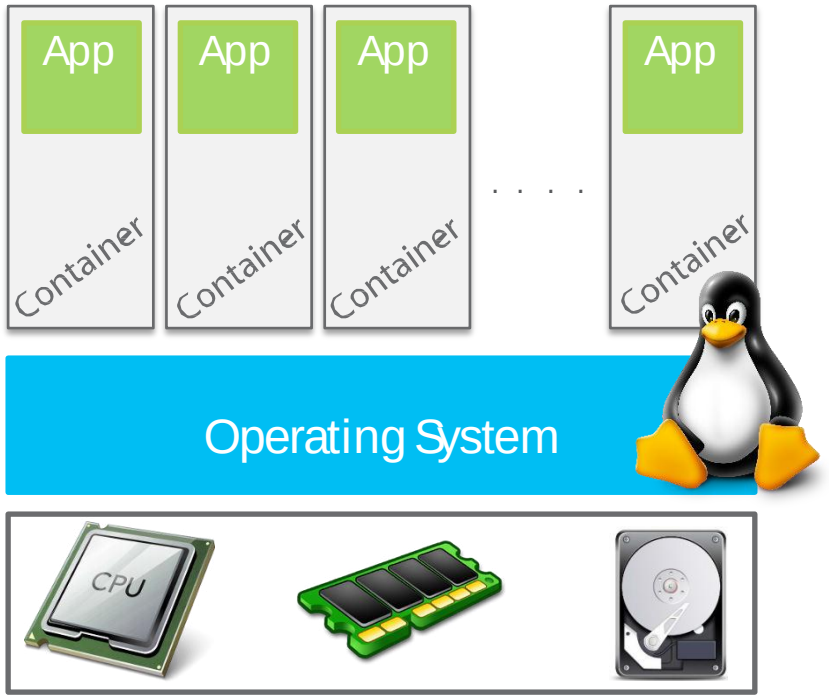
Containers are more lightweight than Virtual Machines



