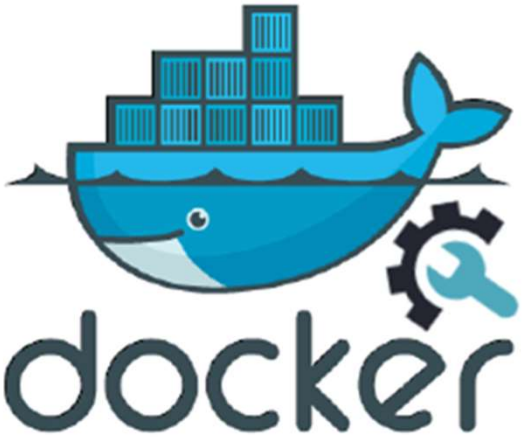
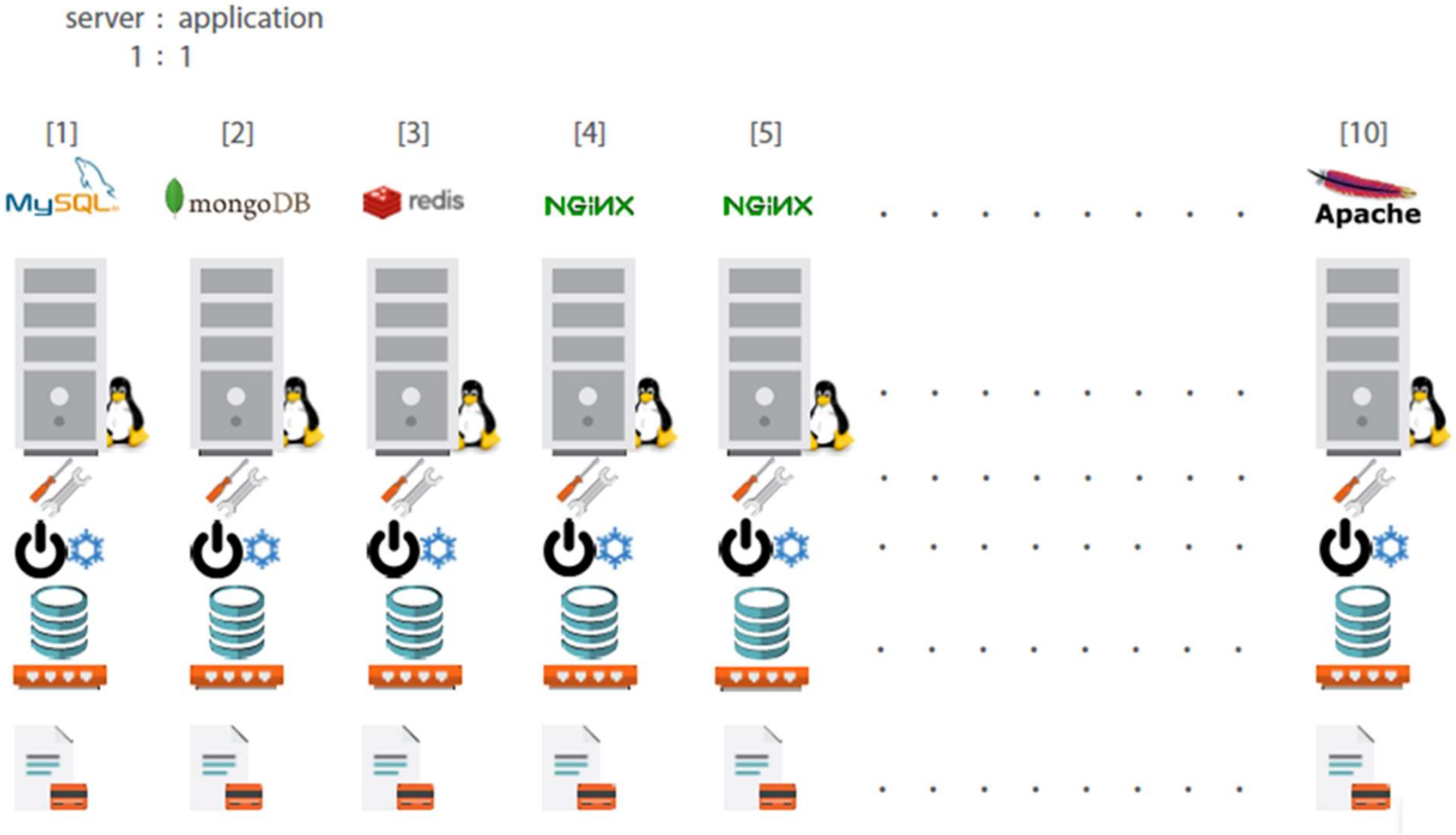
Docker

