Sprints

Containerization and Docker

Hany Hesham

Sprints

Outline

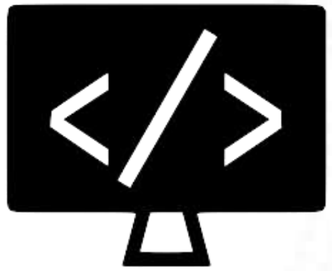
1. What is Containerization?
2. Why it's important for modern application development and deployment ?
3. What is Docker and how it enables containerization ?
4. What are Docker images and containers ?

Sprints

What is a software ?

OS

Run Time Engine



Your Custom Code

Software

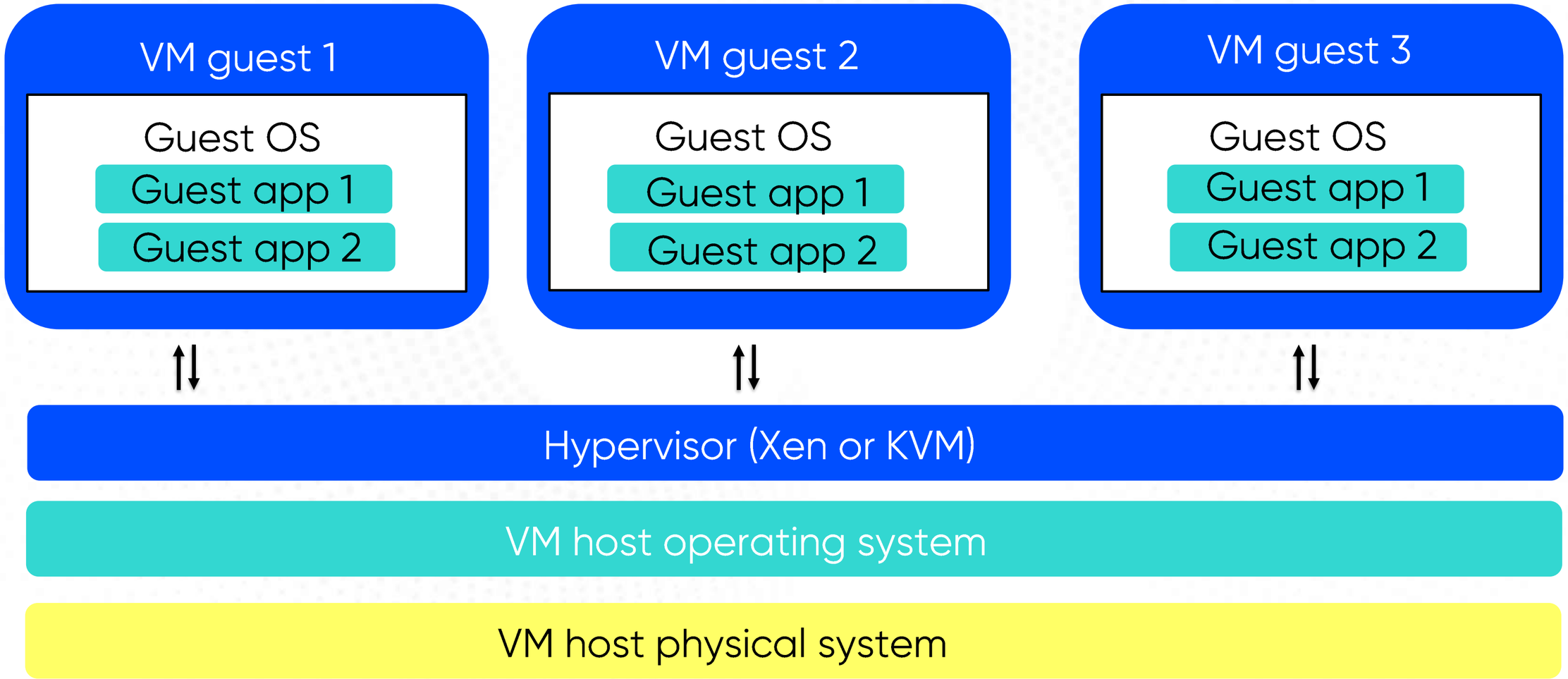
Sprints

Virtualization

Spinning multiple servers on **one** powerful server

Sprints

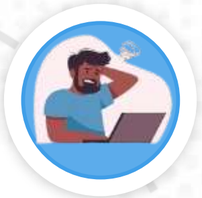
