Containers

Dockers-Docker Hub-ACR-ACI

Rajesh Kolla

Full-stack developer ,Azure Architect

Email: razesh.kolla@gmail.com

Twitter: @RajeshKolla18

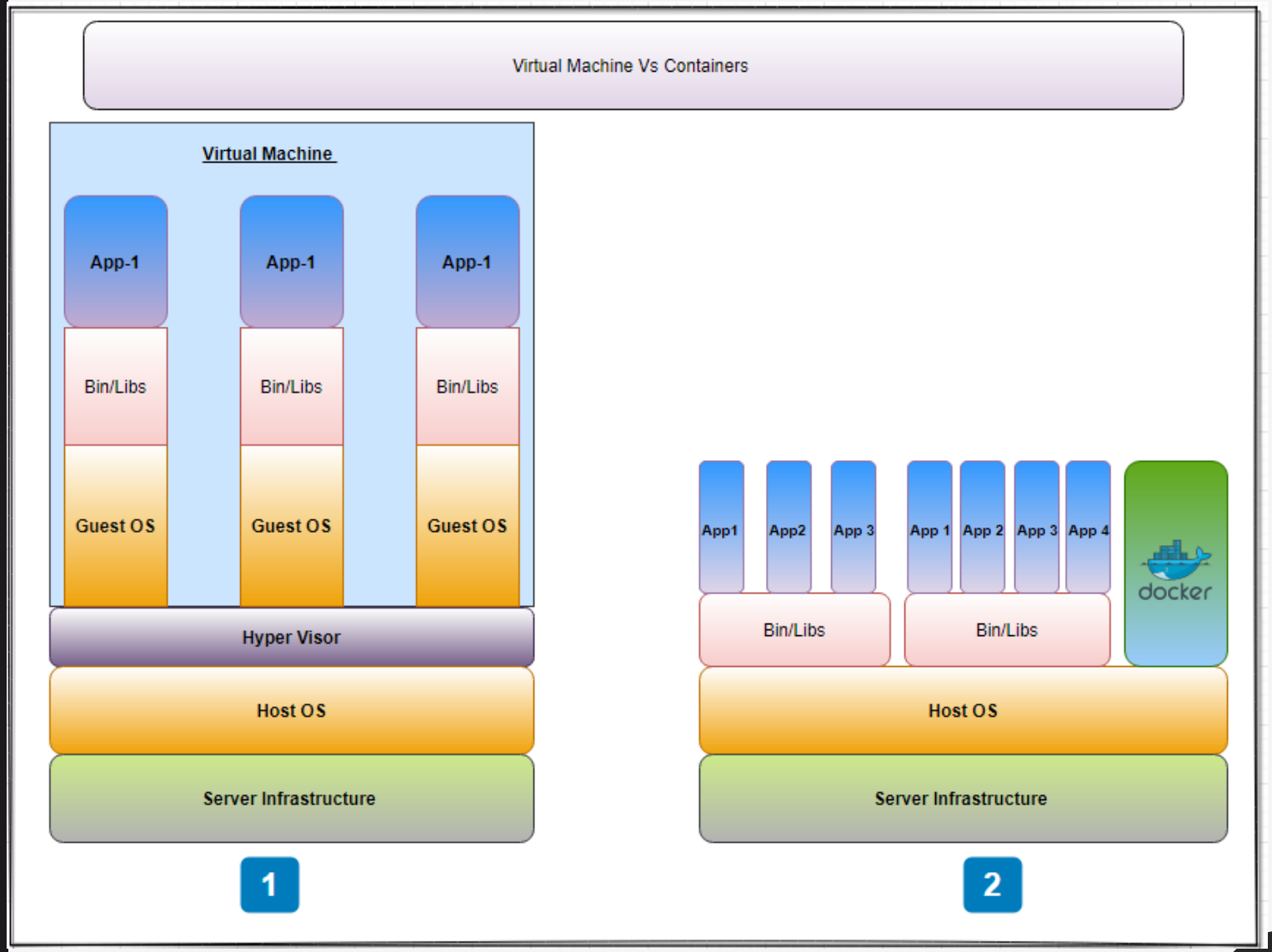
LinkedIn: https://be.linkedin.com/in/razeshkolla

Agenda

- Overview & Basics
- Benefits
- Docker Commands
- Docker Images &Registries
- Running Containers Locally
- Running Containers on Azure
- Demo
- Q&A