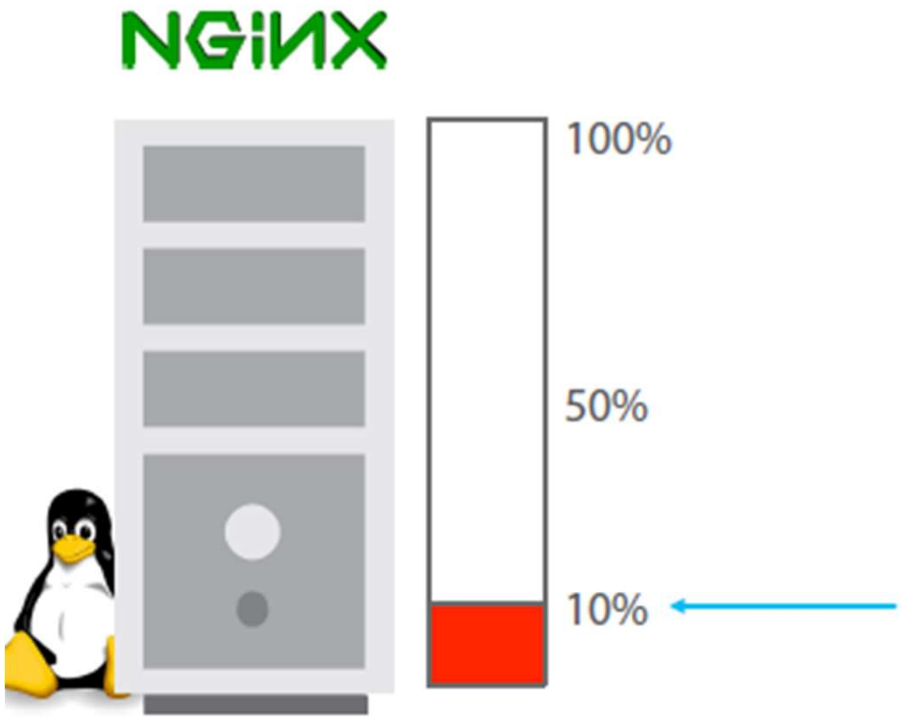
Docker



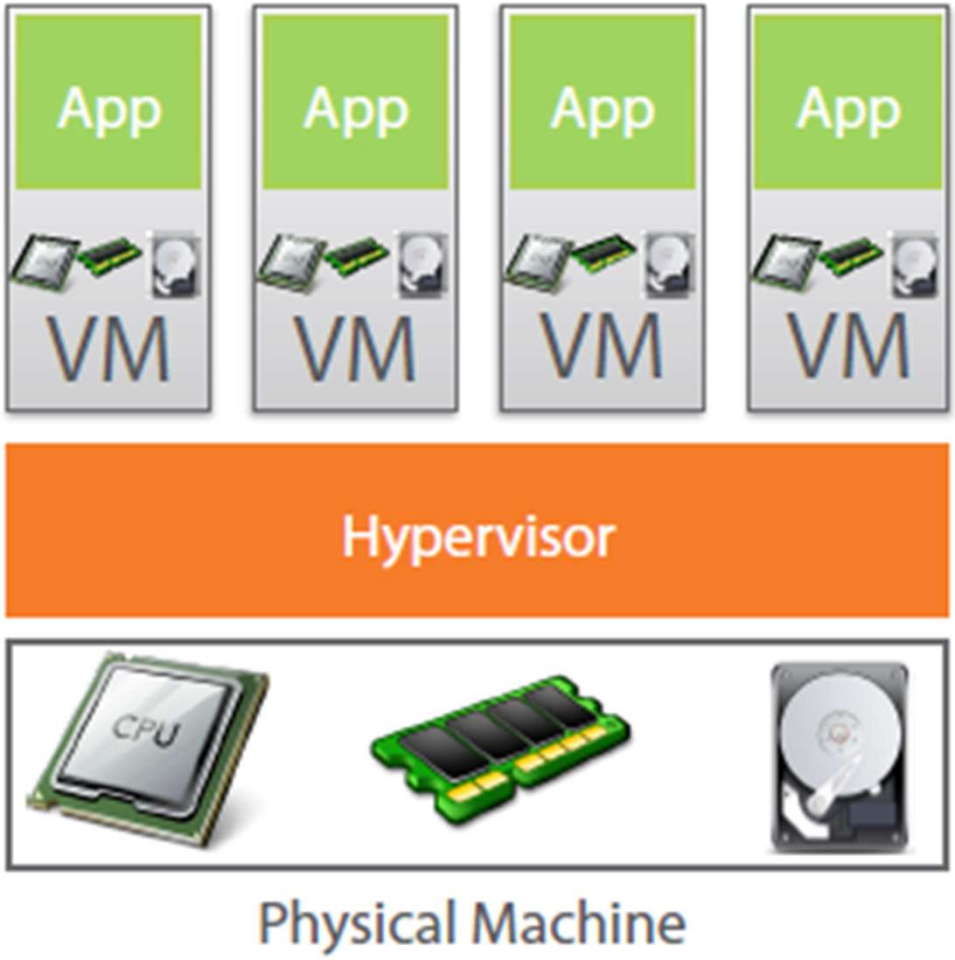
Traditional Deployment Architecture



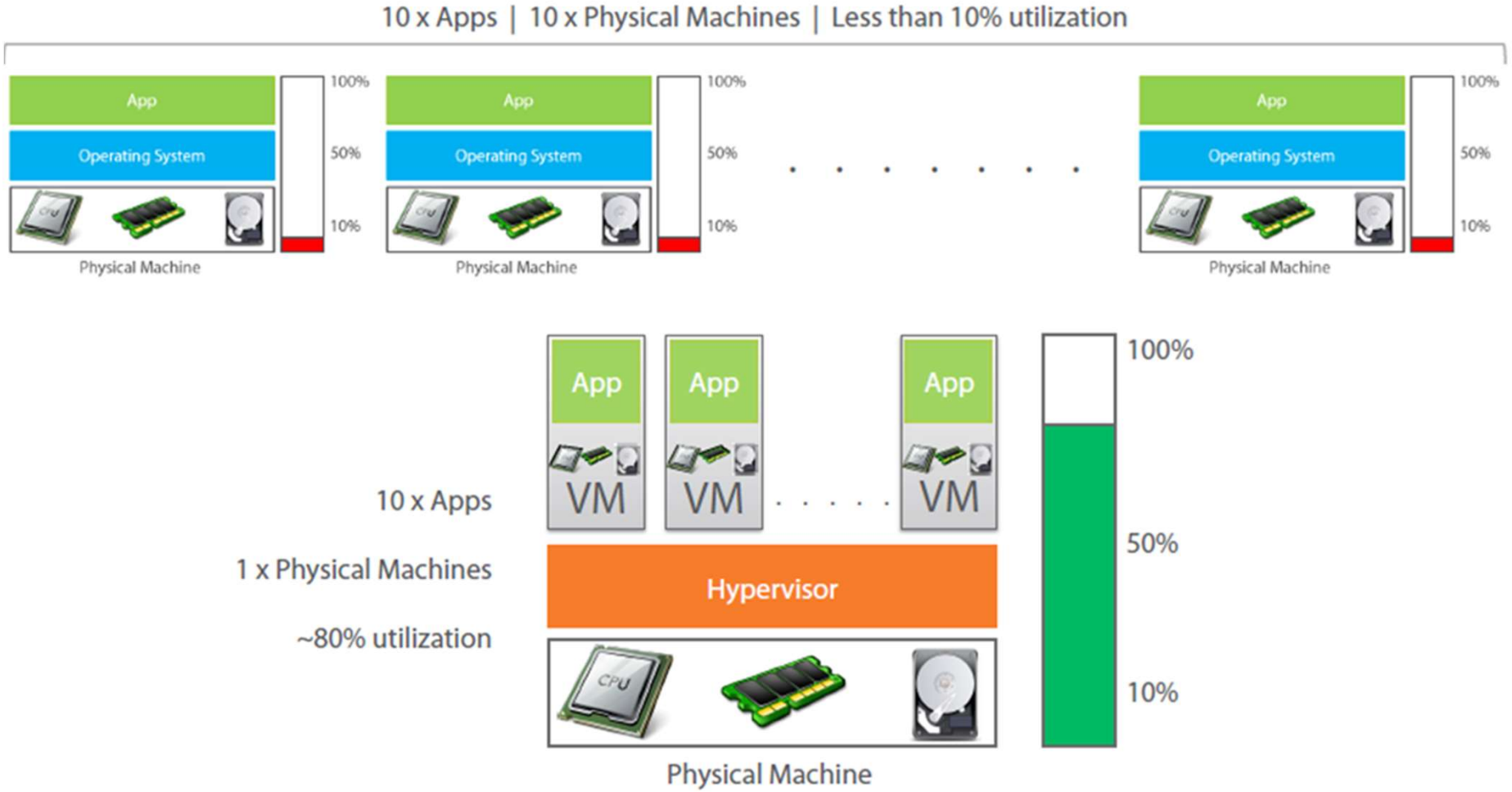
Less Utilization in Traditional Architecture