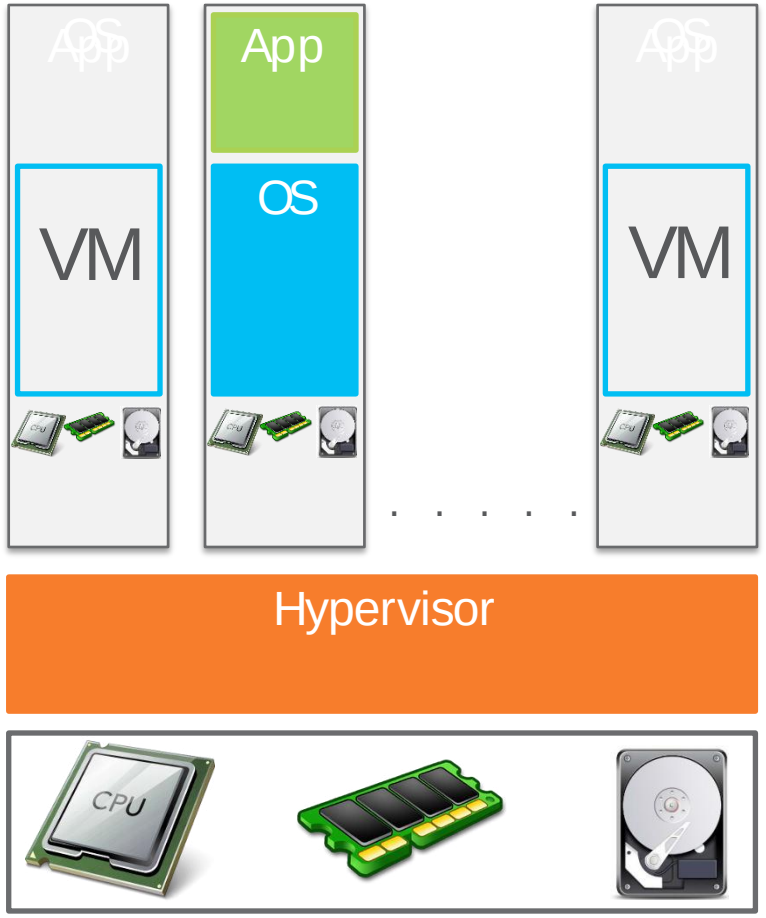
Physical Machine



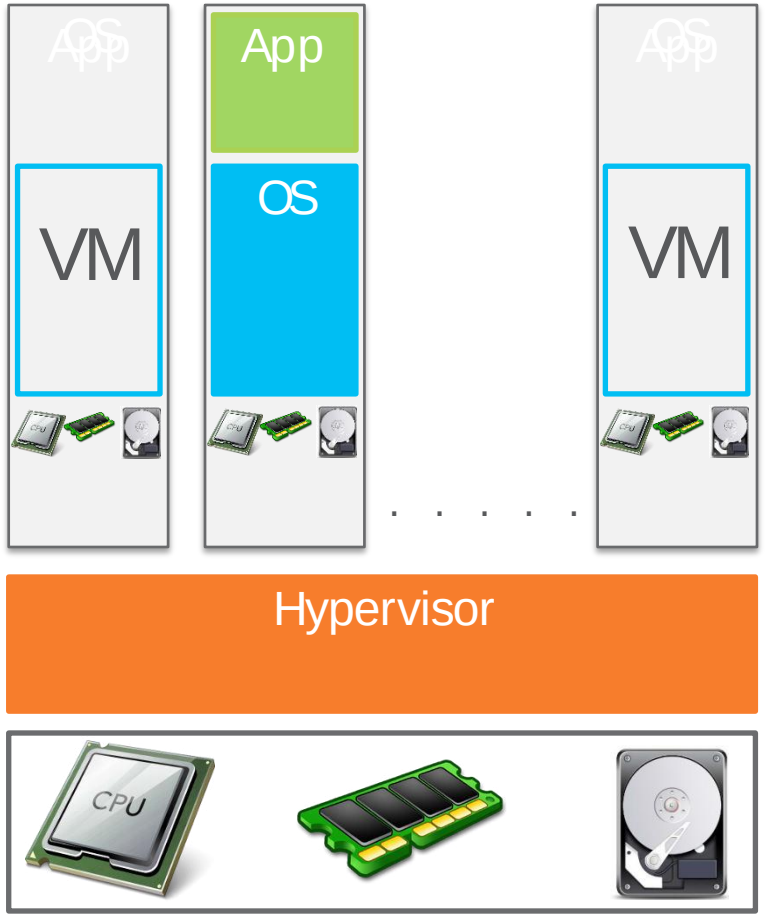
Physical Machine



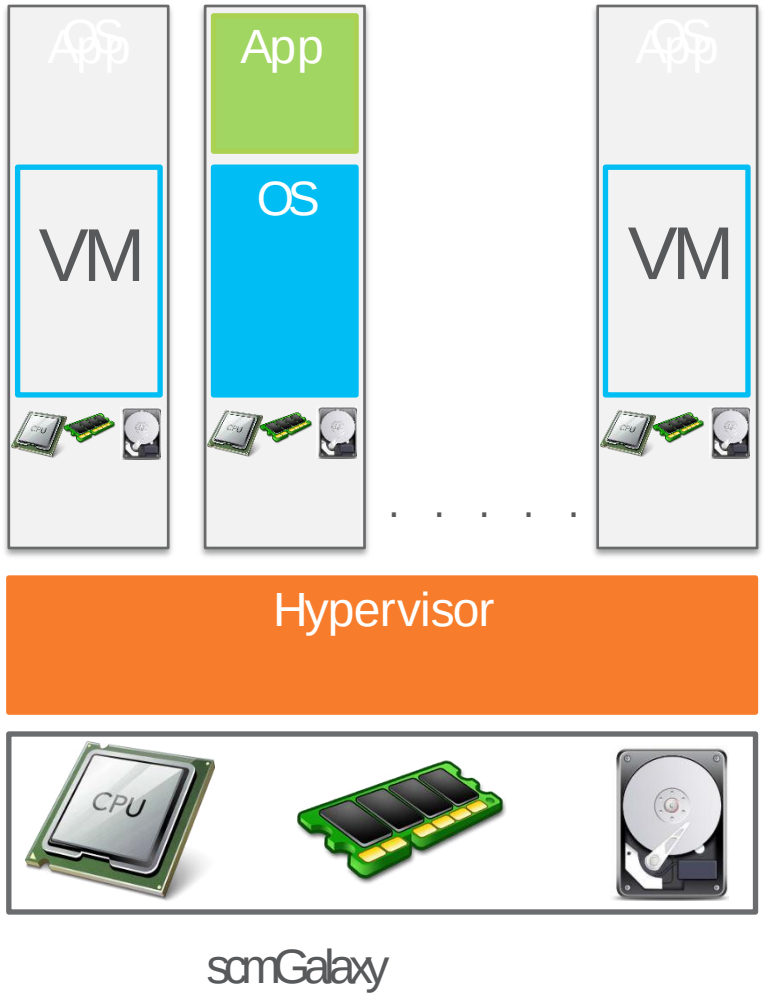
Physical Machine



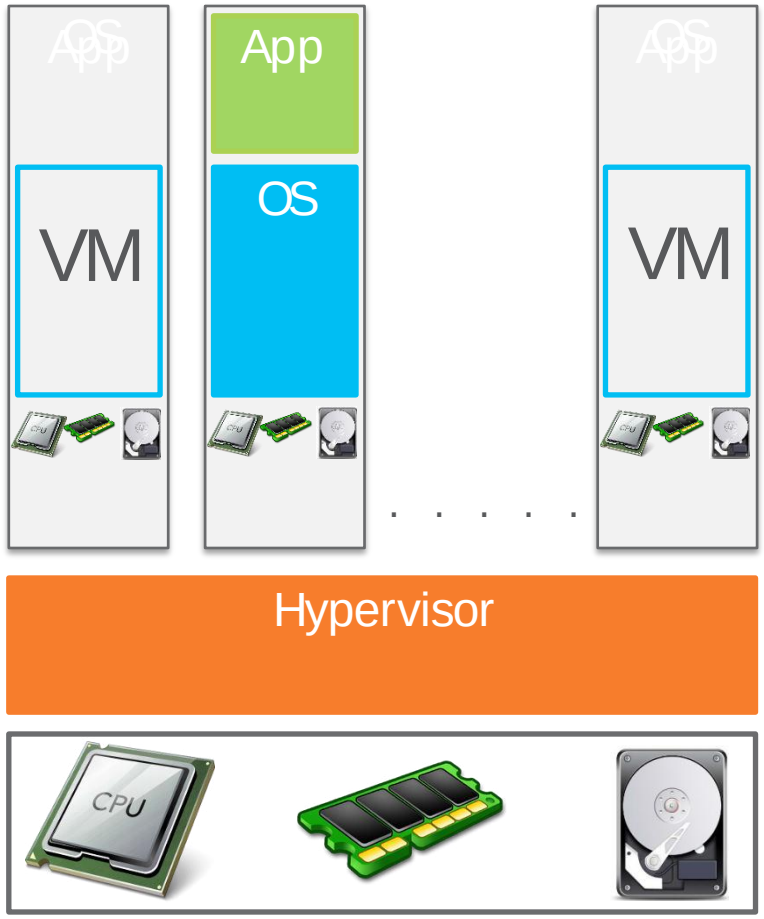
Physical Machine



