

构建微服务云原生应用

架构师杨波



扫码试看/订阅

《Spring Boot & Kubernetes 云原生微服务实践》

CHAPTER

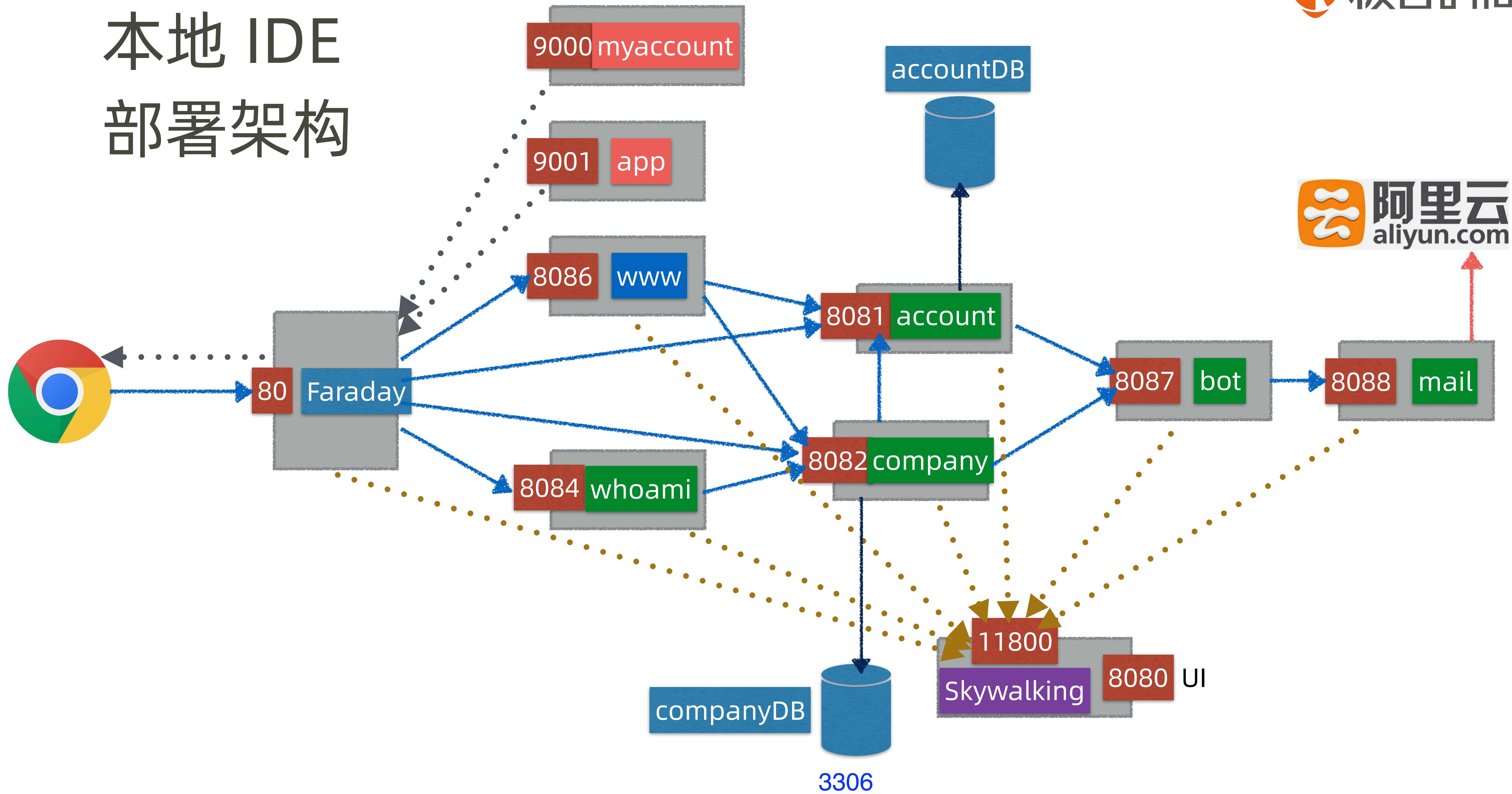
8

服务容器化和 Docker Compose 部署

第 1 部分

本地开发部署架构和需求

本地 IDE 部署架构



软件需求

1. SwitchHosts 域名的方式访问

- <https://github.com/oldj/SwitchHosts>



2. SkyWalking(6.1.0)

- <https://github.com/apache/skywalking/releases>
- <https://github.com/apache/skywalking/blob/master/docs/en/setup/service-agent/java-agent/README.md>

3. MySQL(5.7.25) + Workbench

- <https://dev.mysql.com/downloads/mysql/>

4. 阿里云邮箱账户(可选)