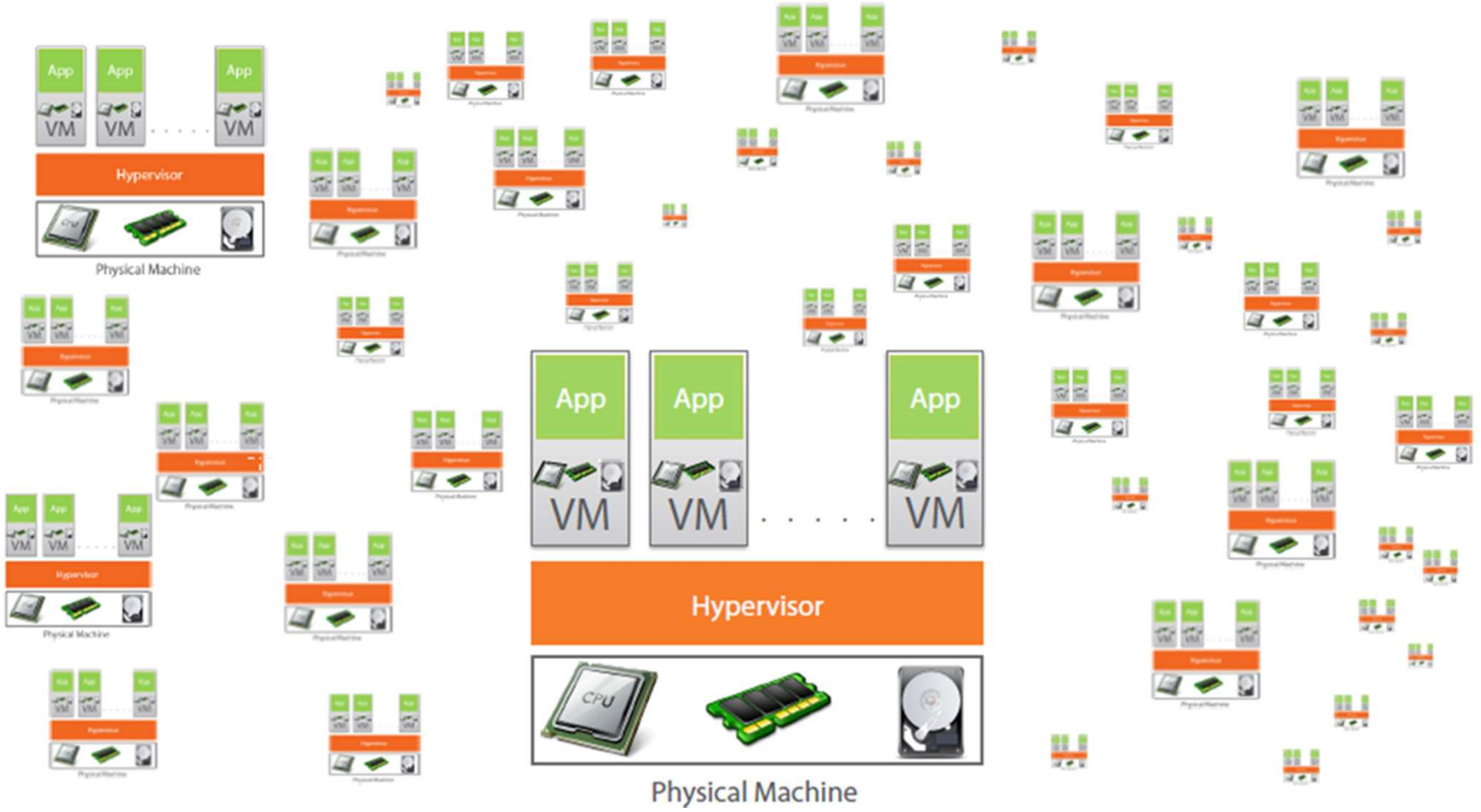
Virtual Machine to the Rescue



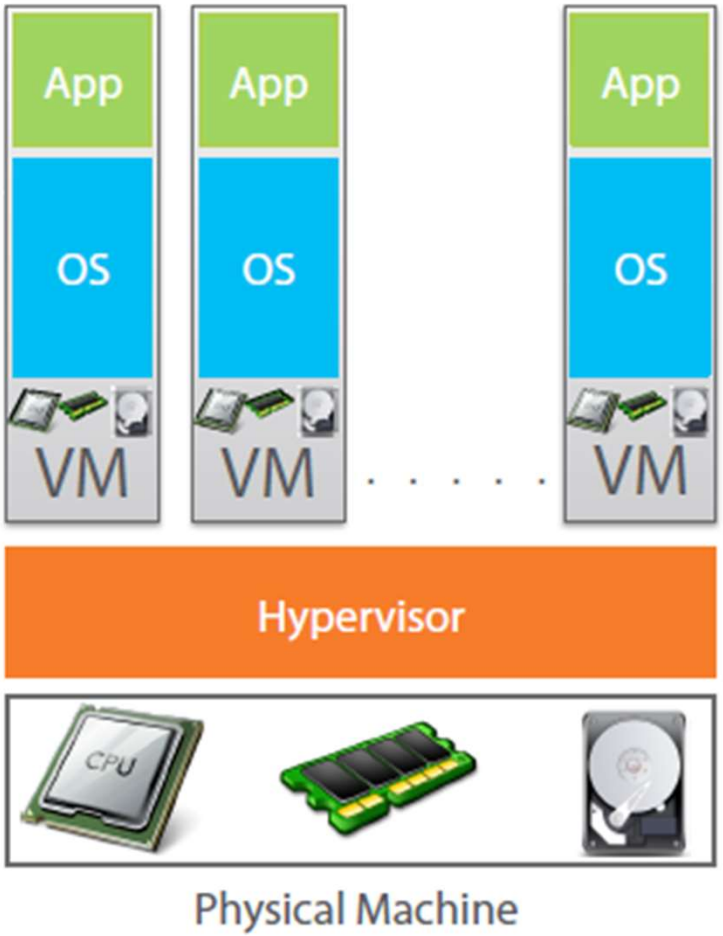
Virtual Machine provides better utilization



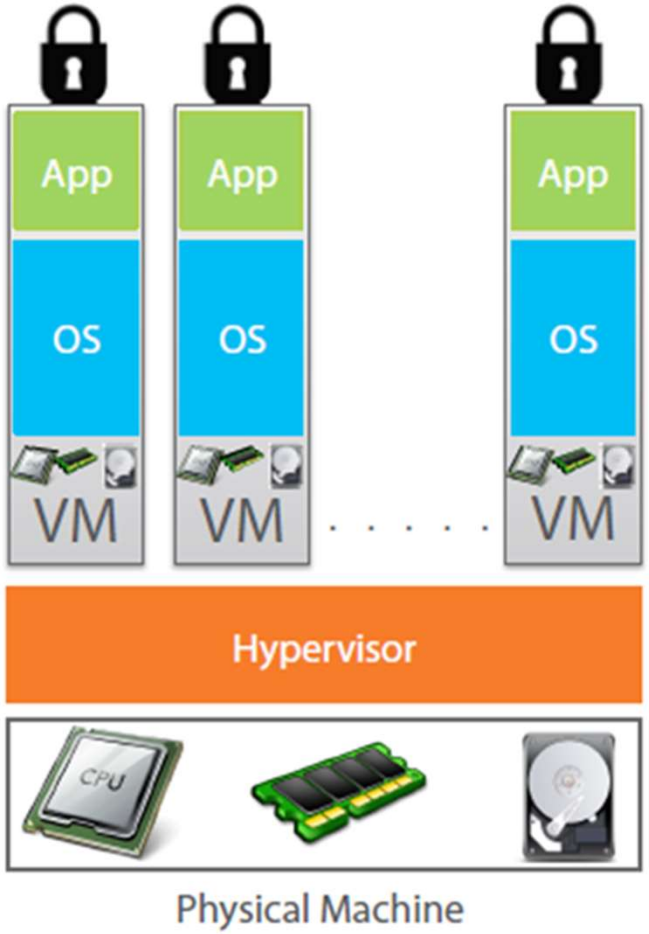
