

# Docker 101

Date **Monday, April 27, 2020 - Thursday, April 30, 2020** Duration **2 hours per day** Time **14:00 - 16:00**

Level **Beginner**

Format **Coding Workshop**

Computer required **YES with Windows 10 Pro or MacOS or Linux 32Bits / 64Bits**

Audience **Manager, Analyst, Programmer, Tester, QA, Project Manager and Interesting Person**

Monday, April 27, 2020	Tuesday, April 28, 2020	Wednesday, April 29, 2020	Thursday, April 30, 2020
<b>Overview</b> <ul style="list-style-type: none"><li>- From Physical to Visual</li><li>- Containerize concept</li><li>- Docker Overview<ul style="list-style-type: none"><li>- Dockerfile</li><li>- Image</li><li>- Container</li></ul></li><li>- Run First Container<ul style="list-style-type: none"><li>- Daemon mode</li></ul></li><li>- Stop, Start, Restart, Remove</li></ul>	<b>Build First Image</b> <ul style="list-style-type: none"><li>- FROM</li><li>- ADD,COPY</li><li>- RUN</li><li>- CMD</li><li>- ENTRYPOINT</li><li>- VOLUME</li><li>- EXPOSE</li></ul> <b>Publish First Image</b> <ul style="list-style-type: none"><li>- Tag</li><li>- Docker hub</li></ul> <b>Manage Image</b> <ul style="list-style-type: none"><li>- Tag</li><li>- Docker hub</li></ul>	<b>Investigate containers</b> <ul style="list-style-type: none"><li>- History</li><li>- Inspect</li></ul> <b>Run Multi-containers</b> <ul style="list-style-type: none"><li>- Docker compose</li><li>- Up</li><li>- Down</li><li>- Build</li><li>- Scale</li></ul>	<b>Healthy Container</b> <ul style="list-style-type: none"><li>- health check</li></ul> <b>Image Optimization</b> <ul style="list-style-type: none"><li>- Keep small</li><li>- Multi-stage build</li></ul> <b>More Use cases</b>



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

สยามชั้นนำคิท  
SIAM CHANNON KIT



Shu Ha Ri



We Love Bug

Docker 101

April 2020





Shu Ha

Ri



SEAL



We Love

Bug



# Agenda

- Why Docker?
- Concept
- Run your first container
- Build your first image
- Share image to the world
- Run multi-containers

Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Why Docker



# Let's talk about History



<https://brewminate.com/wp-content/uploads/2020/02/GeorgeThompsonTeachingHistory01.png>



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Look back to 90s - 2000s



# Physical Server Era



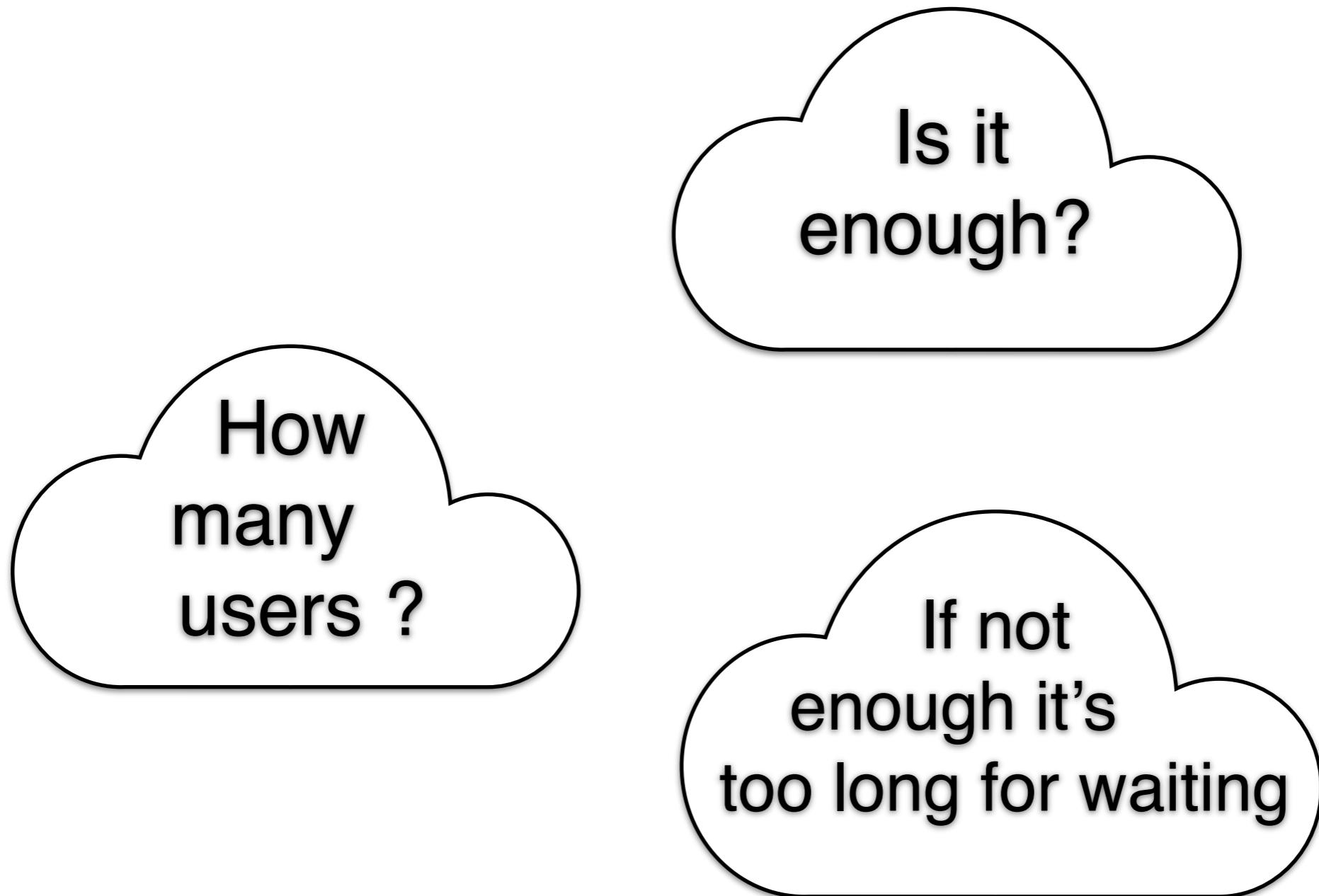
CPU 16 Core  
RAM 32 GB  
HDD 100 GB

Price **250,000.00 Baht**  
Shipping in 1-6 Months

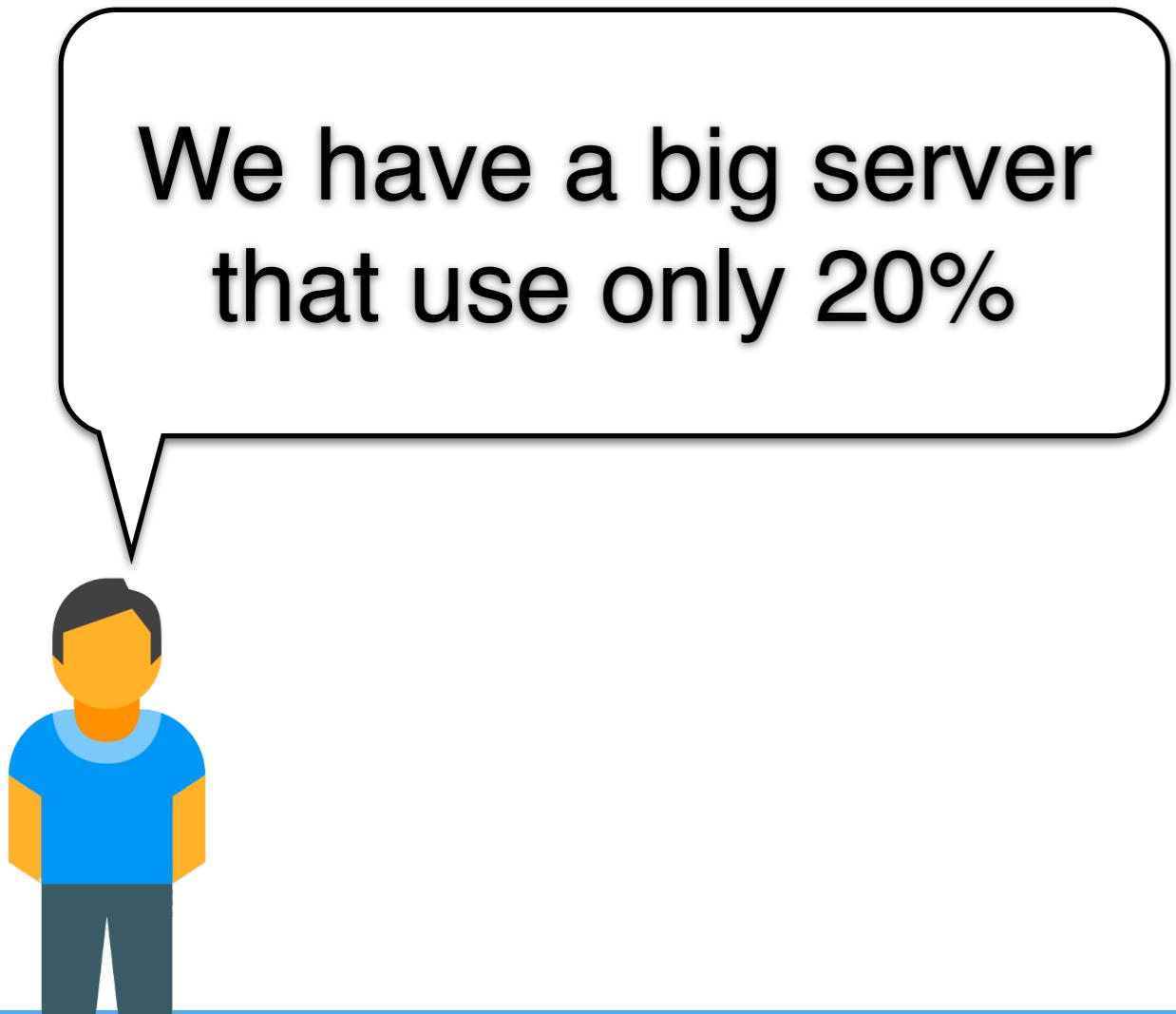
OMG!!



# Physical Server Era



# Shared Server Era



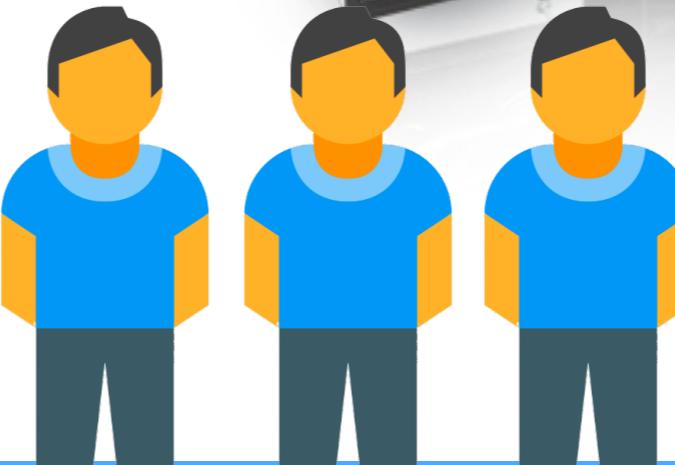
# Shared Server Era

We have a big server  
that use only 20%



Can we share?

Sure



# Shared Server Era

CPU 16 Core  
RAM 32 GB  
HDD 100 GB



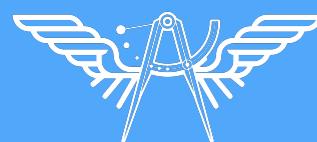
Java 1.8  
NodeJS 6.0  
Python 2.7

I need Java 1.8

I need Java 1.10  
and NodeJS 6.0

NodeJS 6.0 is too old,  
I need 10.0 to enable  
features

This Python is 2.7 but I need  
3.0. If I change Someone  
will broke ?



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Virtual Machine Era

I need Java 1.8



NodeJS 6.0 is too old,  
I need 10.0 to enable  
features



I need Java 1.10  
and NodeJS 6.0



This Python is 2.7 but I need  
3.0. If I change Someone  
will broke ?



CPU 16 Core  
RAM 32 GB  
HDD 100 GB



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Virtual Machine Era

I need Java 1.8



NodeJS 6.0 is too old,  
I need 10.0 to enable  
features



I need Java 1.10  
and NodeJS 6.0



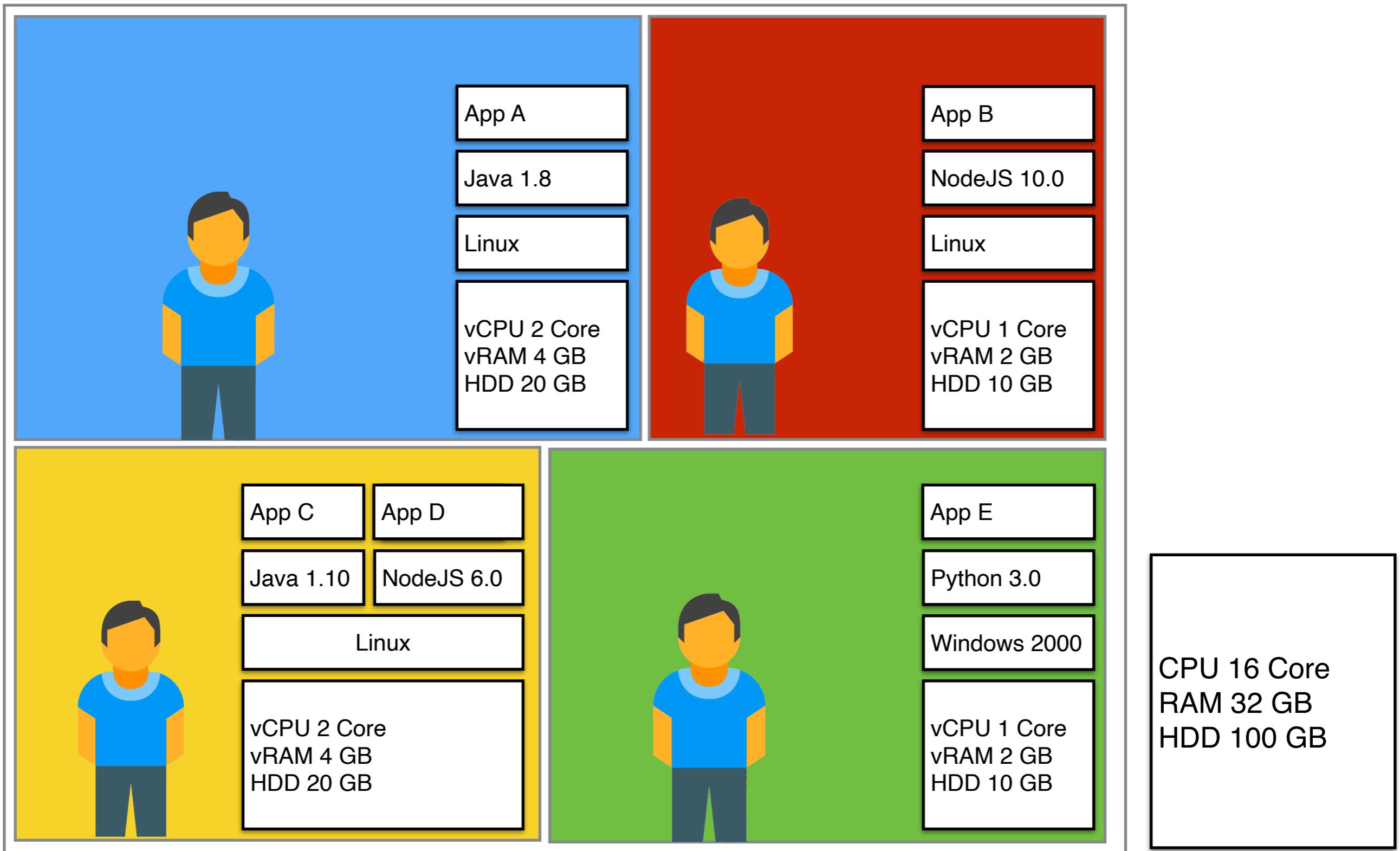
This Python is 2.7 but I need  
3.0. If I change Someone  
will broke ?



