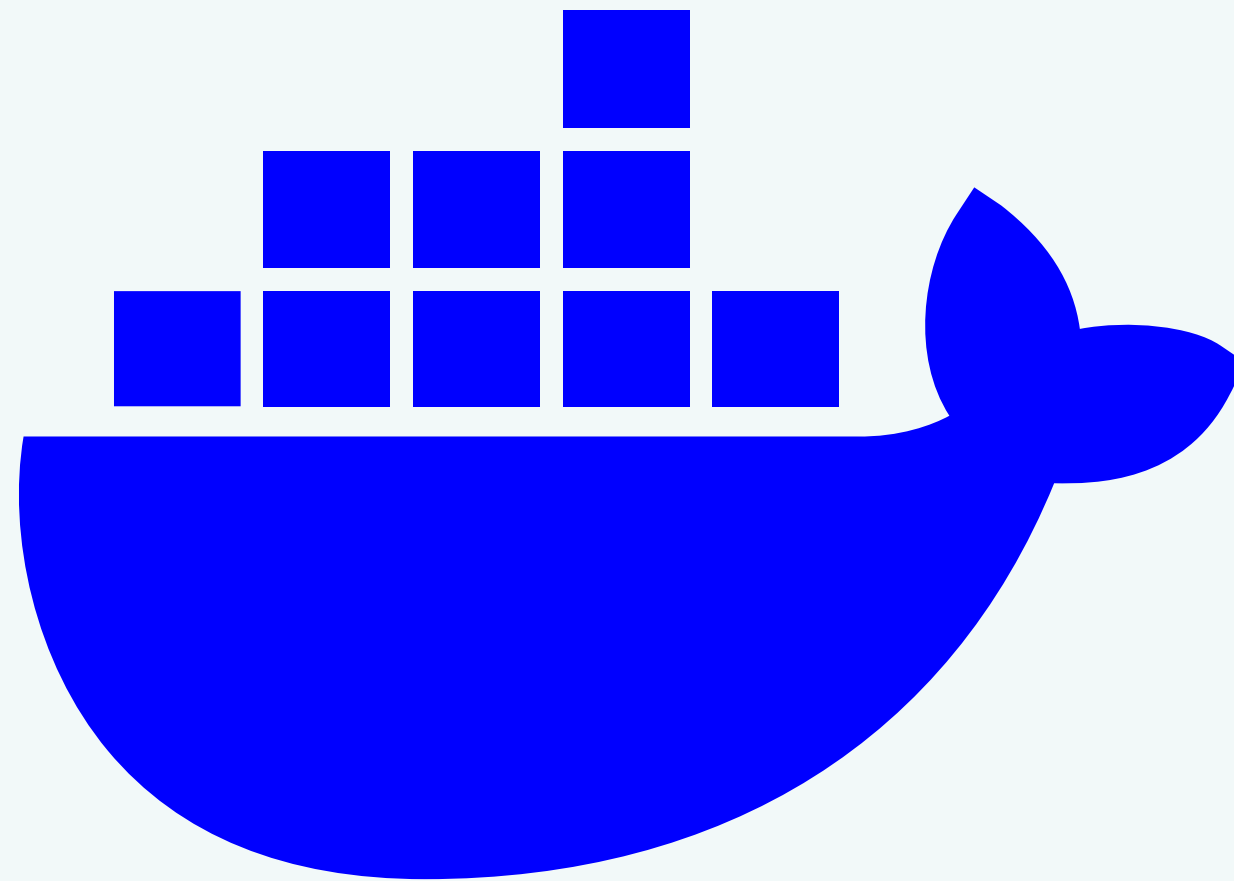


DOCKER



ETHIOPIAN ARTIFICIAL INTELLIGENCE INSTITUTE | 2024

PRESENTATION

CONTENTS

- 1 What is Docker?
- 2 Virtualization Vs Containerization
- 3 What is the of benefits using Docker?
- 4 Docker Concept
- 5 Dockerfile and CLI

WHAT IS DOCKER?

Docker is used to pack an application with all the dependencies it needs into a single, standardized unit for the deployment.

VIRTUALIZATION VS CONTAINERIZATION

VIRTUALIZATION

- Complete, virtualized physical machines.
- Each VM has its BIOS and operating system.
- Runs on top of the host OS.
- Supports different OSs for each VM.

...

- Strong isolation but resource-intensive.
- Slower startup times.
- Requires full OS installation.

...

CONTAINERIZATION

- Isolated environments (containers).
- Containers share the host machine's kernel.
- Efficient resource usage.
- Fast startup times.



- Lightweight containers with minimal OS.
- Easy and fast provisioning.