But Virtual Machine increases Licensing Cost



Each VM needs a separate OS



OS takes most of the Resources

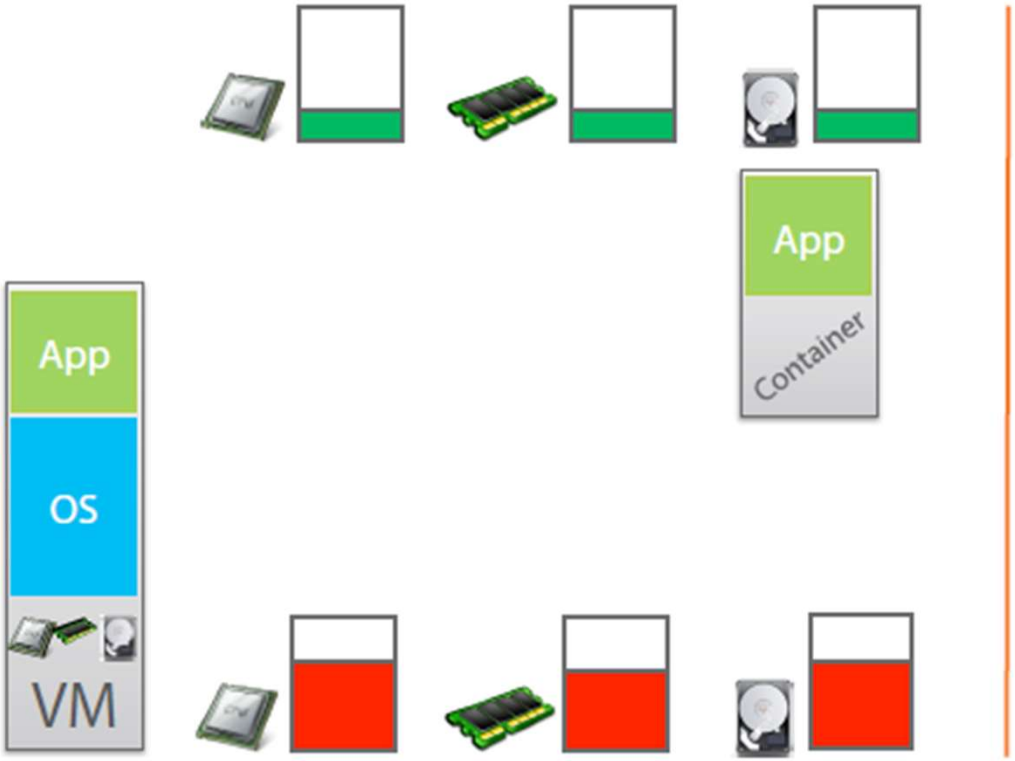


Why use separate OS for each App?