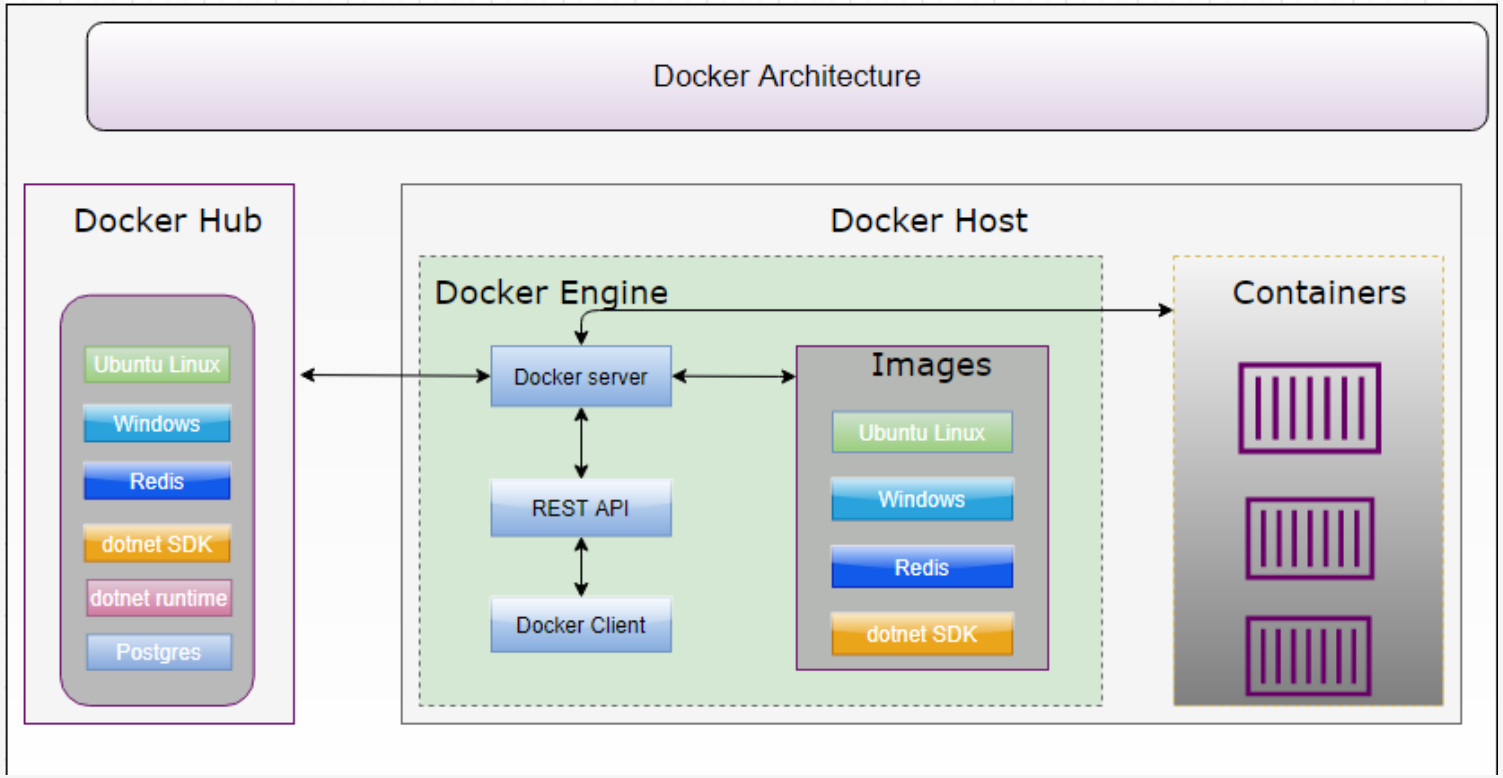
Virtualization

1. Virtual Machines are running on Hypervisor which enables virtualization of OS and manages sharing of physical resources to VMs. Each VM is having its own guest OS.
2. Containers are running directly on host OS. and shares host OS between containers. these are highly portable and isolated.



Docker Architecture

- Docker Host
- Docker Engine
  - Docker Client
  - Docker server
  - REST API
- Docker Hub



Docker Engine: Docker engine is an open source containerization technology for building and containerizing applications.

This consists of several components configured as a client-server implementation where the client and server run simultaneously on the same host.

The client communicates with the server using a REST API, which allows the client to also communicate with a remote server instance.