CPU 16 Core  
RAM 32 GB  
HDD 100 GB



# Virtual Machine Era

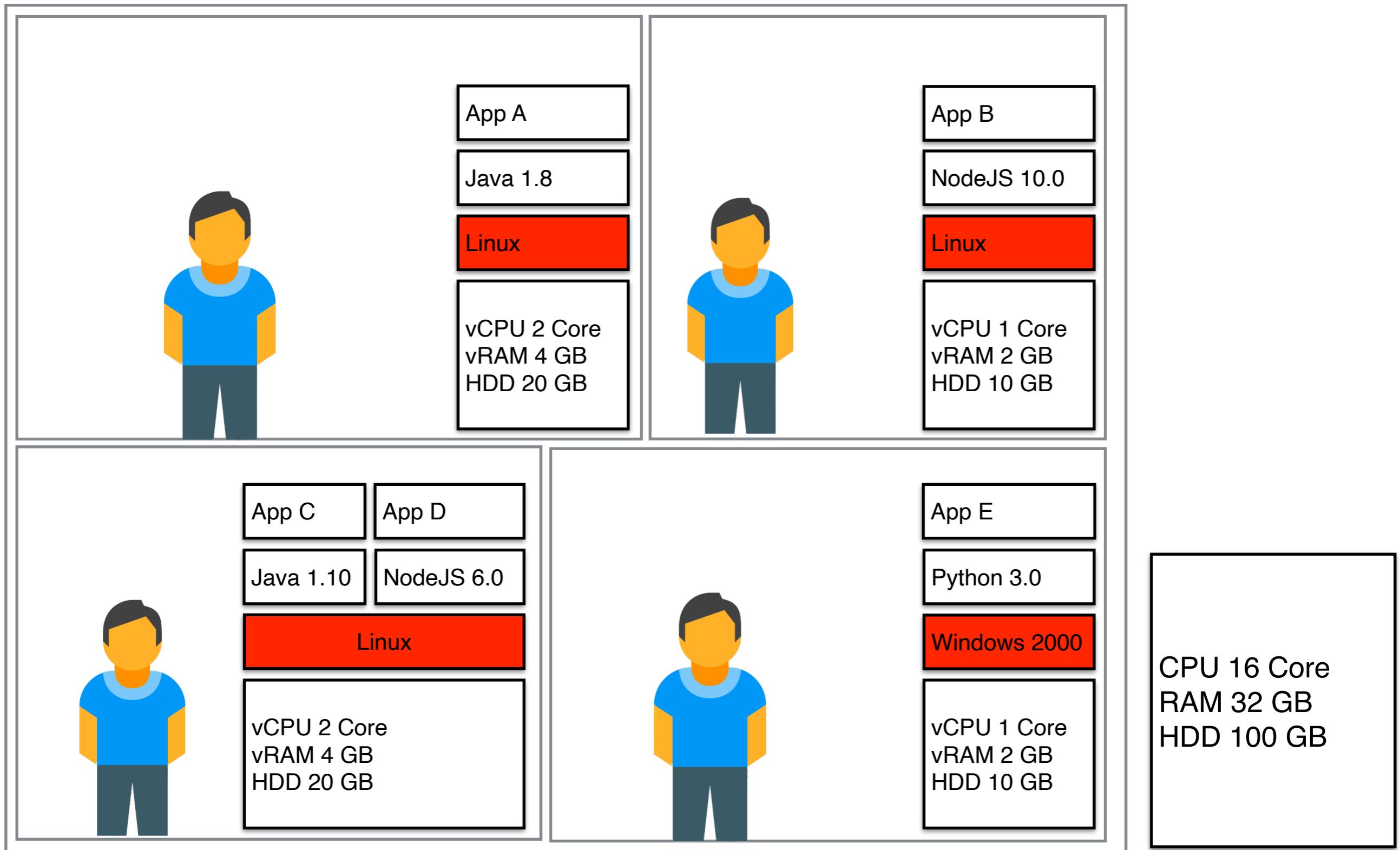


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Virtual Machine Era

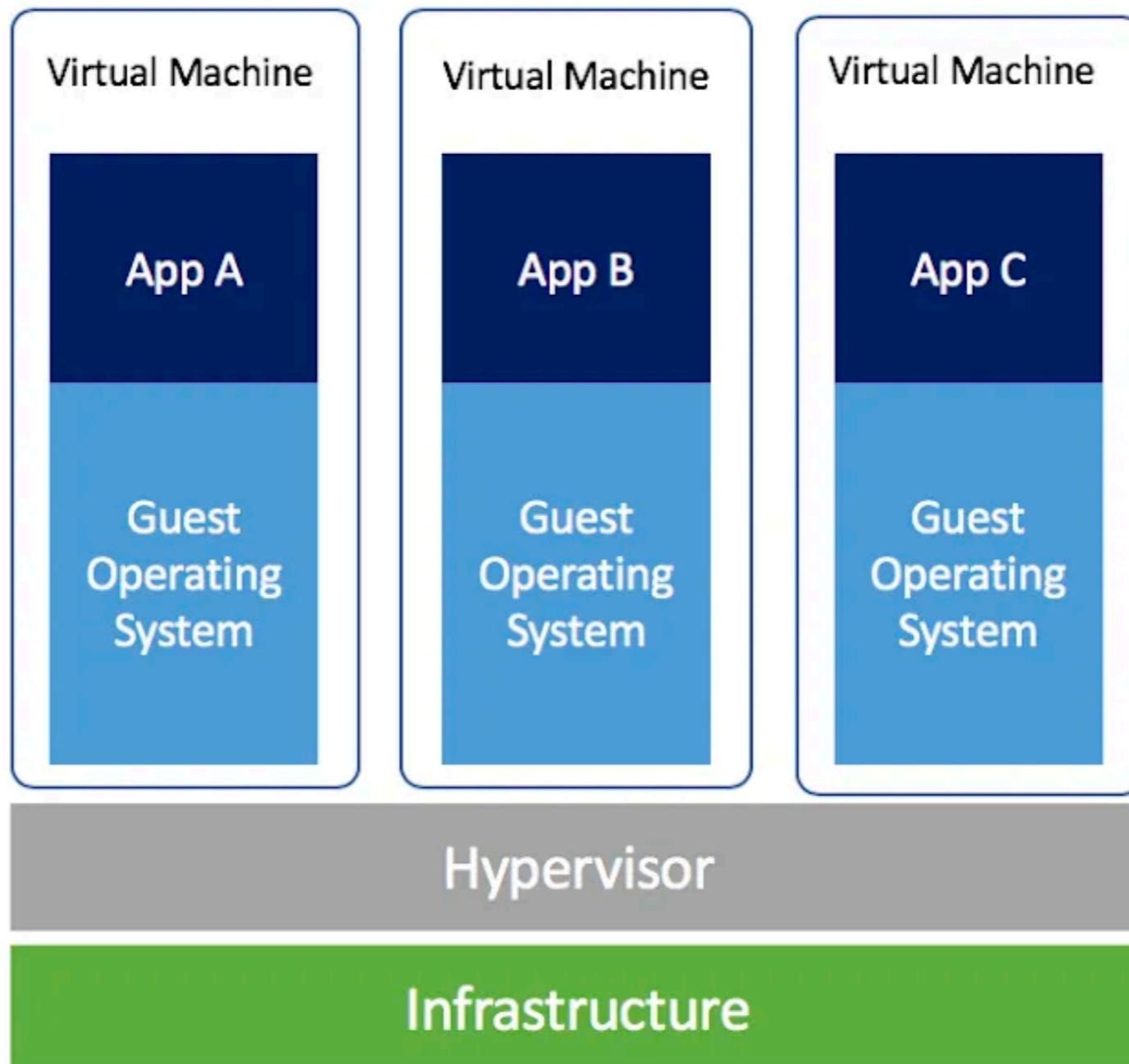


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Virtual Machine Era

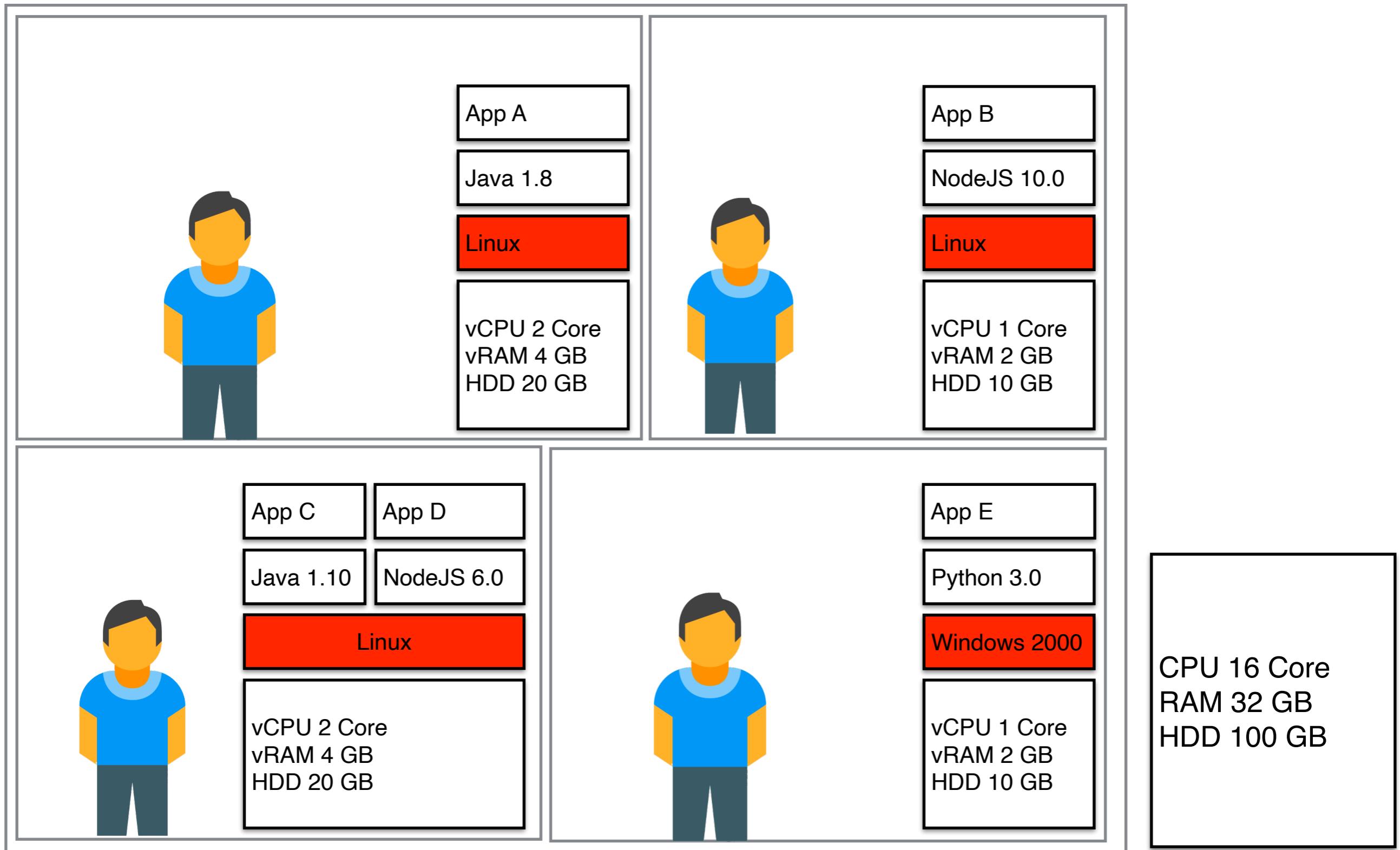


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Container Era

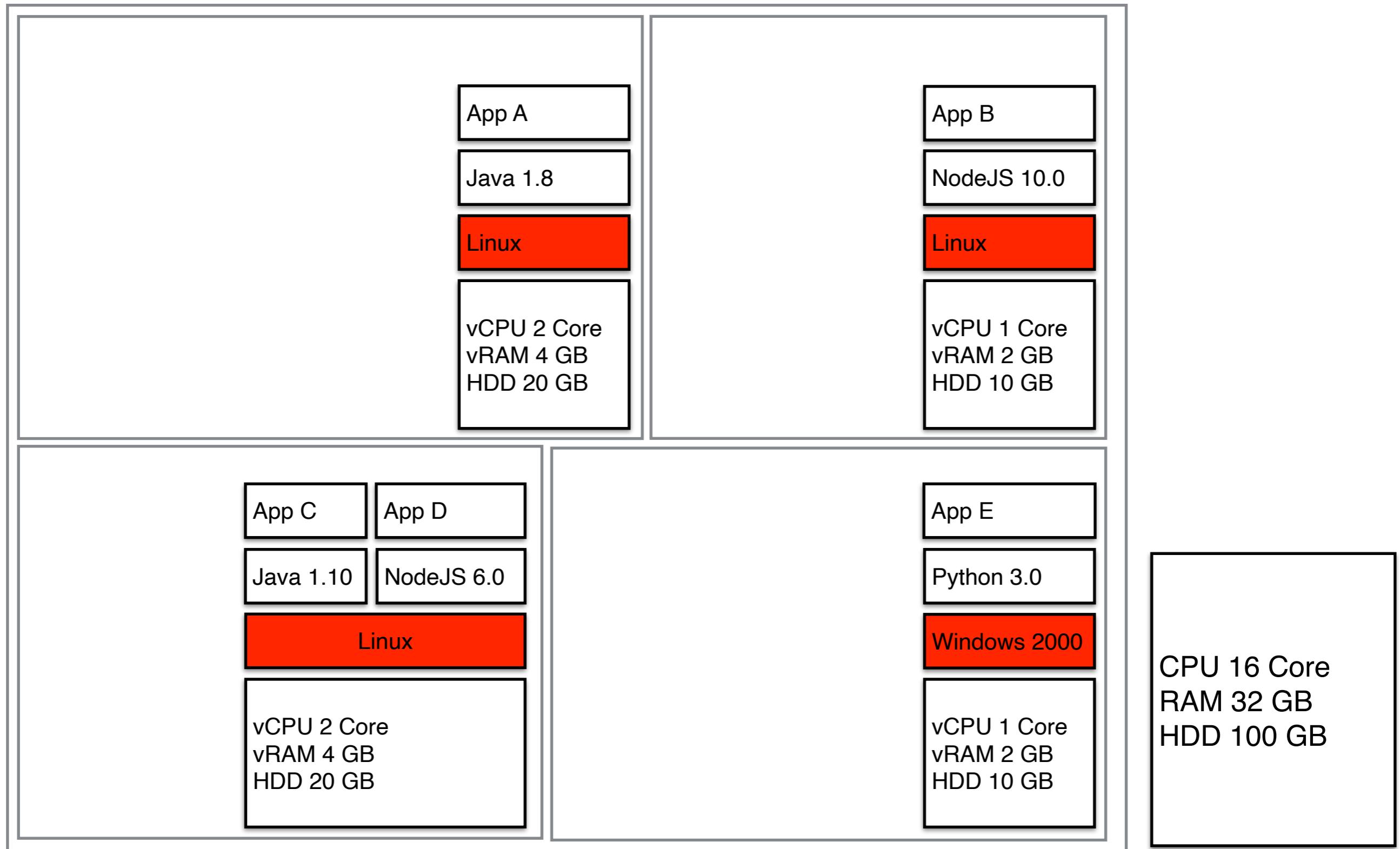


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Container Era



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Container Era

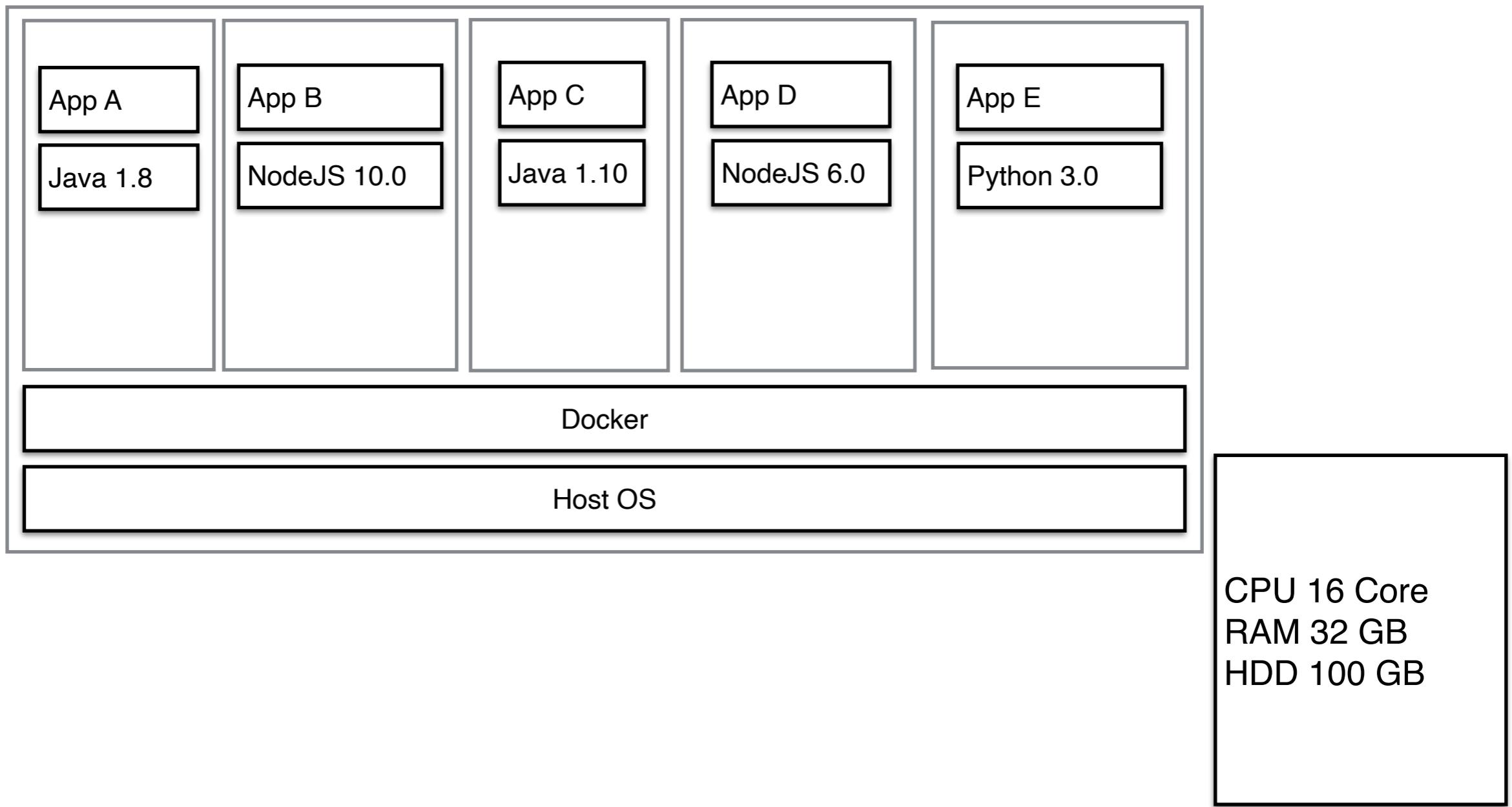


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

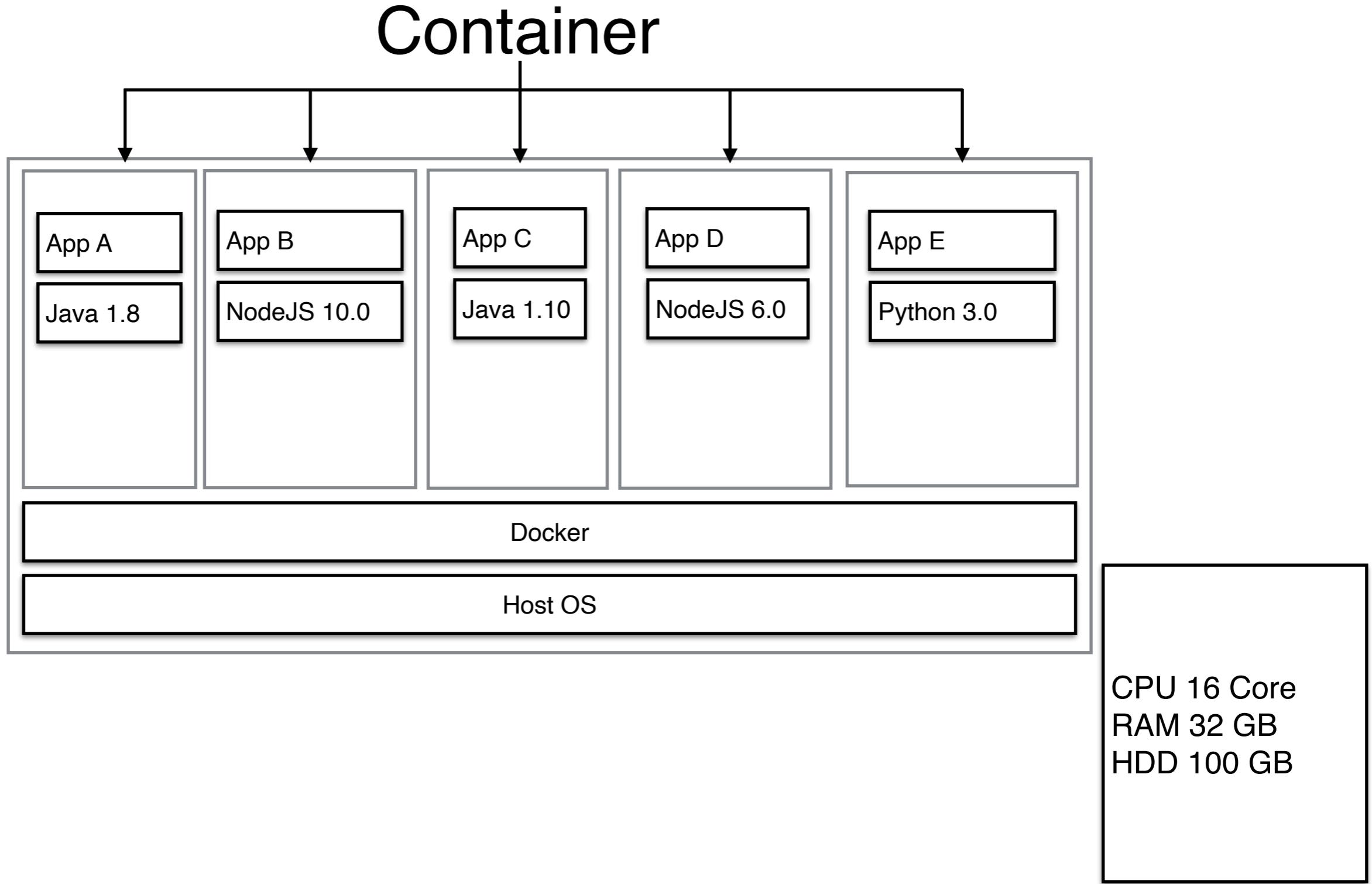
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

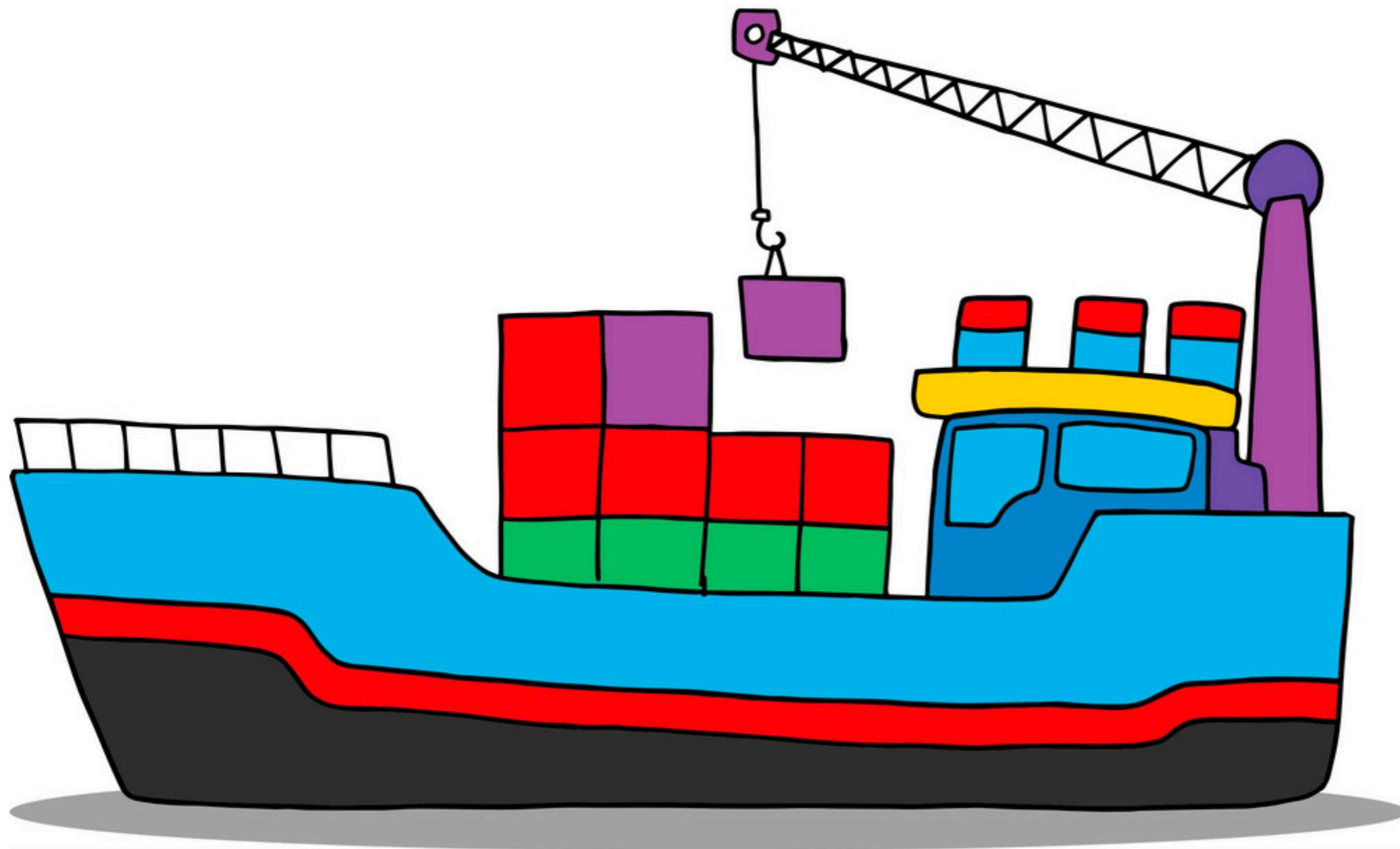
# Container Era



# Container Era



# Container Era



<https://cdn3.vectorstock.com/i/1000x1000/29/37/cartoon-of-container-ship-vector-12882937.jpg>

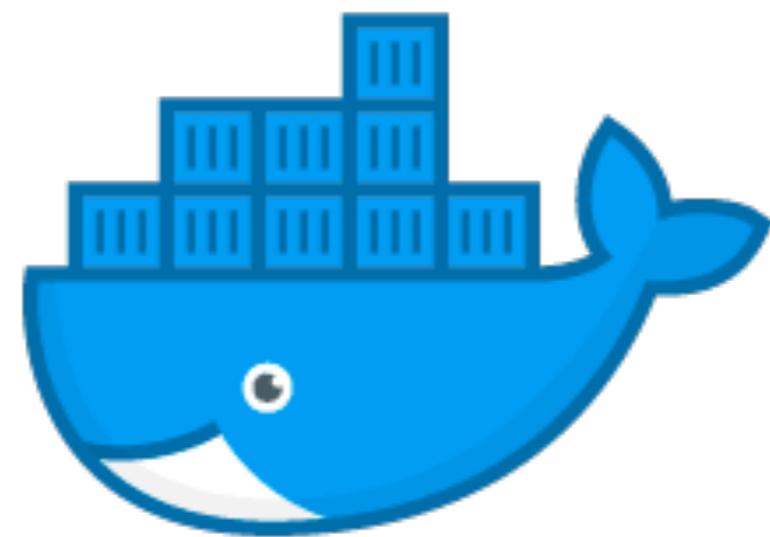


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Container Era



docker

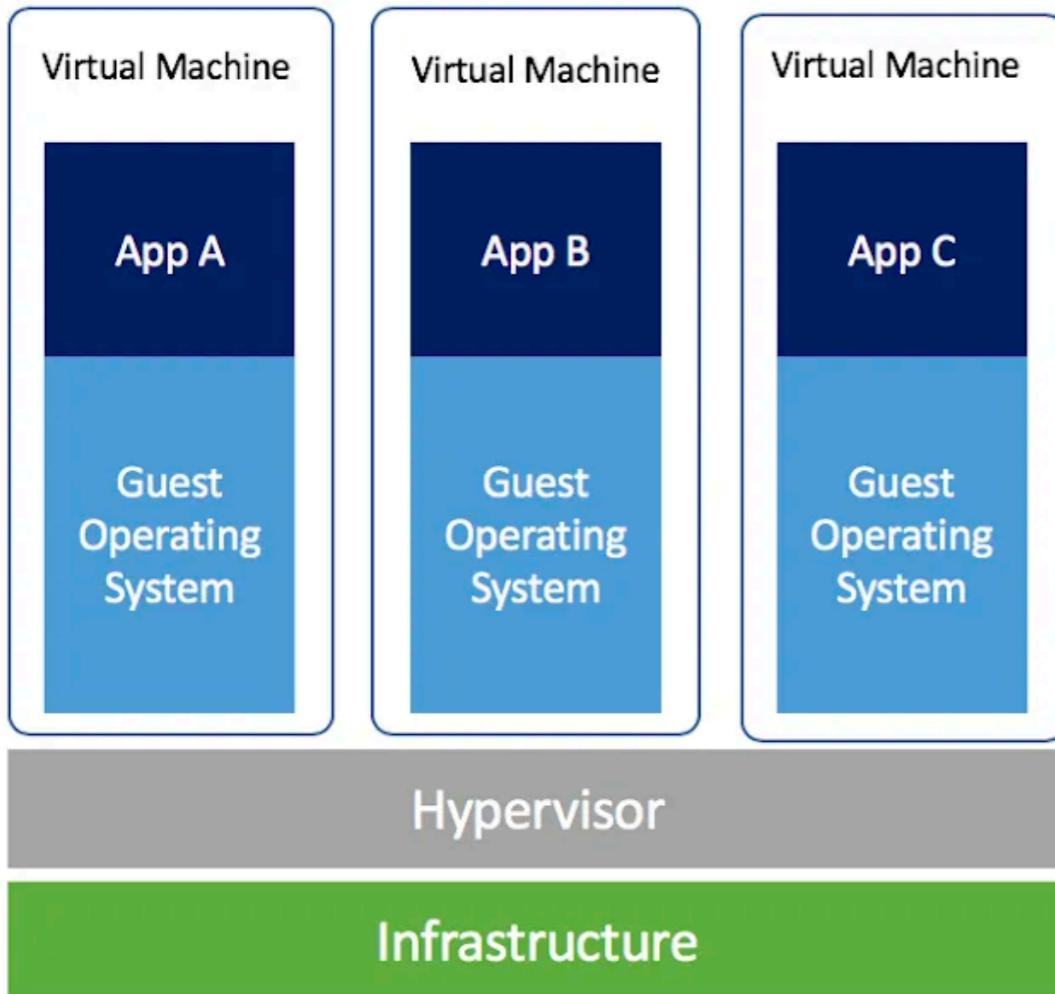


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

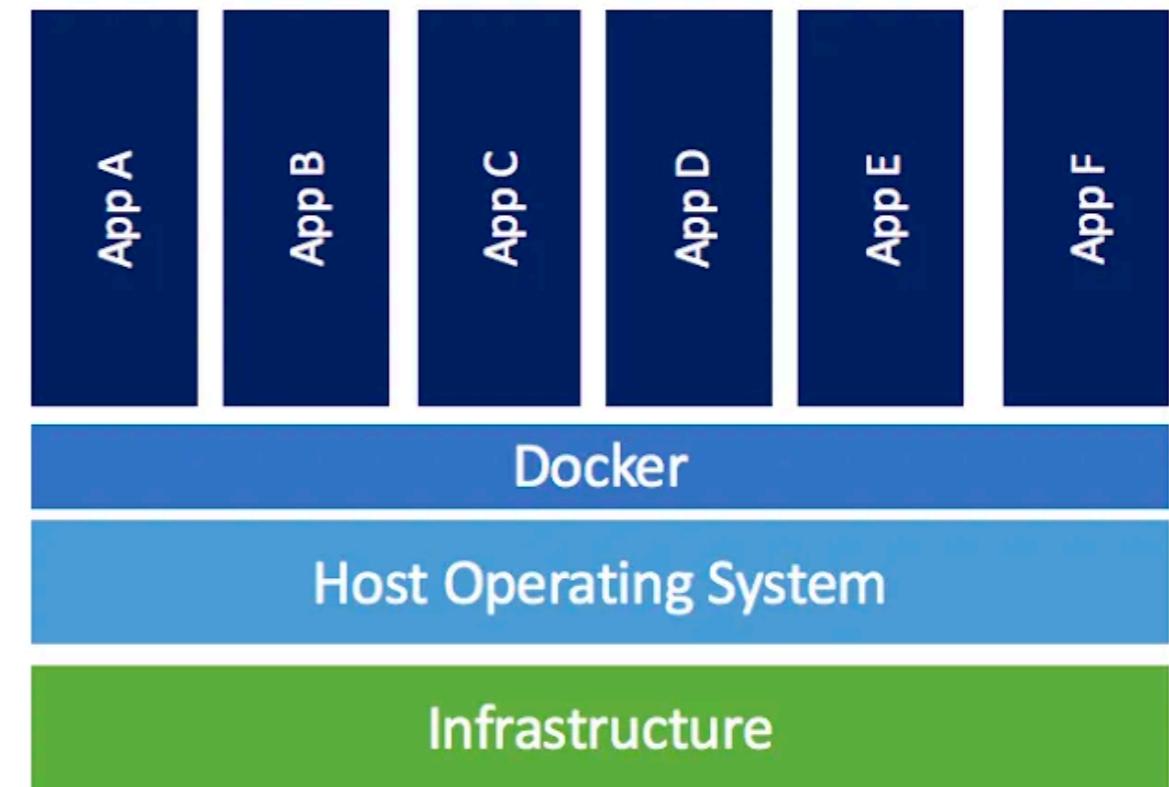
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Container Era