Containerization

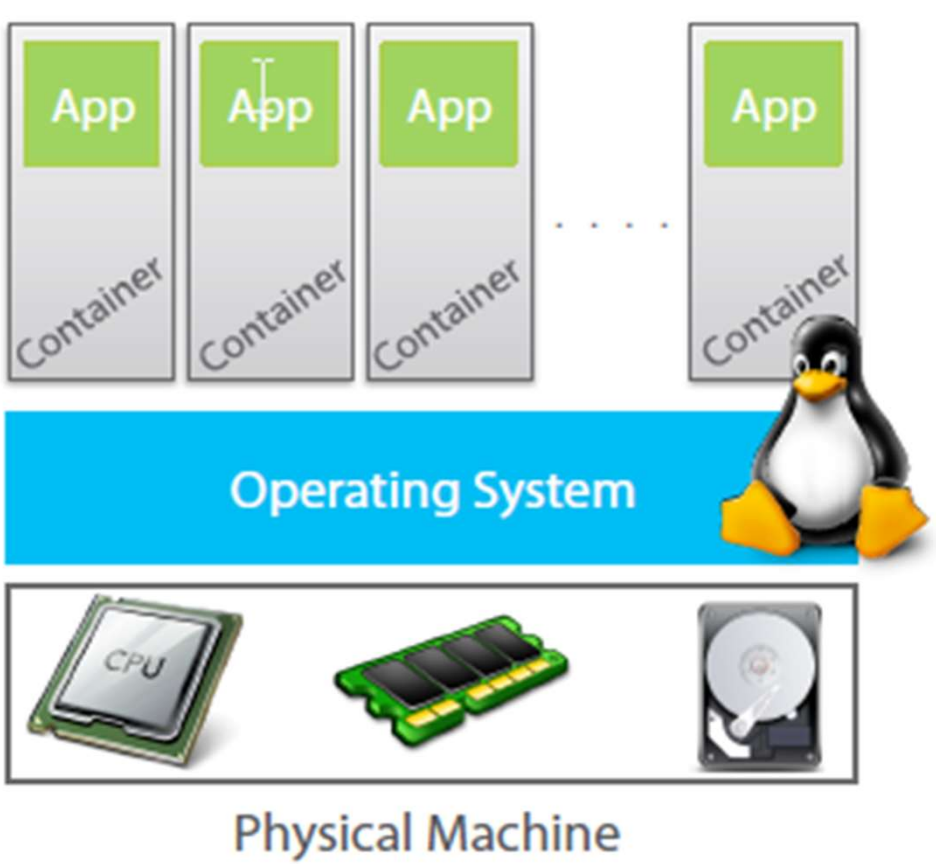
- Encapsulation of an application and its required environment.
- The process of packaging an application along with its required libraries, frameworks, and configuration files together so that it can be run in various computing environments efficiently.