Virtualization Deployments



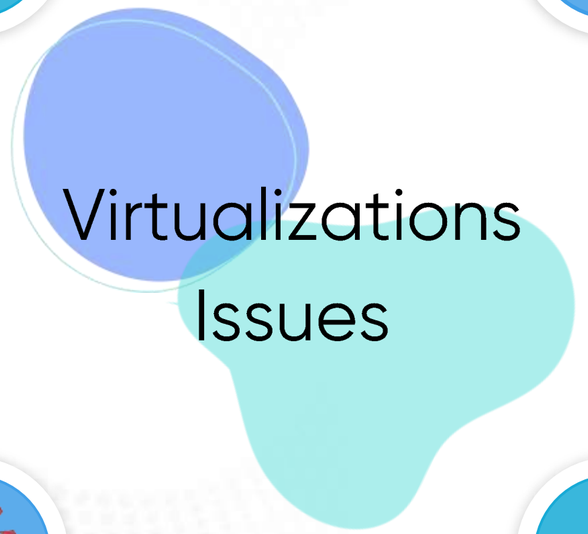
Sprints

Virtualizations Issues

Operation overhead for companies



Hard and very complex to **Scale**



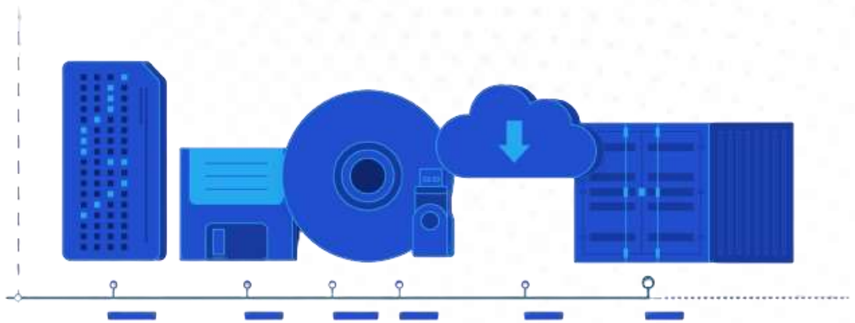
Higher **risk** because of the hidden configurations and custom setup

Higher **cost**

Sprints

Containers

"Containers are packages of software that contain all of the necessary elements to run in any environment" - **Google definition**

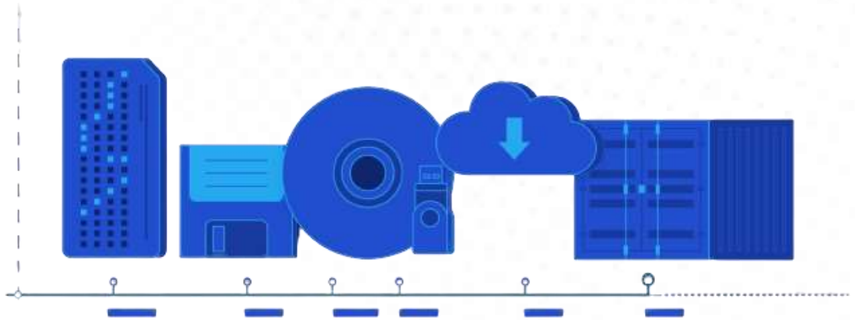


Sprints

What is Containerization ?

