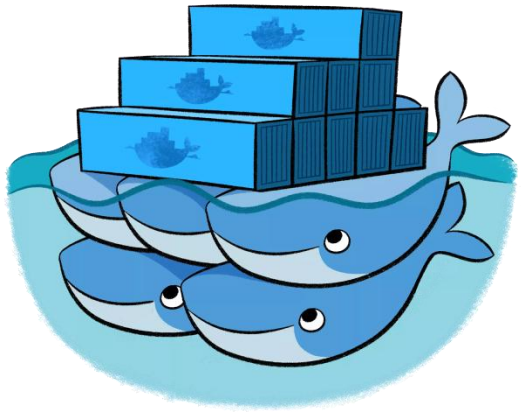


Introduction to Docker

By

Anand Kulkarni

Contents

Sr. No.	Topic
1.	Docker in DevOps
2.	What is Docker?
3.	How Docker works?
4.	Why Docker?
5.	Docker Architecture
6.	Docker Benefits
7.	Docker installation on windows 10
8.	Docker commands
9.	Useful links

Docker in DevOps

There are 4 stages in software development life cycle:

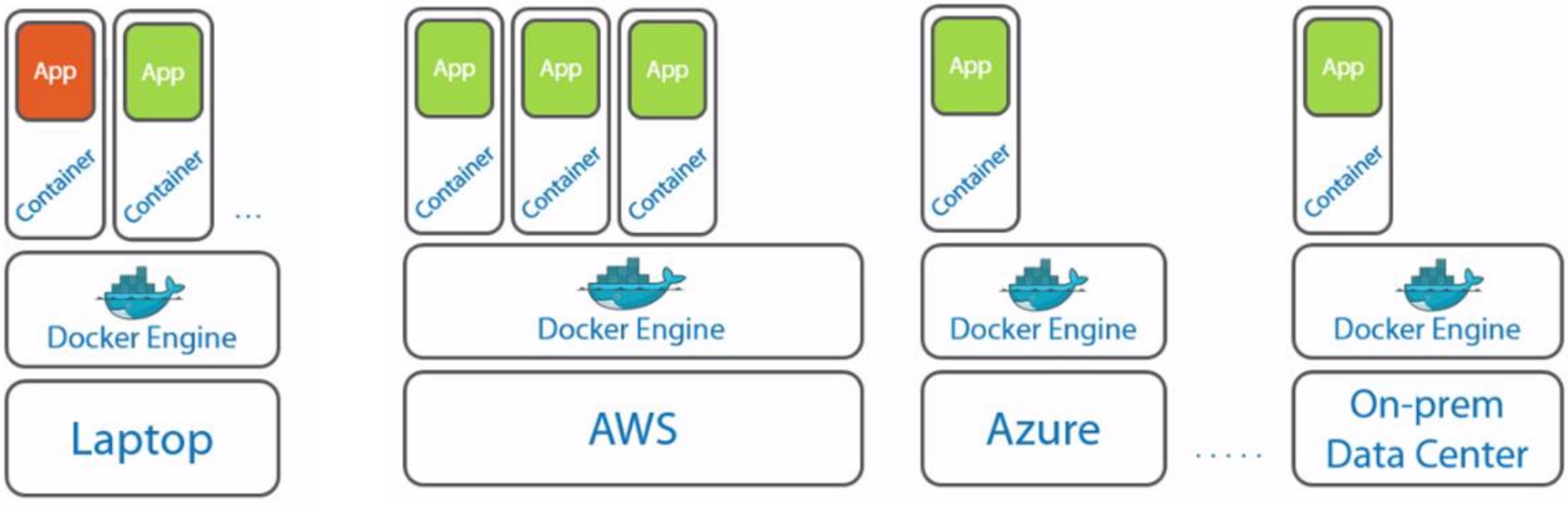
1. Design
2. Development
3. Deployment
4. Testing/Release

Docker is applicable at deployment stage.

