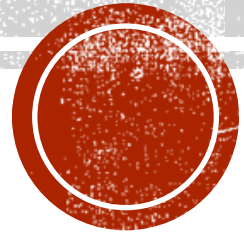
# DOCKER ARCHITECTURE



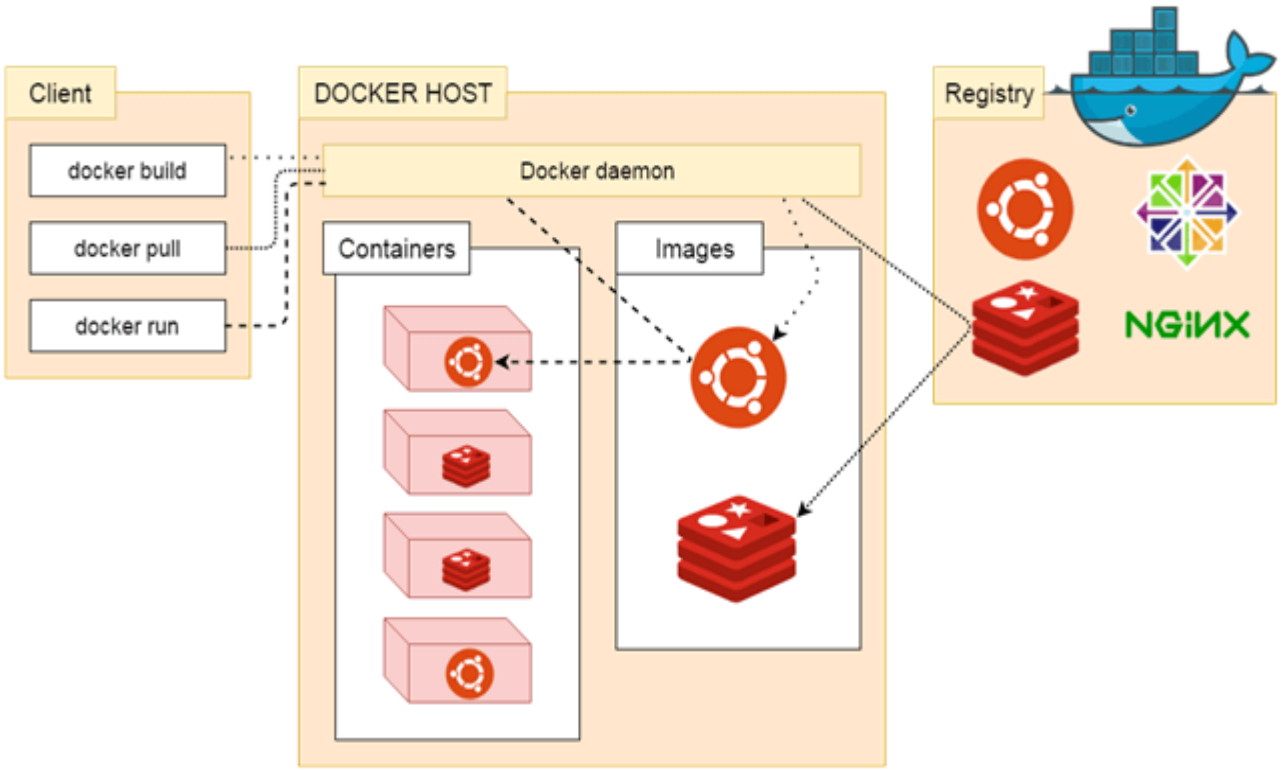
22AIE305 – Introduction to Cloud Computing 2-0-3-3

# WHAT IS DOCKER DAEMON?

- Docker daemon runs on the host operating system.
- It is responsible for running containers to manage docker services.
- Docker daemon communicates with other daemons.
- It offers various Docker objects such as images, containers, networking, and storage.

# DOCKER ARCHITECTURE

- Docker follows Client-Server architecture, which includes the three main components that are **Docker Client**, **Docker Host**, and **Docker Registry**.



# *DOCKER ARCHITECTURE*

### *1. Docker Client*

- Docker client uses **commands** and **REST APIs** to communicate with the Docker Daemon (Server).
- When a client runs any docker command on the docker client terminal, the client terminal sends these docker commands to the Docker daemon.
- Docker daemon receives these commands from the docker client in the form of command and REST API's request.
- Docker Client uses Command Line Interface (CLI) to run the following commands –

  - docker build
  - docker pull
  - docker run

# *DOCKER ARCHITECTURE*

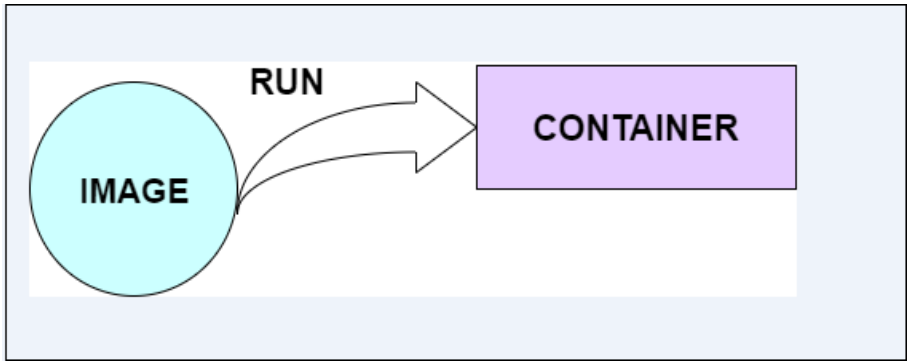
### *2. Docker Host*

- Docker Host is used to providing an environment to execute and run applications. It contains the docker daemon, images, containers, networks, and storage.

### *3. Docker Registry*

- Docker Registry manages and stores the Docker images.
- There are two types of registries in the Docker -
- **Public Registry -** Public Registry is also called a **Docker hub**.
- **Private Registry -** It is used to share images within the enterprise.

# DOCKER ARCHITECTURE




## Docker Objects

- There are the following Docker Objects -

### *Docker Images*

- Docker images are the **read-only binary templates** used to create Docker Containers. It uses a private container registry to share container images within the enterprise and also uses a public container registry to share container images within the whole world. Metadata is also used by docket images to describe the container's abilities.

### *Docker Containers*

- Containers are the structural units of Docker, which is used to hold the entire package that is needed to run the application. The advantage of containers is that it requires very less resources.
- In other words, we can say that the image is a template, and the container is a copy of that template.

# *DOCKER ARCHITECTURE*

***Docker Networking***

- Using Docker Networking, an isolated package can be communicated. Docker contains the following network drivers -

- **Bridge -** Bridge is a default network driver for the container. It is used when multiple dockers communicate with the same docker host.
- **Host -** It is used when we don't need network isolation between the container and the host.
- **None -** It disables all the networking.
- **Overlay -** Overlay offers Swarm services to communicate with each other. It enables containers to run on different docker hosts.
- **Macvlan -** Macvlan is used when we want to assign MAC addresses to the containers.

# *DOCKER ARCHITECTURE*

***Docker Storage***

- Docker Storage is used to store data on the container.
- Docker offers the following options for Storage -

- **Data Volume -** Data Volume provides the ability to create persistence storage. It also allows us to name volumes, list volumes, and containers associated with the volumes.
- **Directory Mounts -** It is one of the best options for docker storage. It mounts a host's directory into a container.
- **Storage Plugins -** It provides the ability to connect to external storage platforms.