CODELAB

第 2 部分

手工服务部署和测试

部署步骤

Step By Step

1. 清理释放内存

- 活动监控器 建议16G
- 关闭Docker

2. 部署 MySQL 数据库

- staffjoy_account
- staffjoy_company

3. 部署 SkyWalking

- bin/startup.sh
- IDE 设置 skywalking-agent

4. 部署服务

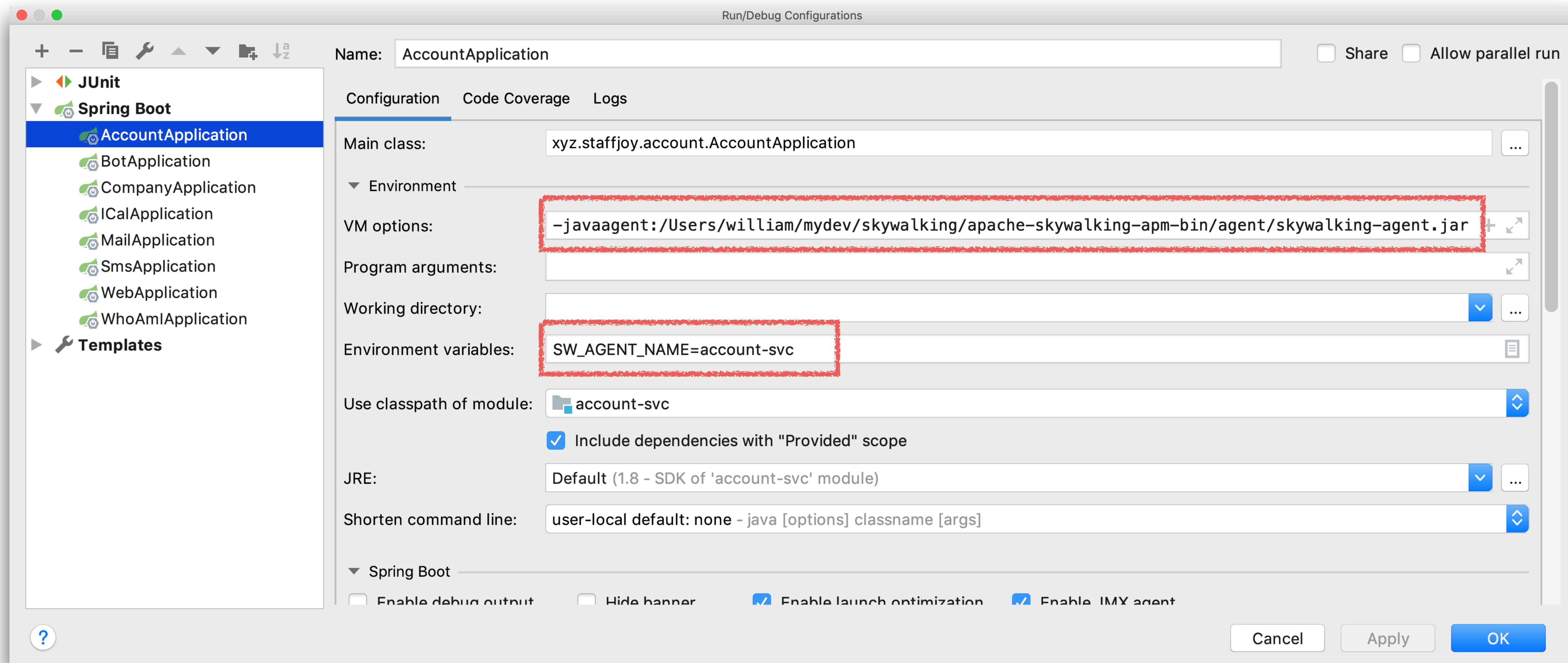
- 本地机密数据配置
- Faraday配置Review
- mail -> bot -> account -> company -> www -> whoami -> faraday

5. 部署单页应用

- npm install
- npm start

6. 启用 SwitchHosts

Run > Edit Configurations



SwitchHosts

```
SwitchHosts!
1 # SwitchHosts!
2
3 # My hosts
4 127.0.0.1 account.staffjoy-v2.local
5 127.0.0.1 faraday.staffjoy-v2.local
6 127.0.0.1 myaccount.staffjoy-v2.local
7 127.0.0.1 whoami.staffjoy-v2.local
8 127.0.0.1 www.staffjoy-v2.local
9 127.0.0.1 ical.staffjoy-v2.local
10 127.0.0.1 staffjoy-v2.local
11 127.0.0.1 app.staffjoy-v2.local
12 127.0.0.1 company.staffjoy-v2.local
13
```

测试步骤

1. 浏览器访问

- www.staffjoy-v2.local

2. 校验业务流程

- 注册管理员和公司，登录
- 注册雇员和登录
- 排班

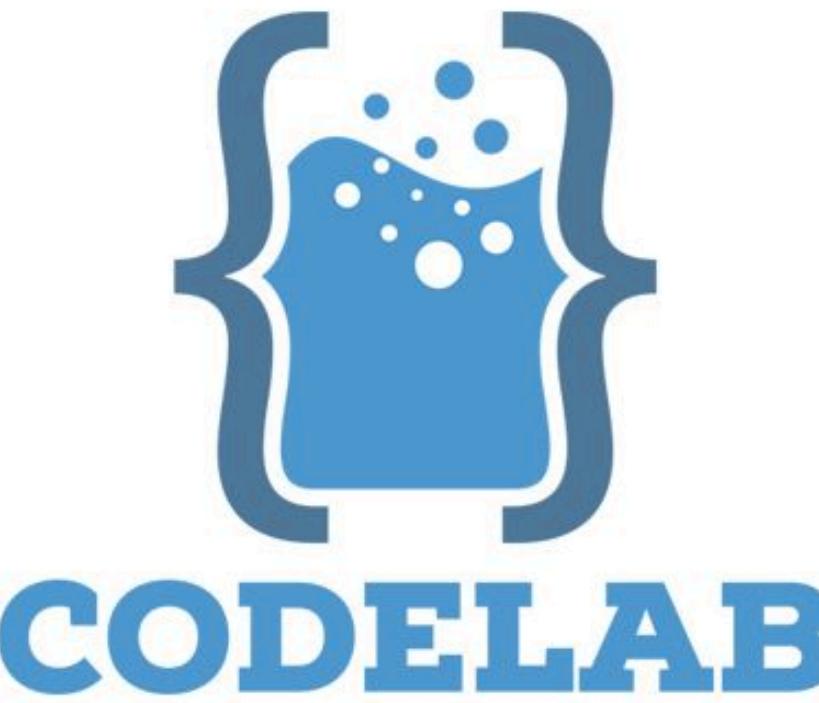
3. 校验 Cookie

4. 校验 DB



Step By Step

A green, rounded rectangular button with the text "Step By Step" written on it in white, tilted diagonally upwards from the bottom-left.



第 3 部分

SkyWalking 调用链监控实验

测试步骤

Step By Step

1. 浏览器访问 Dashboard

- http://localhost:8080
- admin/admin

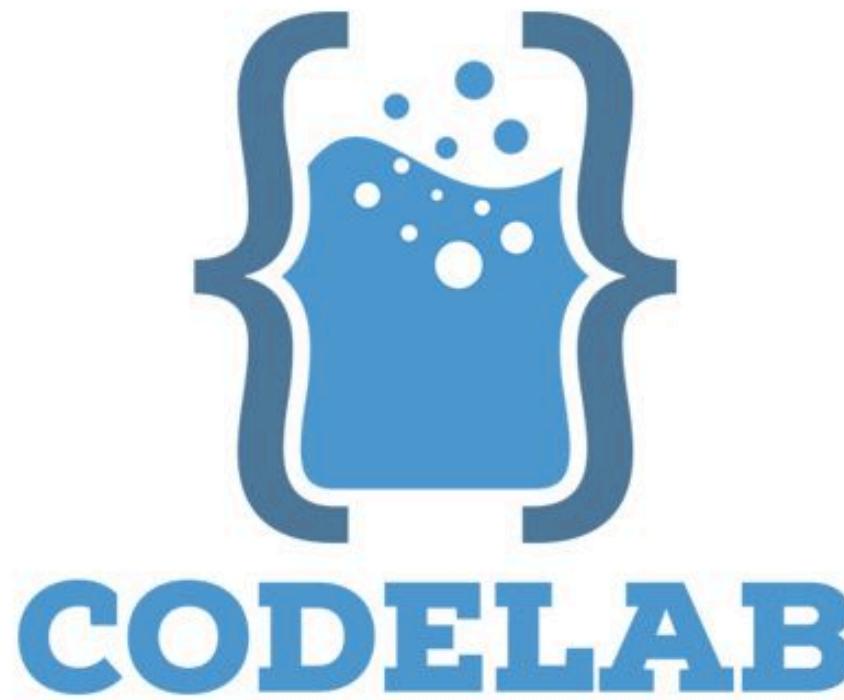
2. 校验拓扑图

3. 校验仪表盘

4. 校验追踪

5. 结束清理

- 关闭服务
- 关闭单页应用
- 关闭 SkyWalking(jps)