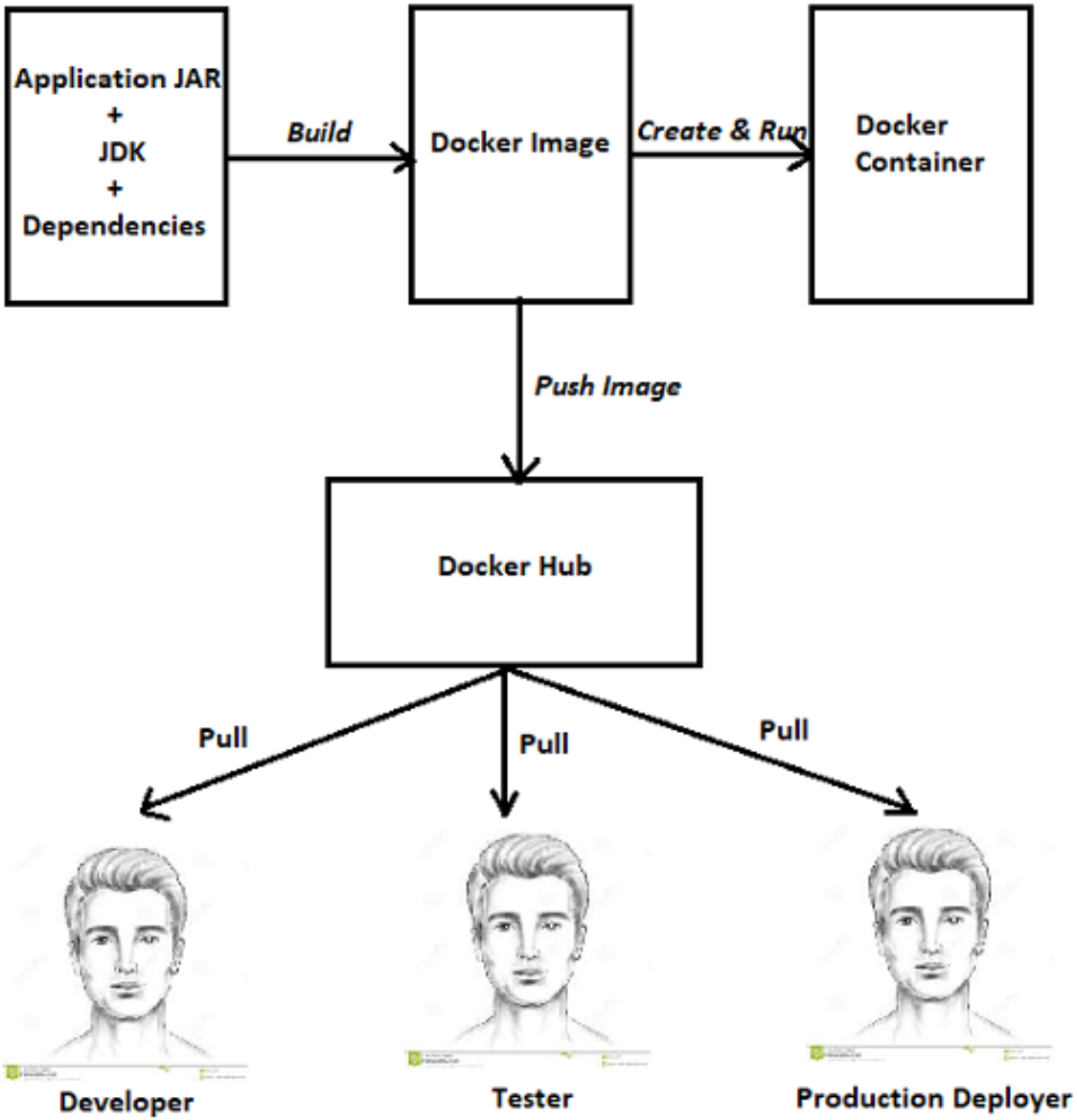
What is Docker?

- Allows you to package an application and all its dependencies into a SINGLE STANDARDIZED unit.
- Docker Image = Code + Runtime + System Tools
- As a result you can run this same image, regardless of the environment.

What is Docker?



How Docker works?



How Docker works?

- Developer defines all his applications and dependencies into a Dockerfile.
- Dockerfile is used to create a docker image.
- Docker images consists of all required applications along with its dependencies.
- Using Docker image, we can create Docker containers.
- Docker container is a runtime instance of Docker image.