Containerized Applications



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

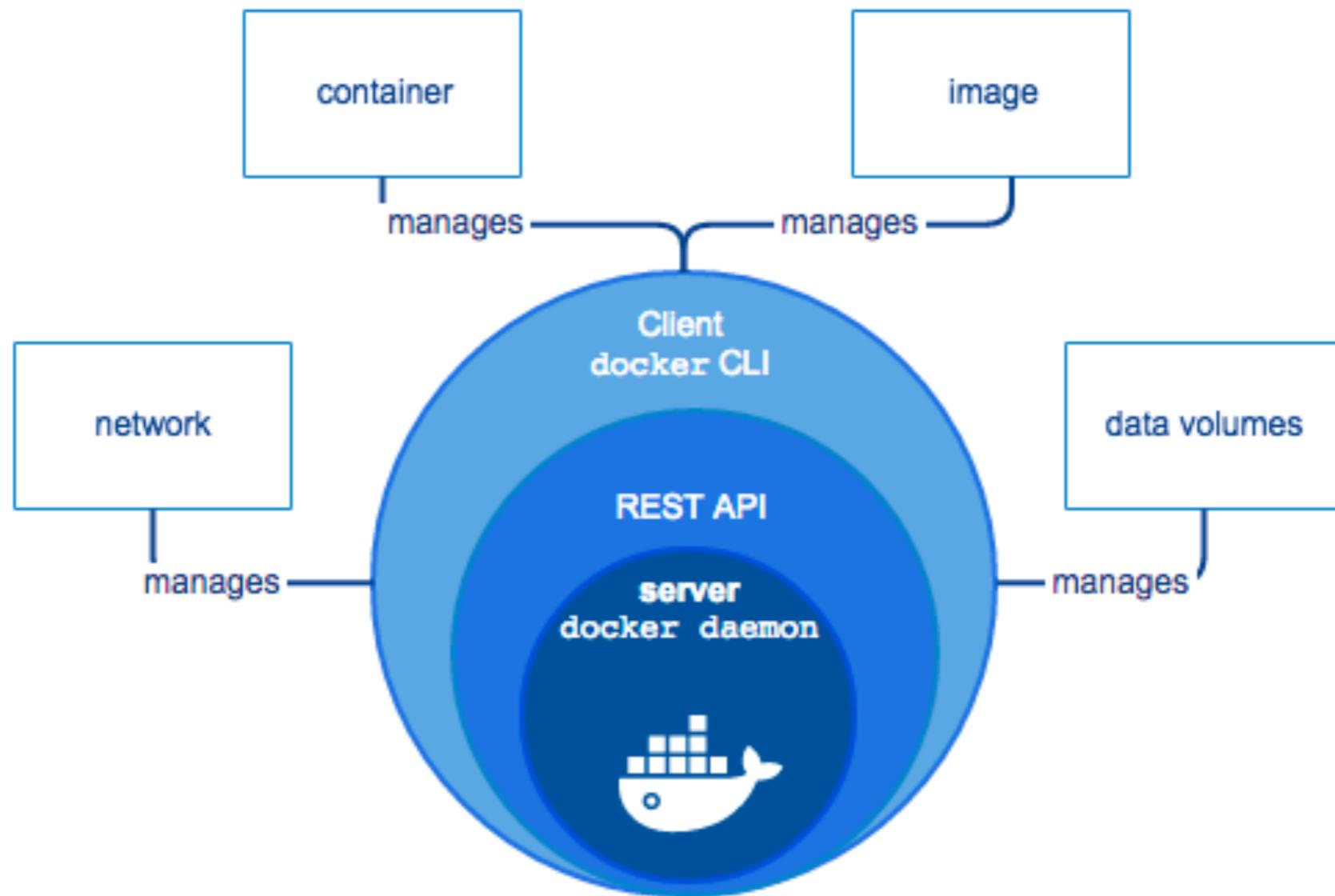
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Concept

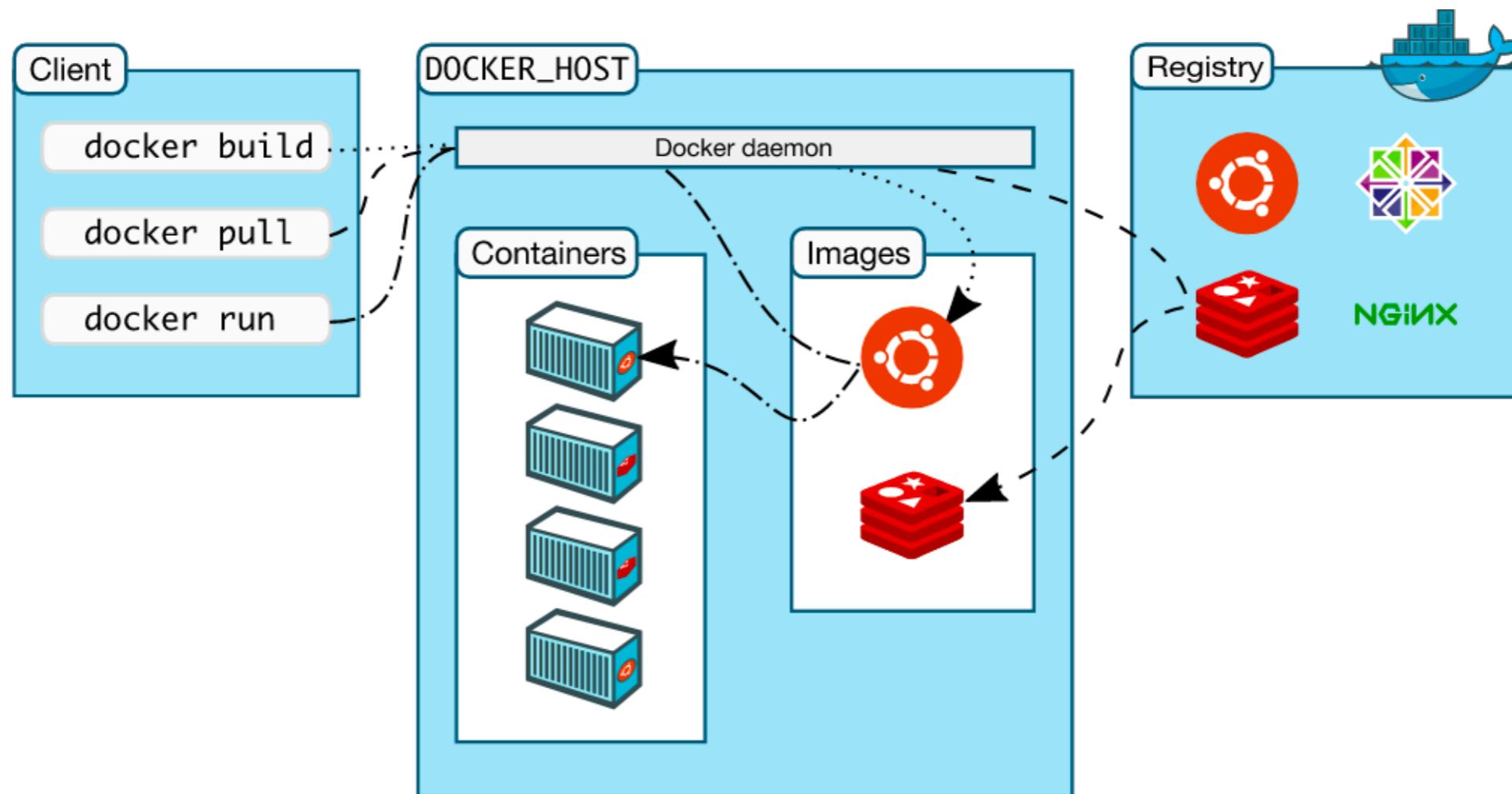


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

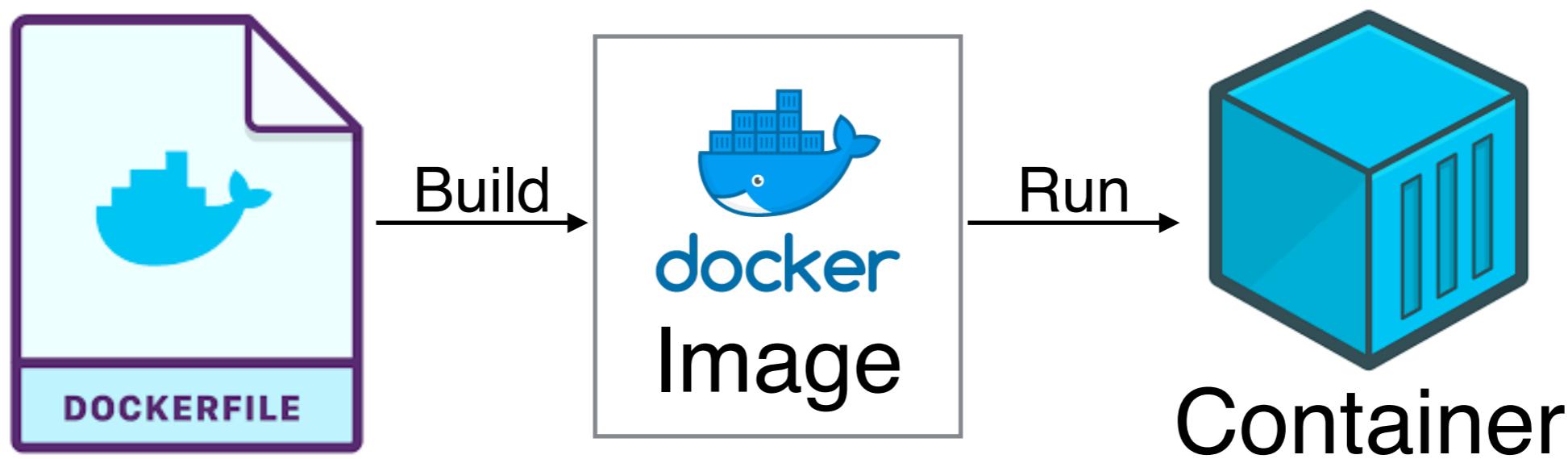
# Docker Engine



# Docker Architecture



# Dockerfile, Image, Container



# Installation



# Installation

For Windows

Windows 10 64-bit: Pro, Enterprise

Hyper-V and Containers Windows features must be enabled

RAM: 4 GB+

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

For Mac

MacBook 2010+

macOS 10.13+

RAM: 4 GB+

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



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

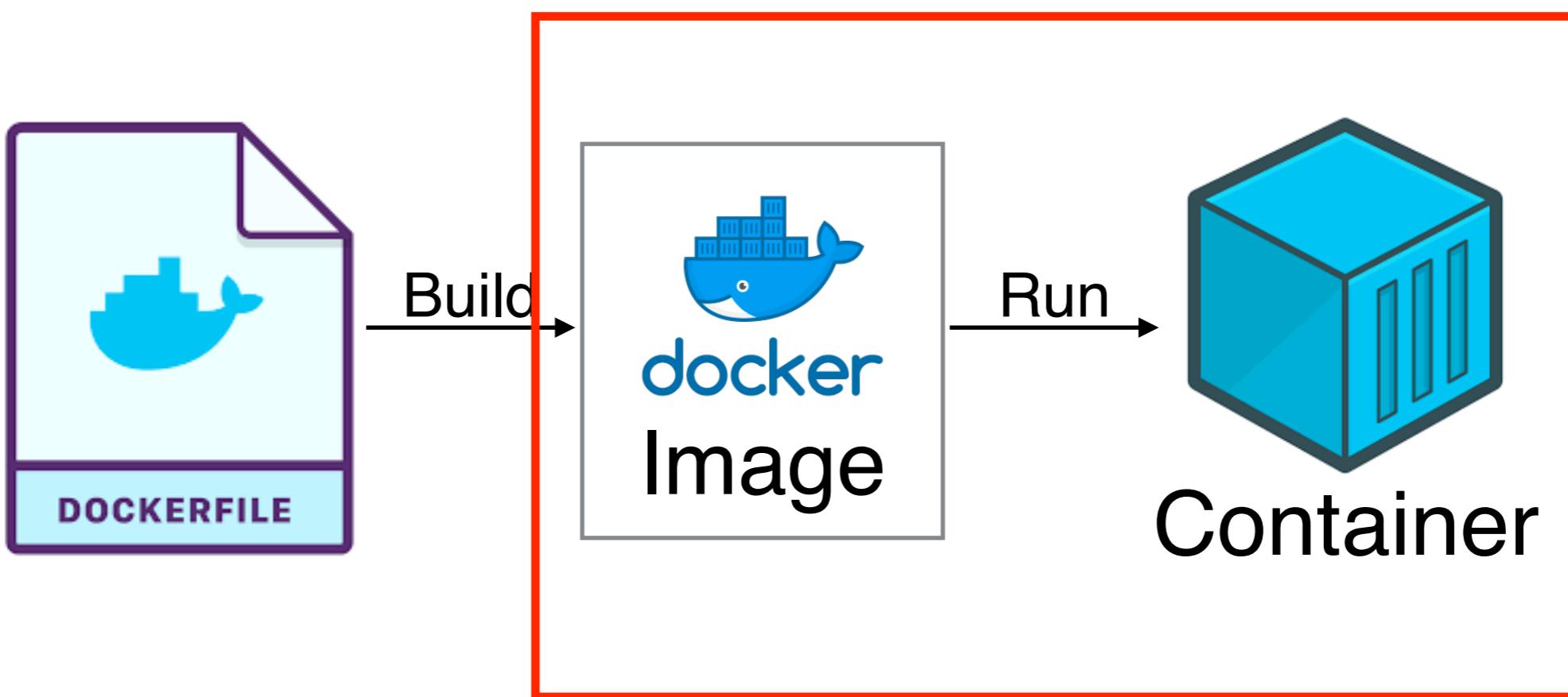
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run your first container



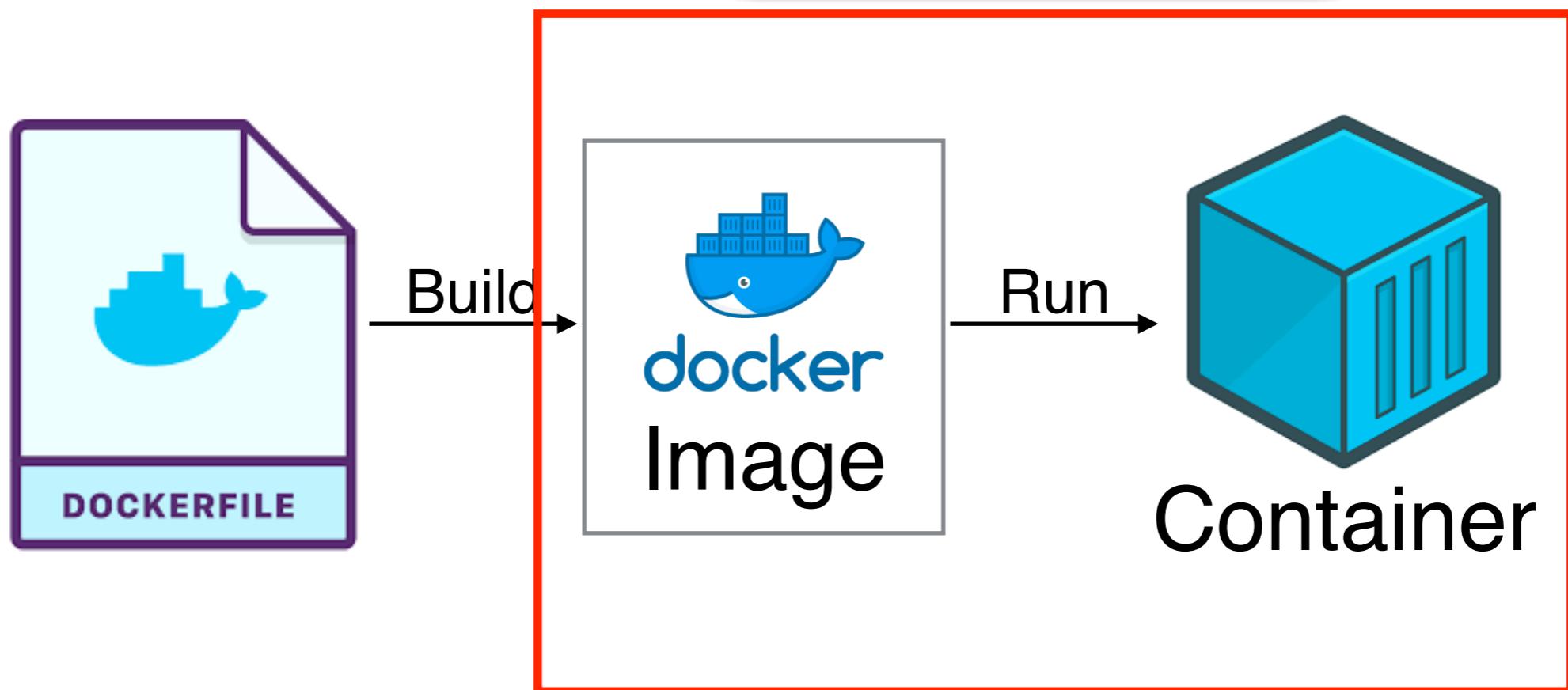
Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Run First Container



# Run First Container

```
$ docker run
```



# First of any new technology



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# First of any new technology

Hello World



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World

```
$ docker run hello-world
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World



# Hello World

Image name



```
$ docker run hello-world  
// alternative  
$ docker container run hello-world
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World

```
$ docker run hello-world
```

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:8e3114318a995a1ee497790535e7b88365222a21771ae7e53687ad76563e8e76
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!  
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:  
`$ docker run -it ubuntu bash`

Share images, automate workflows, and more with a free Docker ID:  
<https://hub.docker.com/>

For more examples and ideas, visit:  
<https://docs.docker.com/get-started/>



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World

```
$ docker run hello-world
```

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:8e3114318a995a1ee497790535e7b88365222a21771ae7e53687ad76563e8e76
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!  
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:  
`$ docker run -it ubuntu bash`

Share images, automate workflows, and more with a free Docker ID:  
<https://hub.docker.com/>

For more examples and ideas, visit:  
<https://docs.docker.com/get-started/>

1. Pull Image

2. Run Container



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World

\$ docker run hello-world

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:8e3114318a995a1ee497790535e7b88365222a21771ae7e53687ad76563e8e76
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!  
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the [Docker Hub](#).  
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.



To try something more ambitious, you can run an Ubuntu container with:  
\$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:  
<https://hub.docker.com/>

For more examples and ideas, visit:  
<https://docs.docker.com/get-started/>

1. Pull Image

2. Run Container



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World (Continue)

```
$ docker run hello-world
```

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:8e3114318a995a1ee497790535e7b88365222a21771ae7e53687ad76563e8e76
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!  
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:  
`$ docker run -it ubuntu bash`

Share images, automate workflows, and more with a free Docker ID:  
<https://hub.docker.com/>

For more examples and ideas, visit:  
<https://docs.docker.com/get-started/>



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World (Continue)

```
$ docker run -it ubuntu bash
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World (Continue)

Specific command

```
$ docker run -it ubuntu bash
```

-i - interactive mode

-t - tty (input, output to device screen)



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Hello World (Continue)

```
$ docker run -it ubuntu bash
```

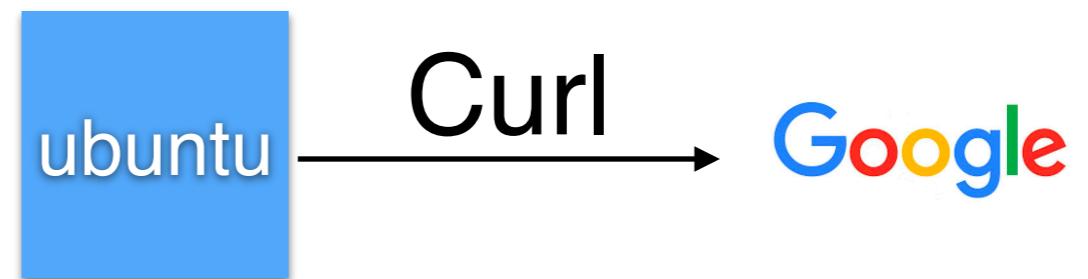
```
root@7dc14327463d:/#
```



# Hello World (Continue)

```
$ docker run -it ubuntu bash
```

```
root@7dc14327463d:/#
```

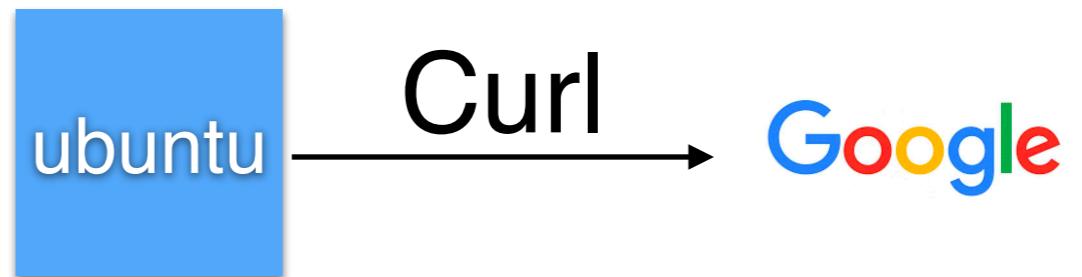


# Hello World (Continue)

```
$ docker run -it ubuntu bash
```

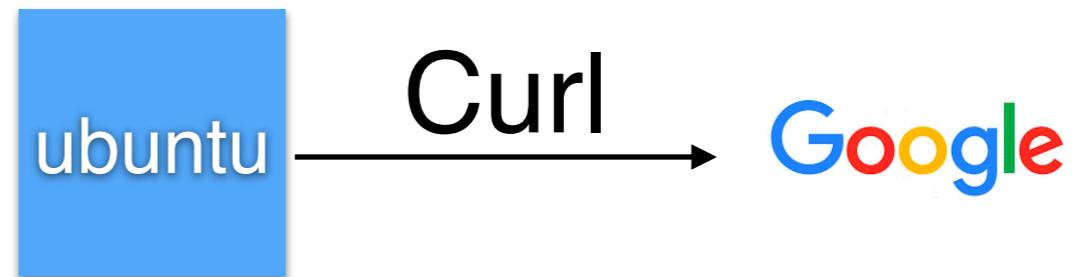
```
root@7dc14327463d:/# apt-get update
```

```
root@7dc14327463d:/# apt-get install curl -y
```



# Hello World (Continue)

```
root@7dc14327463d:/# curl http://www.google.co.th
```



# Hello World (Continue)

```
root@7dc14327463d:/# exit  
// alternative
```

CTRL

D



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Start another one

```
$ docker run -it ubuntu bash
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Why?

```
$ docker run -it ubuntu bash
```

```
root@36add31ad99c:/# curl http://www.google.com  
bash: curl: command not found
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Why?

## New one

```
$ docker run -it ubuntu bash
```

```
root@36add31ad99c:/#
```

## Old one

```
$ docker run -it ubuntu bash
```

```
root@7dc14327463d:/#
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Why?