第 4 部分

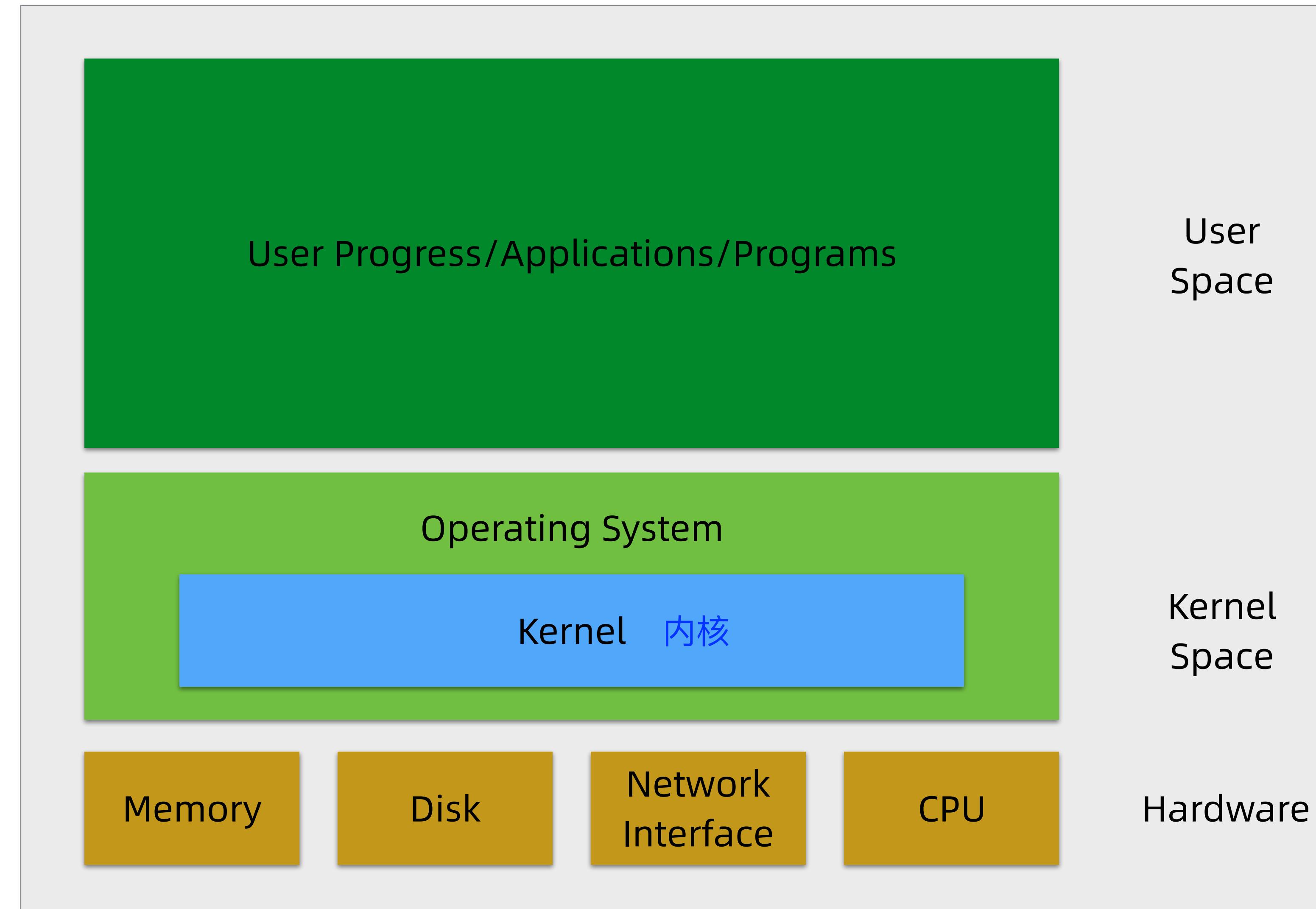
Docker 和 Docker Compose 简介

容器用途

1. 标准化打包
2. 隔离
3. 标准化部署

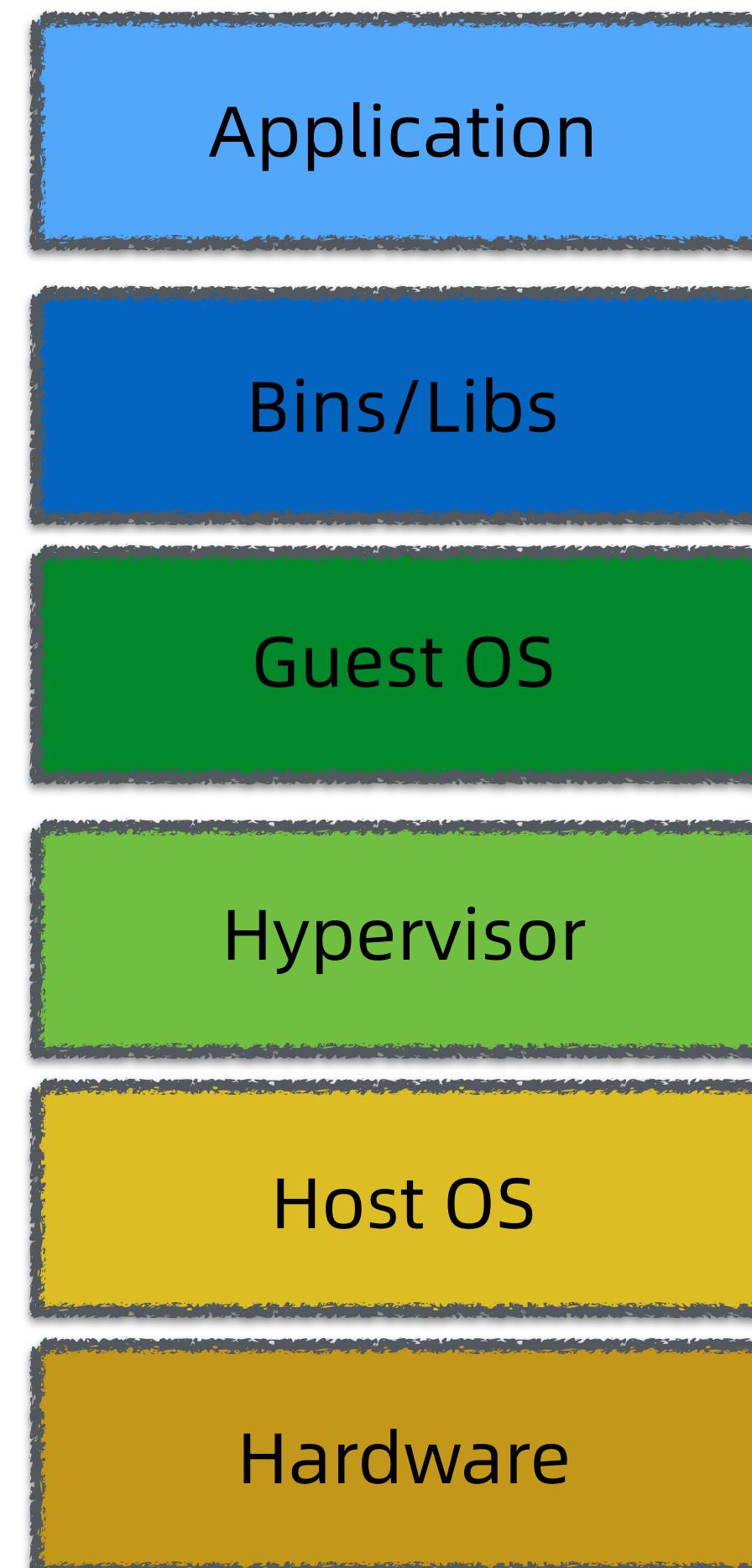


OS & Kernel



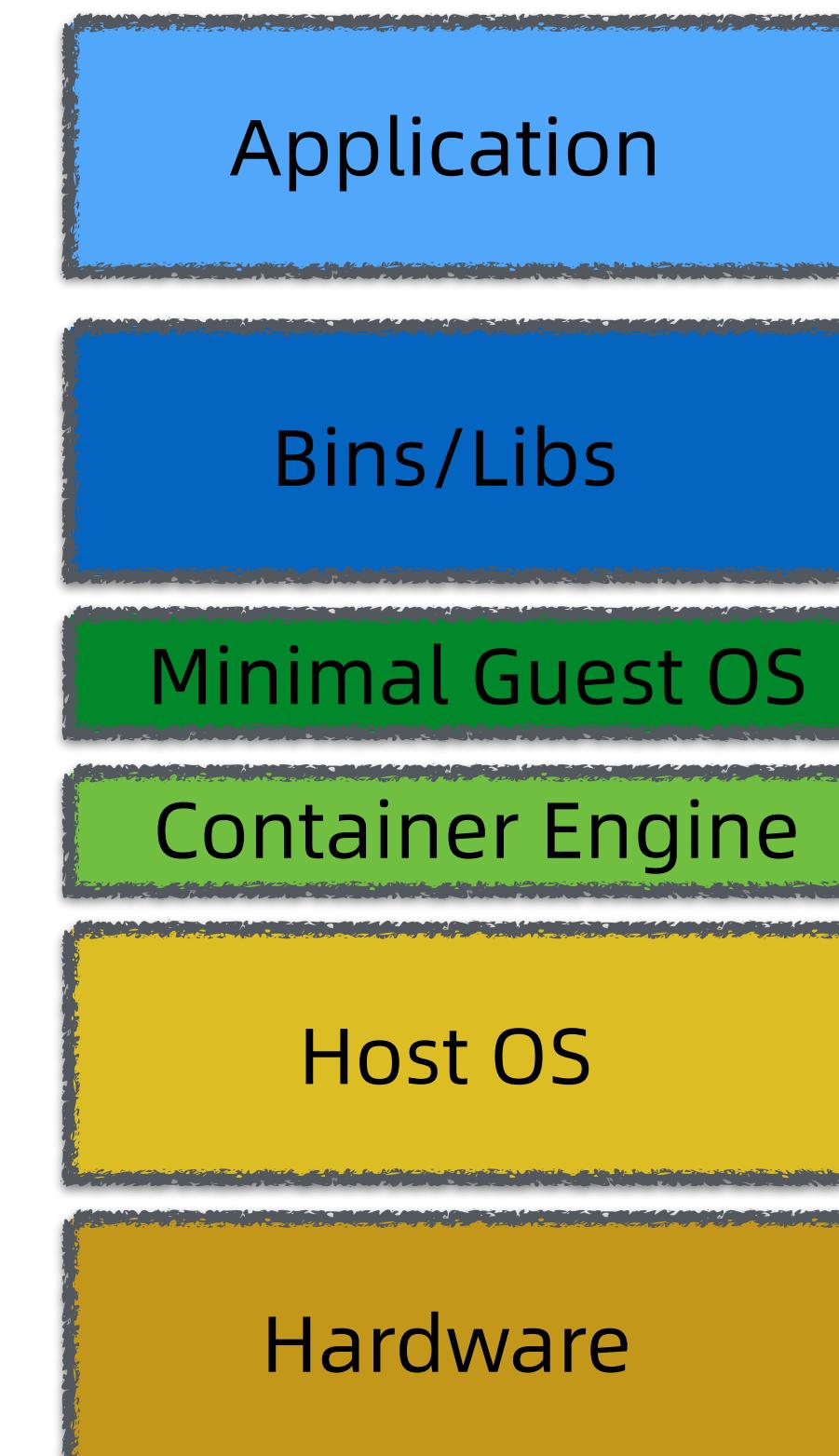
虚拟机和容器

虚拟机



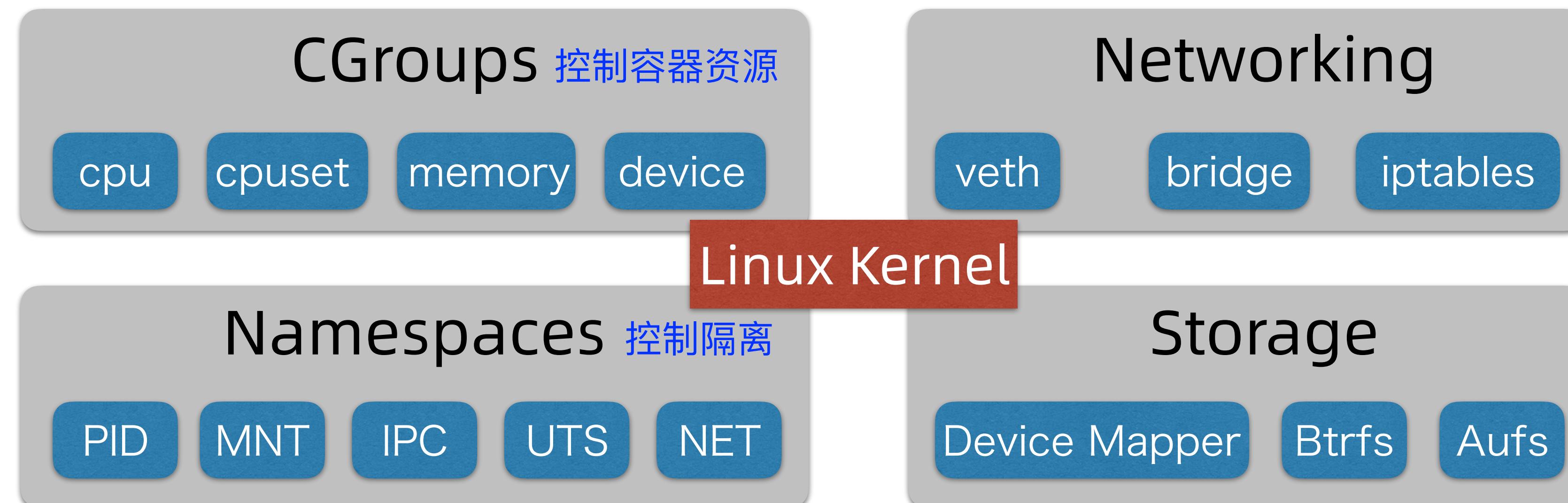
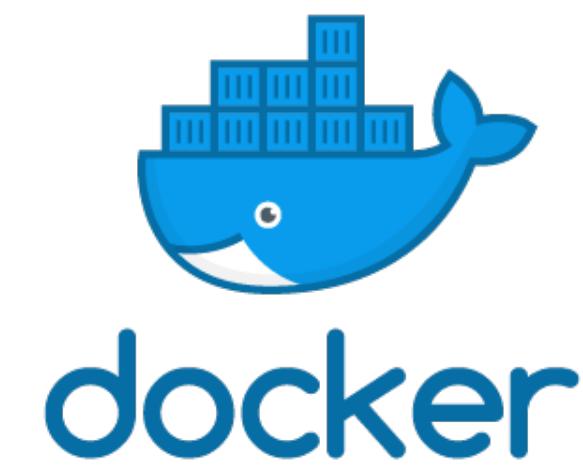
硬件虚拟化 硬件隔离

容器



软件隔离

Docker 容器核心技术



容器镜像

分层(易于重用)

可以认为是固化的“OS”
包含FS和二进制库

Image 1

Thin Read/Write Layer

Application Binary(My Java App)

Apt-get dependencies(Java JDK)

基础镜像(OS)