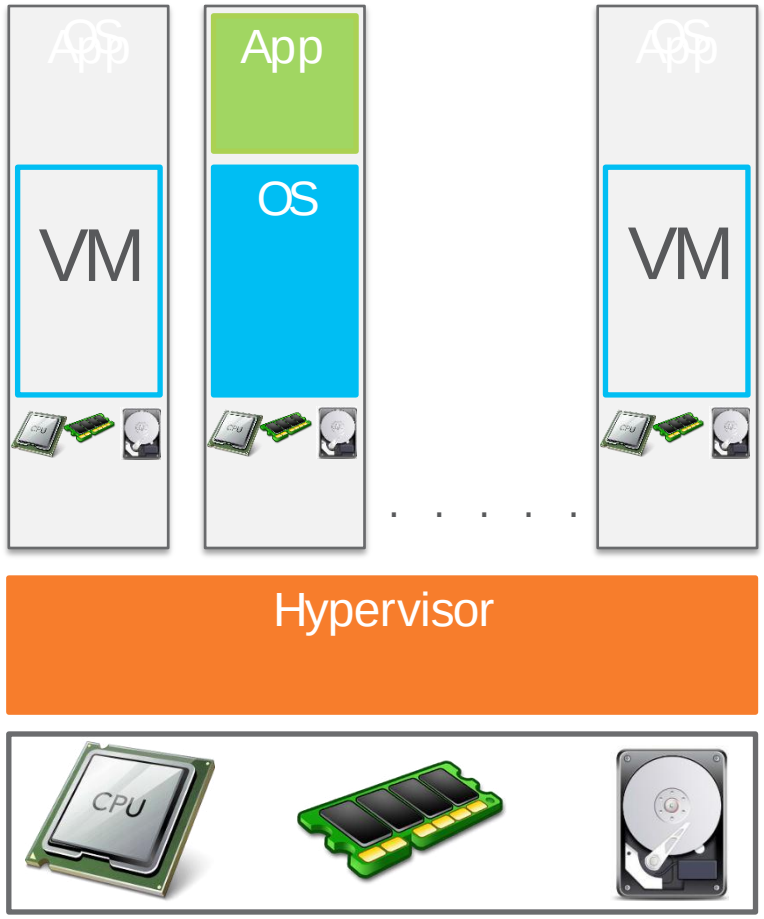
Physical Machine



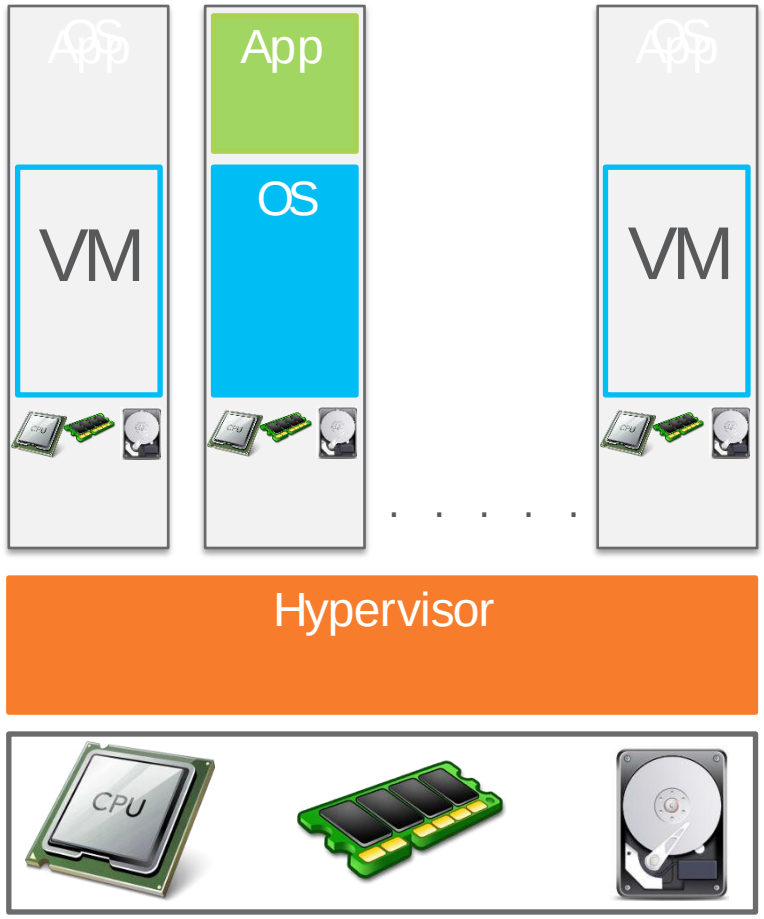
scmGalaxy



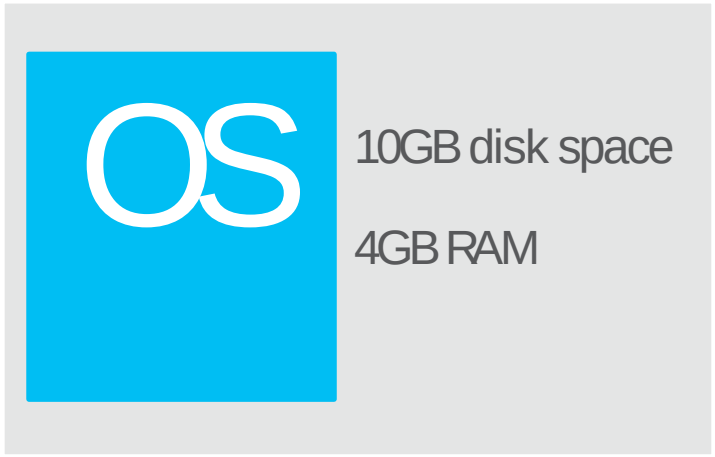
scmGalaxy

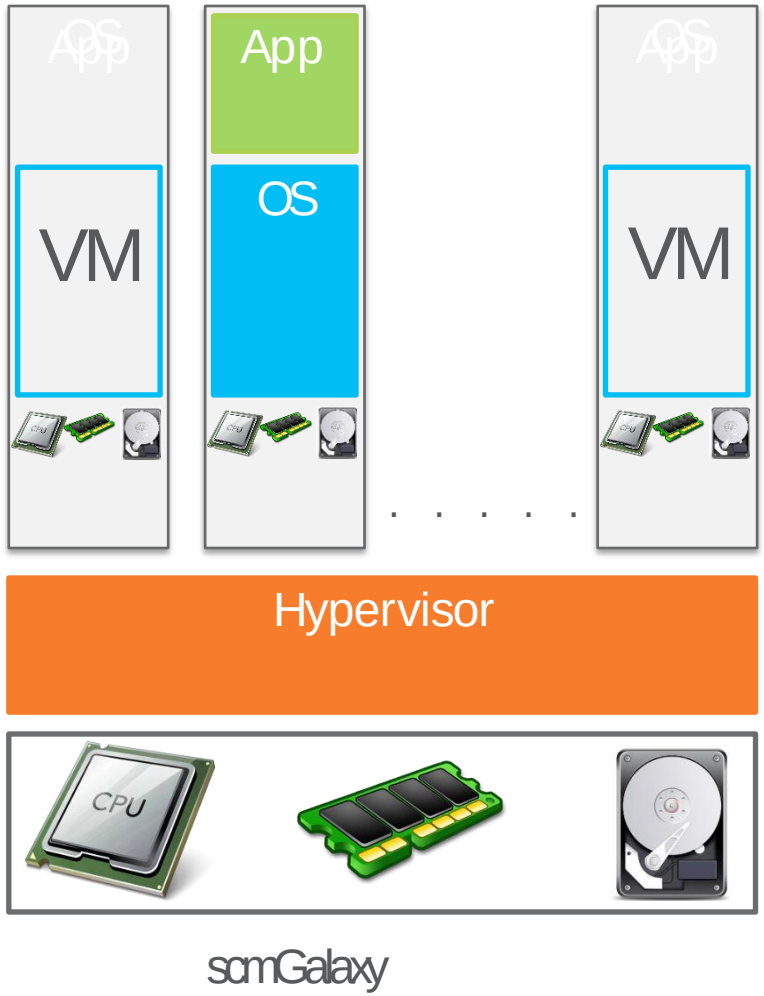


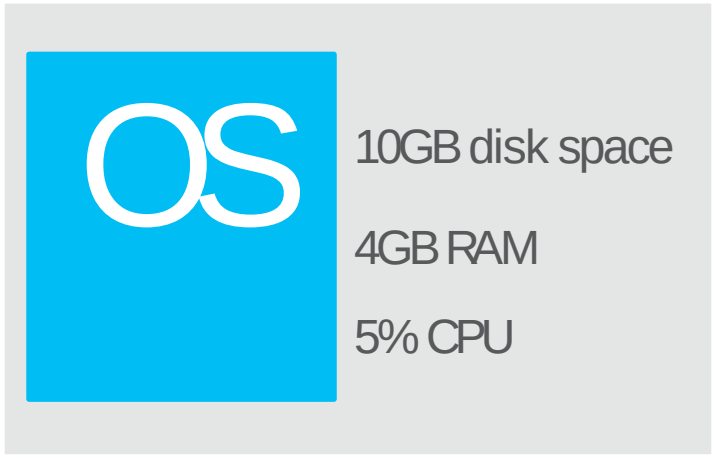
