 **DOCKER TUTORIAL – 1**

**What you will Learn in this Tutorial:**

* **What is Docker**
* **Concepts**
* **Container vs VMs**
* **Understanding of Docker Architecture**
* **Docker File – Docker Image**
* **Docker use in Real Life in Organisation.**
* **Overview of Docker Volumes**
* **Overview of Docker Hub**
* **Create & Containerize your first (Lab)**

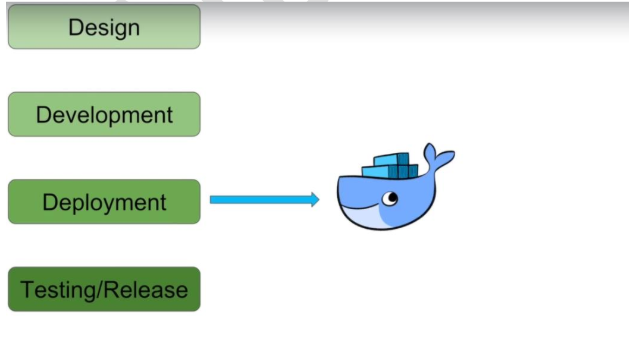
**Docker**

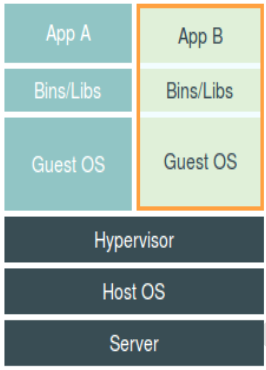
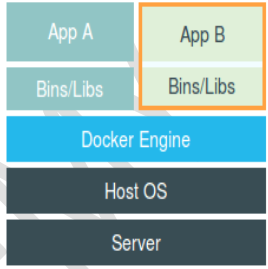
**Container : A container like a virtual machine**

**Docker : Docker is a tool to create those virtual machines**

**What is Docker??**

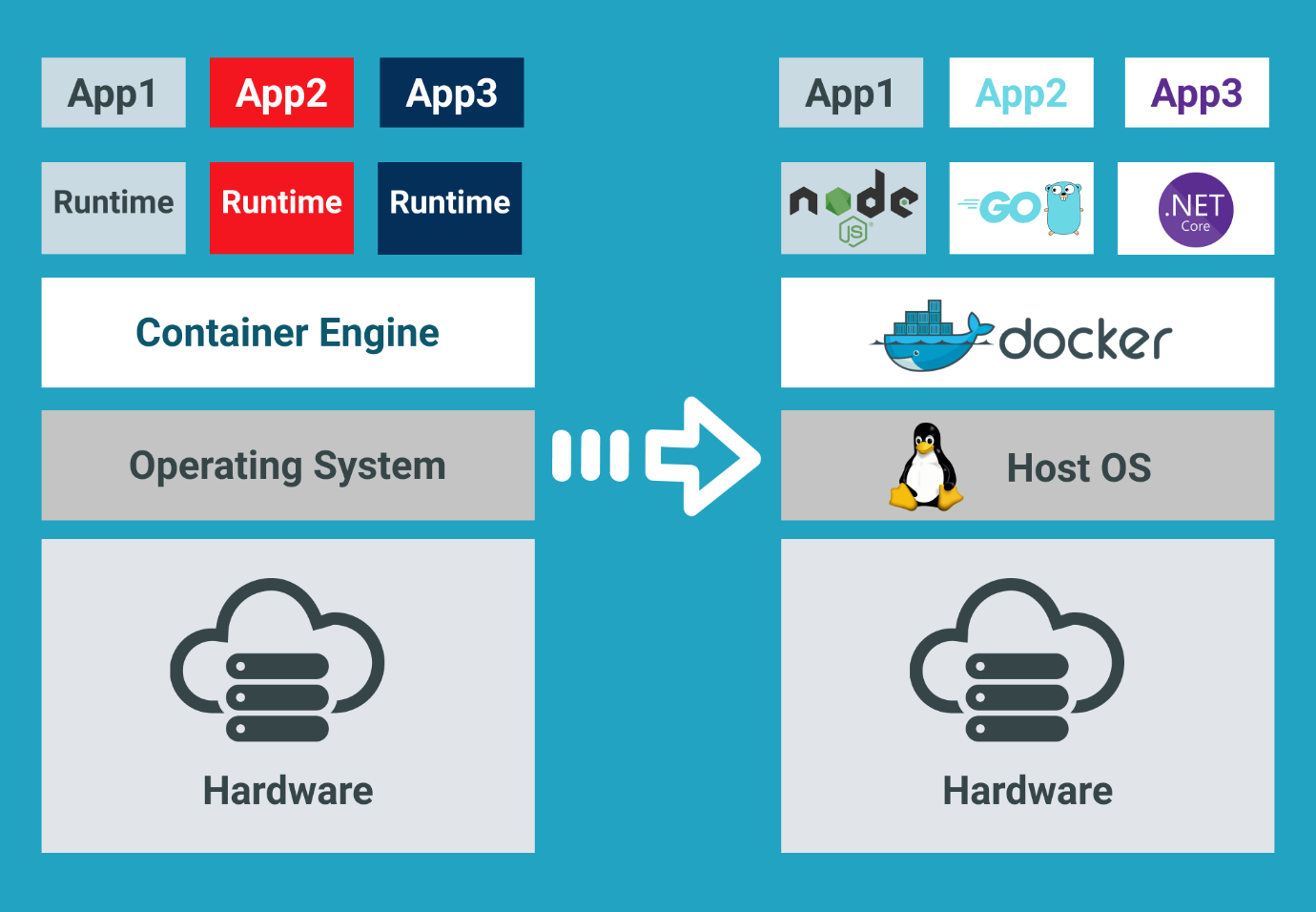
* **Docker is a tool that performs operating-system-level virtualization, also known as "containerization".**
* **It was first released in 2013 and was developed by Docker, Inc**
* **Docker is tool used to create virtual machines called "containers".Took from shipping containers.**
* **Docker is a tool designed to make containers in which we can deploy any type of applications easily.**
* **Docker uses union file system (layered)**
* **Docker performs os-level virtualization**