Image 2

Thin Read/Write Layer

Custom ENV Variables

Configuration
.conf files

3142adfb5ce6

24afa0690fbc

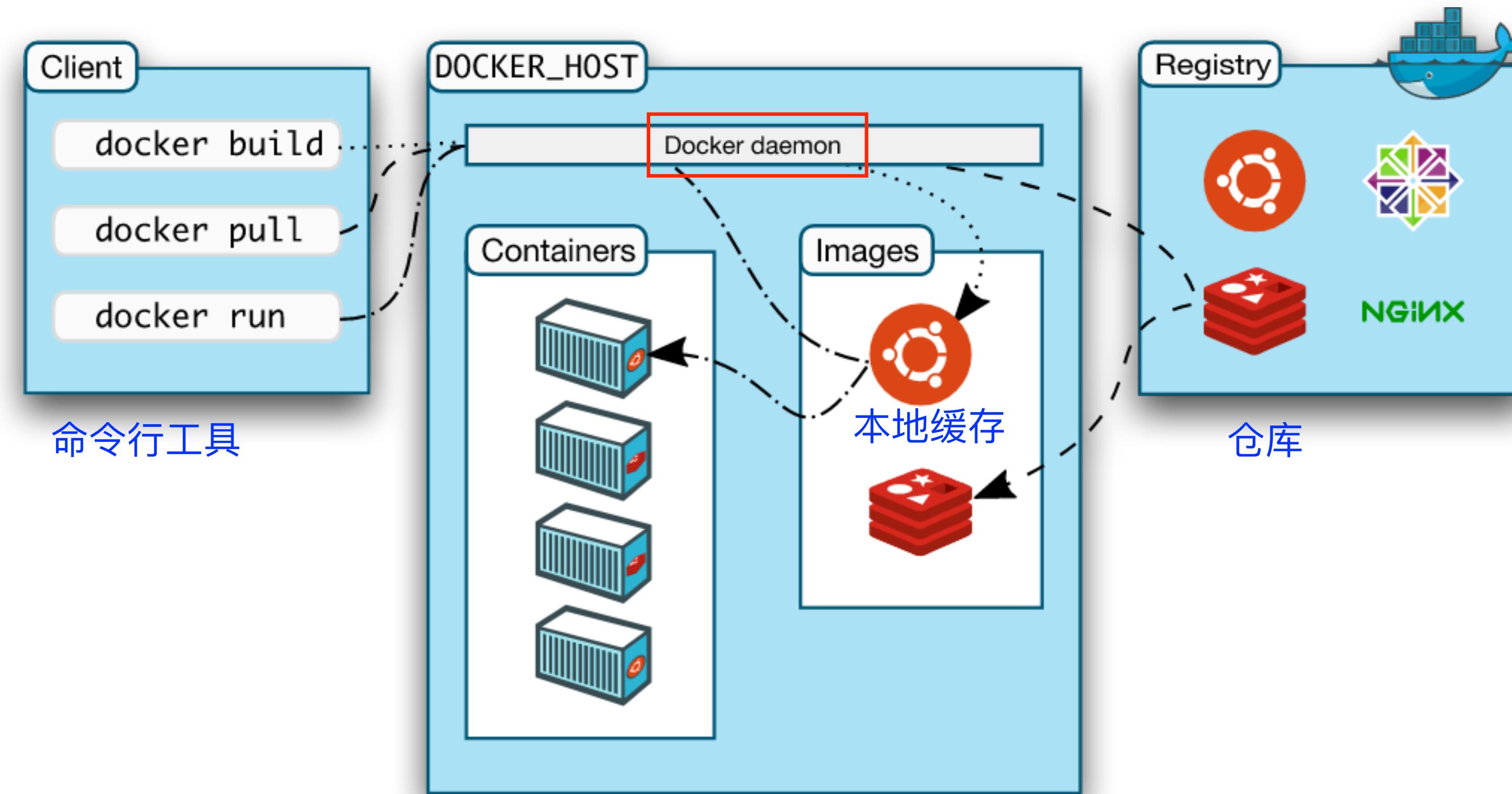
c219a0bb09ec

961467fd729

5287af15dfaf

Docker 架构

3个部分



<https://docs.docker.com/engine/docker-overview/>

Docker Compose

定义和运行多容器应用

\$ docker-compose up

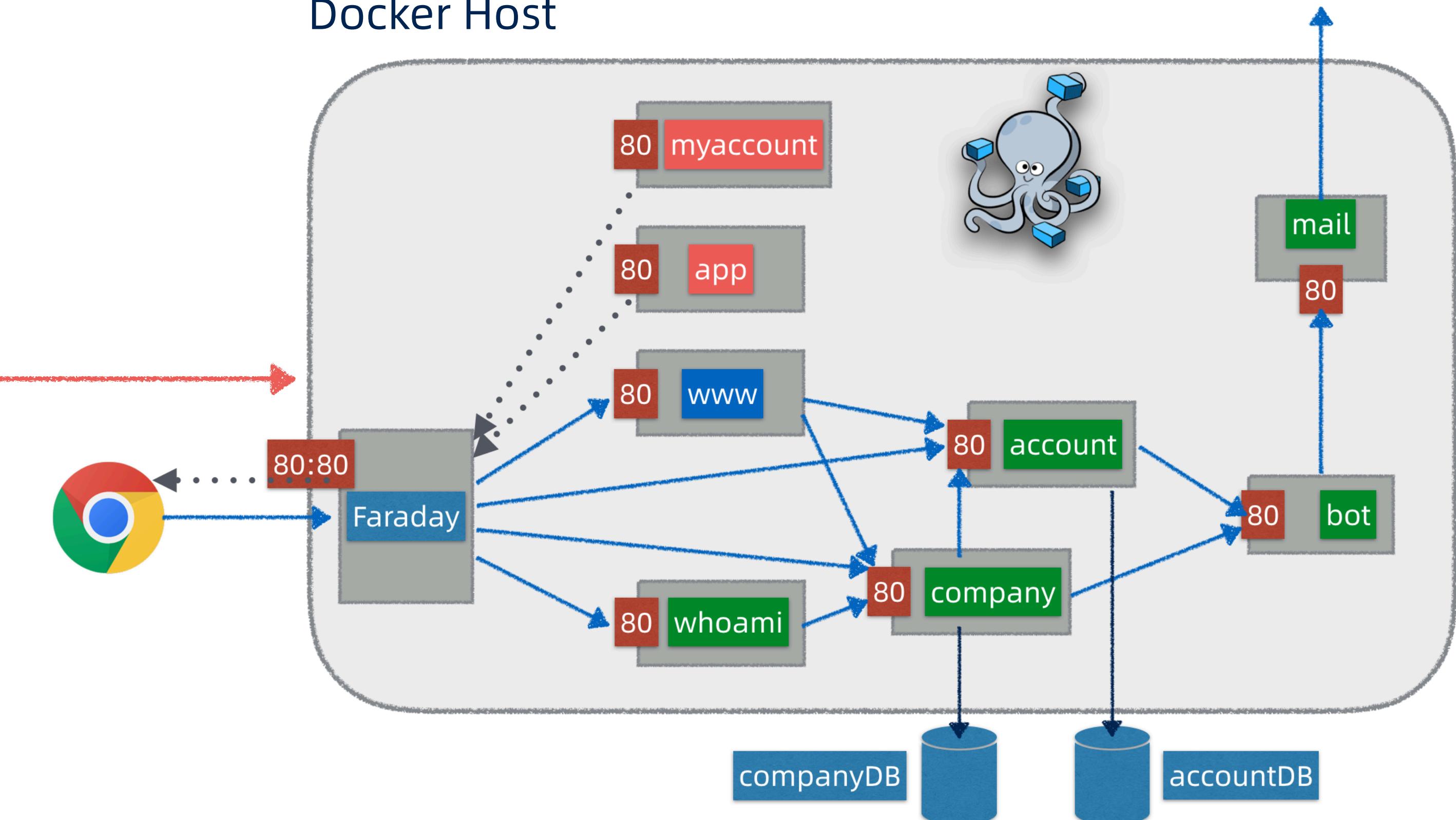
`docker-compose.yml`

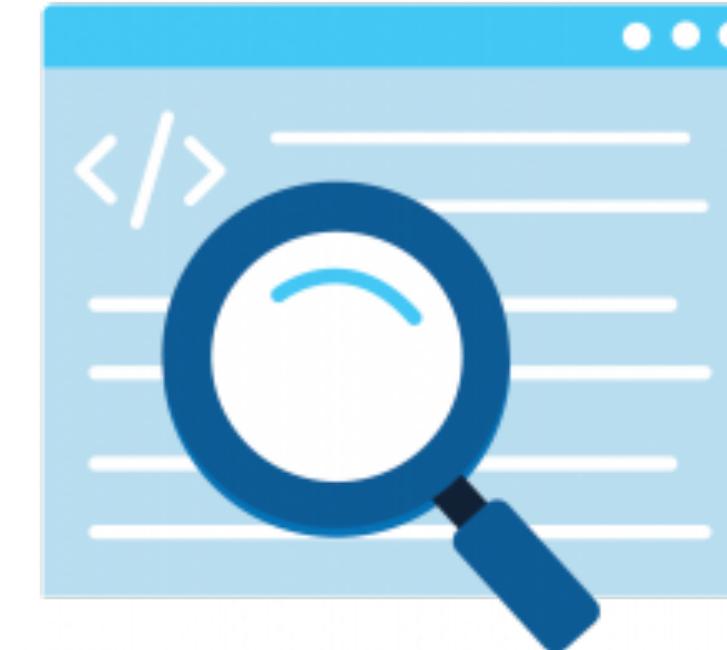
```
version: '3.7'

services:
  account-service:
    build: ./account-svc
    image: boboweike/account-svc
    environment:
      - SPRING_PROFILES_ACTIVE
      - SERVER_PORT
      - SIGNING_SECRET
      - SENTRY_DSN
      - EMAIL_SERVICE_ENDPOINT
      - COMPANY_SERVICE_ENDPOINT
      - BOT_SERVICE_ENDPOINT
      - INTERCOM_ACCESS_TOKEN
      - ACCOUNT_DATASOURCE_URL
      - ACCOUNT_DATASOURCE_USERNAME
      - ACCOUNT_DATASOURCE_PASSWORD
    depends_on:
      - bot-service
      - email-service
    networks:
      - internal_access
      - external_access # db access
```