scmGalaxy

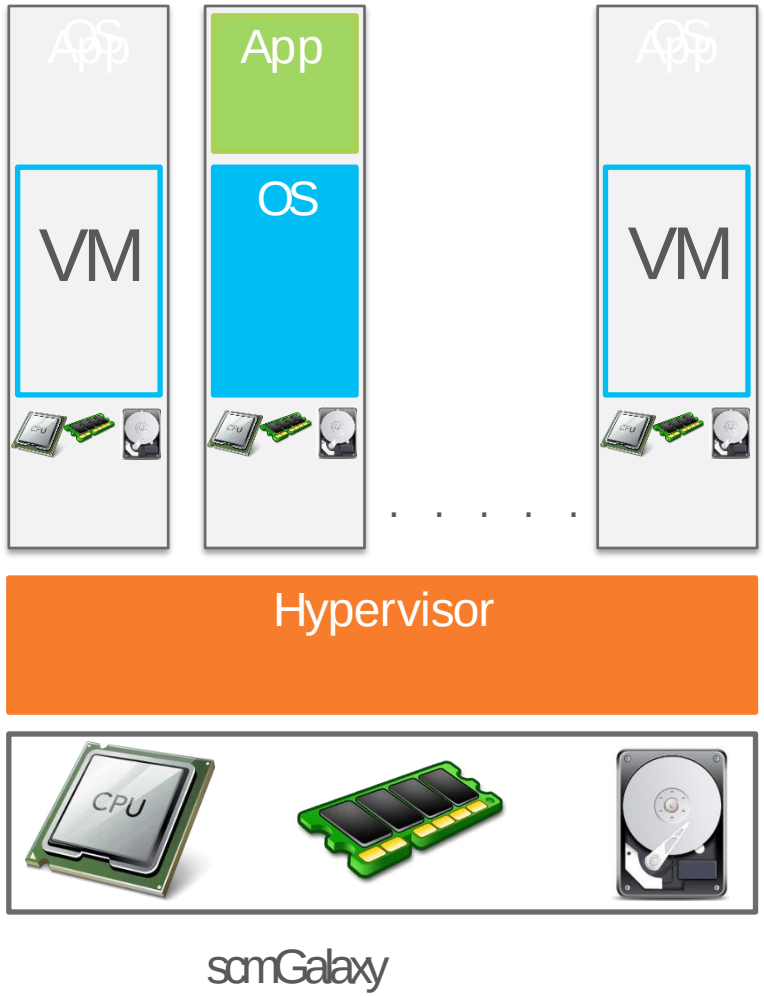


scmGalaxy




App
OS
VM
App
OS
VM
Hypervisor
CPU
scmGalaxy


OS
10GB disk space
4GB RAM
5% CPU



scmGalaxy

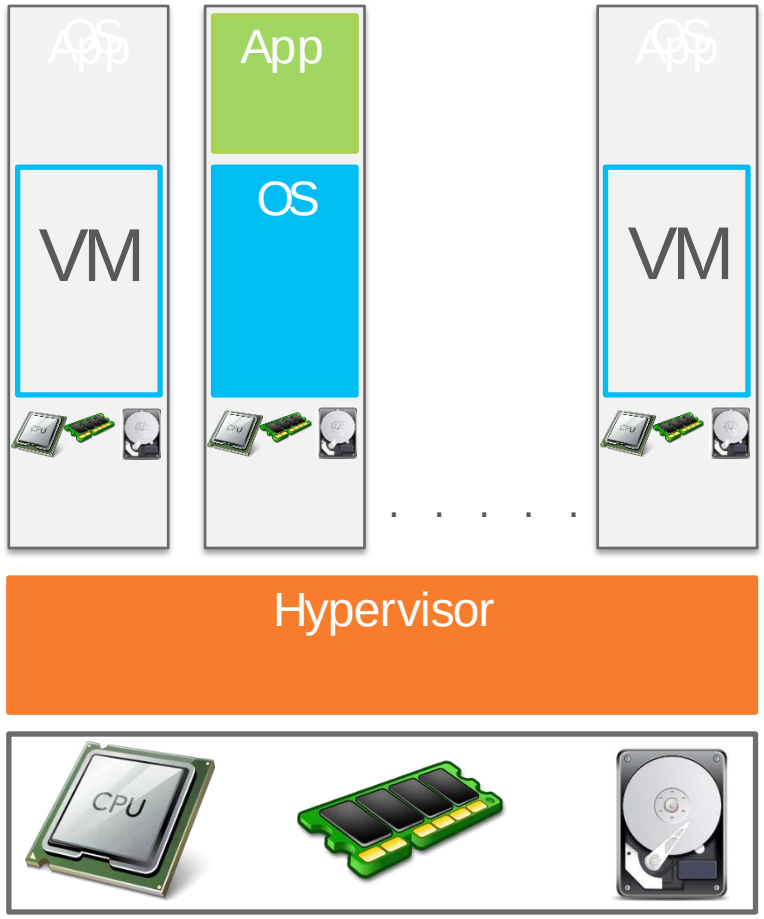
OS

10GB disk space

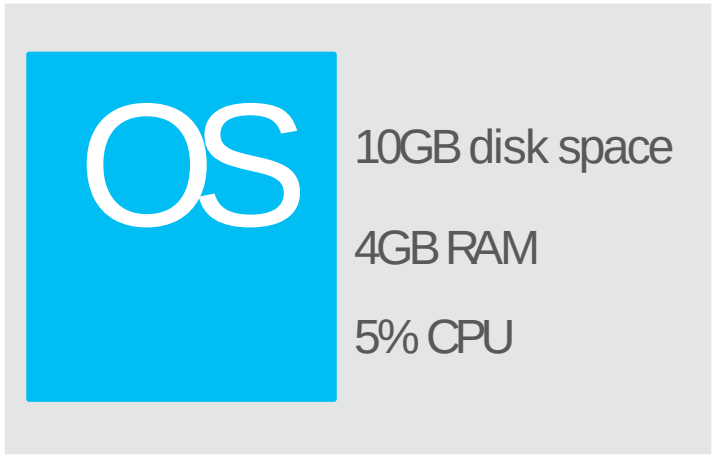