## New one

```
$ docker run -it ubuntu bash
```

```
root@36add31ad99c:/#
```

## Old one

```
$ docker run -it ubuntu bash
```

```
root@7dc14327463d:/#
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Why?

## New one

```
$ docker run -it ubuntu bash
```

```
root@36add31ad99c:/#
```

## Old one

```
$ docker run -it ubuntu bash
```

```
root@7dc14327463d:/#
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# It's Container



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Pet vs Cattle



VS



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Pet Scale Up



VS



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Pet Scale Up



VS



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Cattle Scale Out



VS



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

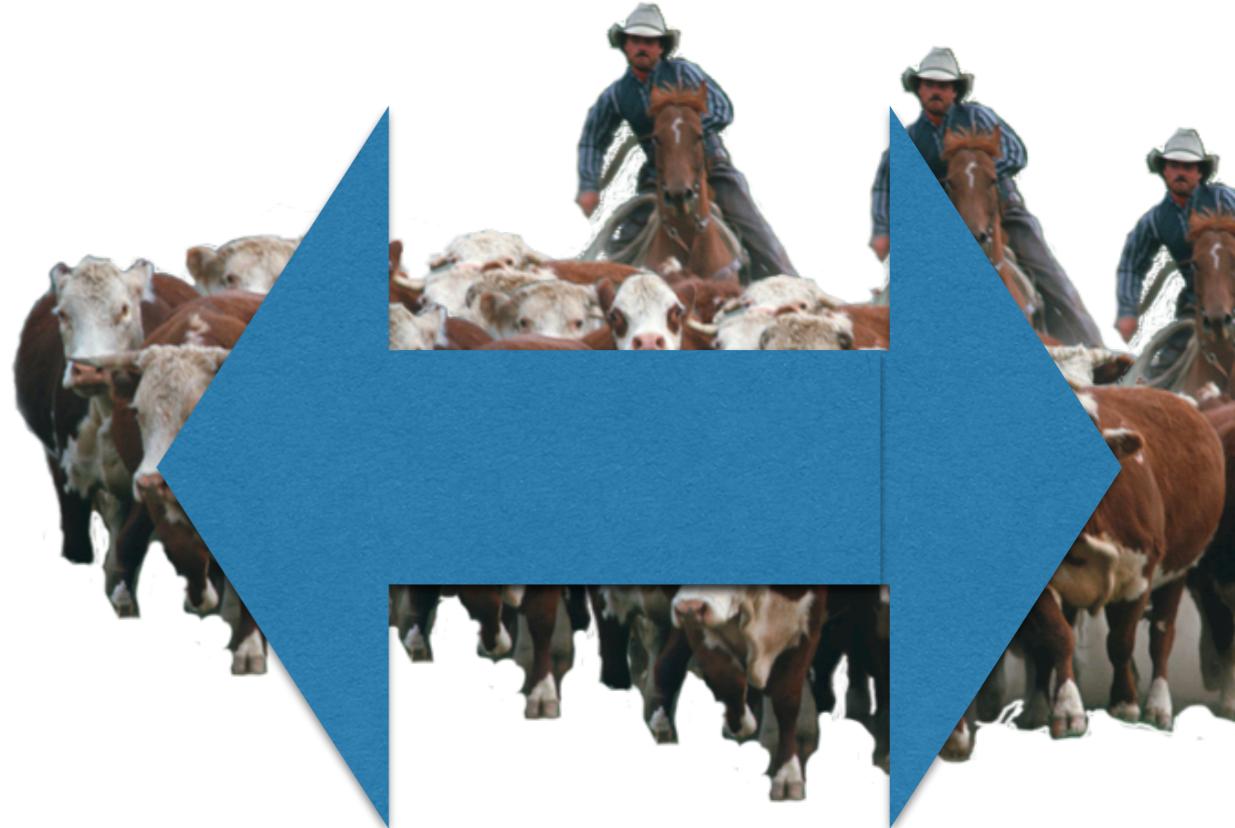
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Cattle Scale Out



VS



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

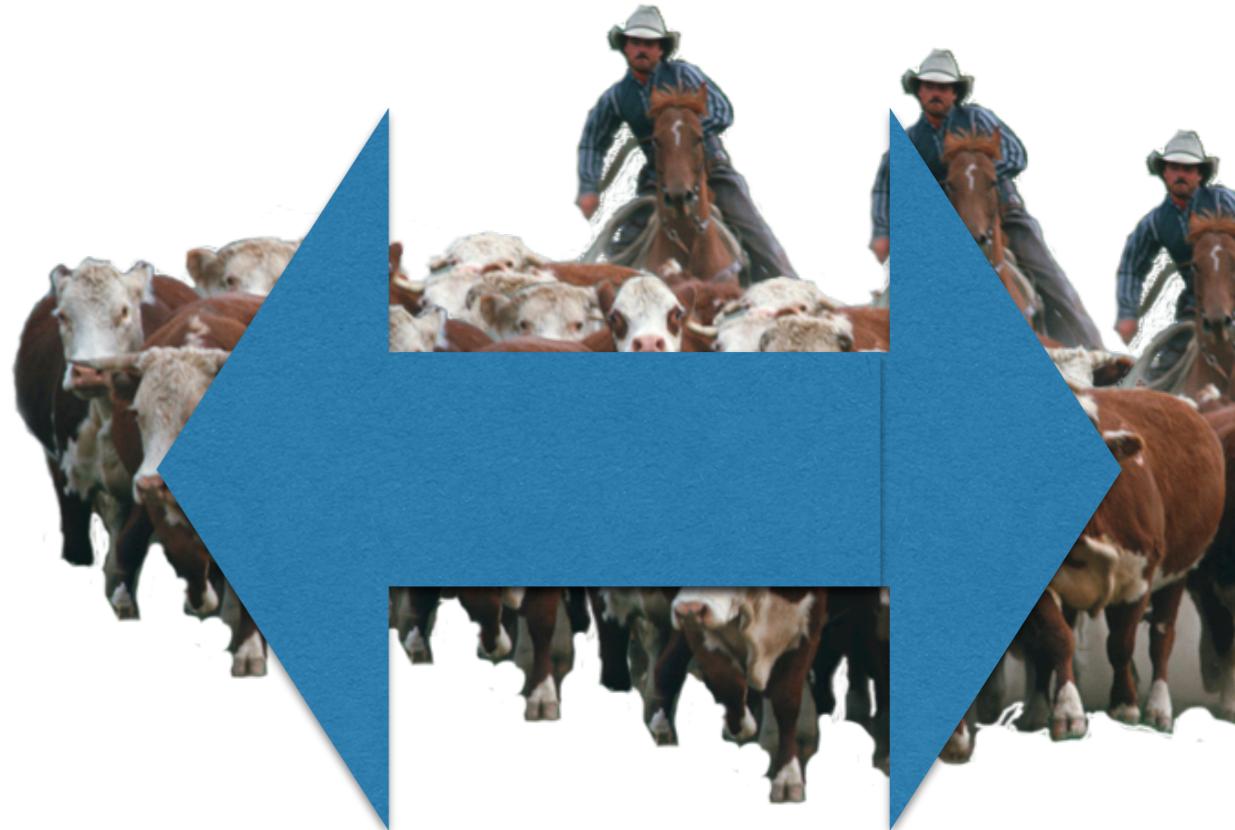
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Cattle Scale Out



VS



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# If we want ubuntu with curl?



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# ubuntu with curl

1. Run container
2. Install software
3. Create new image
4. Share image



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 1.Run container

```
$ docker run -it ubuntu bash
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

## 2. Install Software

```
$ docker run -it ubuntu bash
```

```
root@7dc14327463d:/# apt-get update
```

```
root@7dc14327463d:/# apt-get install curl -f
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

### 3. Create new image

```
root@7dc14327463d:/# exit  
// alternative
```

CTRL

D



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

### 3. Create new image

```
$ docker container ls -a  
// alternative  
$ docker ps -a
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

### 3. Create new image

List all container

```
$ docker container ls -a
```

// alternative

```
$ docker ps -a
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 3. Create new image

```
$ docker container ls -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3da779ba145a	ubuntu	"bash"	16 hours ago	Exited (0)	16 hours ago	reverent_mirzakhani
9624d7c7cc16	ubuntu	"bash"	16 hours ago	Exited (127)	16 hours ago	serene_taussig



### 3. Create new image

```
$ docker diff <id>  
// alternative  
$ docker container diff <id>
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 3. Create new image

Show different that change since container start

```
$ docker diff <id>
```

// alternative

```
$ docker container diff <id>
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

### 3. Create new image

```
$ docker commit <id>  
// alternative  
$ docker container commit <id>
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 3. Create new image

Commit change from container to new image

```
$ docker commit <id>
```

// alternative

```
$ docker container commit <id>
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 3. Create new image

## List image

```
$ docker images  
// alternative  
$ docker image ls
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 3. Create new image

\$ docker images

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
None	none	e0612a51bc6a	3 months ago	13.3kB
hello-world	latest	bf756fb1ae65	3 months ago	13.3kB
Ubuntu	16.04	5e13f8dd4c1a	9 months ago	120MB



# 3. Create new image

## Tag image

```
$ docker tag
```

// alternative

```
$ docker image tag
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

### 3. Create new image

```
$ docker image tag e0612a51bc6a ubuntu-with-curl
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# 3. Create new image

\$ docker images

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ubuntu-with-curl	latest	e0612a51bc6a	3 months ago	13.3kB
hello-world	latest	bf756fb1ae65	3 months ago	13.3kB
Ubuntu	16.04	5e13f8dd4c1a	9 months ago	120MB

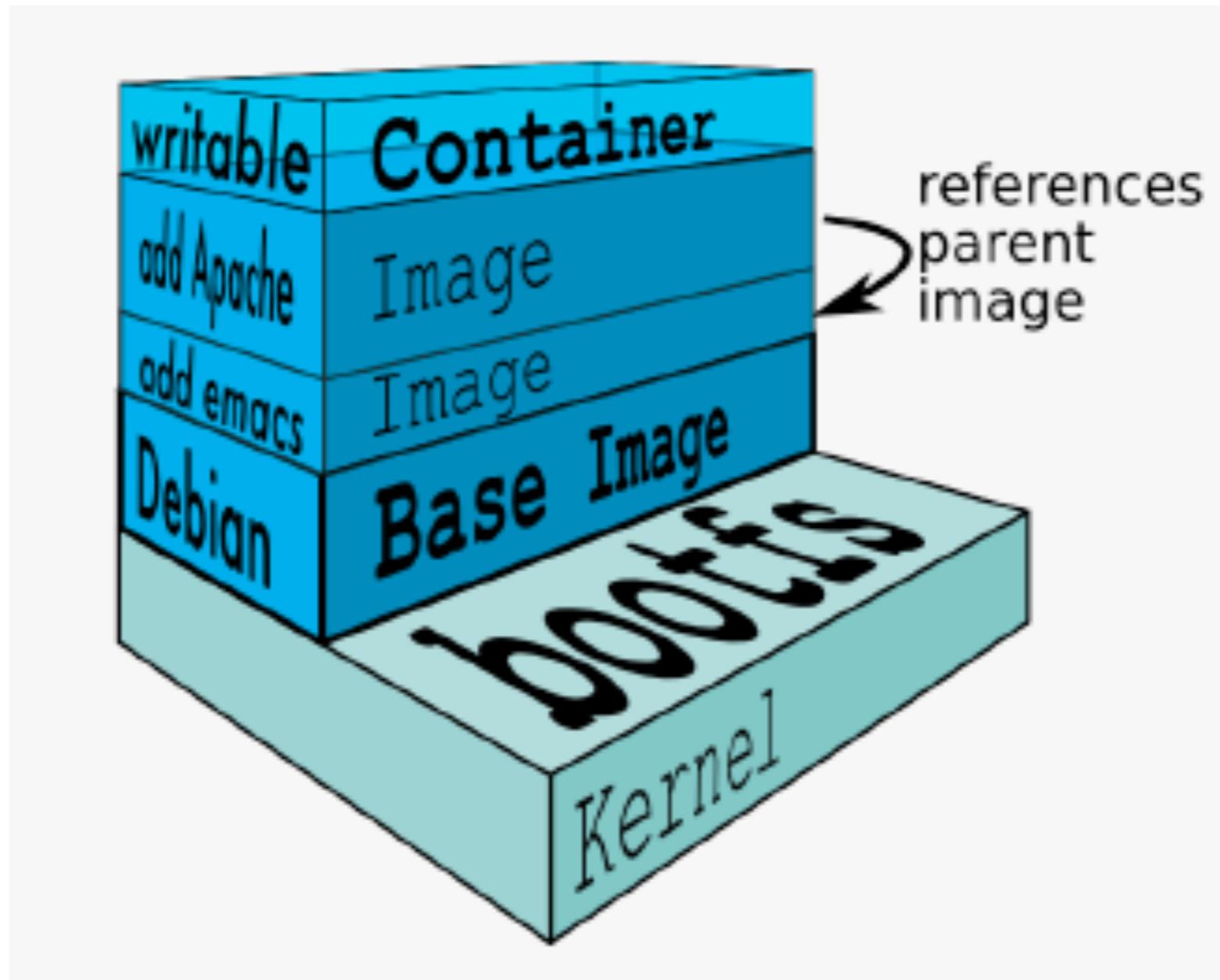


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Image



# Docker Image

```
$ docker images
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Image

```
$ docker images  
// alternative  
$ docker image ls
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Image

\$ docker images

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hello-world	latest	bf756fb1ae65	3 months ago	13.3kB
Ubuntu	16.04	5e13f8dd4c1a	9 months ago	120MB



# Recap

```
$ docker container ls
```

```
$ docker container run
```

```
$ docker container commit <container-id>
```

```
$ docker image ls
```

```
$ docker image tag <id | image:tag> <new image:tag>
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

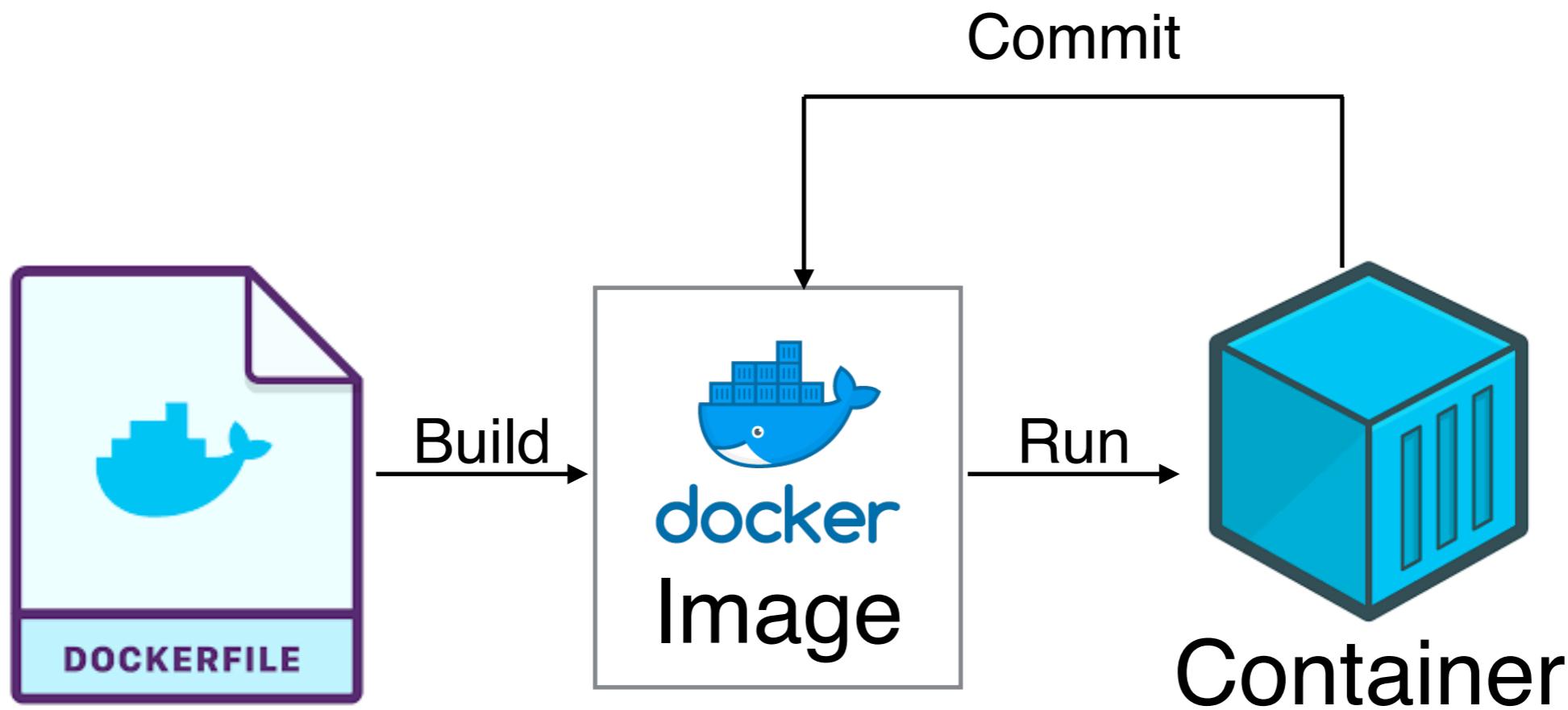
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build your first image

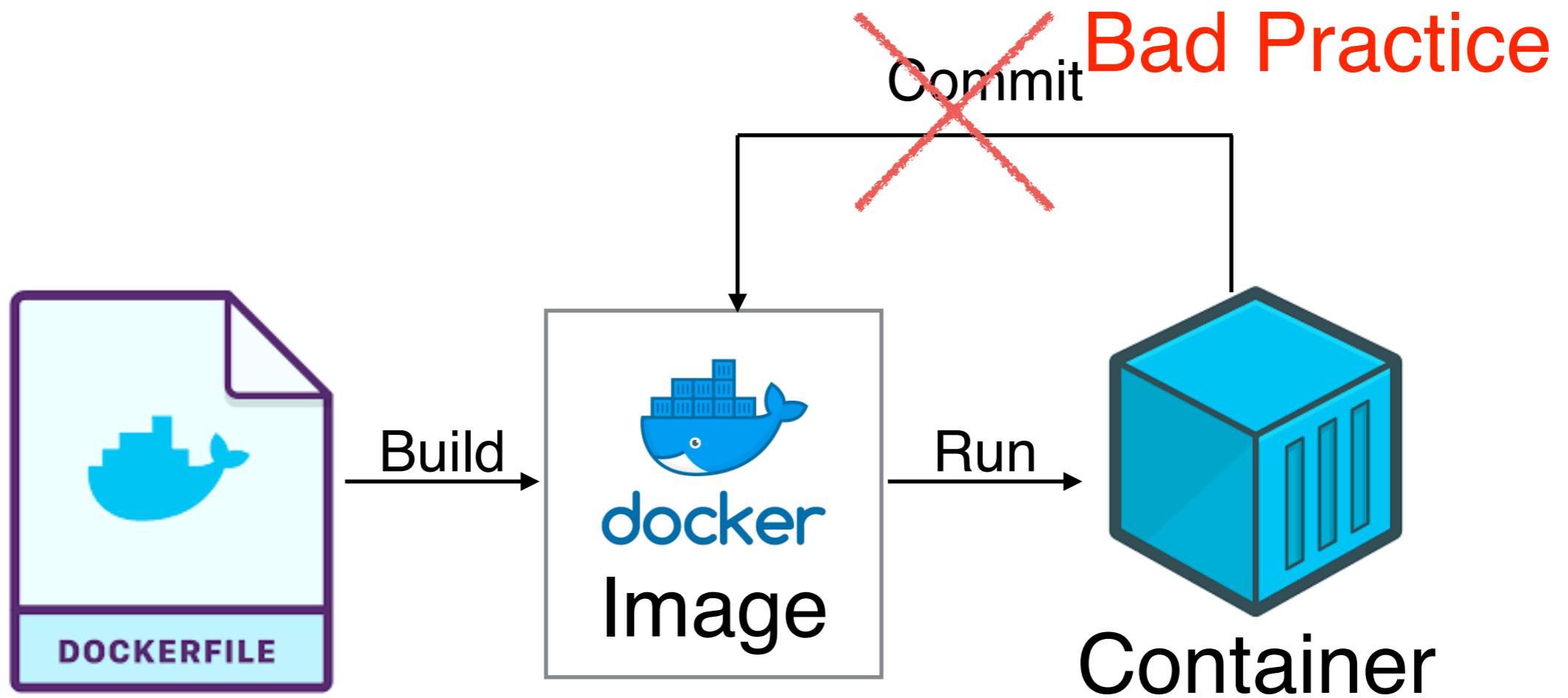


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

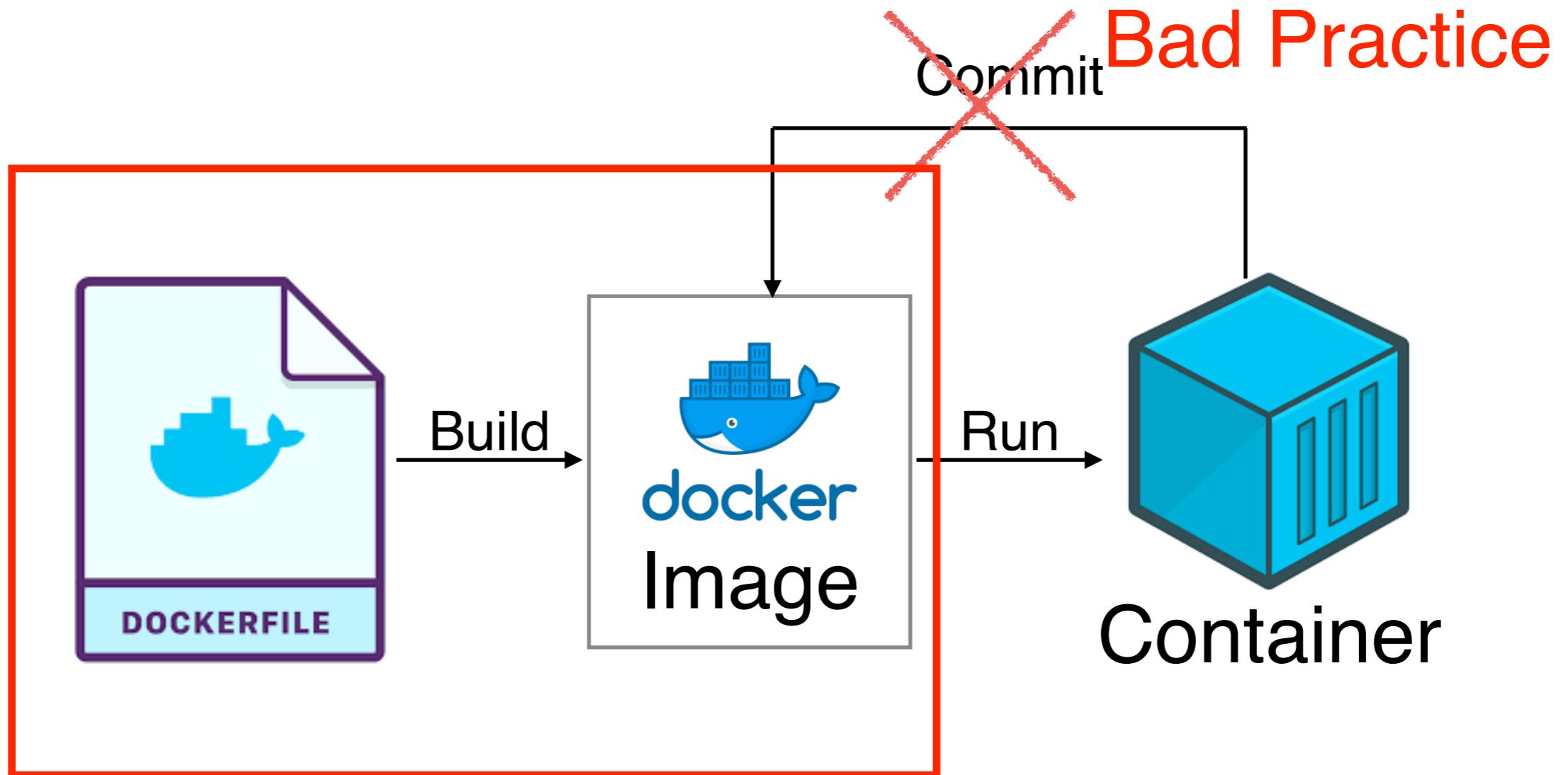
# Build image



# Build image



# Build image



# How to create ubuntu with curl

1. Run container ubuntu
2. Install software curl



# How to create ubuntu with curl

filename: Dockerfile