依赖关系

Docker Host





第 5 部分

容器镜像构建 Dockerfile 解析

Account 服务 Dockerfile

```
FROM java:8-jdk-alpine

COPY ./target/account-svc-1.0.0.jar /usr/app/

WORKDIR /usr/app

ENTRYPOINT ["java", "-jar", "account-svc-1.0.0.jar"]
```

MyAccount 单页应用 Dockerfile

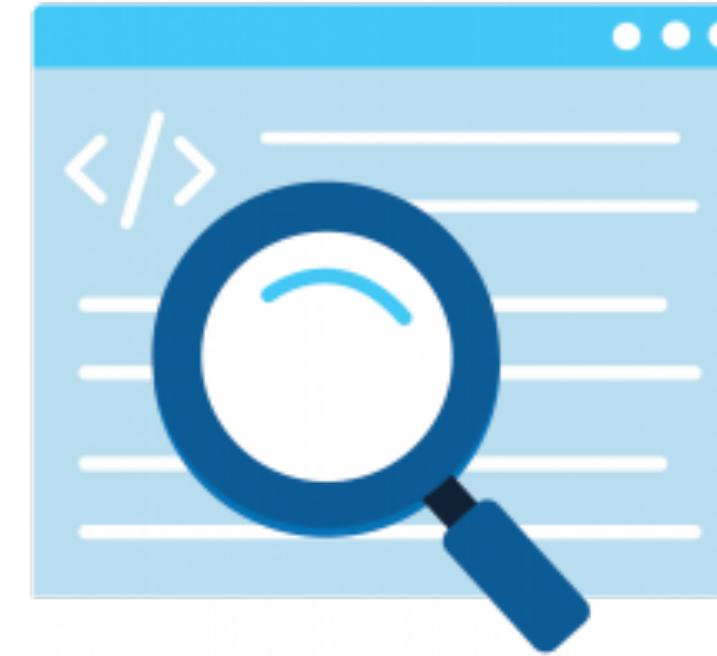
```
FROM node:alpine as builder
WORKDIR '/build'
COPY myaccount ./myaccount
COPY resources ./resources
COPY third_party ./third_party

WORKDIR '/build/myaccount'

RUN npm install
RUN npm rebuild node-sass
RUN npm run build

RUN ls /build/myaccount/dist

FROM nginx
EXPOSE 80
COPY --from=builder /build/myaccount/dist /usr/share/nginx/html
```