4GB RAM

5% CPU

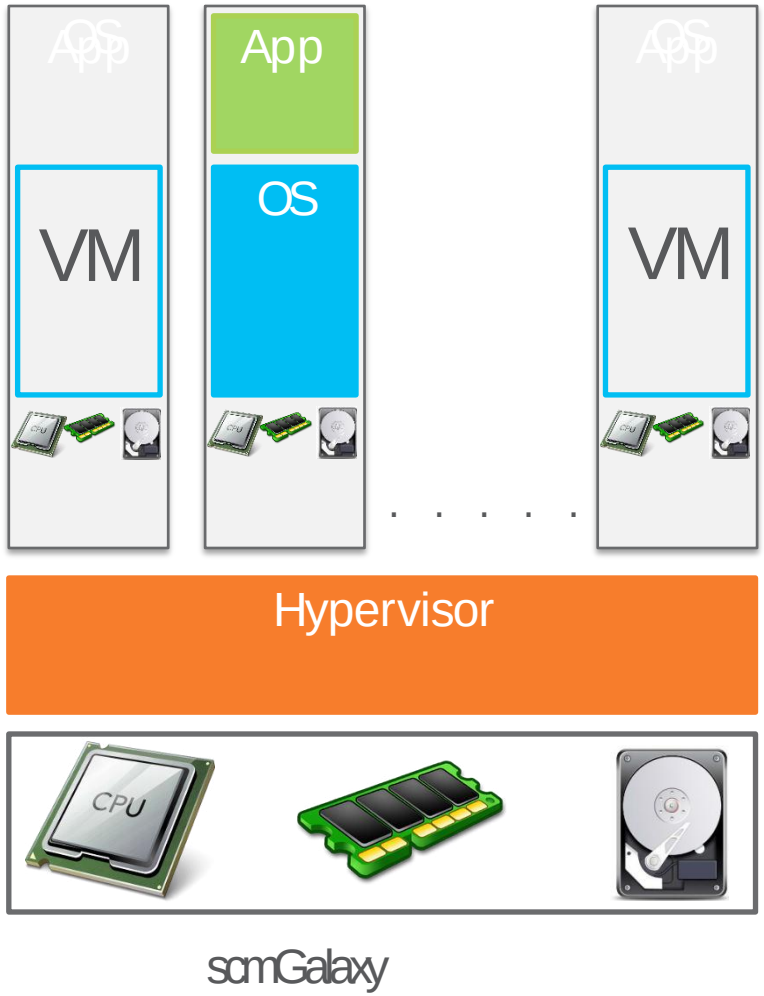
x 10 =

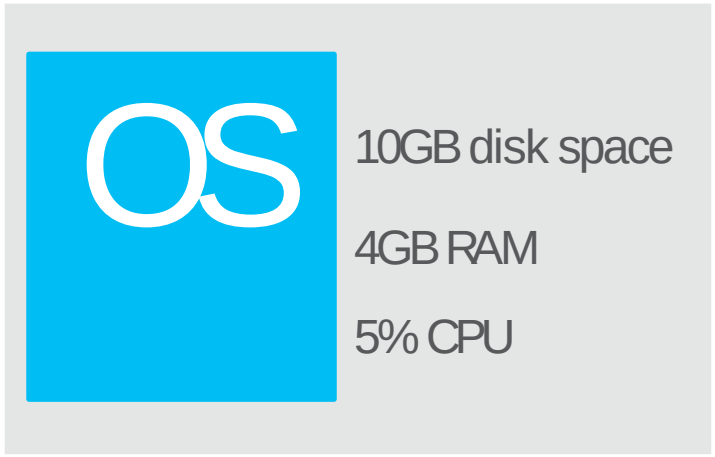


scmGalaxy



x 10 = **100GB** disk space

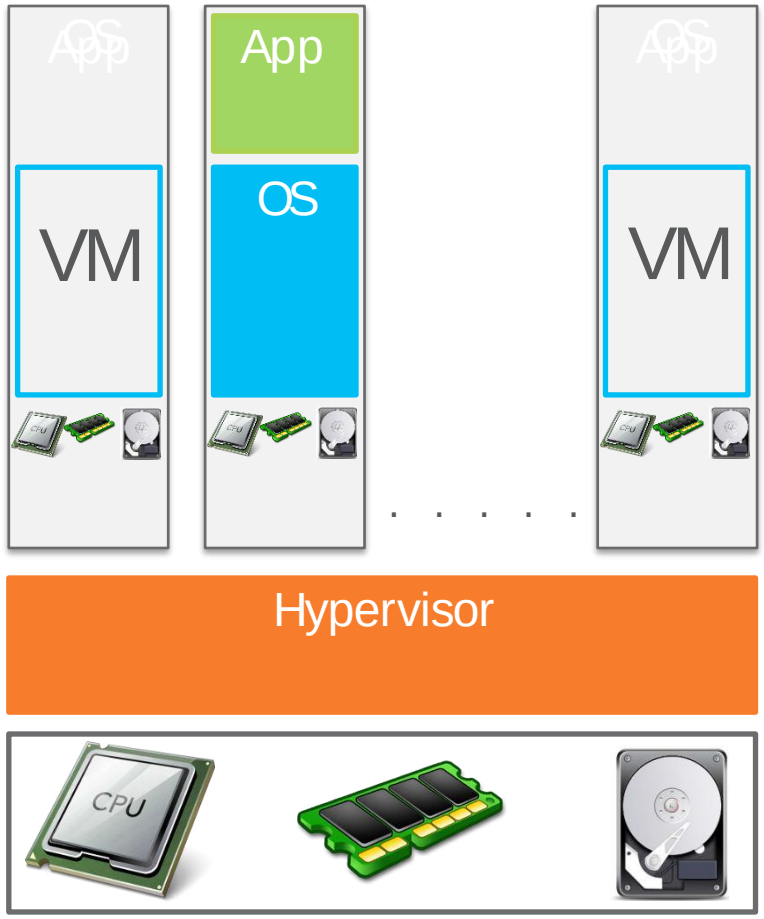


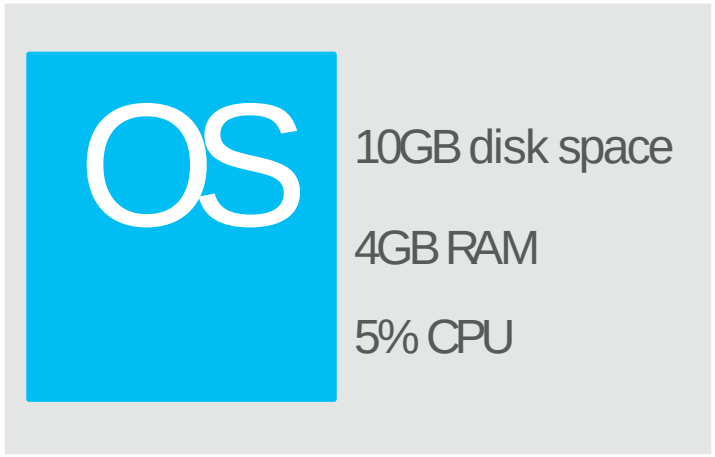