```
FROM ubuntu
RUN apt-get update
RUN apt-get install curl -y
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker build -t <image-name:tag> <context>  
// alternative  
$ docker image build -t <image-name:tag> <context>
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker image build -t first-image .
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker image build -t first-image .
```

Tag name    Context directory



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# List Docker Image

```
$ docker image ls
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Try it

```
$ docker container run first-image bash
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Better way

filename: Dockerfile

```
FROM ubuntu
RUN apt-get update && apt-get install curl -y
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a website

<http://localhost:8080>

Hello World



# We need a website



1. We need web server
2. We copy html to server
3. Running on port 8080



# We need a website



Nginx

1. We need **web server**
2. We copy html to server
3. Running on port 8080



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker hub nginx official

The screenshot shows the Docker Hub interface for the official Nginx repository. At the top, there's a header with a lock icon, the URL 'hub.docker.com/\_/nginx', a two-factor authentication notice, and navigation links for 'Explore', 'Pricing', 'Sign In', and 'Sign Up'. Below the header, the breadcrumb navigation shows 'Explore > nginx'. The main content area features the NGINX logo, the repository name 'nginx ★', and the status 'Docker Official Images'. It describes the image as an 'Official build of Nginx.' Below this, there's a download count of '1B+' and a list of supported architectures: Container, Linux, 386, IBM Z, x86-64, ARM, PowerPC 64 LE, ARM 64, Application Infrastructure, and Official Image. On the right side, a specific tag 'Linux - ARM 64 ( latest )' is highlighted with a copy-to-clipboard button and a command line pull instruction ('docker pull nginx'). A link to 'View Available Tags' is also present. At the bottom of the page, there's a section titled 'Supported tags and respective Dockerfile links' containing a list of tags: 1.17.10, mainline, 1, 1.17, latest, 1.17.10-perl, mainline-perl, 1-perl, 1.17-perl, perl, 1.17.10-alpine, mainline-alpine, 1-alpine, 1.17-alpine, alpine, 1.17.10-alpine-perl, mainline-alpine-perl, 1-alpine-perl, 1.17-alpine-perl, alpine-perl, 1.18.0, stable, 1.18, 1.18.0-perl, stable-perl, 1.18-perl.



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Add html file to image

filename: Dockerfile

```
FROM nginx
```



# We need a website



1. We need web server
2. We copy html to server
3. Running on port 8080



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Add html file to image

filename: Dockerfile

```
FROM nginx
COPY index.html /usr/share/nginx/html
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker image build -t hello-world-web .
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run Web server

```
$ docker container run hello-world-web
```

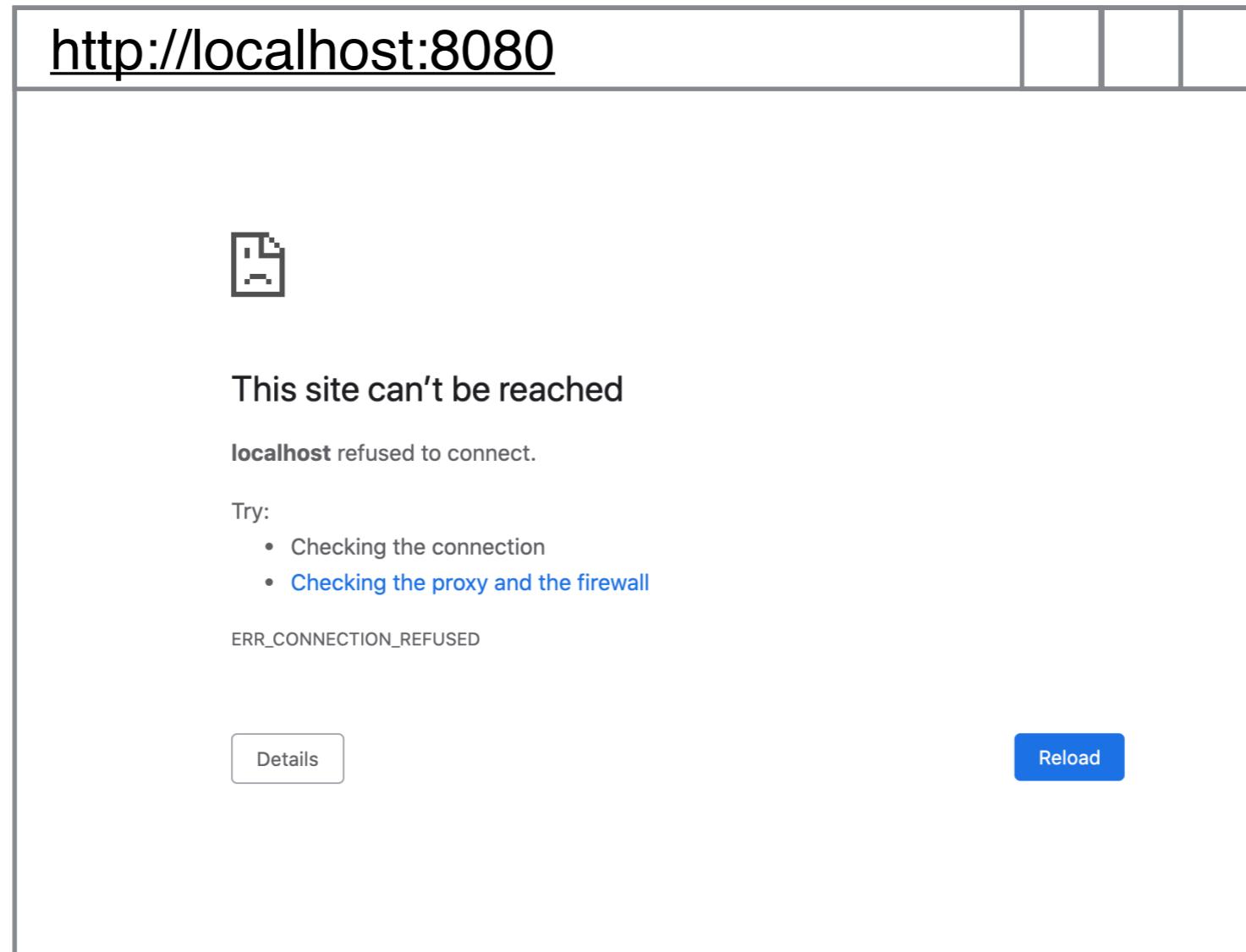


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

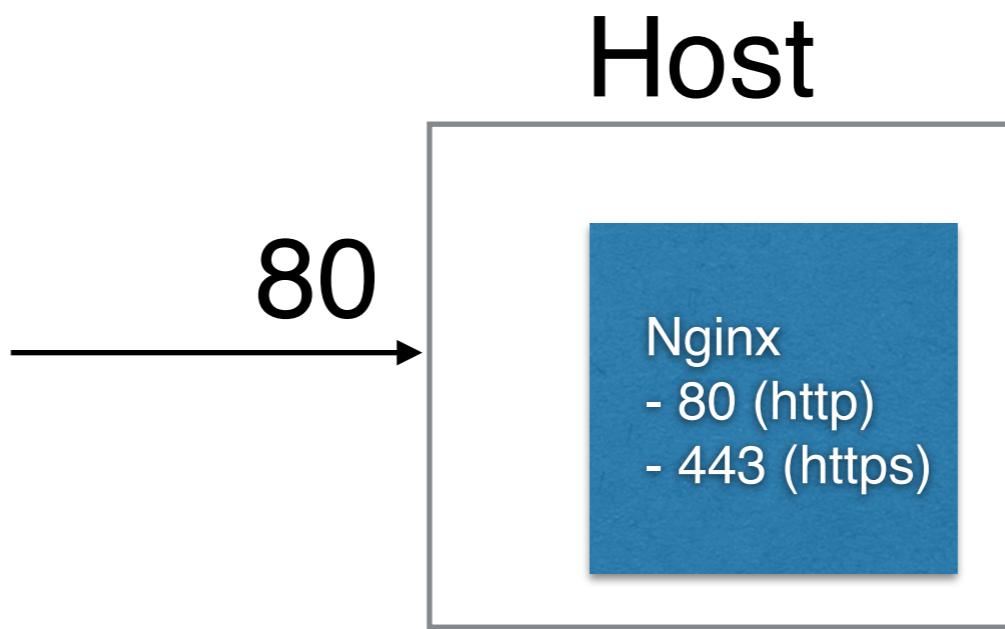
# We need a website



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

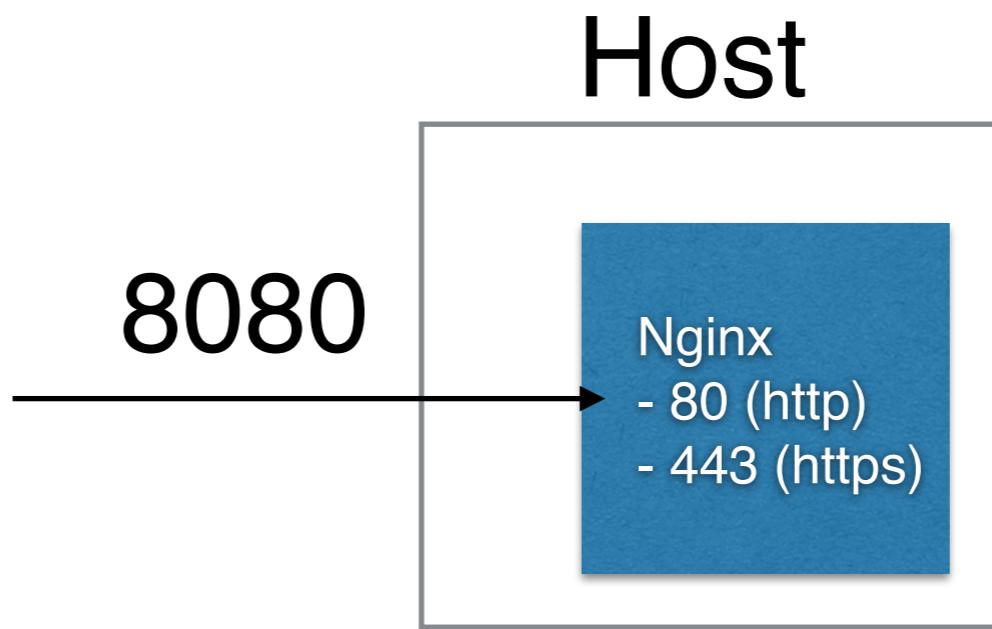
# Run Web server

```
$ docker container run hello-world-web
```



# Run Web server

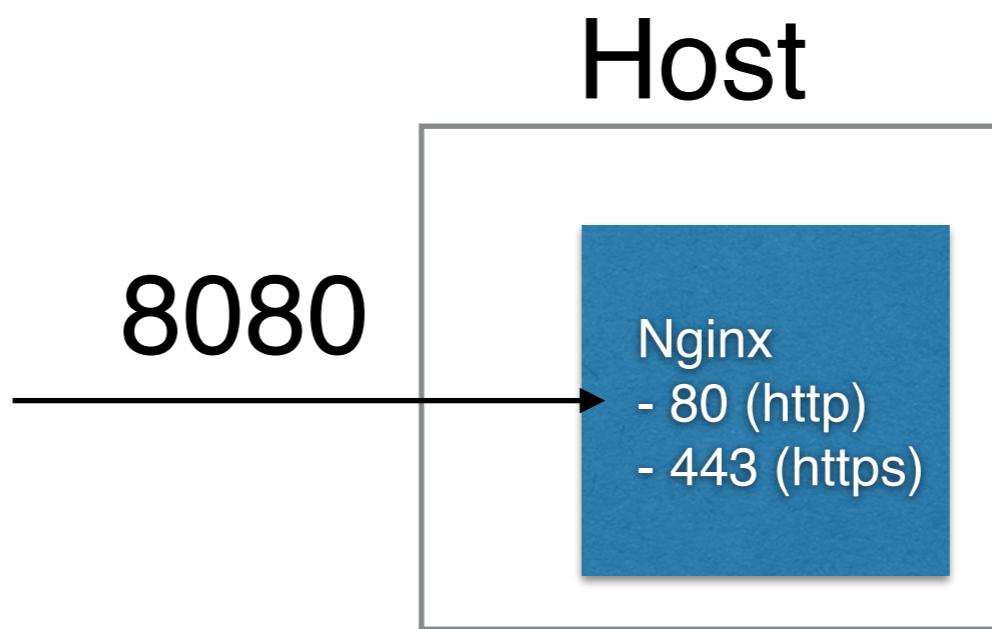
```
$ docker container run -p 8080:80 hello-world-web
```



# Run Web server

Host port: container port

```
$ docker container run -p 8080:80 hello-world-web
```



# Boom!

<http://localhost:8080>

Hello World



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Foreground and Background

## Foreground

```
$ docker container run -p 8080:80 hello-world-web
```

## Interactive Foreground

```
$ docker container run -it ubuntu bash
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Foreground and Background

## Background

```
$ docker container run -d -p 8080:80 hello-world-web
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Foreground and Background

```
$ docker container ls  
$ docker container start <id | name>  
$ docker container stop <id | name>  
$ docker container restart <id | name>
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Foreground and Background

## Container naming

```
$ docker container run -d \  
    -p 8080:80 \  
    --name hello-nginx \  
    hello-world-web
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Foreground and Background

## Container naming

```
$ docker container run -d \  
    -p 8080:80 \  
    --name hello-nginx \  
    hello-world-web
```



# Foreground and Background

Stop container

```
$ docker container stop hello-nginx
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Foreground and Background

Delete container

```
$ docker container rm hello-nginx
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Remove after exit

Remove after exited status

```
$ docker container run -d \  
    -p 8080:80 \  
    --rm \  
    hello-world-web
```

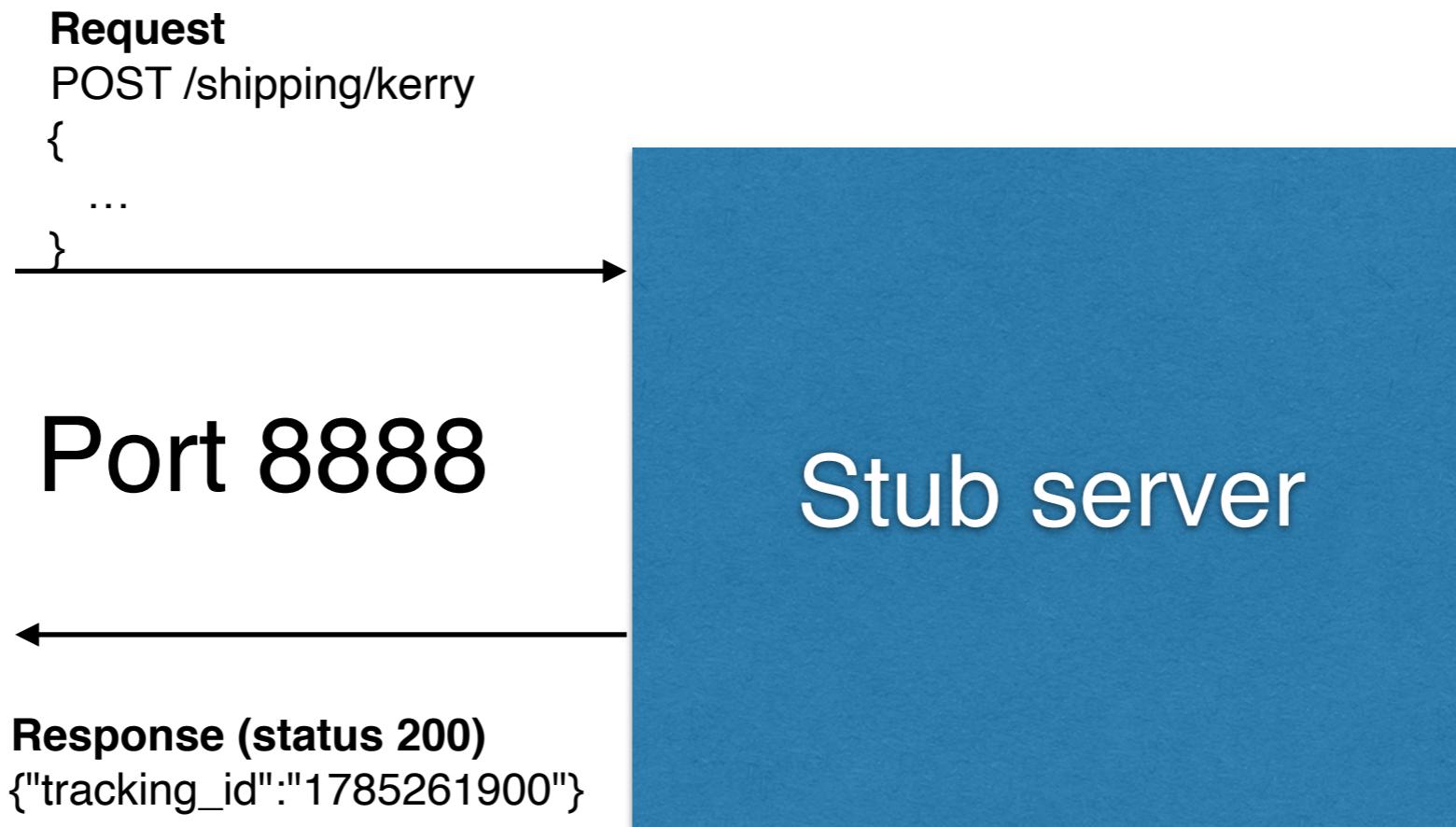


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server



# We need a stub server

Stub server

1. We need stub engine
2. Add Request / Response
3. Run on port 8888



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server

Stub server

1. We need **stub engine**
2. Add Request / Response
3. Run on port 8888



Java JDK8



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server

FROM openjdk:8



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server

```
FROM openjdk:8  
WORKDIR /app
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server

```
FROM openjdk:8  
WORKDIR /app
```

Default directory



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server

```
FROM openjdk:8  
WORKDIR /app  
COPY stubby4j-6.0.2.jar .  
COPY kerry.yml .
```

Jar + stub config



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a stub server

```
FROM openjdk:8  
WORKDIR /app  
COPY stubby4j-6.0.2.jar .  
COPY kerry.yml .  
CMD java -jar stubby4j-6.0.2.jar -l 0.0.0.0 -d kerry.yml
```