- Containerization is a process of packaging an application and its dependencies into a self-contained unit that can run anywhere

- Containerization allows developers to create and deploy applications faster, easier, and more reliably

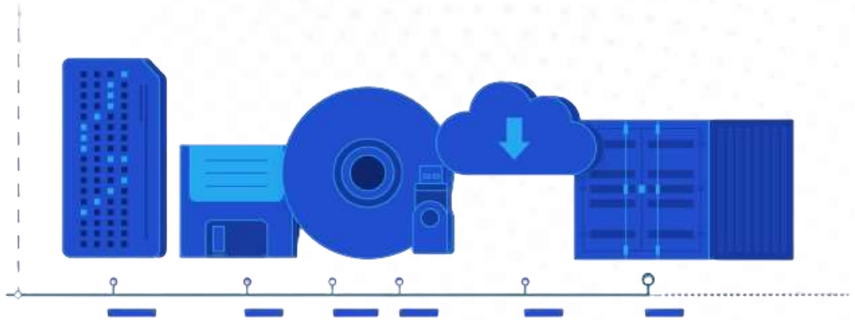


Sprints

What is Containerization ?

- Containerization also enables applications to run in isolated environments, which improves security, performance, and resource utilization

- Containerization is one of the key technologies behind the rise of cloud computing and microservices



Sprints

What is Containerization ?

Control groups

You can use cgroups to limit resources.
This process can only use 256MB of memory and 1 vCPU

\+

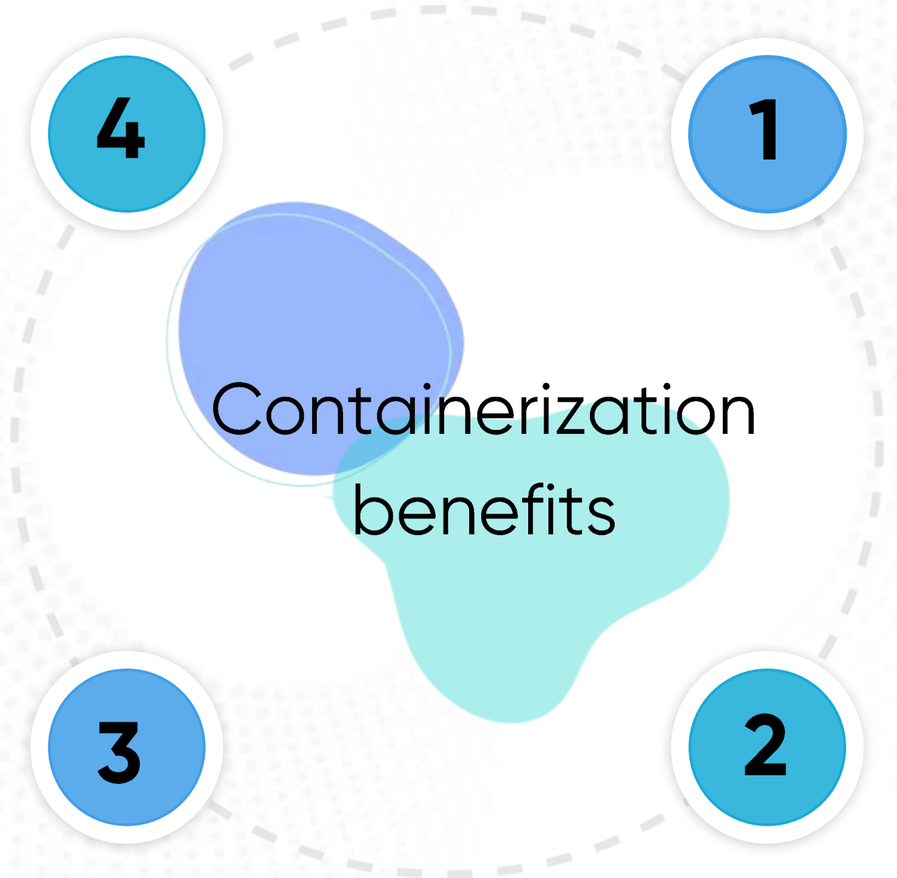
Namespaces

You can use namespace to isolate processes.
This process sees only its network traffic