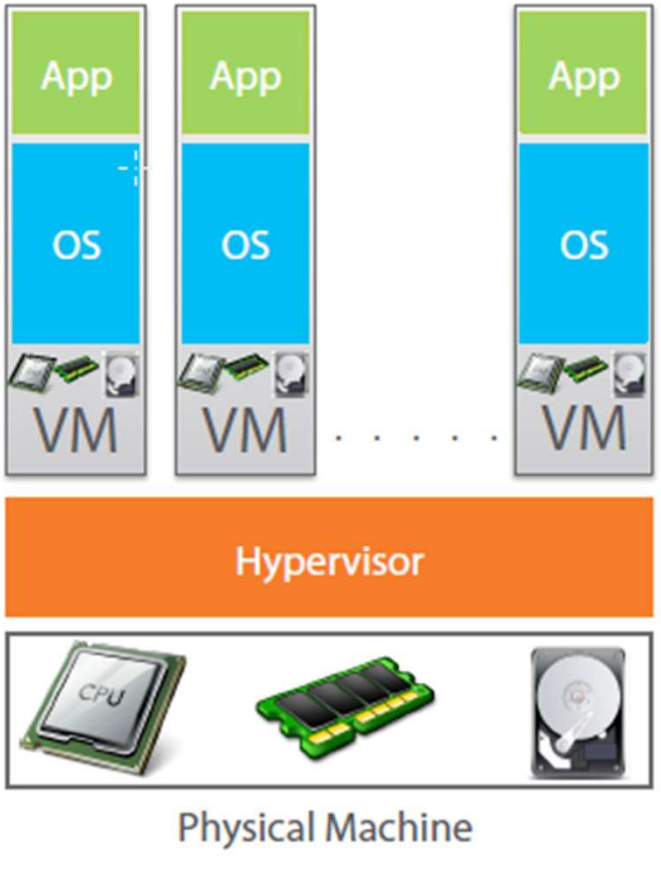
Containers to the Rescue



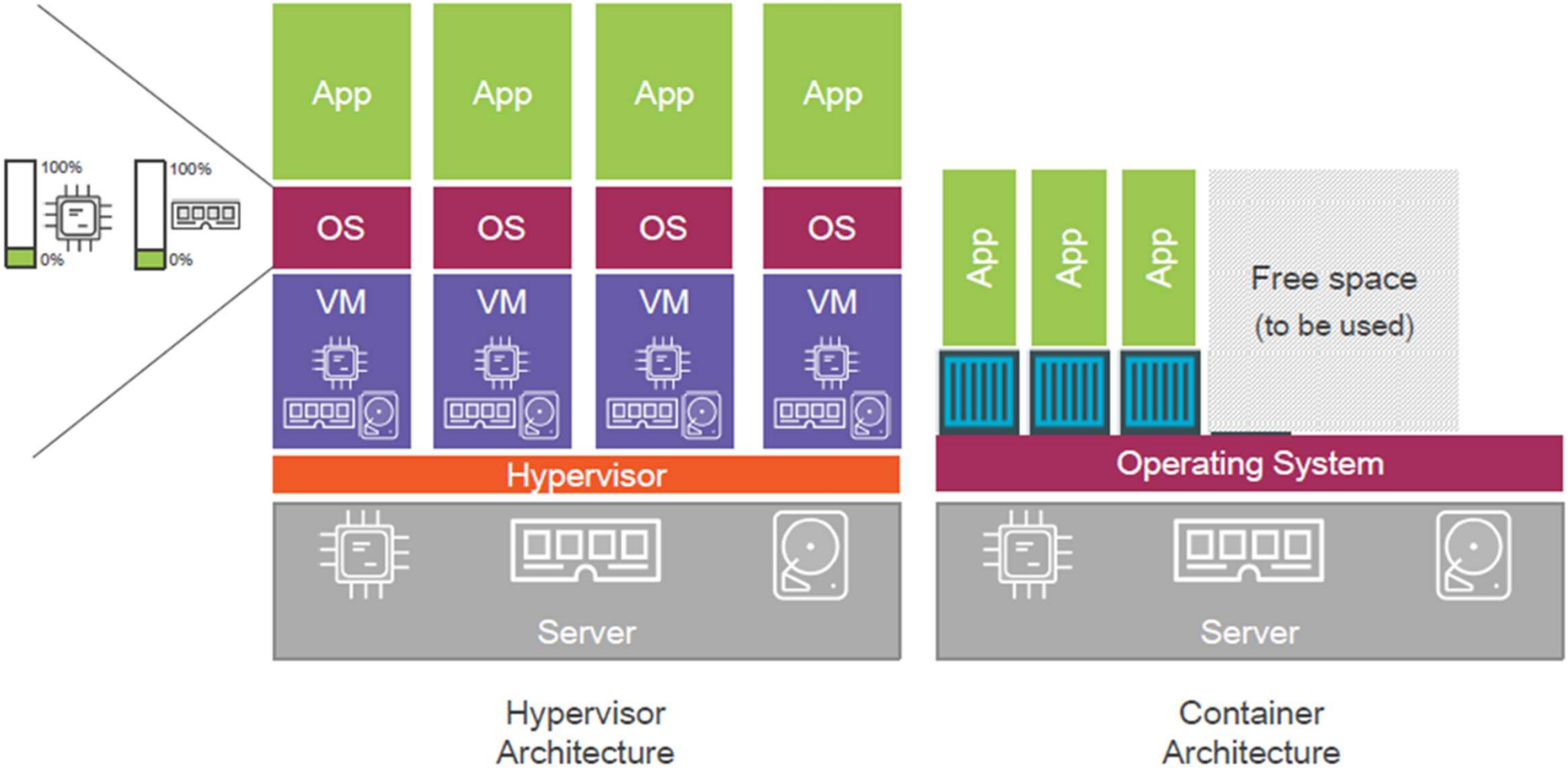
Containers are more lightweight than Virtual Machines