x 10 =

100GB disk space

40GB RAM



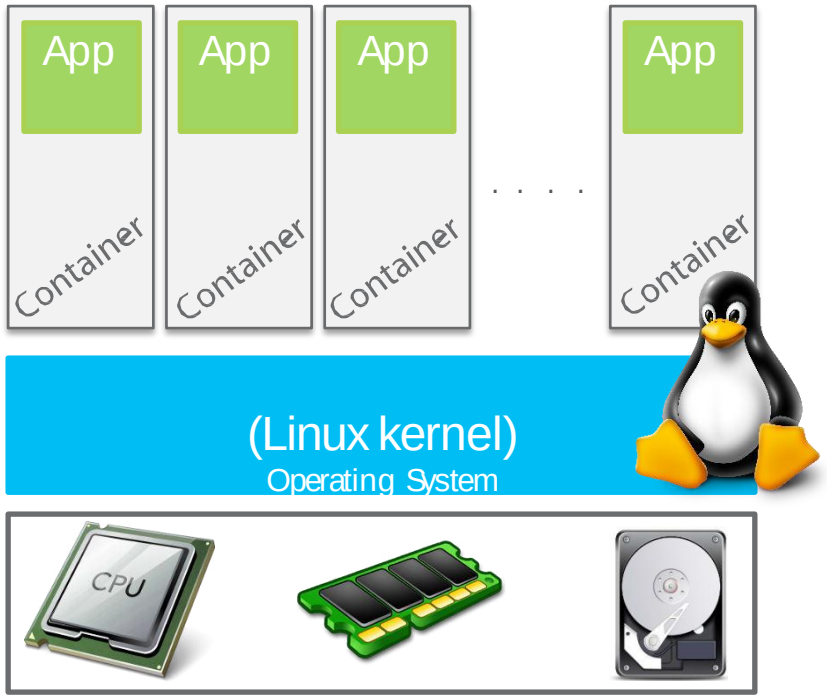


x 10 =

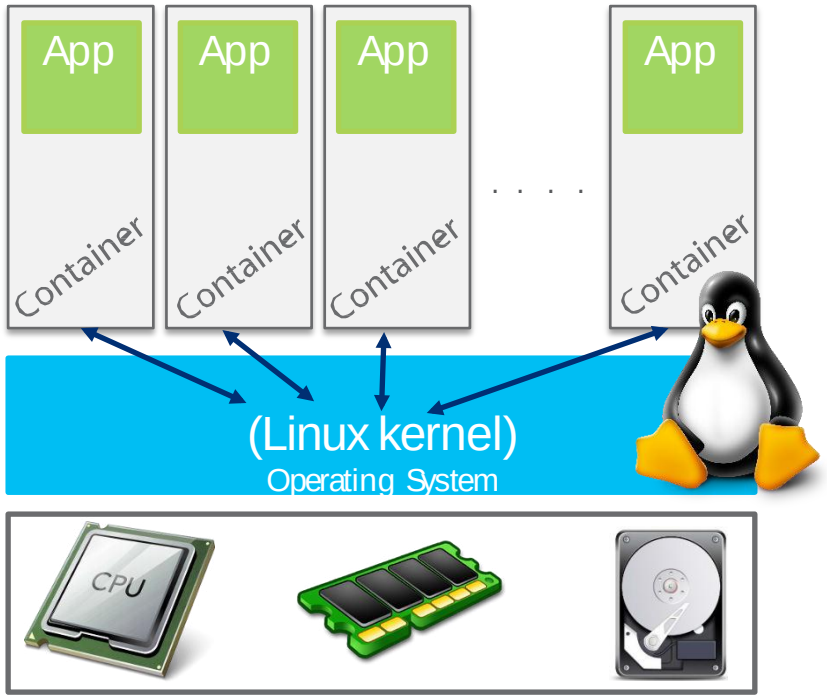
100GB disk space

40GB RAM

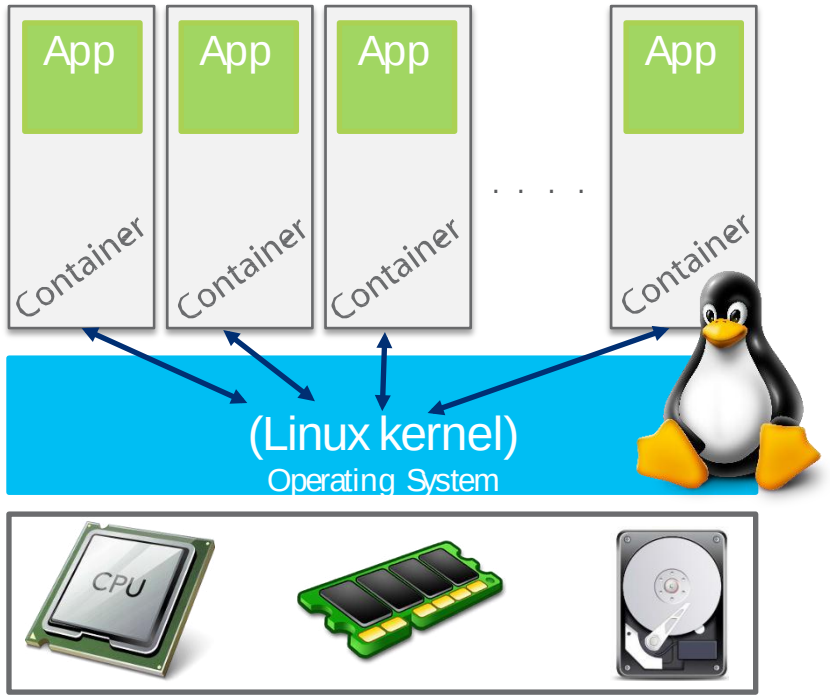
50% CPU



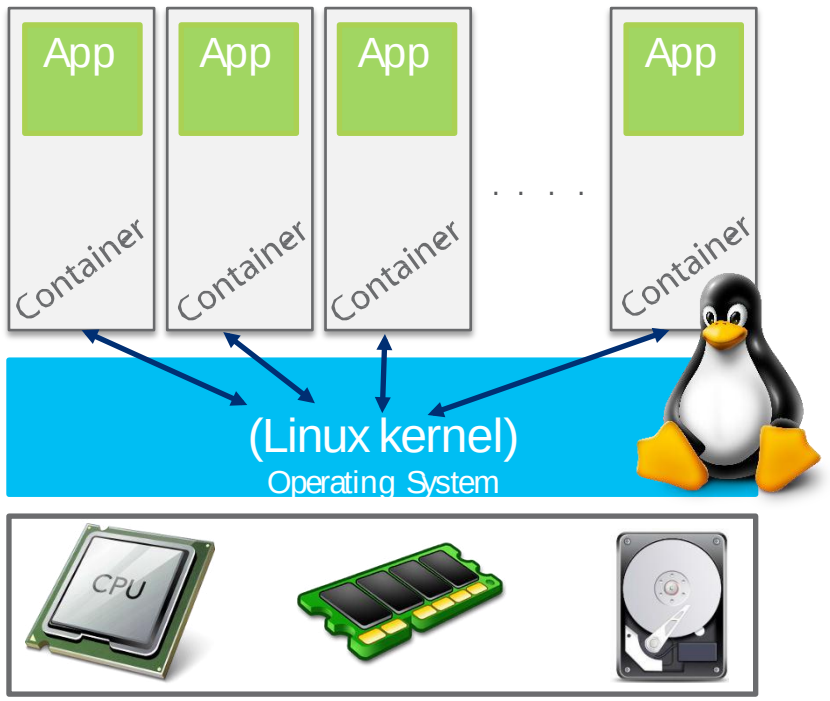