Execute command



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker image build -t stub-server .
```



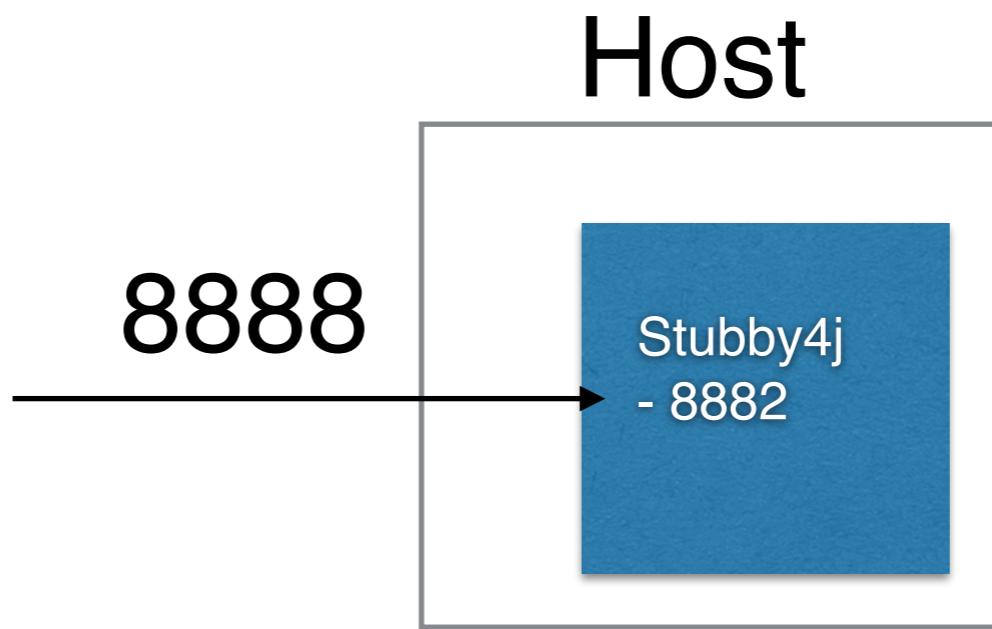
Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

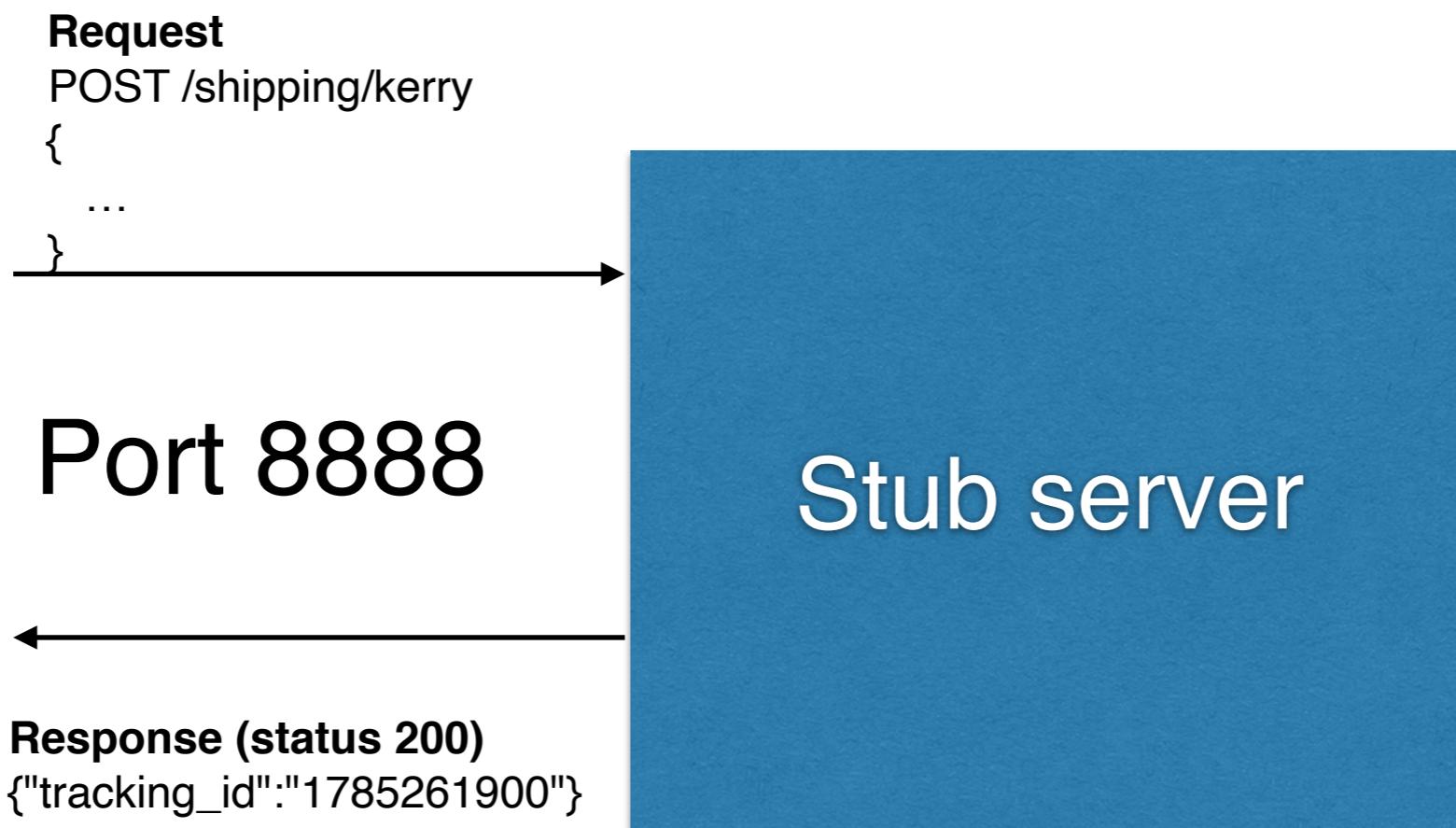
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run Stub server

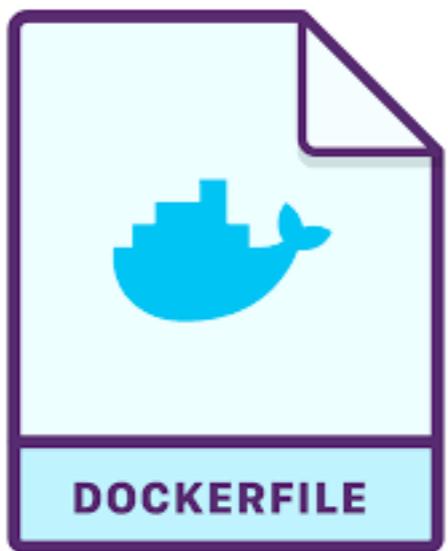
```
$ docker container run -d -p 8888:8882 stub-server
```



# Try Stub server



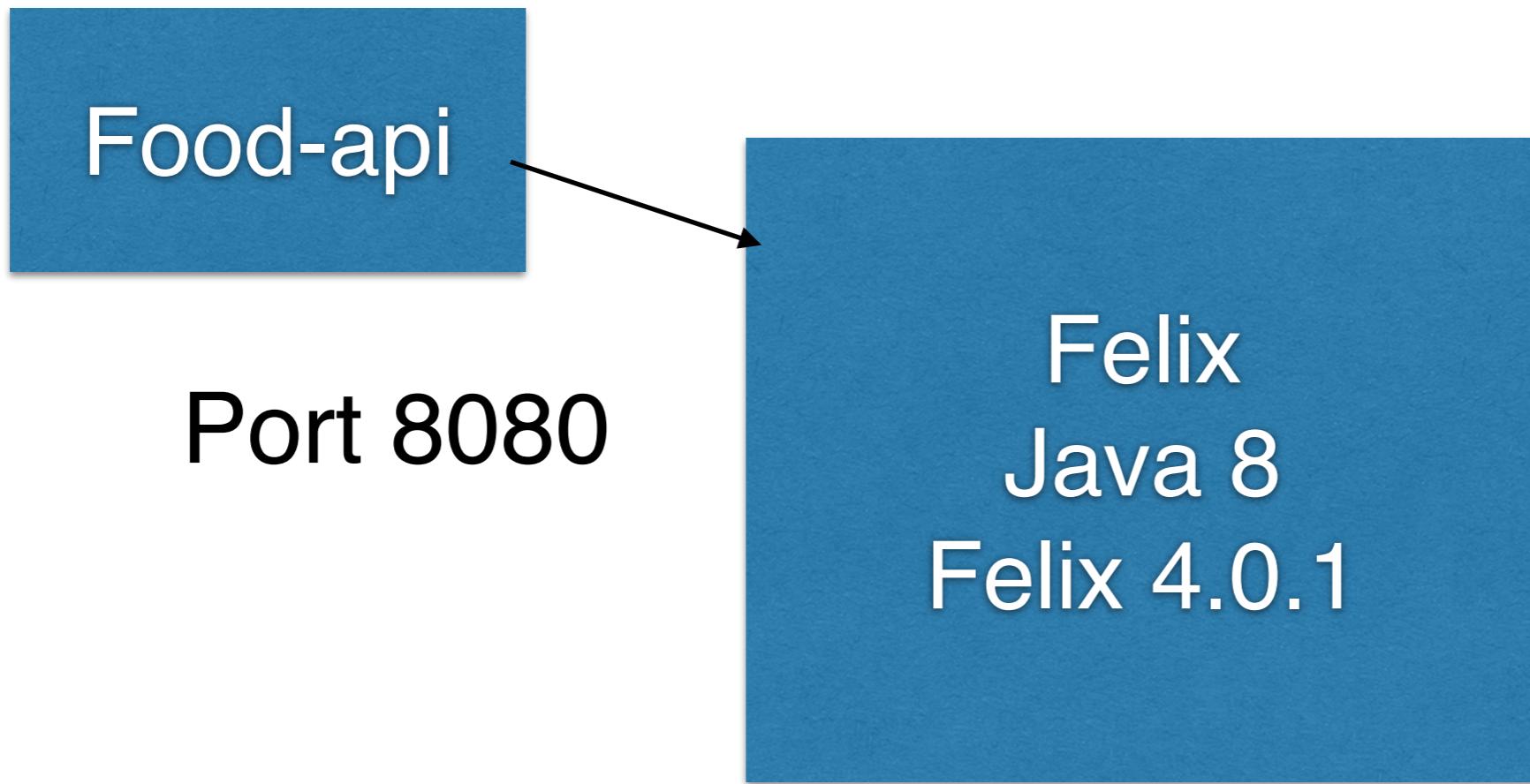
# Dockerfile



FROM - which parent image to use  
ADD, COPY - add file to image  
RUN - execute command when build  
CMD - command use to run container  
WORKDIR - default work directory



# We need a Felix



1. Use Java 8 openjdk:8-jre
2. Install Felix 4.2.1
3. Place food api.jar, food web.jar
4. Run Felix

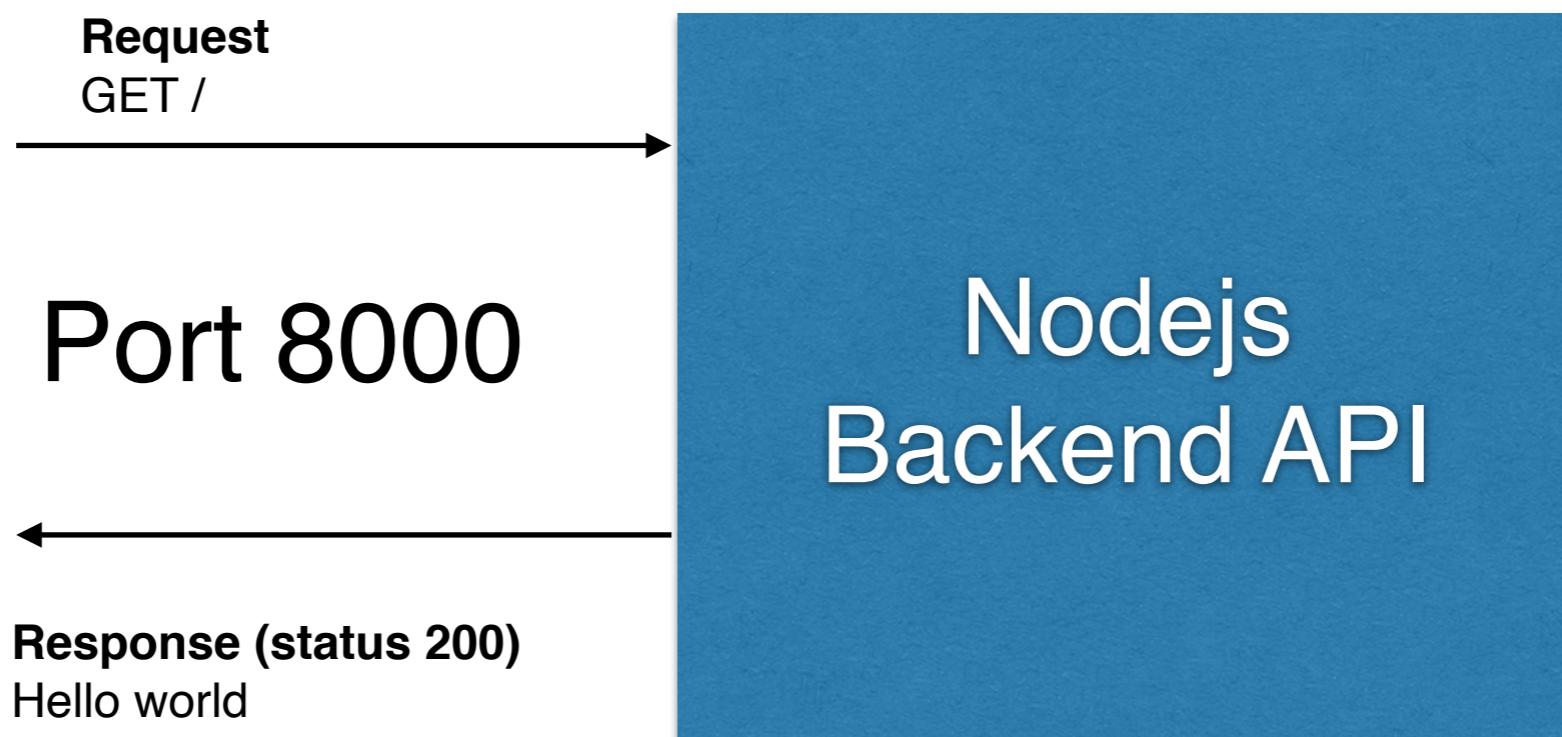


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!



# We need a Nodejs Backend API!

Backend API

1. We need Node version 12
2. Install dependency via npm
3. Copy project
4. Execute server at port 8000



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!

FROM node:12



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!

```
FROM node:12  
WORKDIR /app
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!

```
FROM node:12
WORKDIR /app
COPY package.json .
COPY package-lock.json .
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!

```
FROM node:12
WORKDIR /app
COPY package.json .
COPY package-lock.json .
RUN npm install
```



# We need a Nodejs Backend API!

```
FROM node:12
WORKDIR /app
COPY package.json .
COPY package-lock.json .
RUN npm install
COPY index.js .
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!

```
FROM node:12
WORKDIR /app
COPY package.json .
COPY package-lock.json .
RUN npm install
COPY index.js .
EXPOSE 3000
CMD node index.js
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Nodejs Backend API!

```
FROM node:12
WORKDIR /app
COPY package.json .
COPY package-lock.json .
RUN npm install
COPY index.js .
EXPOSE 3000
CMD node index.js
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker image build -t node-api .
```



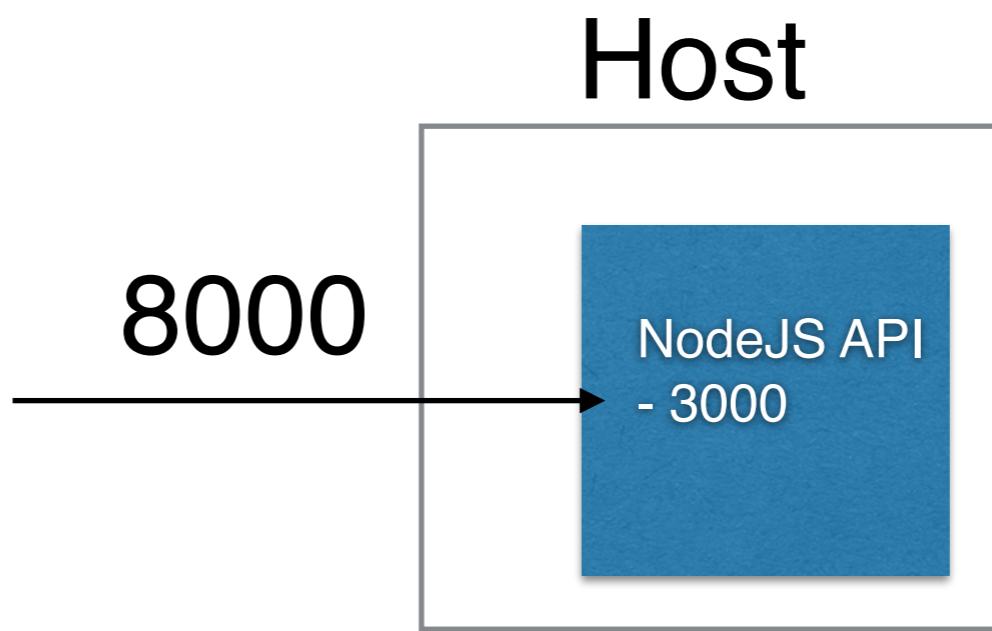
Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

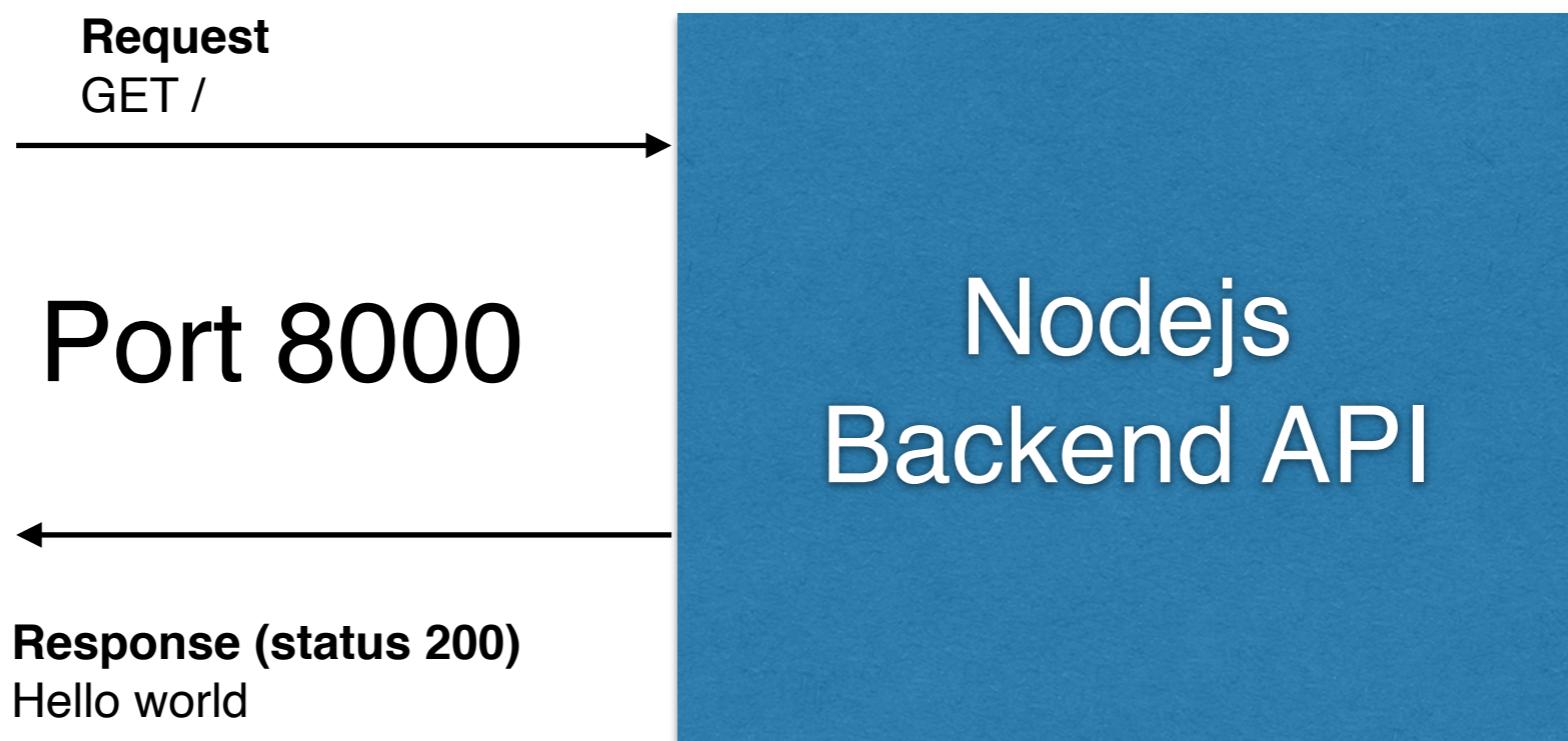
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run NodeJS API

```
$ docker container run -p 8000:3000 node-api
```



# Try!



# Tips: Use Alpine

```
FROM node:12-alpine
WORKDIR /app
COPY package.json .
COPY package-lock.json .
RUN npm install
COPY index.js .
CMD node index.js
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Tips: Use Alpine

## Alpine Linux

---

From Wikipedia, the free encyclopedia

**Alpine Linux** is a [Linux distribution](#) based on [musl](#) and [BusyBox](#), designed for security, simplicity, and resource efficiency.<sup>[3][4][5][6][7]</sup> It used a [hardened kernel](#) until release 3.8 and compiles all [user-space binaries](#) as [position-independent executables](#) with [stack-smashing protection](#).<sup>[8]</sup>

Because of its small size, it is commonly used in [containers](#) providing quick boot-up times.<sup>[9]</sup>

The [postmarketOS](#) project which is designed to run on mobile devices is based on Alpine Linux.



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Tips: Use Alpine

## Alpine Linux

---

From Wikipedia, the free encyclopedia

**Alpine Linux** is a [Linux distribution](#) based on [musl](#) and [BusyBox](#), designed for security, simplicity, and resource efficiency.<sup>[3][4][5][6][7]</sup> It used a [hardened kernel](#) until release 3.8 and compiles all [user-space binaries](#) as [position-independent executables](#) with [stack-smashing protection](#).<sup>[8]</sup>

Because of its small size, it is commonly used in [containers](#) providing quick boot-up times.<sup>[9]</sup>

The [postmarketOS](#) project which is designed to run on mobile devices is based on Alpine Linux.



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Pull Docker Image

```
$ docker pull node:12-alpine  
// alternative  
$ docker image pull node:12-alpine
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Pull Docker image

The screenshot shows the Docker Hub website at [hub.docker.com/\\_/node/](https://hub.docker.com/_/node/). The page displays the Docker Official Images for Node.js. Key elements include:

- node ☆**: The main title and logo.
- Docker Official Images**: A link to the official Docker repository.
- Description**: A brief description stating "Node.js is a JavaScript-based platform for server-side and networking applications."
- Downloads**: 1B+.
- Tags**: Container, Linux, IBM Z, 386, PowerPC 64 LE, x86-64, ARM 64, ARM, Application Infrastructure, Official Image.
- Latest Tag**: Linux - ARM 64 (latest).
- Copy and paste command**: docker pull node
- View Available Tags**: A link to see all available tags.
- Supported tags and Dockerfile links**: A section listing supported tags:
  - 14.0.0-stretch, 14.0-stretch, 14-stretch, stretch, current-stretch, 14.0.0, 14.0, 14, latest, current
  - 14.0.0-stretch-slim, 14.0-stretch-slim, 14-stretch-slim, stretch-slim, current-stretch-slim, 14.0.0-slim, 14.0-slim, 14-slim
  - 14.0.0-buster, 14.0-buster, 14-buster, buster, current-buster
  - 14.0.0-buster-slim, 14.0-buster-slim, 14-buster-slim, buster-slim, current-buster-slim
  - 14.0.0-alpine3.10, 14.0-alpine3.10, 14-alpine3.10, alpine3.10, current-alpine3.10



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Pull Docker Image

```
$ docker image ls
```

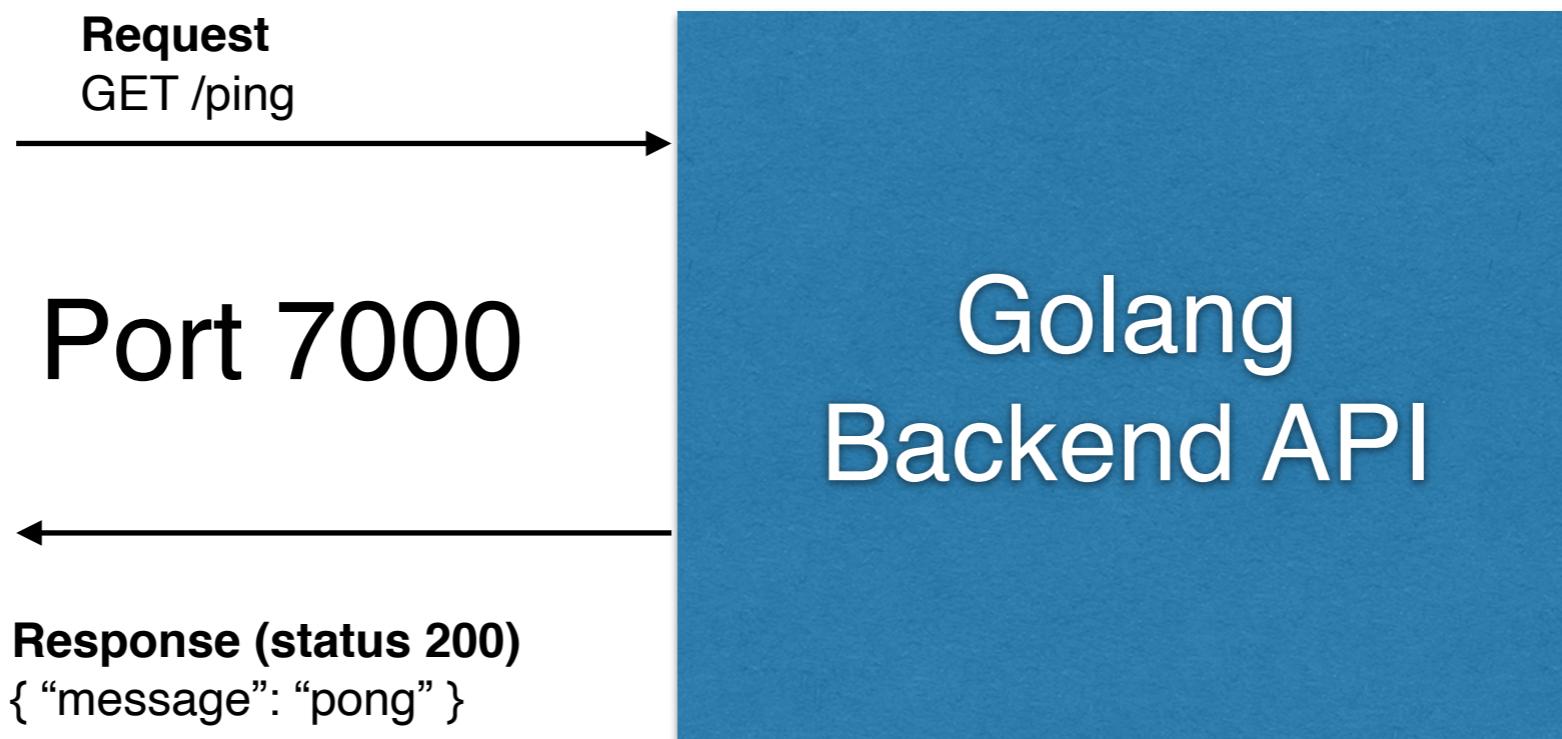


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# We need a Golang Backend API!



# We need a Golang Backend API!

Backend API

1. We need Go version 1.12
2. Install dependency via go mod
3. Copy project
4. Build Binary
5. Execute Binary at port 8080



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Go version 1.12

```
FROM go:1.12
WORKDIR /app
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Install dependency

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Copy project

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Binary

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN go build -o ./bin/app ./main.go
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Execute Binary at port 8080

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN go build -o ./bin/app ./main.go
ENV GIN_MODE release
EXPOSE 8080
CMD ./bin/app
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Execute Binary at port 8080