Docker Commands

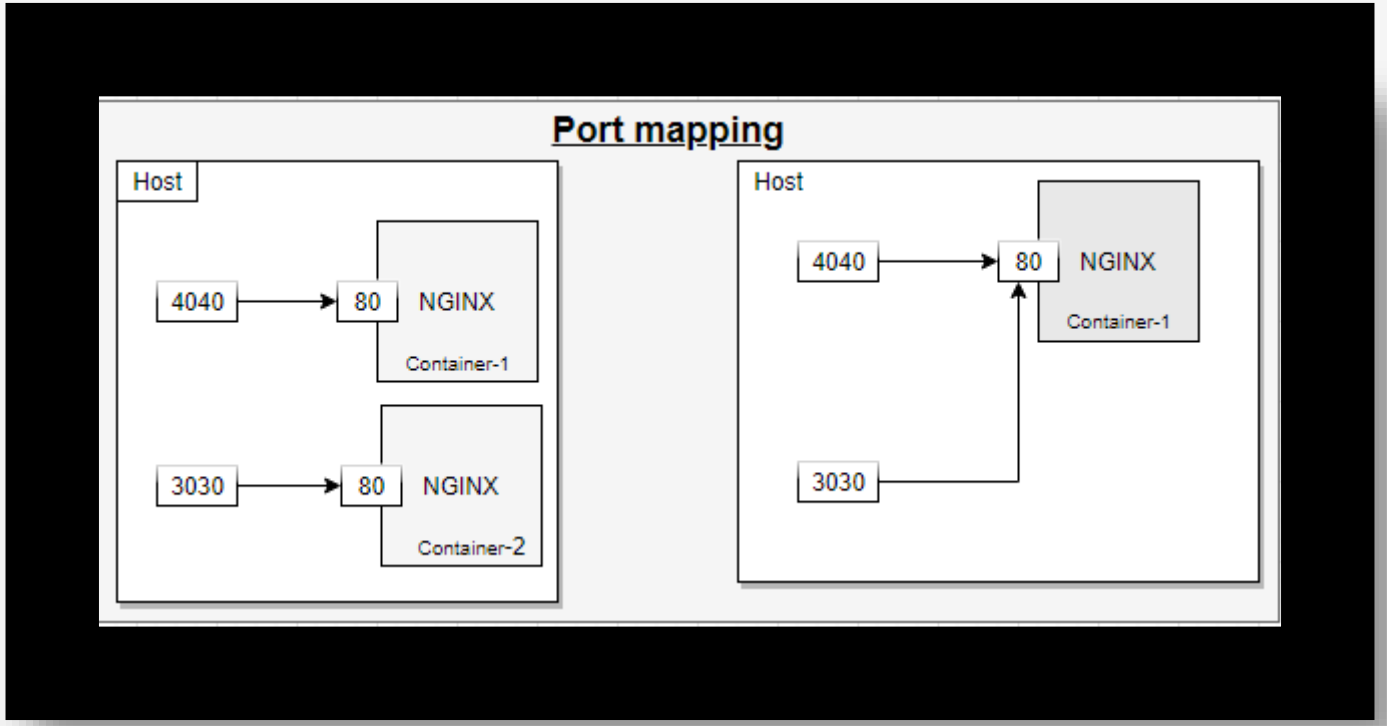
command	Description
docker version	Show client & server version information
docker --version	Show the Docker version information
docker info	Display system-wide information
docker images	List images
docker login	Log in to a Docker registry
docker logout	Log out from a Docker registry
docker ps	List containers
docker pull	Pull an image or a repository from a registry
docker push	Push an image or a repository to a registry
docker run <image>	Run a command in a new container
docker kill <Cont Id>	Kill one or more running containers
docker start	Start one or more stopped containers
docker stop	Stop one or more running containers
docker stats	Display a live stream of container(s) resource usage statistics
docker rm	Remove one or more containers
docker rmi	Remove one or more Images
docker tag	Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
docker exec	Run a command in a running container
docker restart	Restart one or more containers
Docker rename	Rename a container

Benefits with Docker

- **Content Agnostic**
  - Encapsulate content\Payload and dependencies
- **Hardware Agnostic**
  - Can run consistently on any hardware virtually
- **Content isolation**
  - Resource, network and content isolation avoids dependency hell
- **Automation**
  - Standard Operation to run , start ,stop. Well suitable for CI&CD.
- **Highly efficient**
  - Light weight , quick startup , quick to move and scale
- **Separation of Duties**
  - Developer focus on code and Ops focus on infrastructure
- **Build once and run anywhere**
  - Once image is built with all dependencies . This will run in any environment