= Containers

Sprints

Containerization Benefits



4 Less hardware resources

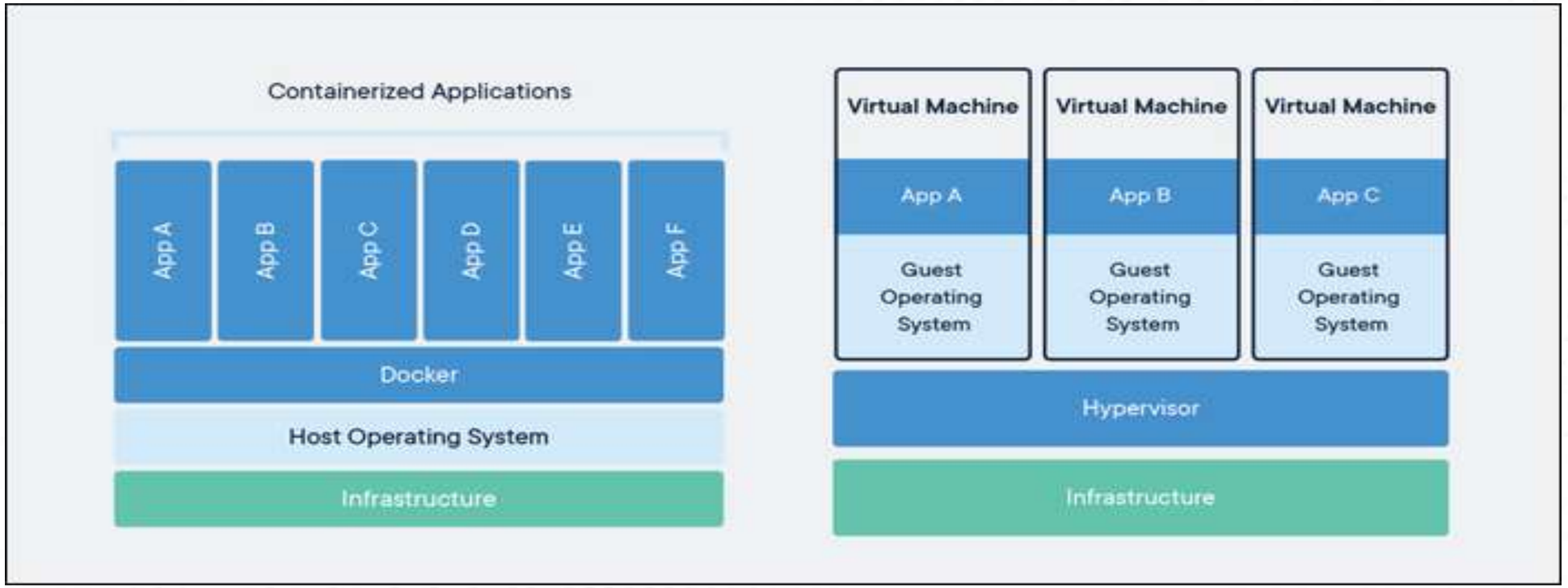
1 Application isolation

3 Quick startup time

2 Constant deployment across multiple OS and environments

Sprints

Virtualization VS Containerization

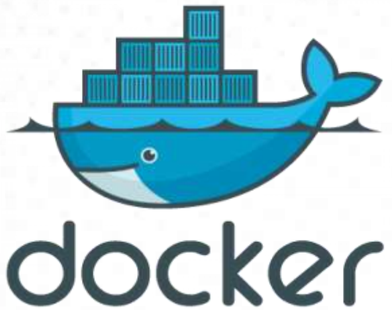


Sprints

What is Docker?