****

** **

**VMs Docker**

* **Docker is a set of platforms as a service (PaaS) products that use Operating system-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries, and configuration files; they can communicate with each other through well-defined channels.**



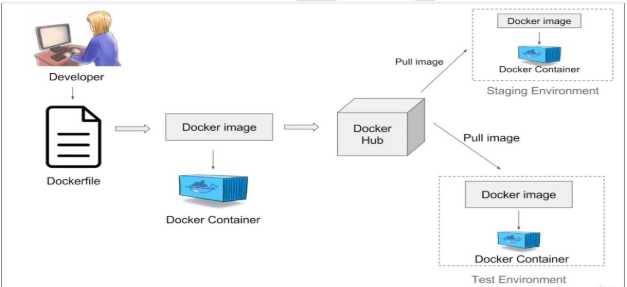
**Docker benefits**

* **Containerization (OS level virtualization)(No need guest OS)**
* **No pre-allocation of RAM**
* **Can replicate same environment**
* **Less cost**
* **Less weight (MB’s in size)**
* **Fast to fire up**
* **Can run on physical/virtual/cloud**
* **Can re-use(same image)**
* **Can create machines in less time.**

**Docker components**

* **Docker image: Contains OS(very small)(almost negligible) + softwares**
* **Docker Container: Container like a machine which is created from Docker image.**
* **Docker file: Describes steps to create a docker image.**
* **Docker hub/registry: Stores all docker images publicly.**
* **Docker daemon: Docker service**

**Docker work flow**

****

**Ways to create Docker Images**

* **Take image from Docker hub**
* **Create image from existing docker containers**
* **Create image from docker file**

**Dockerfile**

**Dockerfile :**

** A text file with instructions to build image**

** Automation of Docker Image Creation**

**o FROM**

**o RUN**

**o CMD**

* **Step 1 : Create a file named Dockerfile**
* **Step 2 : Add instructions in Dockerfile**
* **Step 3 : Build dockerfile to create image**
* **Step 4 : Run image to create container**

**Volumes**

* **Volume is a directory inside your container**
* **First declare directory as a volume and then share volume**
* **Even if we stop container, still we can access volume**
* **Volume will be created in one container**
* **You can declare a directory as volume only while creating**
* **container**
* **You can't create volume from existing container**
* **You can share one volume across any no of containers**
* **Volume will not be included when you update an image**
* **Map volumes in two ways**
* **Share host – container**
* **Share container – container**

**Docker HUB**

**Docker Hub is a cloud-based repository service where users can create, test, store, and distribute Docker container images.**

**It serves as a centralized platform for container image discovery and sharing, allowing developers to automate their build, test, and deployment pipelines with a vast library of trusted and user-contributed images. Docker Hub supports both public and private repositories, integrates with CI/CD tools, and provides features such as automated builds, webhooks, and image tagging.**

**LAB**

**1. AWS Account - Free**

**2. EC2 Create Ubuntu**

**3. Docker installation**

**4. Code App – github repo**

**5. Docker File**

**6. Docker Images**

**7. Docker Container**

**Permissions**

**sudo usermod -aG docker ubuntu**

**sudo usermod -aG docker jenkins**

**groups jenkins**

**sudo chown ubuntu:docker /var/run/docker.sock**

**sudo chmod 660 /var/run/docker.sock**

**sudo systemctl restart jenkins**

**groups jenkins**

**sudo chown root:docker /var/run/docker.sock**

**sudo chmod 660 /var/run/docker.sock**

**sudo systemctl start docker**

**sudo systemctl enable docker**

**sudo docker build . -t flask-app:latest**

**sudo docker run -d -p 7000:7000 flask-app:latest**