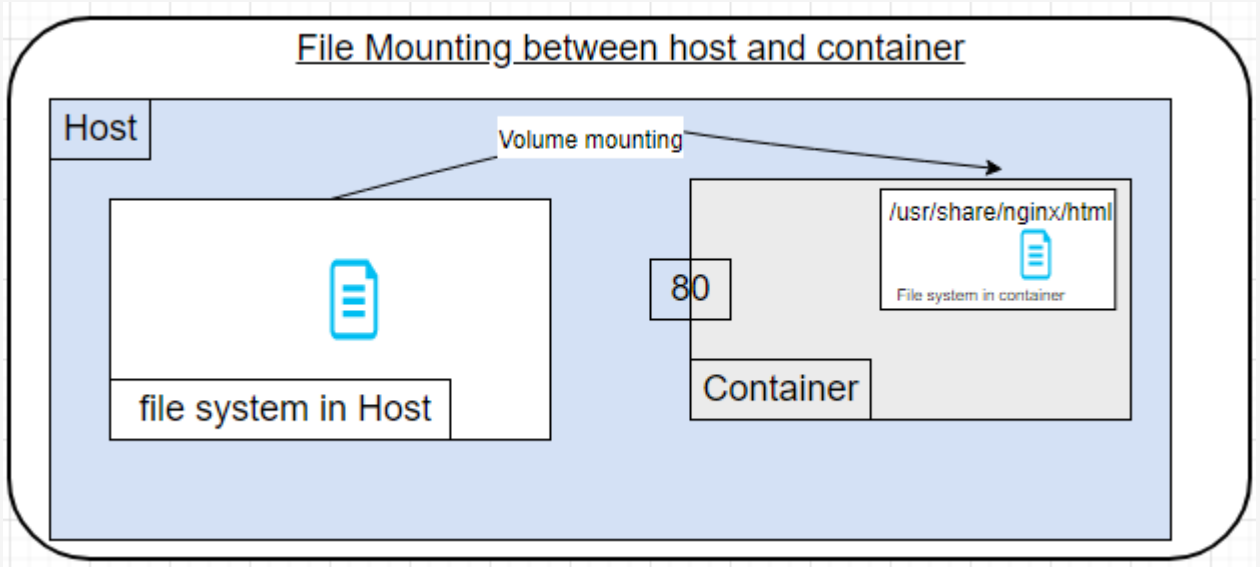
Port mapping

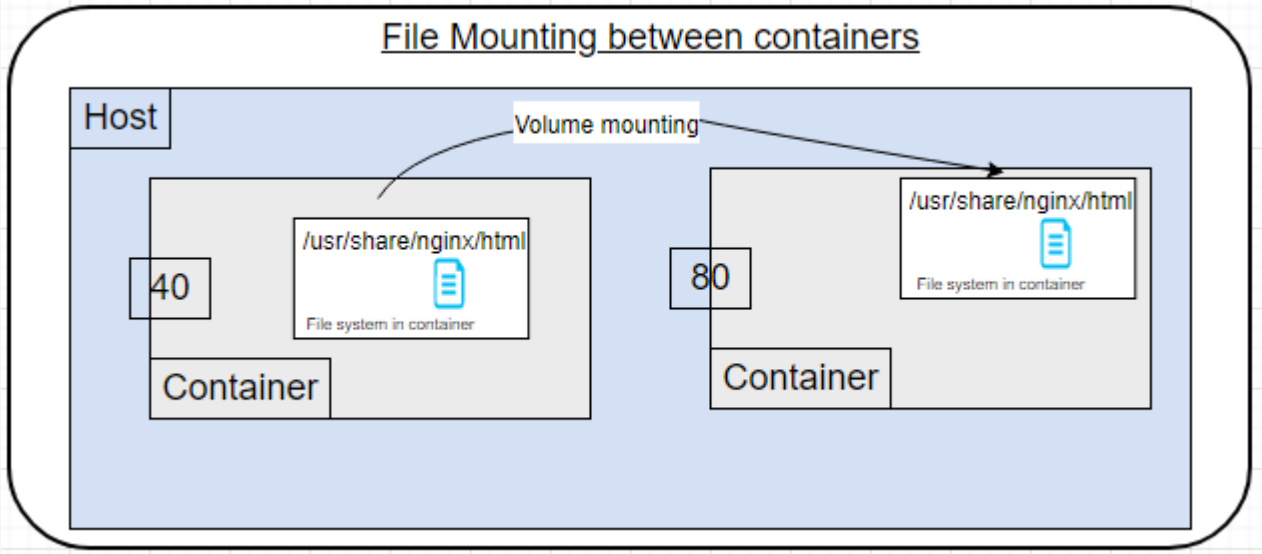
- Every container has its own network . There is provision to assign port to container .
- This port number is unique from host and another containers