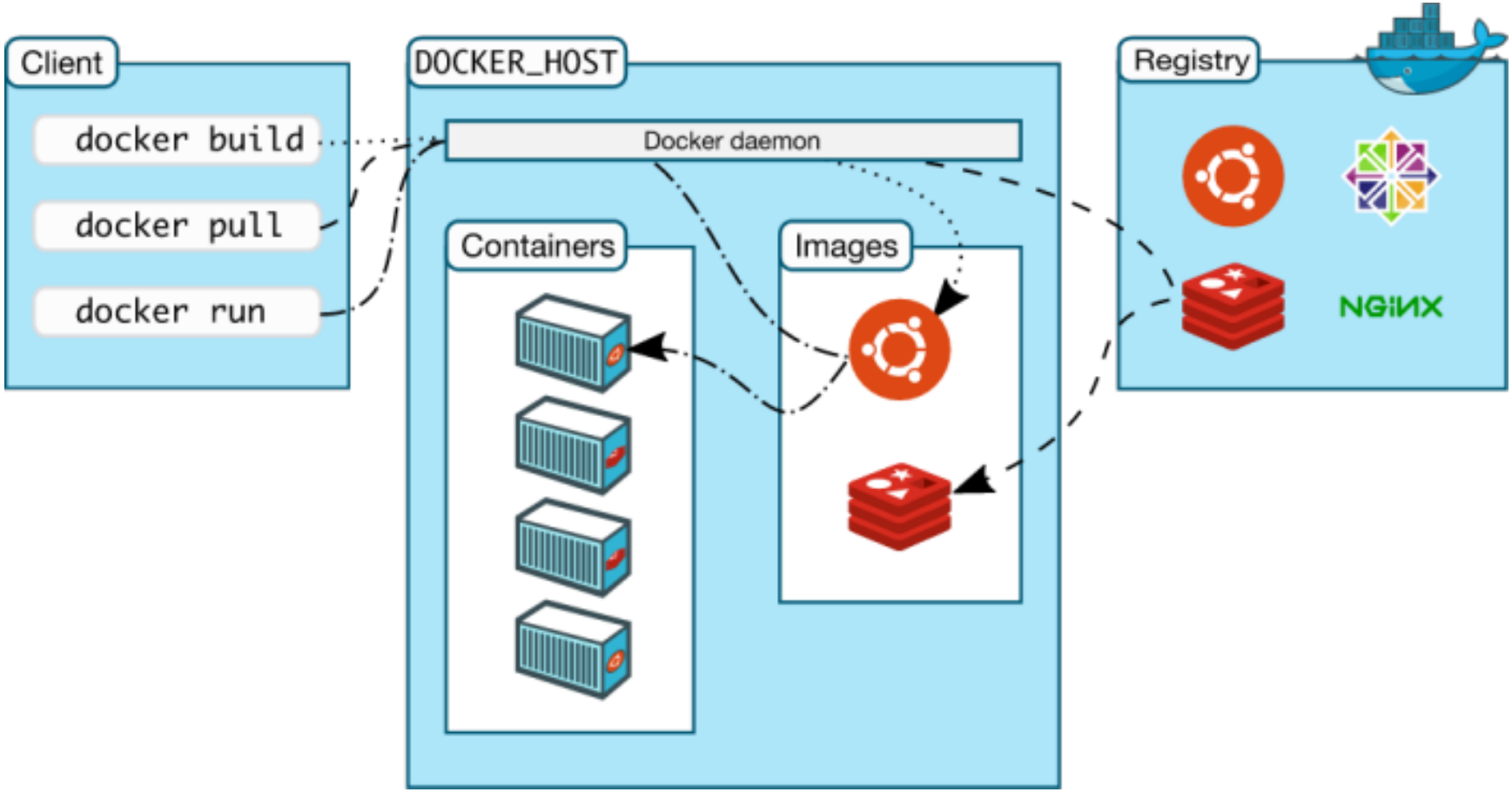
How Docker works?

- These docker images can be stored into a online cloud repository called 'Docker Hub'(https://hub.docker.com). Docker images can be stored into other repositories too.
- Docker images can be pulled into any docker container.

Why Docker?

- Docker makes the process of application deployment very easy and efficient and resolves a lot of issues related to deploying applications.
- Docker is the world's leading software container platform.
- Docker is a tool designed to make it easier to deploy and run applications by using containers. Containers allow developers to package their application with all parts it needs, such as libraries and other dependencies, and ship it all out as one package.

Docker Architecture



Docker Architecture

Docker architecture has following main components:

1) Docker Engine
2) Docker Client
3) Docker Registry
4) Docker Images
5) Docker Containers

Docker Engine

It is the core part of the whole Docker system. Docker Engine is an application which follows client-server architecture. It is installed on the host machine. There are three components in the Docker Engine:

Server: It is the docker daemon called dockerd. It can create and manage docker images. Containers, networks, etc.