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN go build -o ./bin/app ./main.go
ENV GIN_MODE release
EXPOSE 8080
CMD ./bin/app
```

Environment Variable inside Image

Port that open to external network



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Golang pingpong API

```
$ docker build -t pingpong-api .
```



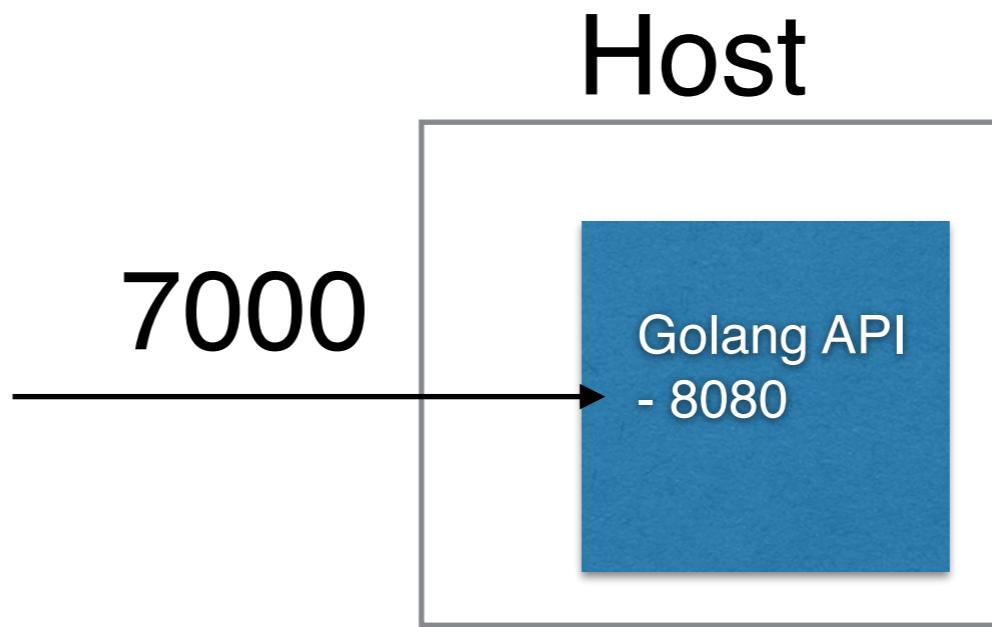
Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run Golang Pingpong API

```
$ docker container run -p 7000:8080 pingpong-api
```



# Multi-stage Build

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN go build -o ./bin/app ./main.go
ENV GIN_MODE release
EXPOSE 8080
CMD ./bin/app
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi-stage Build

```
FROM go:1.12
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN go build -o ./bin/app ./main.go
ENV GIN_MODE release
EXPOSE 8080
CMD ./bin/app
```

Build

Run



# Multi-stage Build

```
FROM go:1.12 as builder
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN CGO_ENABLED=0 GOOS=linux go build -o ./bin/app ./main.go
```

```
FROM alpine:3
WORKDIR /root/
COPY --from=builder /app/bin/app .
ENV GIN_MODE release
EXPOSE 8080
CMD ./app
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi-stage Build

```
FROM go:1.12 as builder
WORKDIR /app
COPY go.mod .
COPY go.sum .
RUN go mod download
COPY ..
RUN CGO_ENABLED=0 GOOS=linux go build -o ./bin/app ./main.go
```

```
FROM alpine:3
WORKDIR /root/
COPY --from=builder /app/bin/app .
ENV GIN_MODE release
EXPOSE 8080
CMD ./app
```

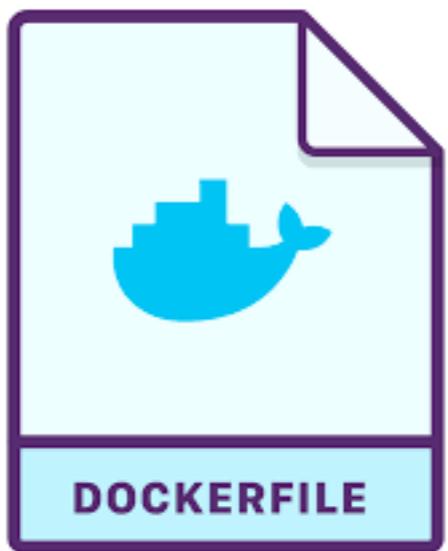


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Dockerfile



FROM - which parent image to use  
ADD, COPY - add file to image  
RUN - execute command when build  
CMD - command use to run container  
WORKDIR - default work directory

EXPOSE - tell which port that image provide  
ENTRYPOINT  
VOLUME  
ENV, ARG  
LABEL



# Share image to the world



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](#).

# Docker hub

NAME	LAST UPDATED	VISIBILITY	
sckseal / toy-store-service	11 days ago	Public	
sckseal / shopping-cart-init	17 days ago	Public	
sckseal / toy-store-shippinggateway	17 days ago	Public	
sckseal / toy-store-bankgateway	17 days ago	Public	
sckseal / toy-store-nginx	17 days ago	Public	
sckseal / toy-store-web	17 days ago	Public	
sckseal / shoppingcart-service	--	Public	
sckseal / shoppingcart-web	--	Public	



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Official Image

The screenshot shows the Docker Hub page for the official Node.js image. The URL is hub.docker.com/\_/node/. The page features the Docker logo and a search bar. The breadcrumb navigation shows 'Explore' and 'node'. The main content area includes the Node.js logo, a star icon, and the text 'Docker Official Images'. It describes Node.js as a JavaScript-based platform for server-side and networking applications. A download count of '1B+' is displayed. Below the description are several tags: Container, Linux, IBM Z, 386, PowerPC 64 LE, x86-64, ARM 64, ARM, Application Infrastructure, and Official Image. To the right, there's a button for 'Linux - ARM 64 (latest)', a copy-to-clipboard link, and a 'docker pull node' command. A 'View Available Tags' link is also present. At the bottom, tabs for 'Description' (which is active), 'Reviews', and 'Tags' are shown. A large section titled 'Supported tags and respective Dockerfile links' lists various tags with their corresponding Dockerfile links:

- 14.0.0-stretch, 14.0-stretch, 14-stretch, stretch, current-stretch, 14.0.0, 14.0, 14, latest, current
- 14.0.0-stretch-slim, 14.0-stretch-slim, 14-stretch-slim, stretch-slim, current-stretch-slim, 14.0.0-slim, 14.0-slim, 14-slim
- 14.0.0-buster, 14.0-buster, 14-buster, buster, current-buster
- 14.0.0-buster-slim, 14.0-buster-slim, 14-buster-slim, buster-slim, current-buster-slim
- 14.0.0-alpine3.10, 14.0-alpine3.10, 14-alpine3.10, alpine3.10, current-alpine3.10



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Unofficial Image

NAME	LAST UPDATED	VISIBILITY	
sckseal / toy-store-service	11 days ago	Public	
sckseal / shopping-cart-init	17 days ago	Public	
sckseal / toy-store-shippinggateway	17 days ago	Public	
sckseal / toy-store-bankgateway	17 days ago	Public	
sckseal / toy-store-nginx	17 days ago	Public	
sckseal / toy-store-web	17 days ago	Public	
sckseal / shoppingcart-service	--	Public	
sckseal / shoppingcart-web	--	Public	



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Share image to the world

1. Sign up account on Docker Hub
2. Login on command line
3. Create your own image
4. Push image to Docker hub



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker hub

The screenshot shows the Docker Hub homepage with a blue header bar. The header includes the Docker Hub logo, a search bar with placeholder text "Search for great content (e.g., mysql)", and navigation links for "Explore", "Pricing", "Sign In", and a prominent "Sign Up" button.

**Build and Ship any Application Anywhere**

Docker Hub is the world's easiest way to create, manage, and deliver your teams' container applications.

**Sign Up Today**

Already have an account? [Sign In](#)

Docker ID

Email

Password

Send me occasional product updates and announcements.

I'm not a robot reCAPTCHA  
Privacy - Terms

**Sign Up**

By creating an account, you agree to the [Terms of Service](#), [Privacy Policy](#), and [Data Processing Terms](#).



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Login to docker hub

```
$ docker login
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Tagging

```
$ docker image tag pingpong-api <user>/pingpong-api:latest  
$ docker image tag <user>/pingpong-api:latest <user>/pingpong-api:1.0.0
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Share Image to the World

```
$ docker image push <user>/pingpong-api:latest  
// alternative  
$ docker push <user>/pingpong-api:latest
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Share Image to the World

```
$ docker image push <user>/pingpong-api:1.0.0
```

// alternative

```
$ docker push <user>/pingpong-api:1.0.0
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

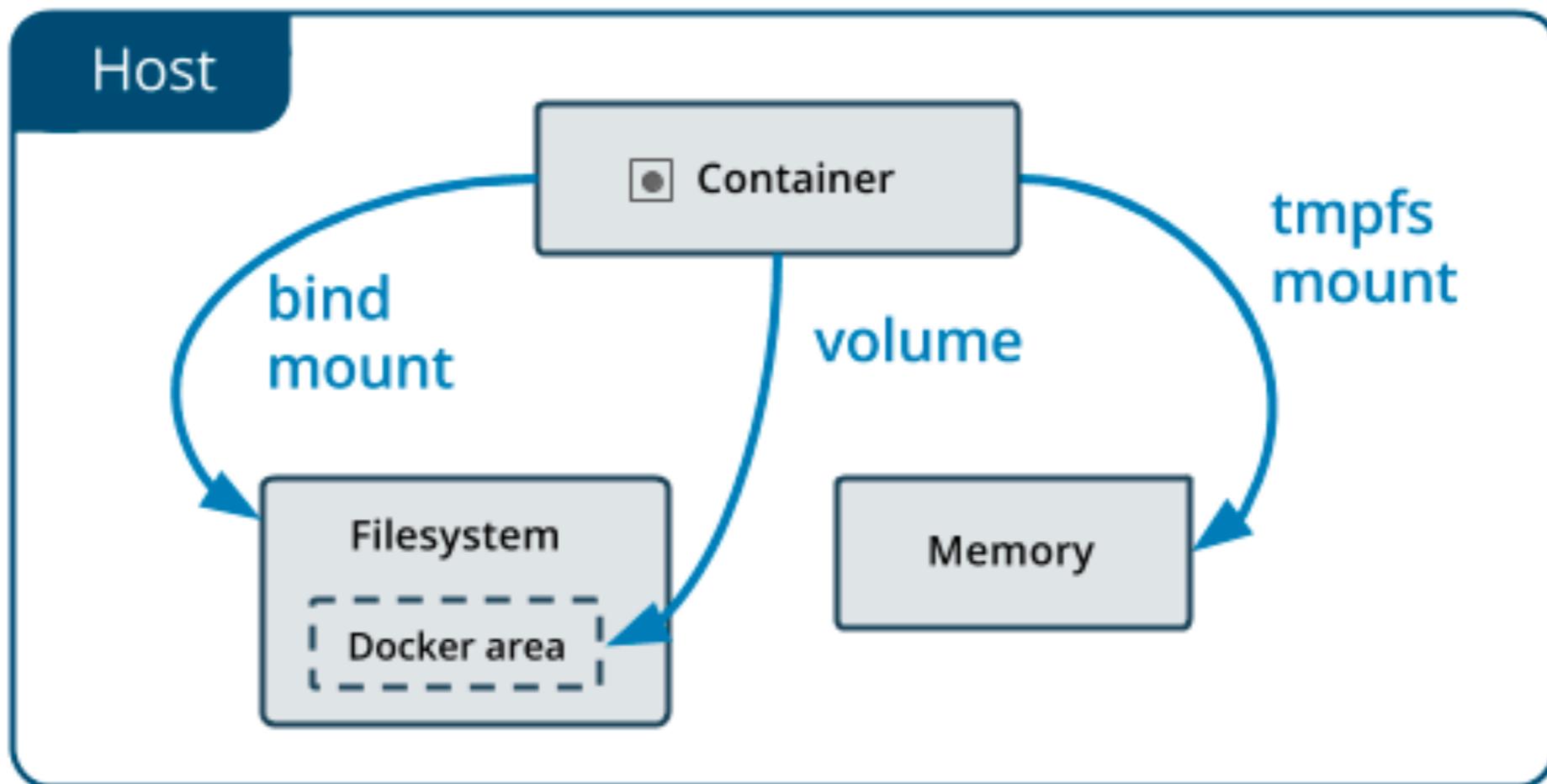
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Data Storage & Volume



# Volume



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website

filename: Dockerfile

```
FROM nginx
COPY index.html /usr/share/nginx/html
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website

filename: Dockerfile

```
$ docker image build -t web .
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website



Edit



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
**NonCommercial** — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website

filename: Dockerfile

```
$ docker image build -t web .
```

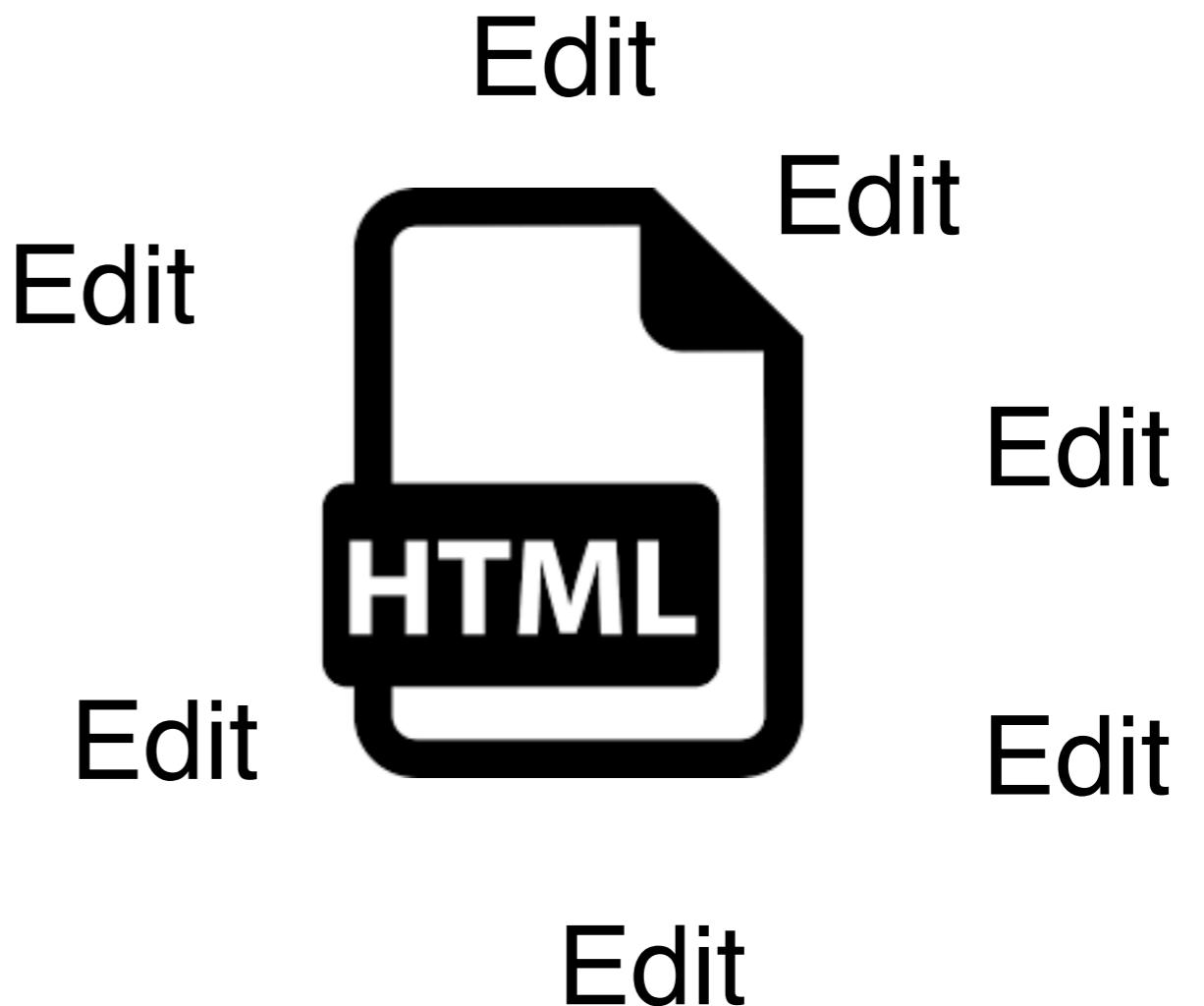


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.  
NonCommercial — You may not use the material for commercial purposes.  
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website

filename: Dockerfile

```
$ docker container run -d -v $(pwd)/web:/usr/share/nginx/html -p 8080:80 nginx
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Website

filename: Dockerfile

Host path:container path

```
$ docker container run -d -v $(pwd)/web:/usr/share/nginx/html -p 8080:80 nginx
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Database Volume Seeding

## seed-data.sql

hub.docker.com/\_/mysql



### Initializing a fresh instance

When a container is started for the first time, a new database with the specified name will be created and initialized with the provided configuration variables. Furthermore, it will execute files with extensions `.sh`, `.sql` and `.sql.gz` that are found in `/docker-entrypoint-initdb.d`. Files will be executed in alphabetical order. You can easily populate your `mysql` services by [mounting a SQL dump into that directory](#) and provide [custom images](#) with contributed data. SQL files will be imported by default to the database specified by the `MYSQL_DATABASE` variable.



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Database Volume Seeding

filename: Dockerfile

```
$ docker container run -d -v $(pwd)/sql:/docker-entrypoint-initdb.d mysql
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Database Volume Store Data

## Where to Store Data

Important note: There are several ways to store data used by applications that run in Docker containers. We encourage users of the `mysql` images to familiarize themselves with the options available, including:

- Let Docker manage the storage of your database data [by writing the database files to disk on the host system using its own internal volume management](#). This is the default and is easy and fairly transparent to the user. The downside is that the files may be hard to locate for tools and applications that run directly on the host system, i.e. outside containers.
- Create a data directory on the host system (outside the container) and [mount this to a directory visible from inside the container](#). This places the database files in a known location on the host system, and makes it easy for tools and applications on the host system to access the files. The downside is that the user needs to make sure that the directory exists, and that e.g. directory permissions and other security mechanisms on the host system are set up correctly.

The Docker documentation is a good starting point for understanding the different storage options and variations, and there are multiple blogs and forum postings that discuss and give advice in this area. We will simply show the basic procedure here for the latter option above:

1. Create a data directory on a suitable volume on your host system, e.g. `/my/own/datadir`.

2. Start your `mysql` container like this:

```
$ docker run --name some-mysql -v /my/own/datadir:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=my-secret-pw -d mysql:tag
```

The `-v /my/own/datadir:/var/lib/mysql` part of the command mounts the `/my/own/datadir` directory from the underlying host system as `/var/lib/mysql` inside the container, where MySQL by default will write its data files.



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

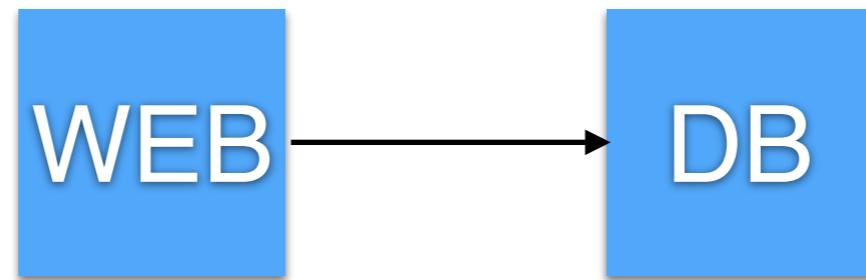
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run Multi-containers



# Multi containers



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi containers



```
$ docker container run --rm -d --name db training/postgres  
$ docker container run --rm --name web --it training/webapp bash
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi containers



```
$ docker container run --rm --name web --it training/webapp bash
```

```
root@7dc14327463d:/# ping db
```

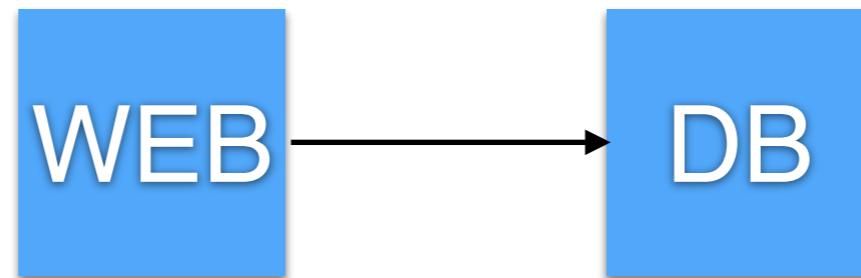


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi containers



```
root@7dc14327463d:/# ping db
```

```
...
```

```
root@7dc14327463d:/# exit
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi containers



```
$ docker container run --rm --link db --name web --it training/webapp bash
```

```
root@7dc14327463d:/# ping db
```

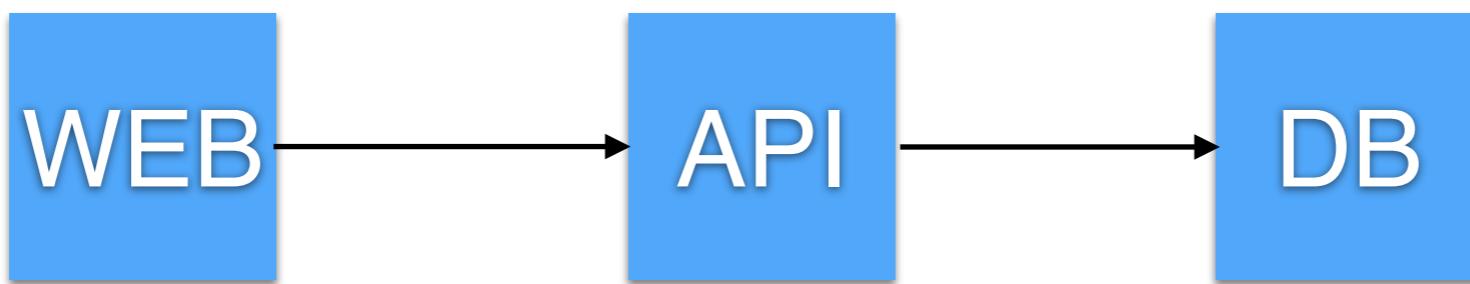


Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

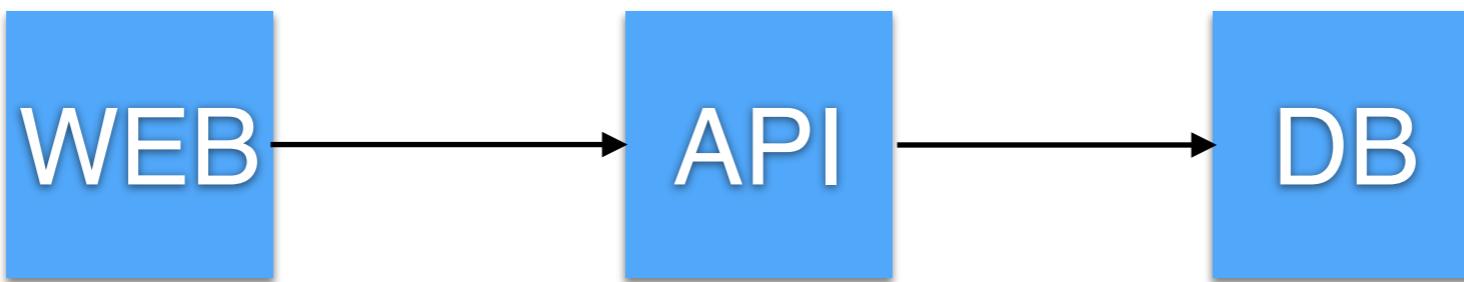
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi containers



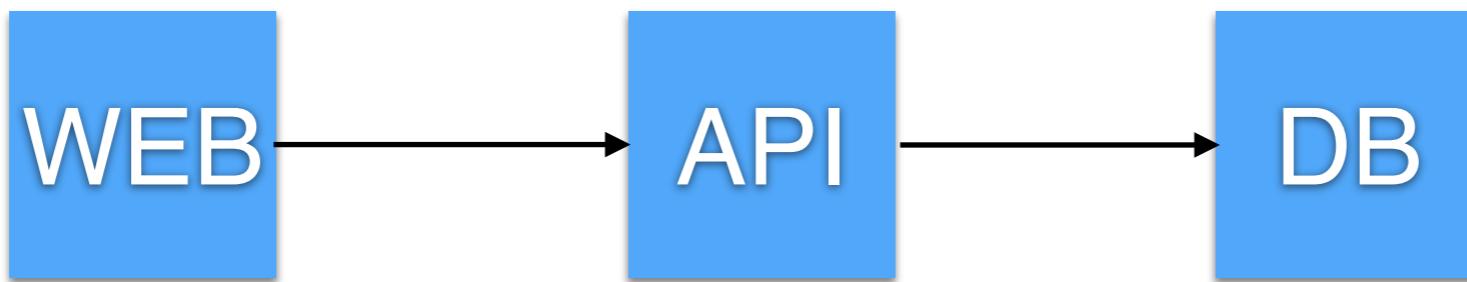
# Multi containers