Containers vs VM

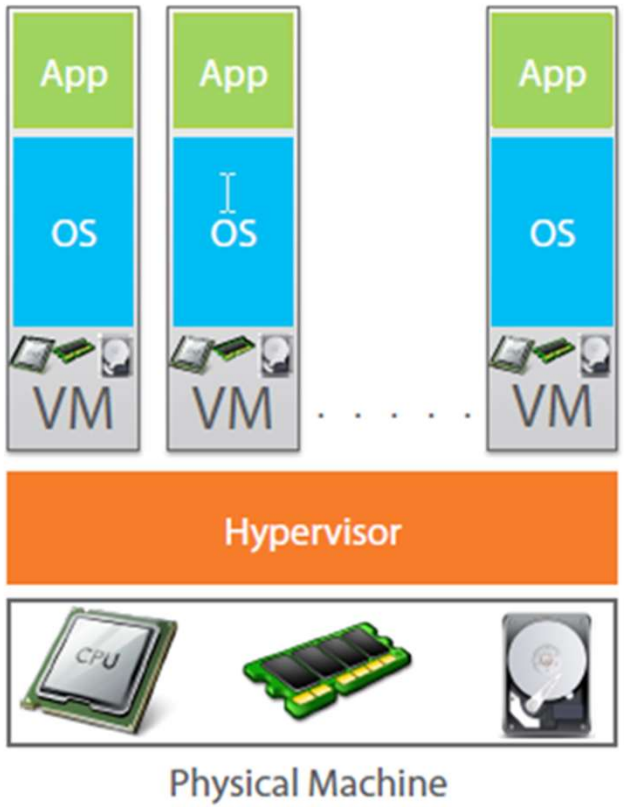


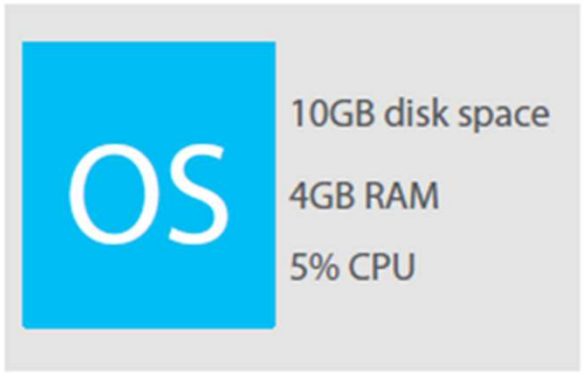


Containers vs VM



OS takes more resources and Licensing cost





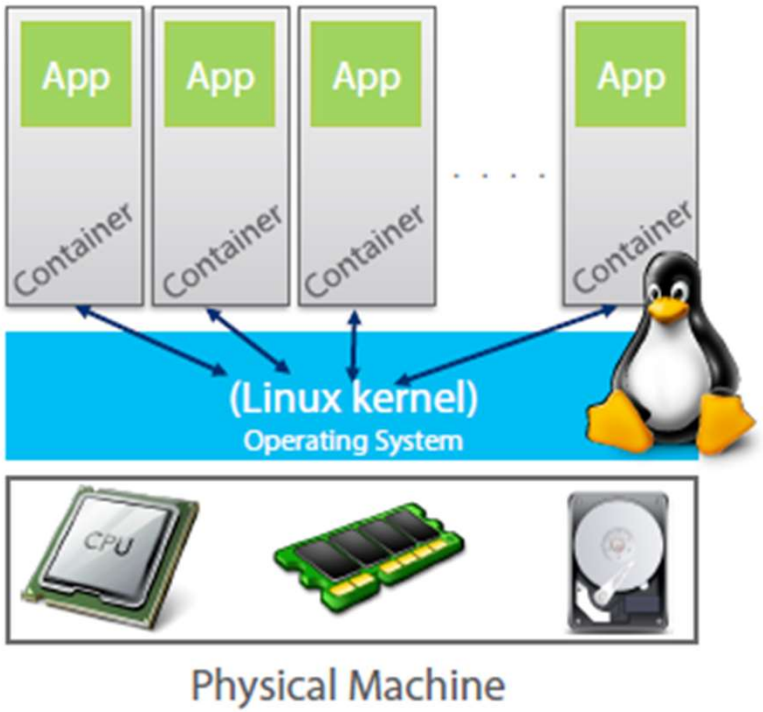
x 10 =

100GB disk space

40GB RAM

50% CPU

Containers takes less resources



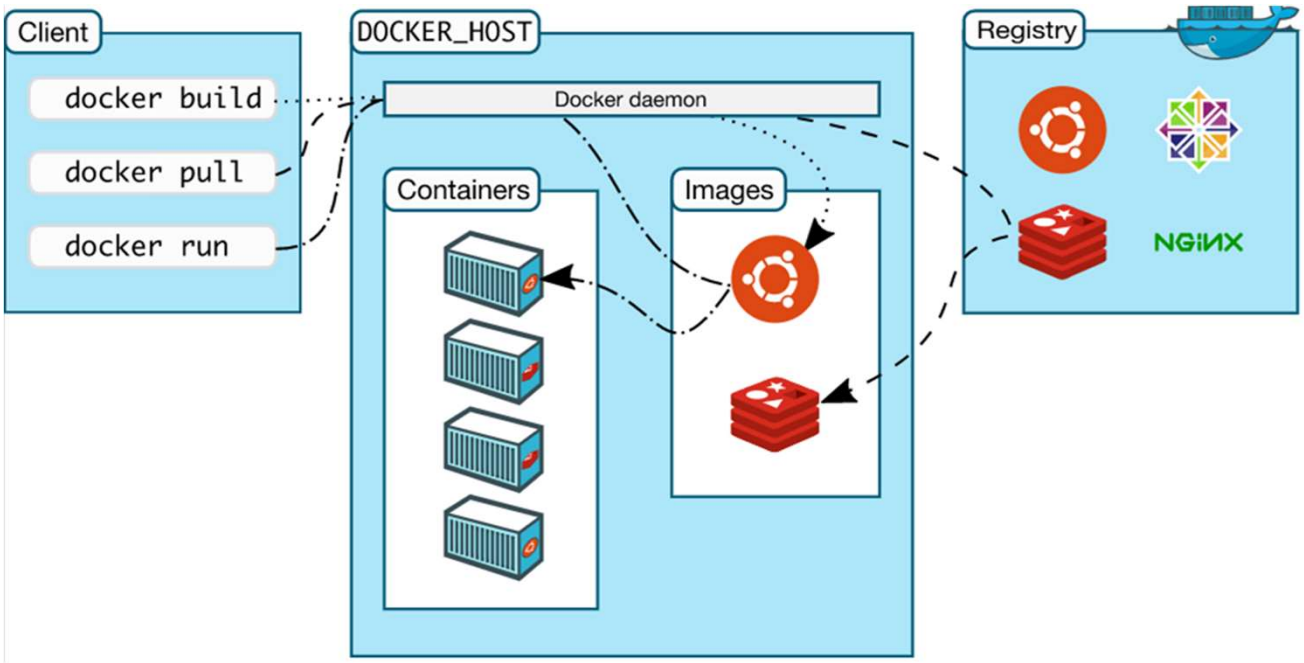
Containers consume less CPU, RAM and disk resource than Virtual Machines

What is Docker?

- Docker is an open-source project
  - that automates the deployment of applications inside software containers,
  - by providing an additional layer of abstraction and
  - automation of operating system–level virtualization on Linux.

Practical

Docker Architecture



- Docker uses a client-server architecture.
- Docker client talks to the Docker daemon
- The Docker client and daemon can run on the same system, or can connect a client to a remote Docker daemon.
- The Docker client and daemon communicate using a REST API