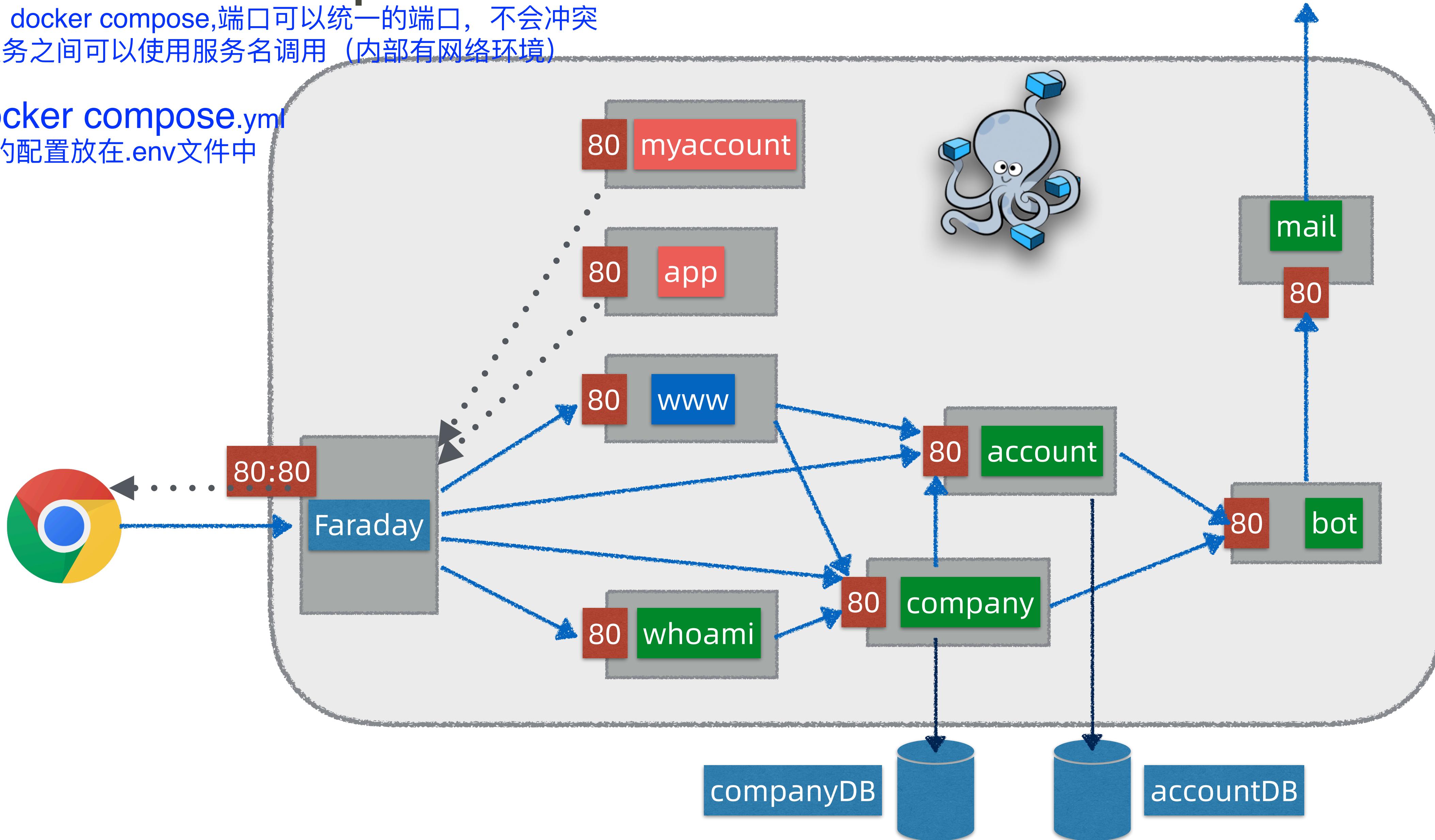
第 6 部分

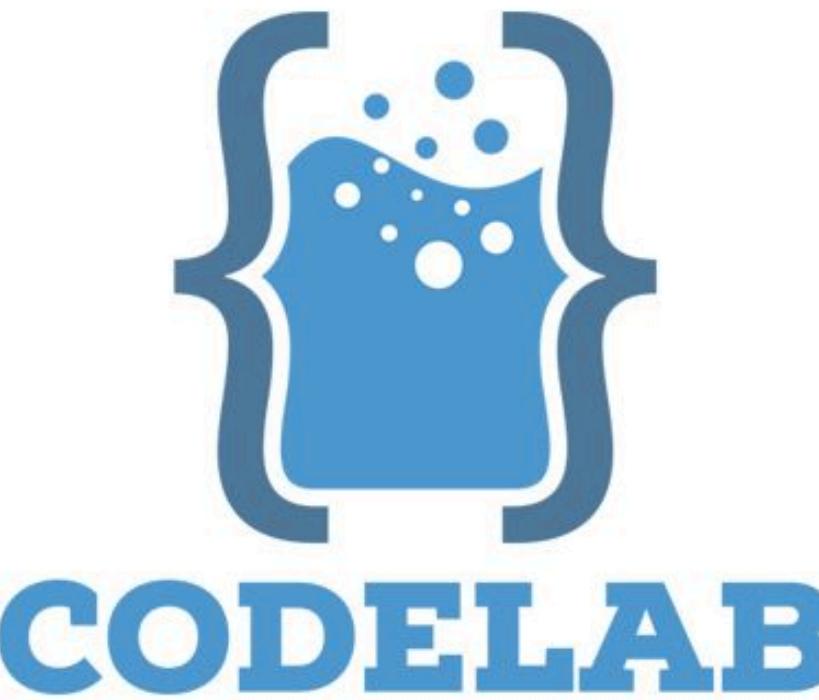
Docker Compose 部署文件解析

Docker Compose 部署架构

利用 docker compose, 端口可以统一的端口, 不会冲突
服务之间可以使用服务名调用 (内部有网络环境)

参见 [docker compose.yml](#)
私密的配置放在.env文件中

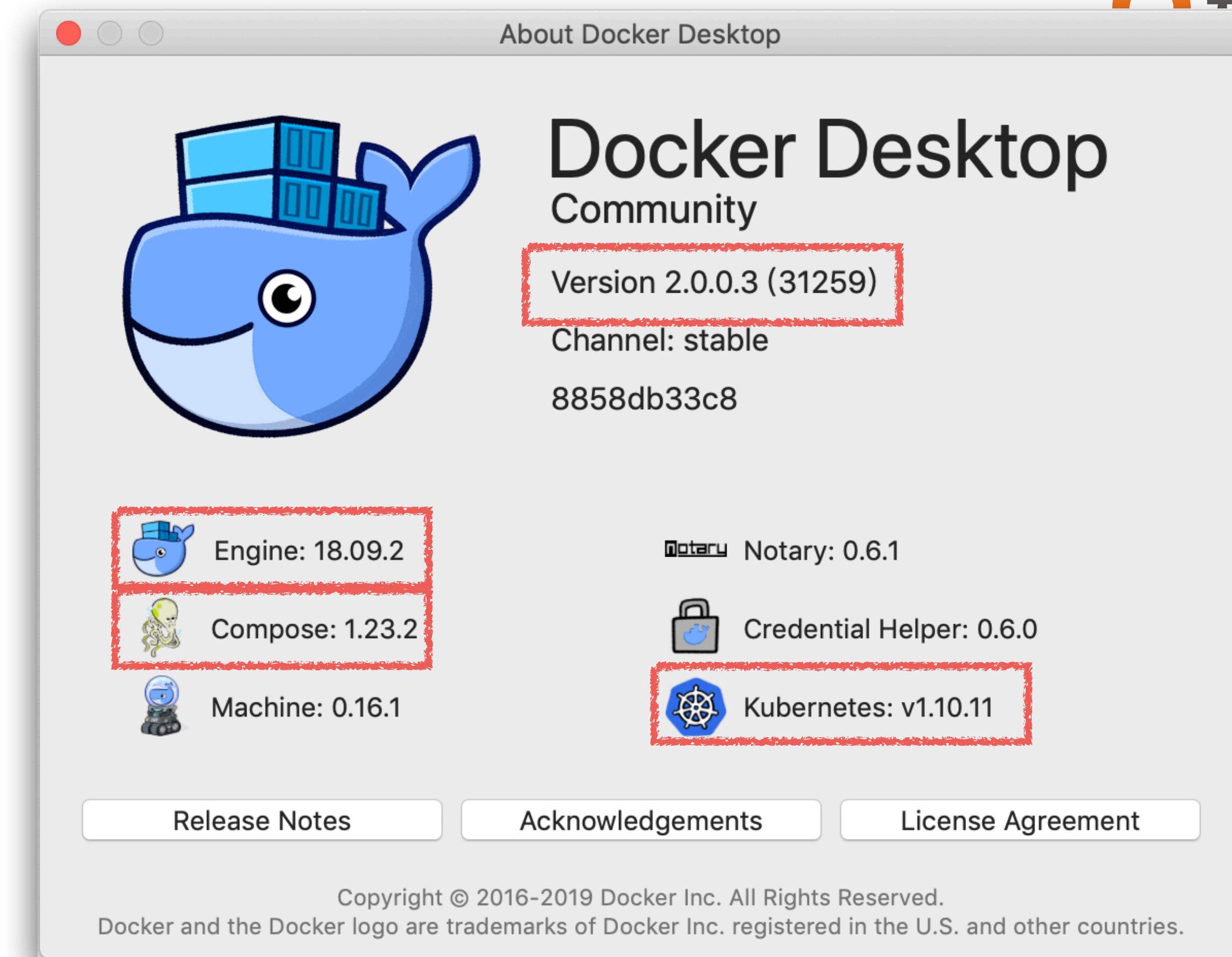




第 7 部分

将 Staffjoy 部署到本地 Docker Compose 环境

Docker Desktop for Mac/Win



<https://docs.docker.com/docker-for-mac/install/>

<https://docs.docker.com/docker-for-windows/install/>

构建和部署

Step By Step

1. 镜像构建

- mvn clean package -DskipTests
- docker-compose build
- docker images

2. 部署 MySQL 数据库

- staffjoy_account
- staffjoy_company

3. 部署 Staffjoy

- docker-compose up
- docker-compose ps

4. 启用 SwitchHosts

5. 校验 Staffjoy

6. 清理

- docker-compose down



扫码试看/订阅

《Spring Boot & Kubernetes 云原生微服务实践》