Physical Machine



Physical Machine



Physical Machine

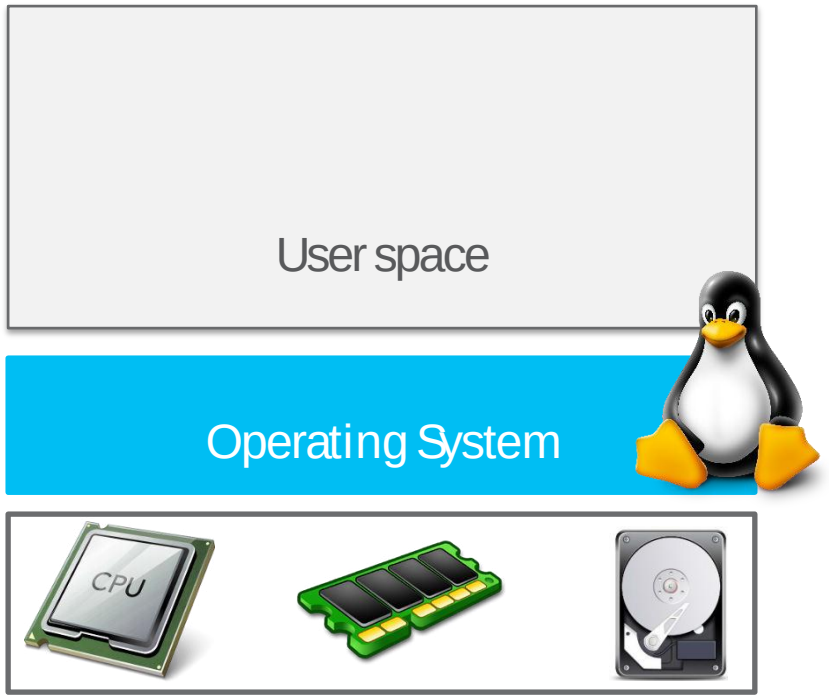


Physical Machine

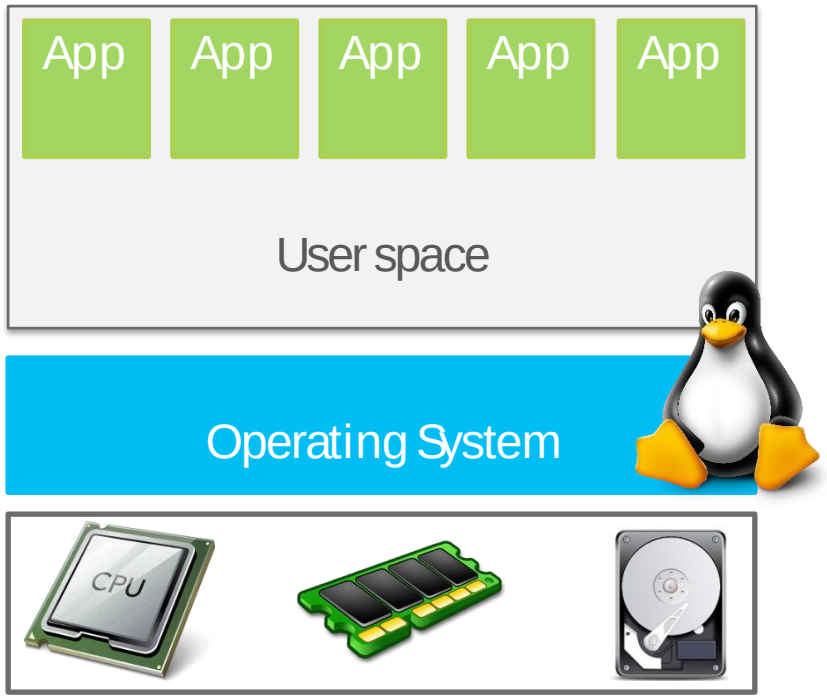
Containers consume less CPU, RAM and disk resource than Virtual Machines