```
$ docker container run -d --name db training/postgres  
$ docker container run -d --link db      --name john training/webapp  
$ docker container run -d --link john:api --name web training/webapp
```



# Multi containers



```
$ docker container run -d --name db training/postgres  
$ docker container run -d --link db --name john training/webapp  
$ docker container run -d --link john:api --name web training/webapp
```

--link <name | id>:<alias>



# Multi containers

```
$ docker image build -t api api/  
$ docker image build -t web web/
```

```
$ docker container run -d --name db training/postgres  
$ docker container run -d --link db --name api training/webapp  
$ docker container run -d --link api --name web training/webapp
```

```
$ docker container stop web  
$ docker container stop api  
$ docker container stop db
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Multi containers

```
$ docker image build -t api api/  
$ docker image build -t web web/
```

```
$ docker container run -d --name db training/postgres  
$ docker container run -d --link db --name api training/webapp  
$ docker container run -d --link api --name web training/webapp
```

```
$ docker container stop web  
$ docker container stop api  
$ docker container stop db
```

TOO MANY REPEAT COMMAND



# Multi containers

```
$ docker image build -t api api/  
$ docker image build -t web web/
```

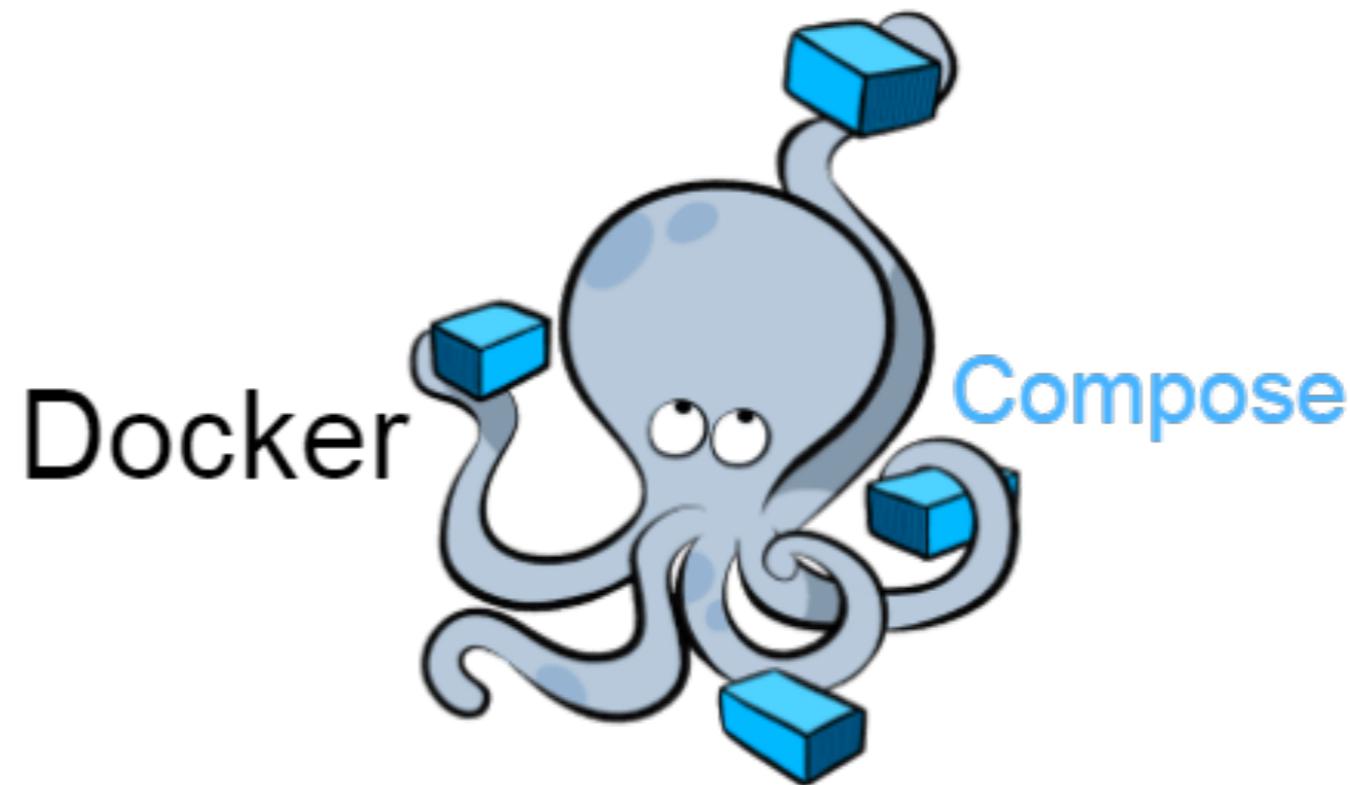
```
$  
$ Can we do one command per action ?  
$
```

```
$ docker container stop web  
$ docker container stop api  
$ docker container stop db
```

TOO MANY REPEAT COMMAND



# Docker Compose

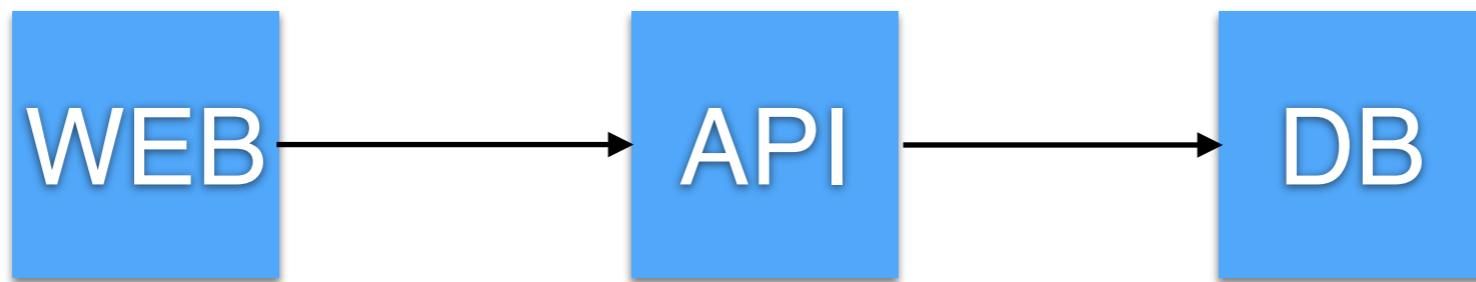


Tool for defining and running multi-container Docker applications. use a YAML file to configure your application's services. Then, with a single command, you create and start all the services from your configuration  
Compose works in all environments: production, staging, development, testing, as well as CI Server



# Services

## Services



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services

## Services



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

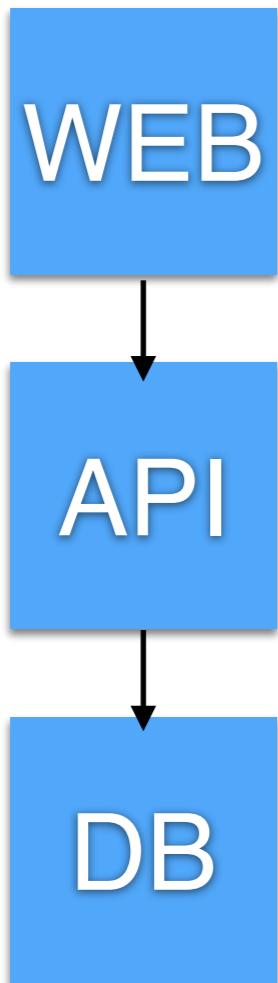
NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services mapping to docker compose

## Services

## docker-compose.yml



version: "3.8"

services:

web:

...

api:

...

db:

...



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services mapping to docker compose

```
$ docker container run -d --name db training/postgres  
$ docker container run -d --link db --name api training/webapp  
$ docker container run -d --link api --name web training/webapp
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services mapping to docker compose

```
$ docker container run -d --name db training/postgres  
$ docker container run -d --link db --name api training/webapp  
$ docker container run -d --link api --name web training/webapp
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```

## docker-compose.yml

```
version: "3.8"
```

```
services:
```

```
web:
```

```
  container_name: web  
  image: training/webapp
```

```
api:
```

```
  container_name: api  
  image: training/webapp
```

```
db:
```

```
  container_name: db  
  image: training/postgres
```



# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```

## docker-compose.yml

version: "3.8"

### services:

#### web:

  container\_name: web  
  image: training/webapp

#### api:

  container\_name: api  
  image: training/webapp

#### db:

  container\_name: db  
  image: training/postgres



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```

docker-compose.yml

version: "3.8"

services:

web:

  container\_name: web  
  image: training/webapp

api:

  container\_name: api  
  image: training/webapp

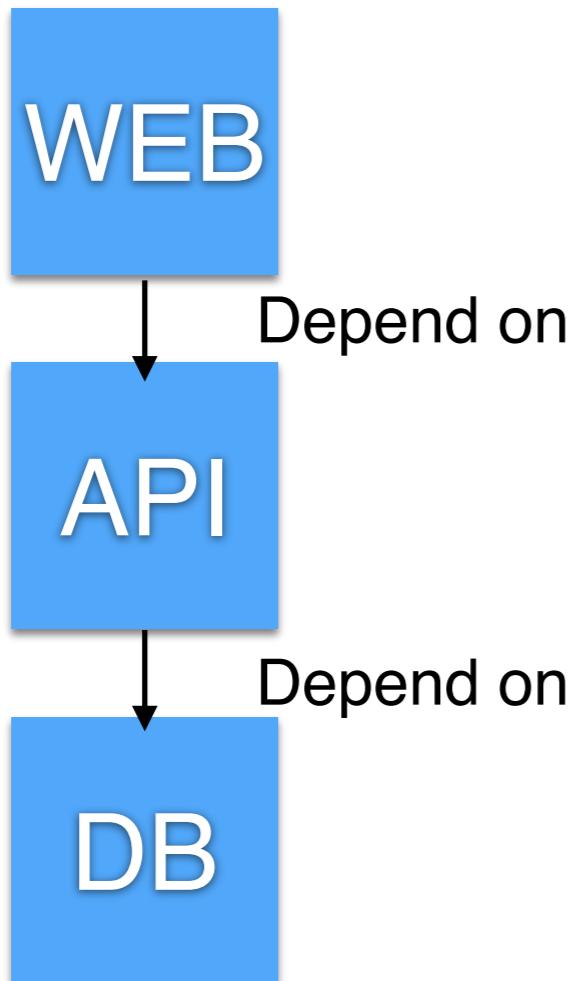
db:

  container\_name: db  
  image: training/postgres



# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```



docker-compose.yml

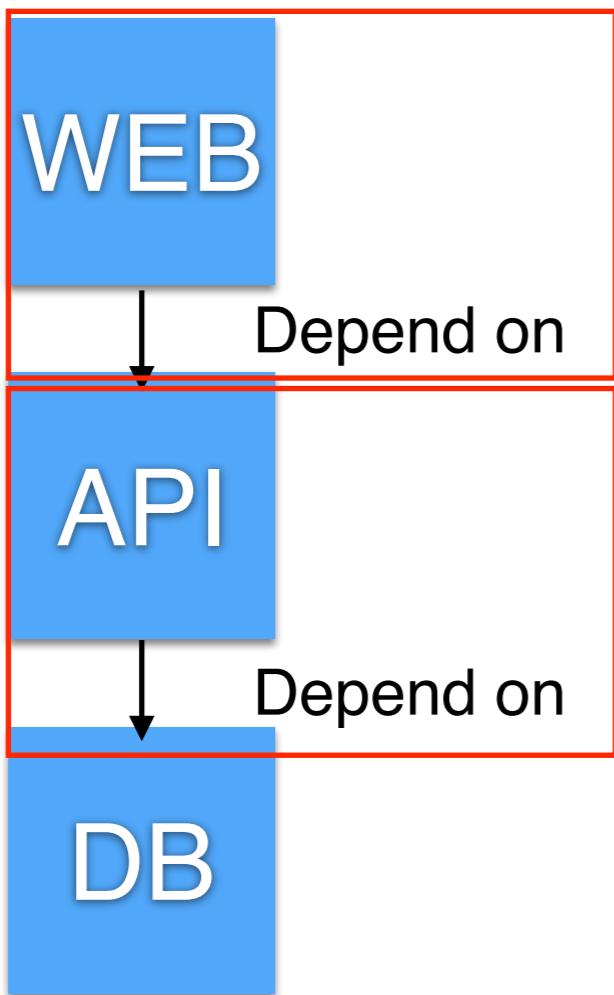
version: "3.8"

```
services:  
  web:  
    container_name: web  
    image: training/webapp  
  api:  
    container_name: api  
    image: training/webapp  
  db:  
    container_name: db  
    image: training/postgres
```



# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```



## docker-compose.yml

version: "3.8"

services:

web:

```
  container_name: web  
  image: training/webapp
```

api:

```
  container_name: api  
  image: training/webapp
```

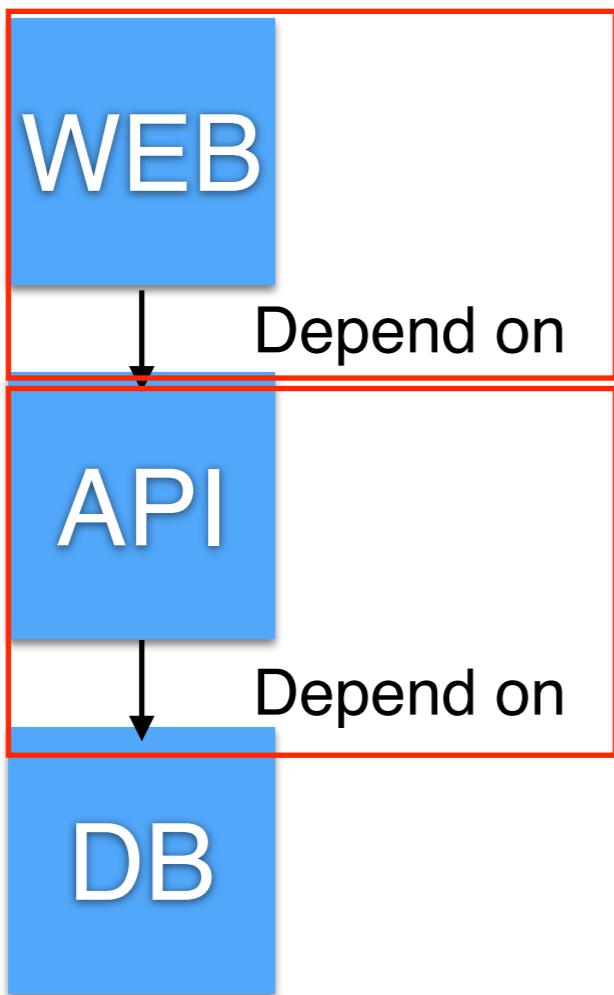
db:

```
  container_name: db  
  image: training/postgres
```



# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```



## docker-compose.yml

version: "3.8"

services:

web:

```
  container_name: web  
  image: training/webapp  
  depends_on:  
    - api
```

api:

```
  container_name: api  
  image: training/webapp  
  depends_on:  
    - db
```

db:

```
  container_name: db  
  image: training/postgres
```



This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

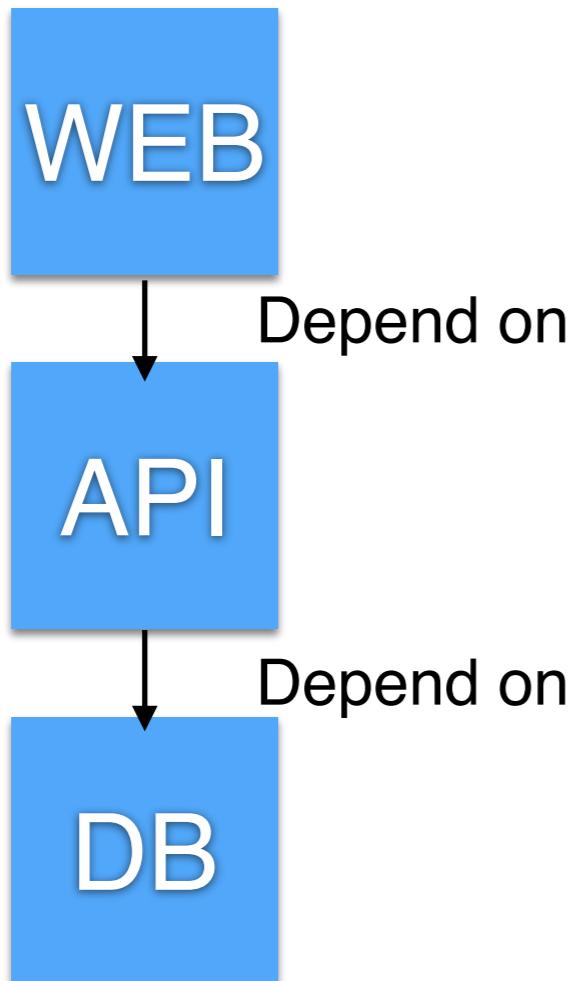
Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

# Services mapping to docker compose

```
--name db training/postgres  
--link db --name api training/webapp  
--link api --name web training/webapp
```



## docker-compose.yml

version: "3.8"

services:

web:

```
  container_name: web  
  image: training/webapp  
  depends_on:  
    - api
```

api:

```
  container_name: api  
  image: training/webapp  
  depends_on:  
    - db
```

db:

```
  container_name: db  
  image: training/postgres
```



# Run All Service !

```
$ docker-compose up
```



Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Run All Service !

## Foreground

```
$ docker-compose up
```

## Background

```
$ docker-compose up -d
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Stop All Service

```
$ docker-compose down
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Management Command

// Start, stop, and rebuild services

docker-compose up

docker-compose down

docker-compose build

// View the status of running services

docker-compose ps

// Stream the log output of running services

docker-compose logs



Share — copy and redistribute the material in any medium or format.

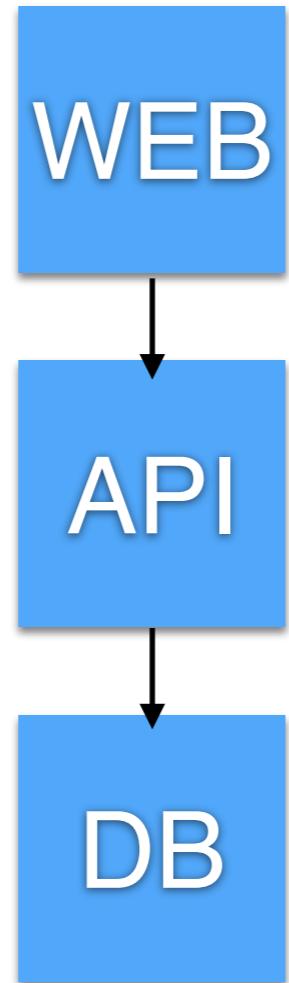
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Services mapping to docker compose

Port  
80:80  
3000:3000



docker-compose.yml

version: "3.8"

services:

web:

  container\_name: web  
  image: training/webapp

  depends\_on:

  - api

api:

  container\_name: api  
  image: training/webapp

  depends\_on:

  - db

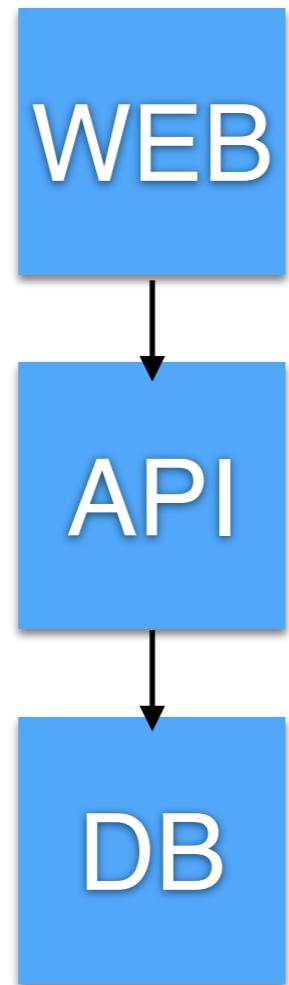
db:

  container\_name: db  
  image: training/postgres



# Services mapping to docker compose

Port  
80:80  
3000:3000



## docker-compose.yml

```
version: "3.8"

services:
  web:
    container_name: web
    image: training/webapp
    depends_on:
      - api
    ports:
      - "80:80"
  api:
    container_name: api
    image: training/webapp
    depends_on:
      - db
    ports:
      - "3000:3000"
  db:
    container_name: db
    image: training/postgres
```



# Services mapping to docker compose

What if Database is not ready ?  
Application should not still running

## docker-compose.yml

```
version: "3.8"

services:
  web:
    container_name: web
    image: training/webapp
    restart: always
    depends_on:
      - api
    ports:
      - "80:80"
  api:
    container_name: api
    image: training/webapp
    restart: always
    depends_on:
      - db
    ports:
      - "3000:3000"
  db:
    container_name: db
    image: training/postgres
```



# Services mapping to docker compose

What if Database is not ready ?  
Application should not still running

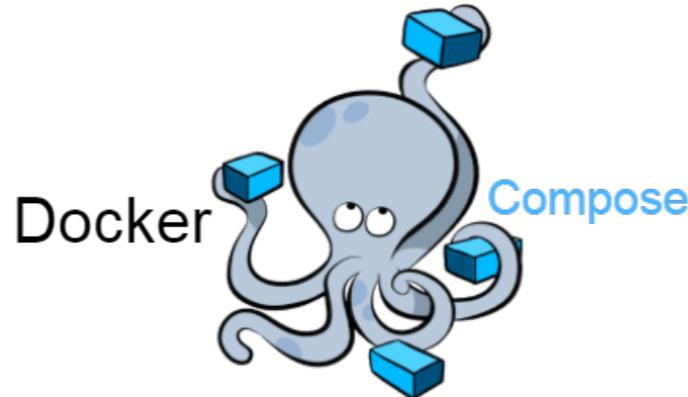
## docker-compose.yml

```
version: "3.8"

services:
  web:
    container_name: web
    image: training/webapp
    depends_on:
      - api
    ports:
      - "80:80"
  api:
    container_name: api
    image: training/webapp
    restart: always
    depends_on:
      - db
    ports:
      - "3000:3000"
  db:
    container_name: db
    image: training/postgres
```



# Docker Compose



`docker-compose.yml`



`docker-compose.override.yml`



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Compose Specific File

```
$ docker-compose -f <specific-file> <command>  
// docker-compose -f docker-compose.local.yml up
```



# Docker Compose Build

```
FROM openjdk:8
WORKDIR /app
COPY stubby4j-6.0.2.jar .
COPY kerry.yml .
CMD java -jar stubby4j-6.0.2.jar -l 0.0.0.0 -d kerry.yml
```



# Build Docker Image

```
$ docker image build -t stub-server .
```



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Build Docker Image

```
$ docker image build -t stub-server .
```

`docker-compose.yml`

`version: "3.8"`

`services:`

`kerry-tracking-service:`

`image: stub-server:latest`

`build:`

`context: 2-stub-server`

`dockerfile: Dockerfile`



Share — copy and redistribute the material in any medium or format.  
Adapt — remix, transform, and build upon the material.

NonCommercial — You may not use the material for commercial purposes.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

# Docker Compose Build Challenge

## docker-compose.yml

```
version: "3.8"

services:
  kerry-tracking-service:
    image: stub-server:latest
    build:
      context: 2-stub-server
      dockerfile: Dockerfile
    ...
  web:
    ...
  node-service:
    ...
  go-service:
    ...
  ...
```