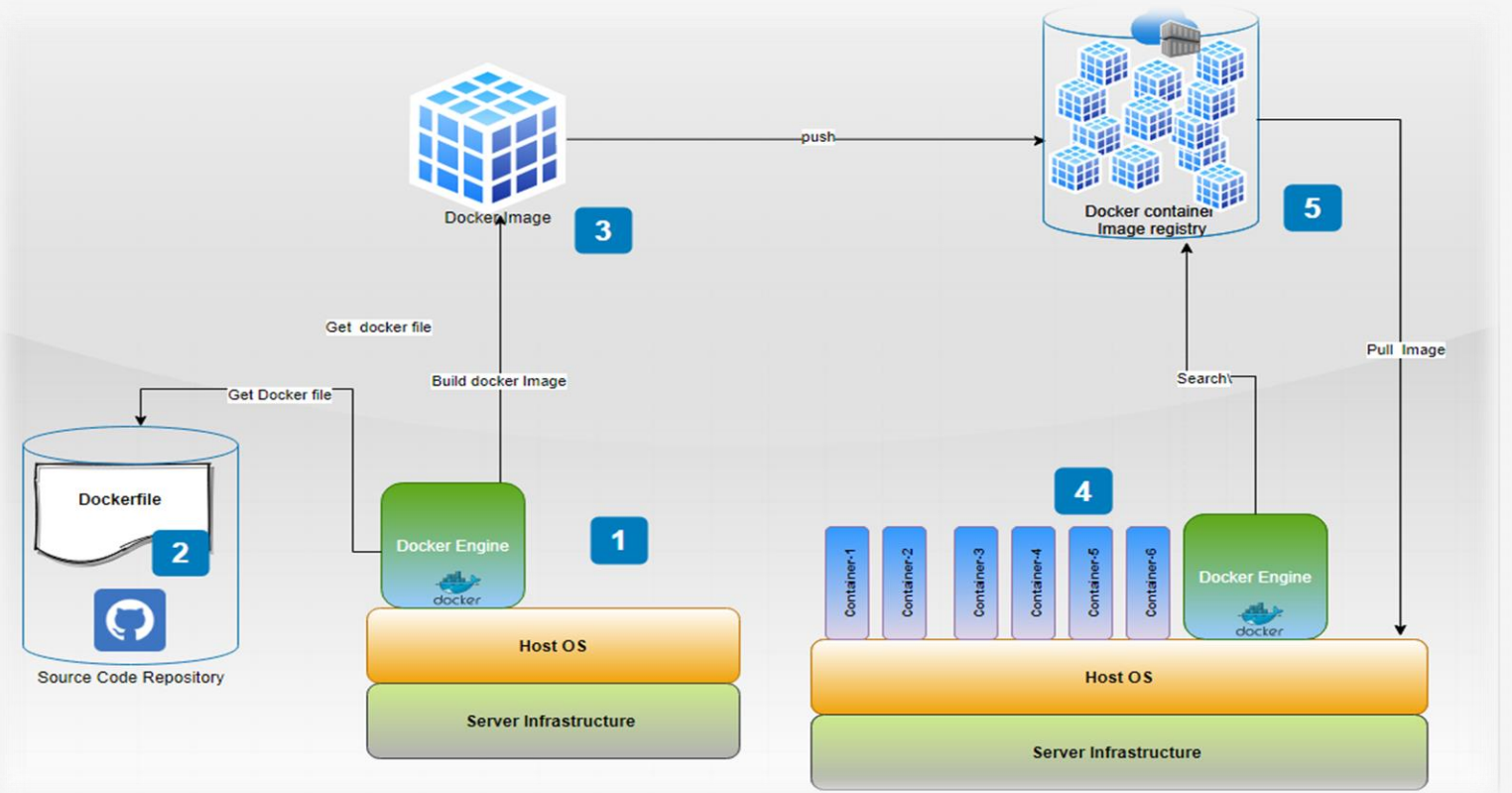
Volume Mounting

- Each container has its own isolated file system .
- When we dispose container Data in container will be lost
- Volume mounting is one of the solution for this issue.
- Volume mounting is nothing mapping external file system to container file system





Docker Image workflow



Docker file:-

- This is plain text file with set of instructions how to create container image.
- It uses set of keyword to create image in Layered architecture.
- It also support multi stages

Docker Host:- a physical computer system or virtual machine The component on the host that does the work of building and running containers is the Docker Daemon.