How Containers Work

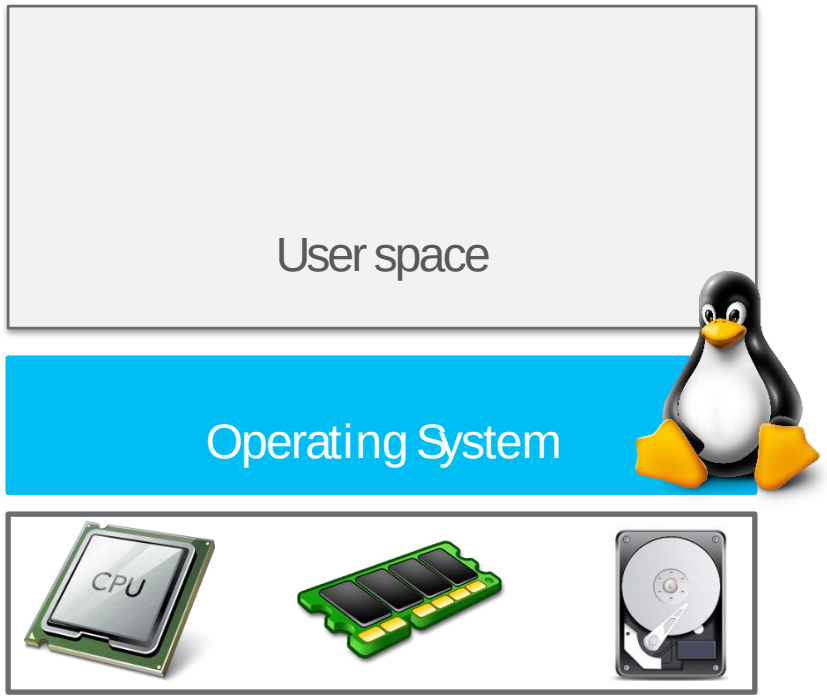
How Containers Work



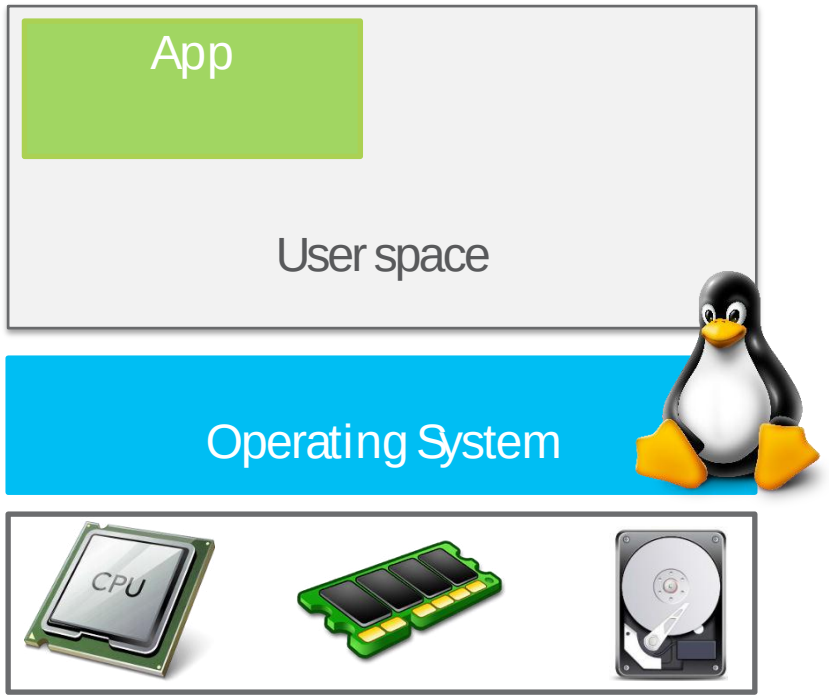
How Containers Work



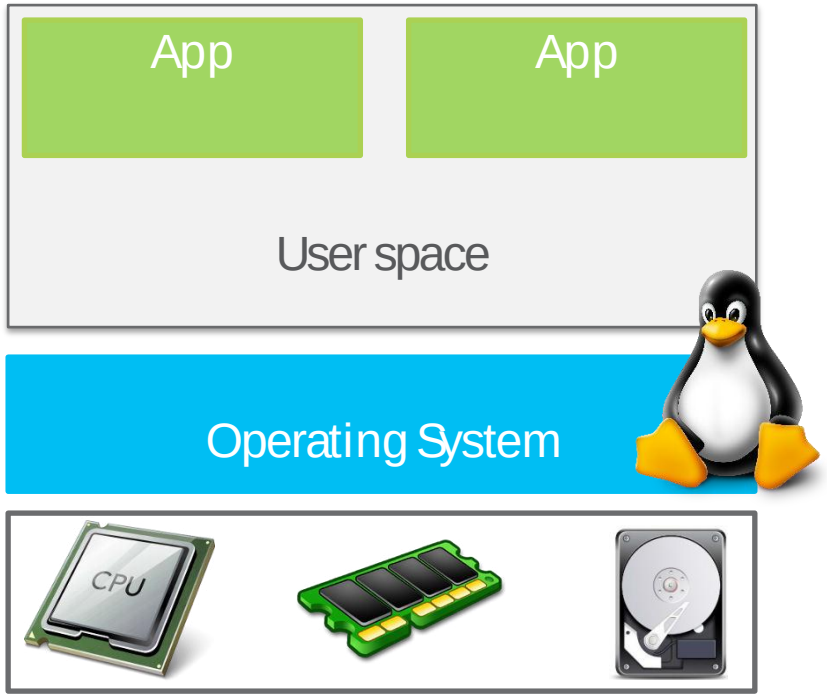
How Containers Work



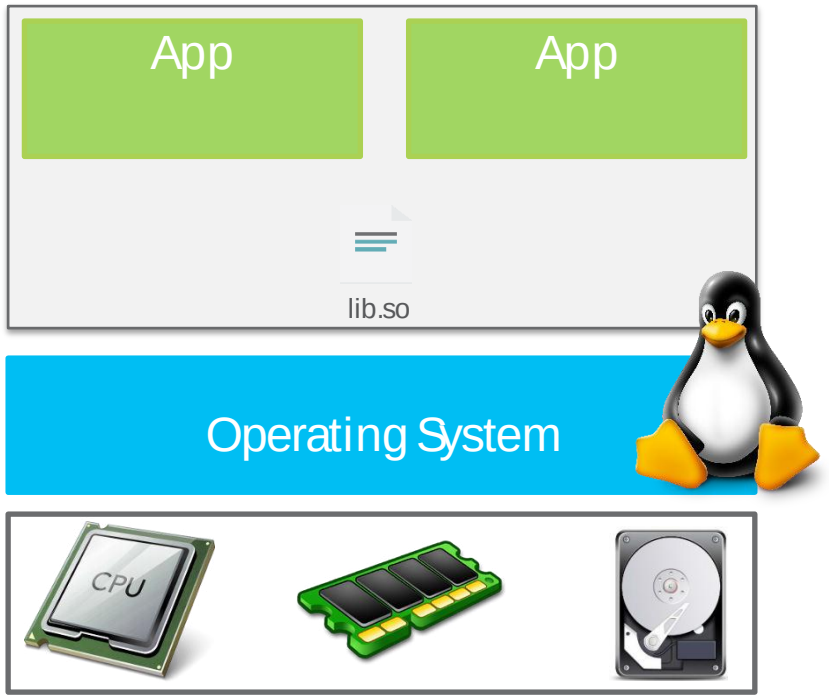
How Containers Work



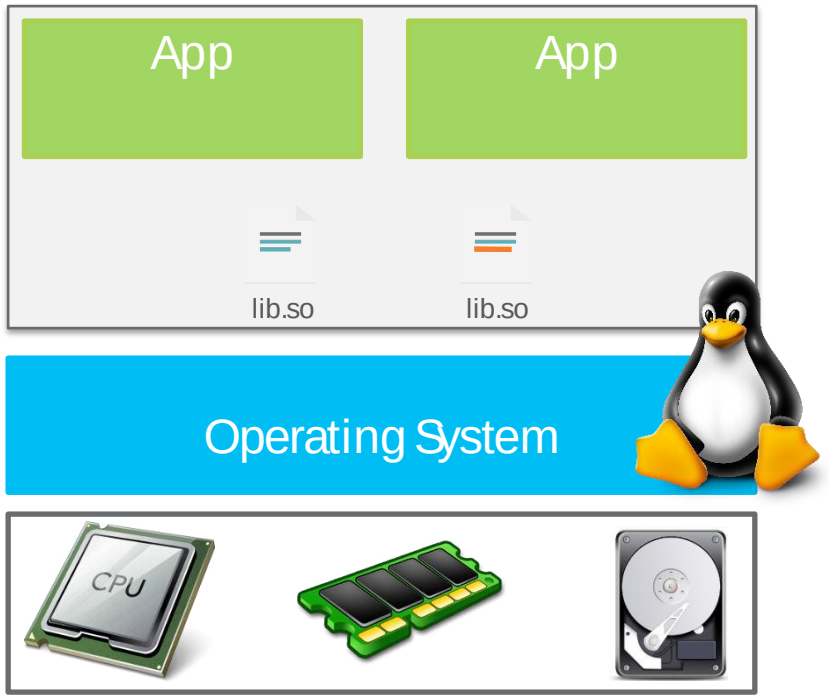
How Containers Work



How Containers Work



How Containers Work



Physical Machine

How Containers Work