

Docker Compose

Divyansh Rajesh Jain

Problem?

- Docker is designed to work with isolated containers
- How do I build multiple containers
 - Have to run many build and run commands
 - NOT SCALABLE!

Docker Compose!

- Tool for defining multi-container application configuration
- One YAML file is the source of truth for the entire build configuration
- Simplified CLI (my opinion)

Steps of using Docker Compose

- Define Dockerfile(s) for all of your different containers
- Define the logical “services” and their environments
(Dockerfile)
 - Can configure many options in the Compose file
- Use one command to build/run all containers!

Basic Components

- Environment → Dockerfile
- Services → The invocation of a Dockerfile with all options
- Volumes → Attach services to volumes
- Networking → Configure Local Connectivity for services

Networking in Docker Compose

- Docker → Containers are separated
- Docker Compose → Services belong to the same network
- What are possible use cases of this?

Why Docker Compose?

- Easier configuration for docker containers
 - Docker configurations are baked into the compose file
- Compatible with Deployment Tools
 - Swarm Mode → Orchestration Platform from Docker
 - Kubernetes → Open Source Container Orchestration Platform

Example of Docker Compose File

```
api:
  build:
    dockerfile: Dockerfile.dev
    context: ./maisie_api
  volumes:
    - /app/deps
    - ./maisie_api:/app
  environment:
    - PGUSER=postgres
    - PGHOST=postgres
    - PGDATABASE=postgres
    - PGPASSWORD=postgres_password
    - PGPORT=5432
    - GUARDIANSECRET=8IW0WCEymrsSAXKoRpyEz2
```


Docker Compose Commands

- `docker compose build` → Build all the images in compose configuration
- `docker compose up [-d]` → Build Images and/or start all containers in the compose configuration
- `docker compose ps` → See all running contains in the compose configuration

Docker Compose Live Demo

Thank You!

Next Time: Kubernetes