WHAT IS THE OF BENEFITS USING DOCKER?

- Fast Performance and Short Provisioning Time
- Resource Efficiency and Lightweight Containers
- Portability and Easy Distribution
- Support for Microservices

...

- Abstraction Layer for Platform Independence
- Reproducible Testing and Parallel Testing
- Consistency in Development, Test, and Production Environments
- Simplified Management and Quick Deployment

DOCKER CONCEPT

IMAGE

- Images are read-only templates used to create containers.
- Images serve as a base foundation for containers, containing everything an application needs to run.

...

- Images can start from a base image (e.g., Ubuntu) or use pre-prepared images from the internet.
- Docker images are created using Dockerfiles, which are plain text files with a series of instructions.

...

- Each instruction in a Dockerfile creates a new layer in the image.
- Images are highly portable across hosts and operating systems.

...

LAYERS

- Images consist of a series of layers stacked on top of one another.
- Each layer is an intermediate image and contains an ID and a pointer to its parent layer.

...

- Layers are reusable, cacheable, and additive.
- Reusable layers save time, bandwidth, and disk space.
- The additivity of layers can result in growing image size, so optimization is important.

...

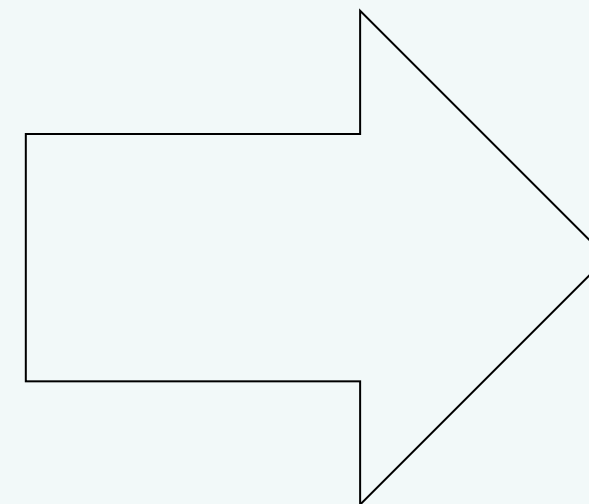
File2.txt

File1.txt

File2.txt

Pom.xml

Config.json



File2.txt

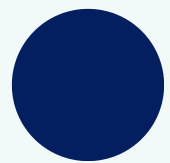
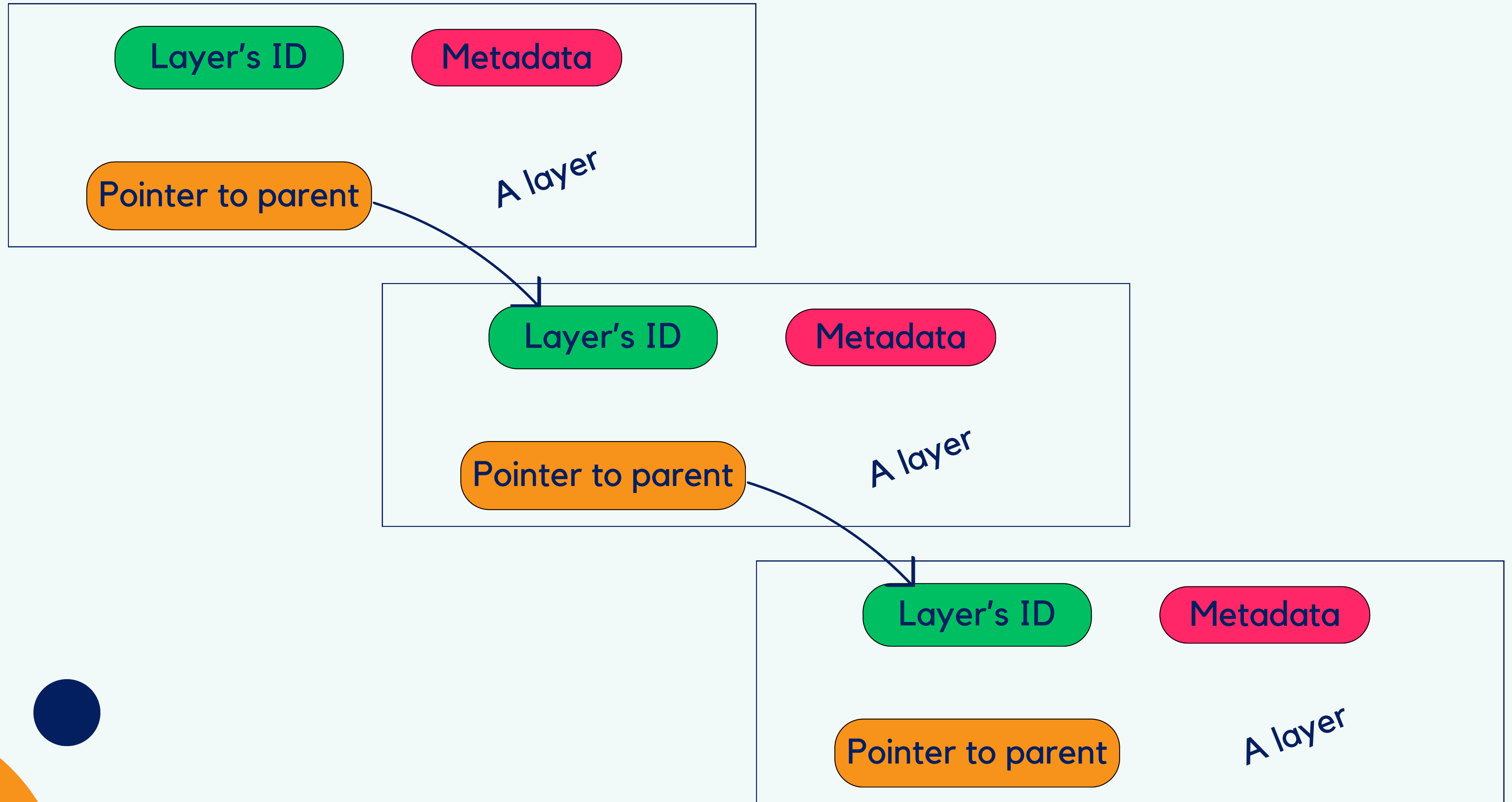
File1.txt