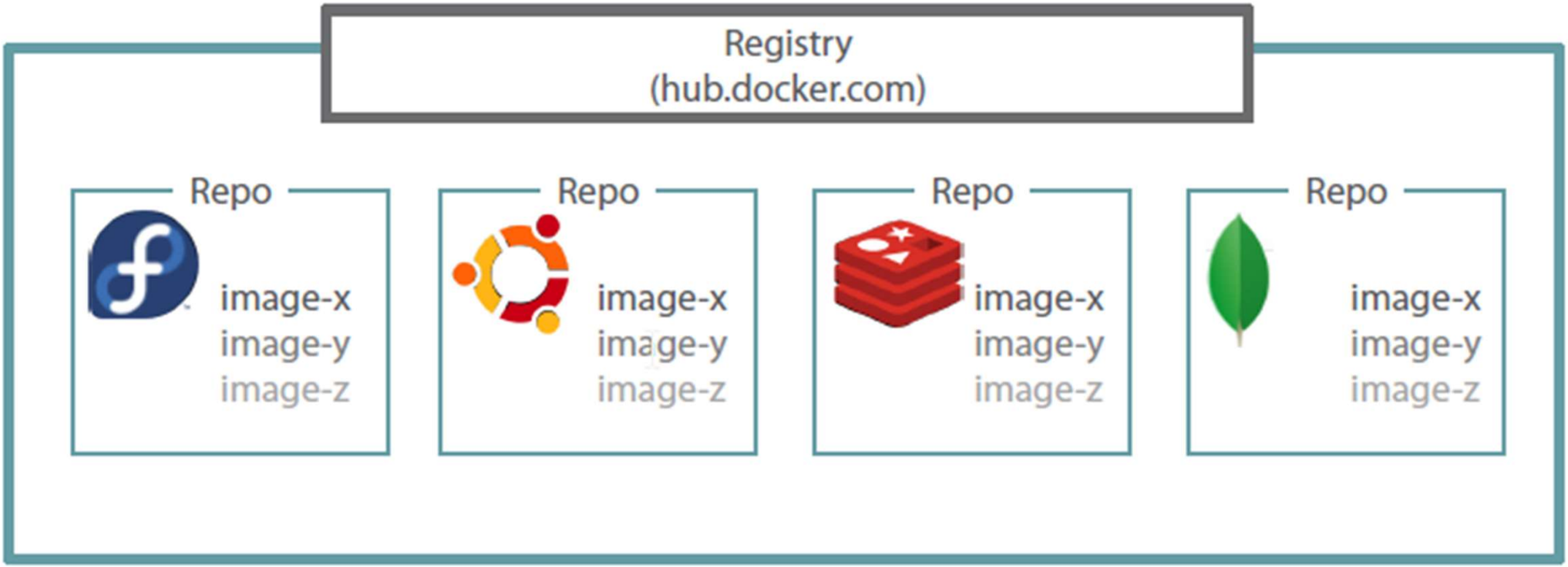
Image

- Persisted snapshot that can be run
- Common Docker Commands:
  - images: List all local images
  - run: Create a container from an image and execute a command in it
  - tag: Tag an image
  - pull: Download image from repository
  - rmi: Delete a local image

Container

- Runnable instance of an image
- Common Docker Commands
  - ps: List all running containers
  - ps –a: List all containers (incl. stopped)
  - top: Display processes of a container
  - start: Start a stopped container
  - stop: Stop a running container
  - pause: Pause all processes within a container
  - rm: Delete a container
  - commit: Create an image from a container

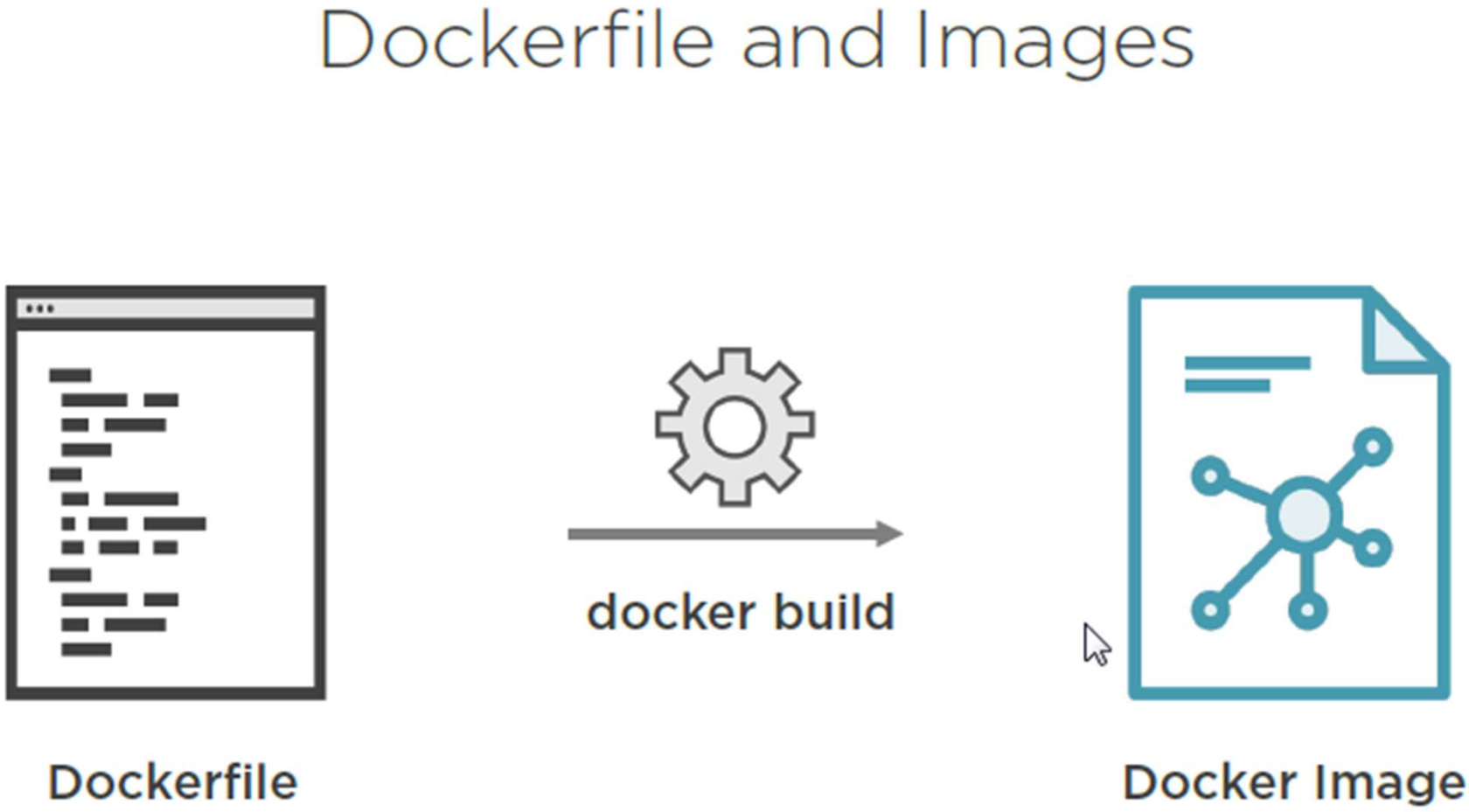
Docker Registry



Create Dockerized Application

- We can dockerize our application using dockerfile
  - Dockerfile Create images automatically using a build script: «Dockerfile»
  - It Can be versioned in a version control system like Git
  - Docker Hub can automatically build images based on dockerfiles on Github
- This is a basic Dockerfile we need to dockerize a node application
  - FROM node:4-onbuild
  - RUN mkdir /app
  - COPY . /app/
  - WORKDIR /app
  - RUN npm install
  - EXPOSE 8234
  - CMD ["npm", "start"]

Dockerfile


Dockerfile and Images
docker build
Dockerfile
Docker Image

Dockerfile Template

Build Image

- Now once we have our Dockerfile ready lets build an image out of it.
- Assuming you all have docker installed on your system lets follow some simple steps:-
  - Navigate to directory containing Dockerfile.
  - Run the following command on your terminal:-
    - docker build -t myimage .
- docker images
- docker run -p 8234:8234 'your image name'


thanks