- Testing Docker is an open-source platform that provides tools and services for building running and managing containers

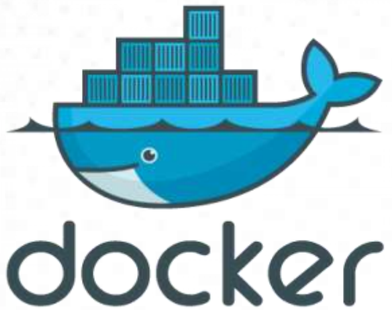
- Docker uses a client-server architecture, where the Docker client communicates with the Docker daemon (or server) to perform various operations on containers



Sprints

What is Docker?

- Docker provides a registry services called Docker Hub, where you can store and share your container images
- Docker supports multiple operating systems such as Linux , windows and macOS which make it perfect for development and deployment



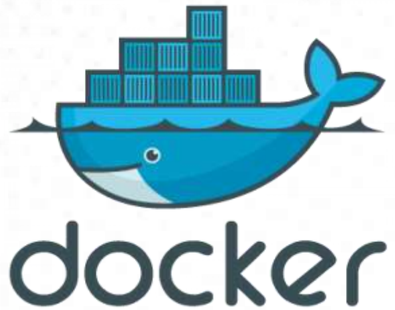
Sprints

What is Docker Image?

- Docker images are the building blocks of containers, they are files that contain all information needed to create and run a container

- Docker images are composed of layers, which are snapshots of the file system at different stages of the container creation process

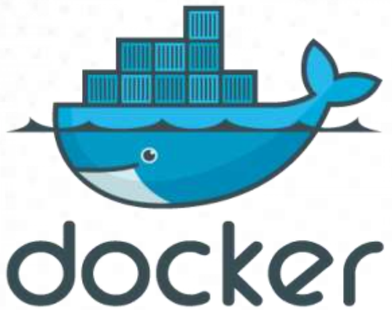
- Docker images are immutable, meaning they cannot be modified once they are created, instead we create different image tags



Sprints

What is Docker Image?

- Docker images can be created using a Dockerfile, which is a text file that contains instructions for building an image
- Docker images can be shared and transferred to any registry
- You can run as many containers as you can from one docker image