Rest API: It is used to instruct docker daemon what to do.

Command Line Interface (CLI): It is a client which is used to enter docker commands.

Docker Client

Docker users can interact with Docker through a client. When any docker commands runs, the client sends them to dockerd daemon, which carries them out. Docker API is used by Docker commands. Docker client can communicate with more than one daemon.

Docker Registry

- Docker registry is the location where the Docker images are stored. It can be a public docker registry or a private docker registry. Docker Hub is the default place of docker images, its stores' public registry. You can also create and run your own private registry.
- When you execute docker pull or docker run commands, the required docker image is pulled from the configured registry. When you execute docker push command, the docker image is stored on the configured registry.

Docker Images

- Docker images are read-only templates with instructions to create a docker container.
- Docker image can be pulled from a Docker.
- You can create your own docker images also using a dockerfile.
- Docker image has a base layer which is read-only, and the top layer can be written. When you edit a dockerfile and rebuild it, only the modified part is rebuilt in the top layer.

Docker Containers

- After you run a docker image, it creates a docker container. All the applications and their environment run inside this container. You can use Docker API or CLI to start, stop, delete a docker container.

Docker Benefits

1) Build application only once: An application inside a container can run on any system that has docker installed. So there is no need to build/configure your application on different platforms.

2) More sleep less worry: With docker you test your application inside a container and ship it inside a container. This means the environment in which you test is identical to the one on which the application will run on production.

3) Portability: Docker containers can run on any platform. They can run on your local system, Amazon ec2, Google cloud platform, Rackspace server, VirtualBox etc.

Docker Benefits

4) Version control: Like GIT, Docker has in-built version control system. Docker containers work just like GIT repositories, allowing you to commit changes to Docker images and version control them.

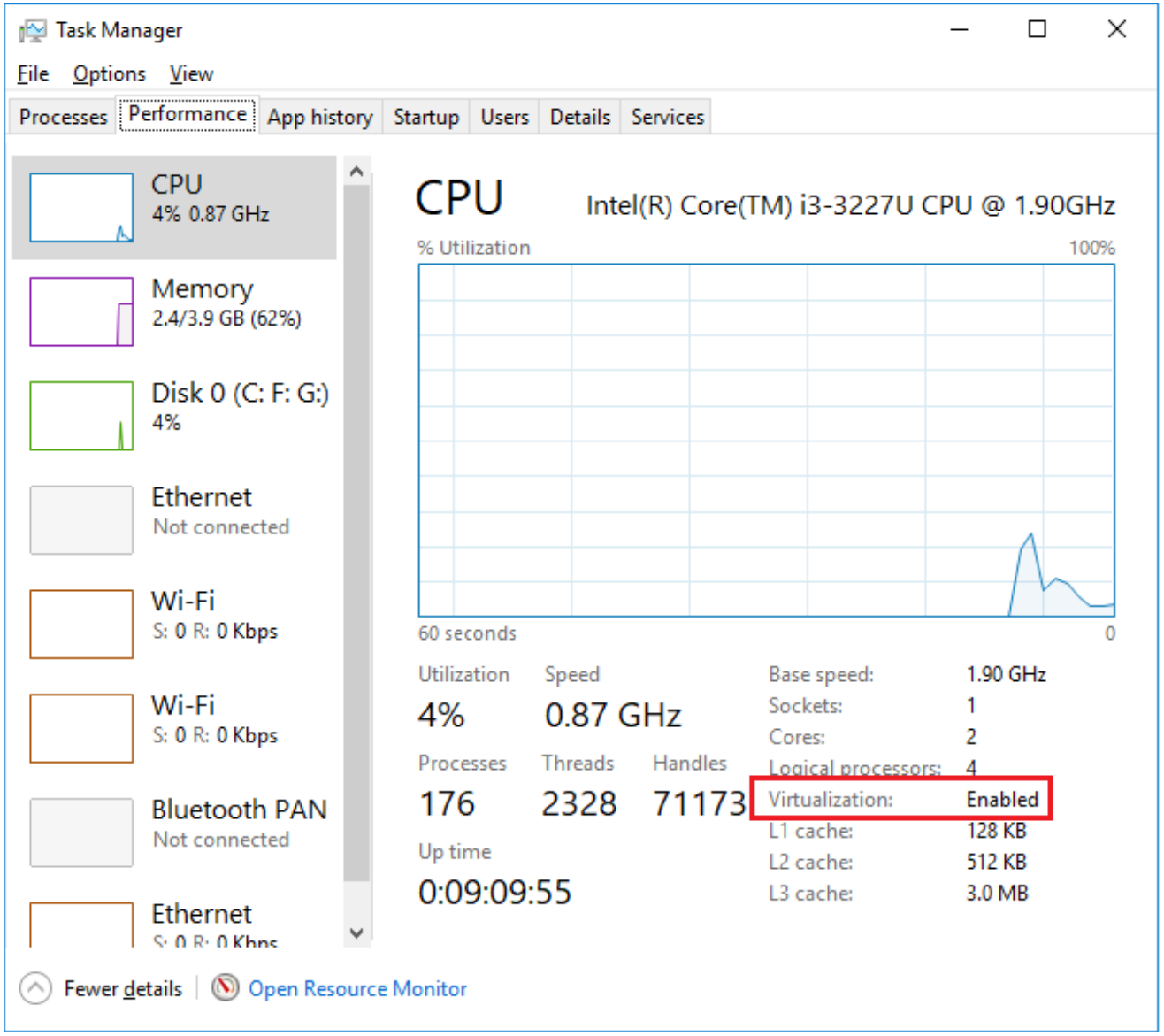
5) Isolation: Every application in docker works in perfect isolation using docker container. And it never interferes with other application running on the same system.