Pom.xml

Config.json

Union Filesystem

...



...

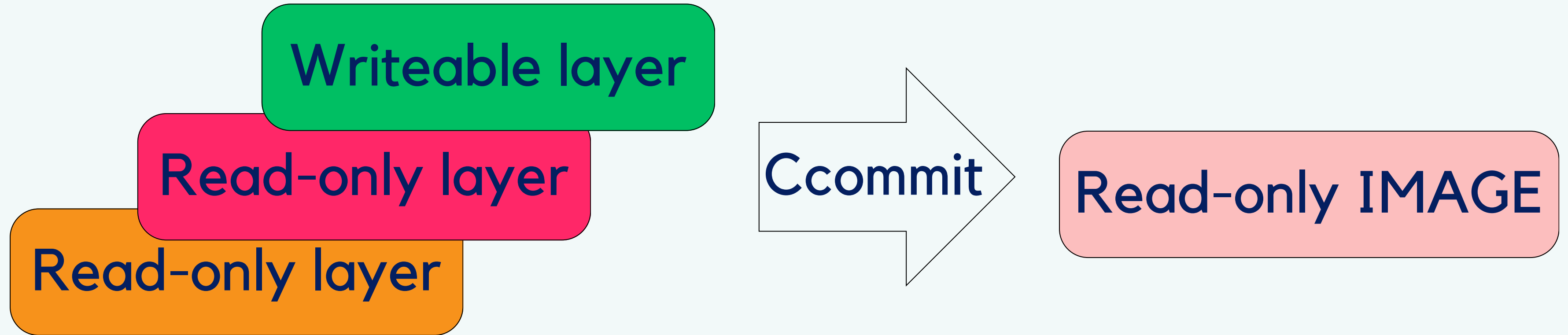
CONTAINER

- Containers are running instances of images.
- Containers are created from images and have a writable layer on top of the image's read-only layers.

...

- Changes to a running container are made in the writable layer and can be committed to create a new image.
- Stopping a container retains its settings and filesystem changes in the writable layer.
- Containers are managed using commands like `docker run`, `docker stop`, and `docker rm`.

...



DOCKERFILE FOR REACT JS

Base image

FROM node:14

Working directory

WORKDIR /app

Copy package files

COPY package.json .

Install dependencies

RUN npm install

...

Copy application code

COPY ..

Expose port

EXPOSE 3000

Command to run the
app

CMD ["npm", "start"]

DOCKER CLI

- Build an Image from a Dockerfile
`docker build -t <image_name>`
- List local images
`docker images`
- Delete an Image
`docker rmi <image_name>`
- Publish an image to Docker Hub
`docker push <username>/
<image_name>`

...

- Publish an image to Docker Hub

`docker pull <image_name>`

- Run a container with and publish a container's port(s) to the host

`docker run -p <host_port>:`

`<container_port> <image_name>`

- Get help with Docker

`docker --help`

THANK YOU!



ETHIOPIAN ARTIFICIAL INTELLIGENCE INSTITUTE | 2024