

Composing Applications with docker-compose



Wes Higbee

COFFEE CONNOISSEUR

@g0t4 www.weshigbee.com



From Commands to docker-compose.yml

```
docker run --name db `
  -d `
  -p 3306:3306 `
  -e MYSQL_ROOT_PASSWORD=my-secret-pw `
  -v db:/var/lib/mysql `
  mysql
```

```
docker inspect db # extract ip address
```

```
docker run --name web `
  -d `
  -p 8080:80 `
  -e MY_DB_PORT=3306 `
  -e MY_DB_HOST=? `
  -v /my/php/app:/usr/share/nginx/html `
  nginx
```



From Commands to docker-compose.yml

```
docker run --name db -d -p 3306:3306 -e MYSQL_ROOT_PASSWORD=my-secret-pw -v  
db:/var/lib/mysql mysql
```

```
docker inspect db # extract ip address
```

```
docker run --name web -d -p 8080:80 -e MY_DB_PORT=3306 -e MY_DB_HOST=? -v  
/my/php/app:/usr/share/nginx/html nginx
```



From Commands to docker-compose.yml

version: '2'

services:

db:

image: mysql

ports:

- 3306:3306

environment:

- MYSQL_ROOT_PASSWORD=my-secret-pw

volumes:

- db:/var/lib/mysql

web:

image: nginx

ports:

- 8080:80

environment:

- MY_DB_PORT=3306

- MY_DB_HOST=db

volumes:

- /my/php/app:/usr/share/nginx/html



docker-compose.yml



Key Takeaways



Dockerfile : images

docker-compose : containers (and more)

User defined networks