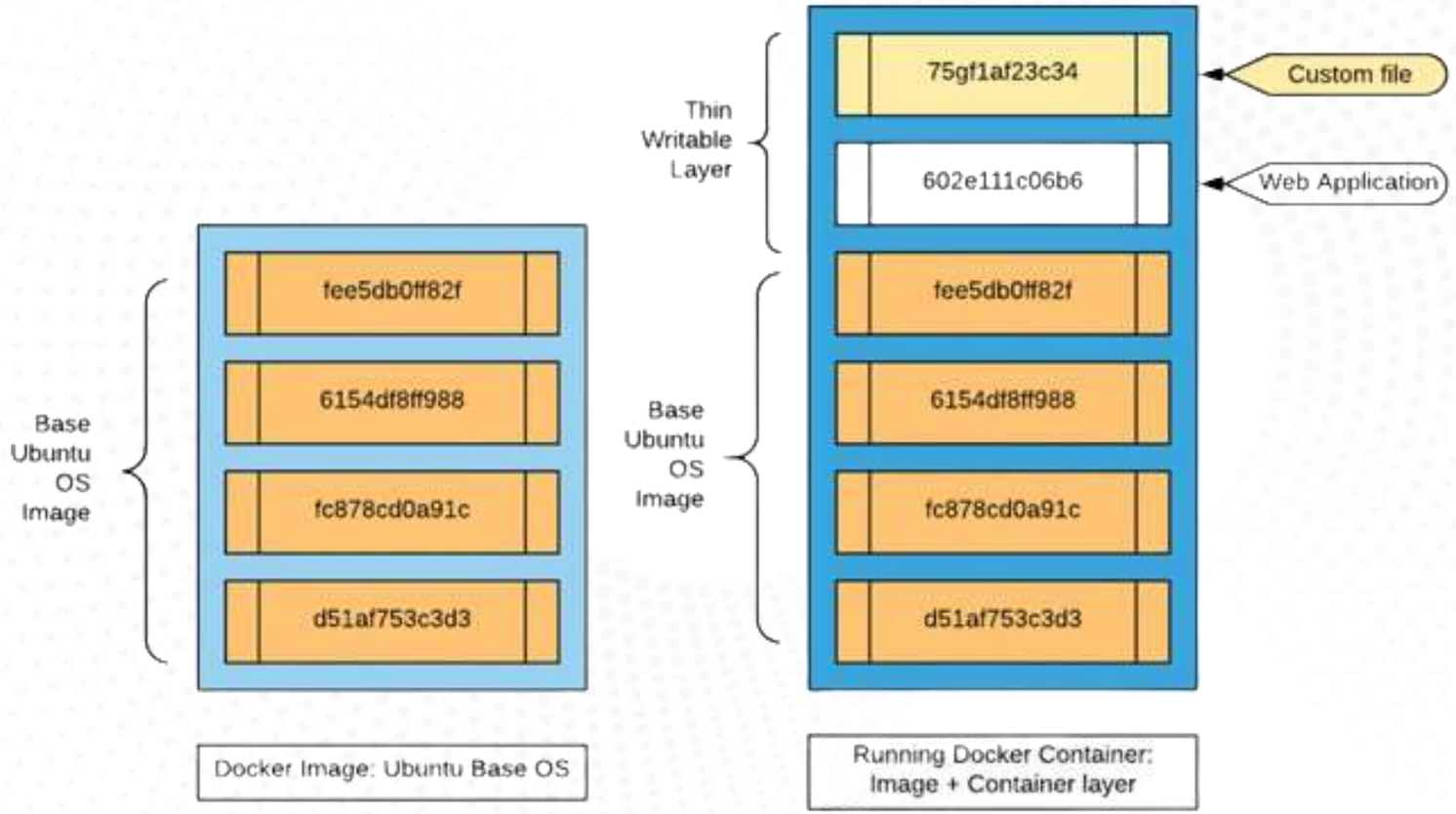
Sprints

Dockerfile Example

```dockerfile
FROM python:3

# set a directory for the app
WORKDIR /usr/src/app

# copy all the files to the container
COPY . .

# install dependencies
RUN pip install --no-cache-dir -r requirements.txt

# define the port number the container should expose
EXPOSE 5000

# run the command
CMD ["python", "./app.py"]
```

Sprints

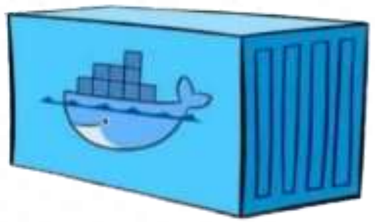
Dockerfile Example



Sprints

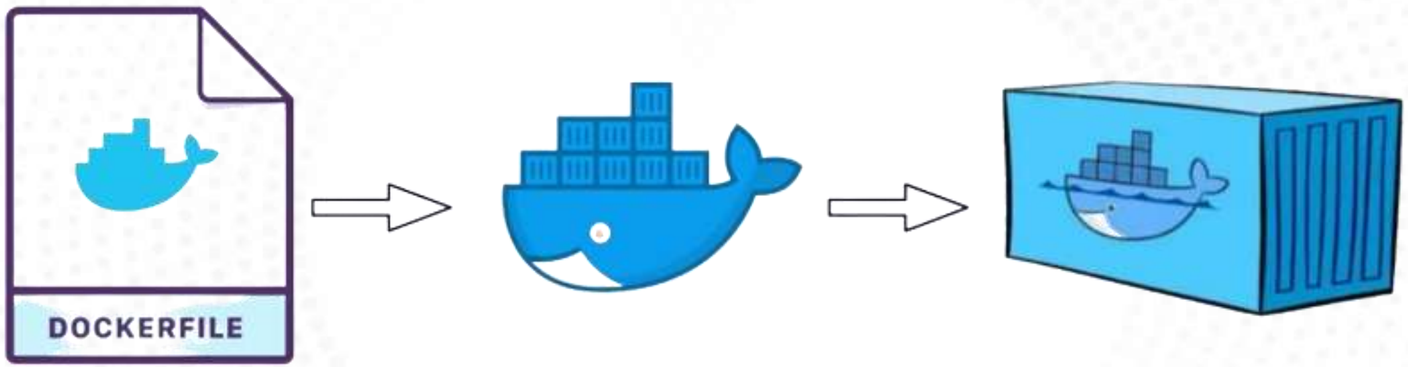
What is Docker Container?