Docker Image:- is a standard unit of shippable software that packages up code and all its dependencies

Docker Container:- A container is running instance of image(standard unit of software that packages up code and all its dependencies) so the application runs quickly and reliably from one computing environment to another

Dockerfile

Build Docker image using Docker file

- Base Image: Image derived in docker file to build custom image
- Tag: it will create versioning to image
- Single stage Docker file:
  -Copies pre-build application
- Multi-stage Docker file:
  - Stage1:build app in container with SDK
  -Stage2: run app in container with runtime using stage1 build output

docker **build –t webapp:dev .**

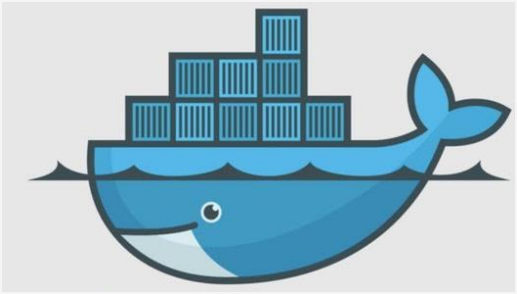
Singlestage docker file

```
#See https://aka.ms/containerfastmode to understand how Visual

FROM mcr.microsoft.com/dotnet/core/sdk:3.1-buster AS build
WORKDIR /src
COPY ["WebApp1.csproj", ""]
RUN dotnet restore "./WebApp1.csproj"
COPY . .
WORKDIR "/src/."
RUN dotnet build "WebApp1.csproj" -c Release -o /app/build
```

Multistage docker file

```
#See https://aka.ms/containerfastmode to understand how Visual Studio uses

FROM mcr.microsoft.com/dotnet/core/aspnet:3.1-buster-slim AS base
WORKDIR /app
EXPOSE 80
EXPOSE 443

FROM mcr.microsoft.com/dotnet/core/sdk:3.1-buster AS build
WORKDIR /src
COPY ["WebApp1.csproj", ""]
RUN dotnet restore "./WebApp1.csproj"
COPY . .
WORKDIR "/src/."
RUN dotnet build "WebApp1.csproj" -c Release -o /app/build

FROM build AS publish
RUN dotnet publish "WebApp1.csproj" -c Release -o /app/publish

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "WebApp1.dll"]
```

Container Registries



- Docker Images can be stored locally
- Docker Images can be shared by using Container registries
- Docker Hub is SaaS Application and most popular for host public images and Privates images
- ACR - Azure Container Registry is private registry provided by in Azure to store images in Azure
  - Can be build images automatically
  - It will reduce ingress and egress network latency

Azure Container Instances

- ACI is Server less PaaS component
- It enables the quickest and easiest way to run a container in azure
- It provides Per-second billing model
  -Pay only while container is running
- It is good for experiments , short runs and CI build.
- Good option for Batch jobs which are running for a few hours over night
- It is not good option for long running web server & database

- ACI Features
  - Easy to create and Manage using
    Azure CLI, Powershell, C# SDK, ARM
  - Networking
    - Public IP Address
    - Domain Name prefix
    - Expose ports
  - Mount Volumes
    - Azure file share
    - Secrets
  - CPU and Memory
    - By Default 1 CPU and 1.5GB Memory
  - Supports both windows & Linux containers
    - Linux containers are spin up very faster rather than windows containers
    - default :Linux
  - Configure environment variable
  - access container logs
  - Container groups
    - one or more containers
    - Run on the same server and share resources

Q&A

Email: razesh.kolla@gmail.com
Twitter: @RajeshKolla18
LinkedIn: https://be.linkedin.com/in/razeshkolla

Thank you!

Email: razesh.kolla@gmail.com
Twitter: @RajeshKolla18
LinkedIn: https://be.linkedin.com/in/razeshkolla