6) Productivity: Docker allows faster & more efficient deployments without worrying about running your application on different platforms. It increases overall productivity.

Docker Installation on Windows 10

- Set ‘Virtualization Technology’ = ‘Enabled’ inside BIOS settings.
- Go to Control Panel > Windows features > Click on ‘Hyper-V’ option. Then restart the system.
- Make sure you get ‘Virtualization’ enabled into Task Manager. You may need to restart the system 2/3 times to get Enabled value for Virtualization.

Docker Installation on Windows 10



Docker Installation on Windows 10

- Download the stable version of Docker Desktop for Windows at https://hub.docker.com/editions/community/docker-ce-desktop-windows/
- Install the Docker desktop & restart the system.
- Start the Docker desktop service.
- In order to confirm, open the console & run ‘docker version’ command. It should display your ‘Server’ details.
- Note that Docker desktop takes huge memory & hence if not required, you should shutdown Docker desktop service along with Hyper-V services.

Docker Commands

- Docker command gives instruction to Docker Daemon to perform certain action & they can be fired using console.
- You can find complete manual of docker comands at https://docs.docker.com/engine/reference/commandline/docker/
- You can play with docker commands without installing it on your machine. For this access https://labs.play-with-docker.com
- Docker commands are broadly divided into 4 categories:

1) Basic
2) Images
3) Containers
4) System

Basic commands

Sr. No.	Command	Description	Usage
1.	Version	It shows the Docker version information	>docker version
2.	Info	Displays docker system level detailed information	>docker info
3.	Login	Allows to login into docker registry.	>docker login

Images commands

Sr. No.	Command	Description	Usage
1.	images	It shows the list of all images.	>docker images –a
2.	pull	It pulls an image from docker repository.	>docker pull ubuntu
3.	rmi	Removes one or more images	>docker rmi fd484f19954f

Container commands

Sr. No.	Command	Description	Usage
1.	ps	It lists down all containers	>docker ps -a
2.	run	It first creates a container & then runs it.	>docker run –p 9999:8080 anand0204/spring-boot-order-docker
3.	start	Starts one or more containers	>docker start container_id
4.	stop	Stops one or more containers	>docker stop container_id

System commands

Sr. No.	Command	Description	Usage
1.	stats	Display a live stream of container(s) resource usage statistics.	>docker stats
2.	system df	Shows the docker’s disk usage.	>docker system df
3.	system prune	Removes unused data.	>docker system prune

Useful links

- Docker Hub: https://hub.docker.com/
- Playing with docker commands: https://labs.play-with-docker.com
- Download Docker for windows 10:

  https://hub.docker.com/editions/community/docker-ce-desktop-windows/
- Docker commands:

  https://docs.docker.com/engine/reference/commandline/docker/

Thank You