- Docker containers are instances of Docker images that run in an isolated environment
- Contains can be created using the docker run command which takes an image name as an input and it start a docker container based on that image
- You can run multiple containers and they can communicate with each others using the docker network



Sprints

What is Docker Container?



Dockerfile Docker Image Docker Container

Sprints

What is Docker Benefits?

Portability

Docker containers can run on any platform that supports Docker, without requiring any change to the application code or configuration

Security

Docker containers provide isolation and sandboxing for application which reduces the risk of malicious attacks or accidental errors

Scalability

Docker containers can be scaled up or down according to the demand or resource availability

Efficiency

Docker containers use less resources than traditional virtual machines, as they share the same kernel and do not need to run a separate operating system for each application

Sprints

Docker Use Cases

Development

Docker can be used to create consistent and reproducible development environment that match production environment

Production

Docker can be used to deploy application in production environments with high availability, reliability and scalability

Testing

Docker can be used to automate and streamline the testing process by creating and destroying containers on demand

Cloud Computing

Docker can be used to leverage the benefits of cloud computing, such as elasticity, flexibility and cost-effectiveness

Sprints

Conclusion

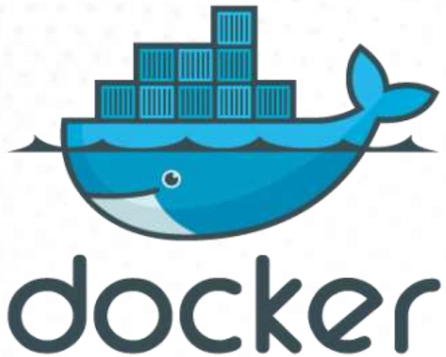
- Containerization is a powerful technology that enables modern application development and deployment
- Docker is a leading platform that provides tools and services for containerization
- Docker offers many benefits for developers and operators such as portability, scalability, security and efficiency
- Docker can be used for various scenarios, such as development, testing, and production

Sprints

Workshop

Installation

Please follow the steps from this link to download and install Docker desktop on your laptop



Sprints

Common Docker CLI

- **docker ps** - list running containers
- **docker ps –a** - list running & stopped containers
- **docker images** - list available images
- **docker stop** - stop container, provide Container ID
- **docker start** - start stopped container, provide Container ID
- **docker restart** - stop then restart container, provide Container ID
- **docker rm** - remove stopped container, provide Container ID
- **docker rmi** - remove image provide Image ID
- **docker kill** - forcibly stop (kill) container, provide Container ID

https://dockerlabs.collabnix.com/docker/cheatsheet/

Sprints

Exercise 1

- Pull Nginx image
- Run Nginx container
- View running containers
- Stop Nginx container
- Restart Nginx container
- Remove the Nginx container

- Nginx: is open source software for web serving, reverse proxying, caching, load balancing, media streaming, and more

Sprints

Exercise 2

Build and run a custom nginx Docker image with your own home page

- Create a Dockerfile
- Create a new file index.html
- Build your image
- Run a new container from your image

Sprints

THANK YOU

www.sprints.ai