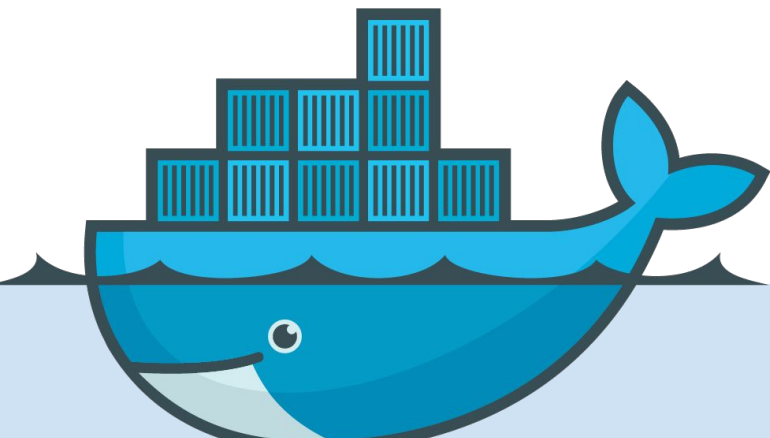


Docker

for WebDevs



@cassilup, May 2017

timjis





<http://www.example.com>



`http://www.example.com`

`www.example.com:80`



`http://www.example.com`

`www.example.com:80`

`127.0.0.1:80`



`http://www.example.com`

`www.example.com:80`

`127.0.0.1:80`

`./node index.js`



How we used to do it

XAMPP anyone? (2002)

Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P)

Manual installs.

Hard to Config.

Bloated OS.

It did the job, though.



How we used to do it (contd.)

Vagrant anyone? (2013)

*“**Person** without a settled home or regular work who wanders from place to place and lives by begging.”* -- Google

Clean OS. Took a lot of space.

Several projects → no space left on HDD.

Resource consuming → CPU + RAM

Matter of Minutes

Dev Envs



How the cool kids do it (2017)

Docker (2013)

Containers

Namespace in the OS

Just run your process(es) carefree

Dockerfile



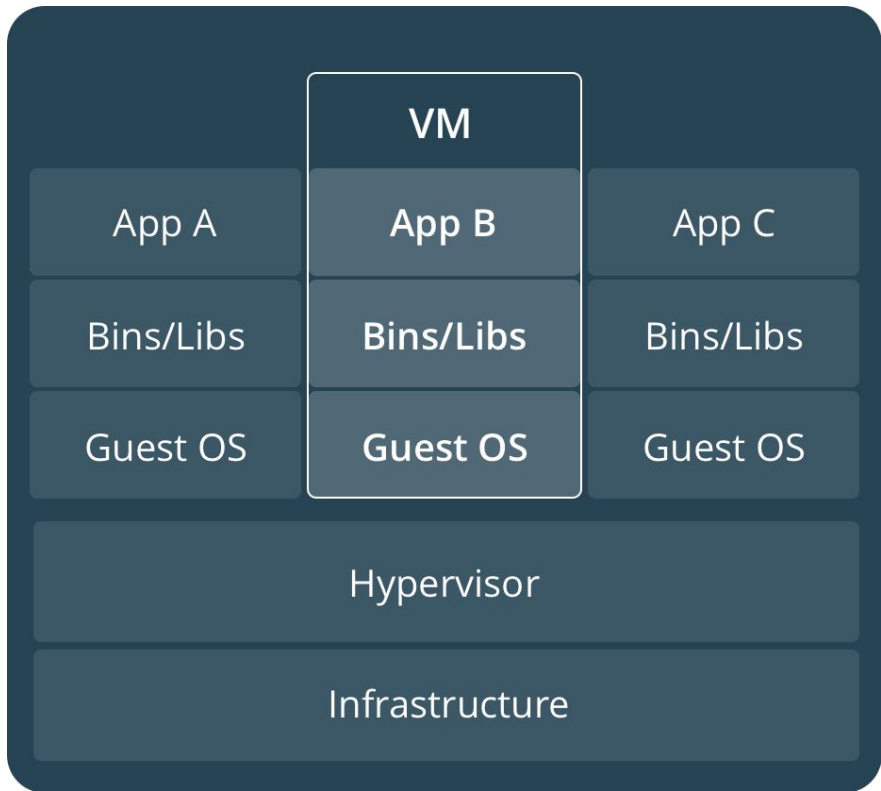
VAGRANT



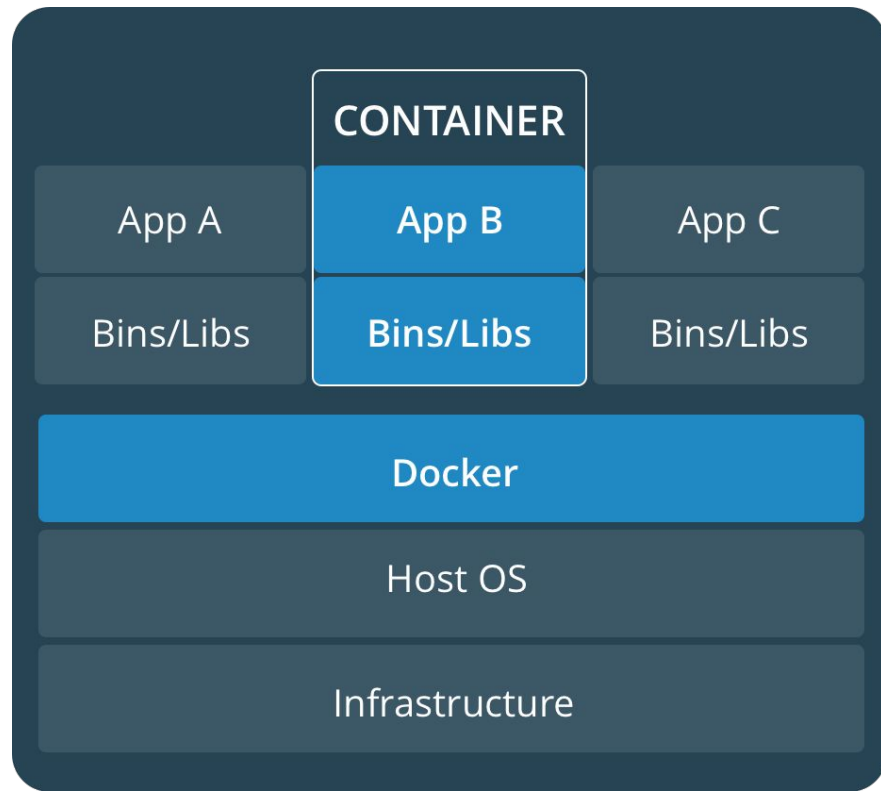
docker



Virtual Machine Stack



Container Stack

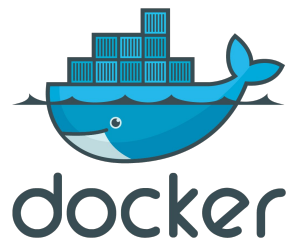




Hours

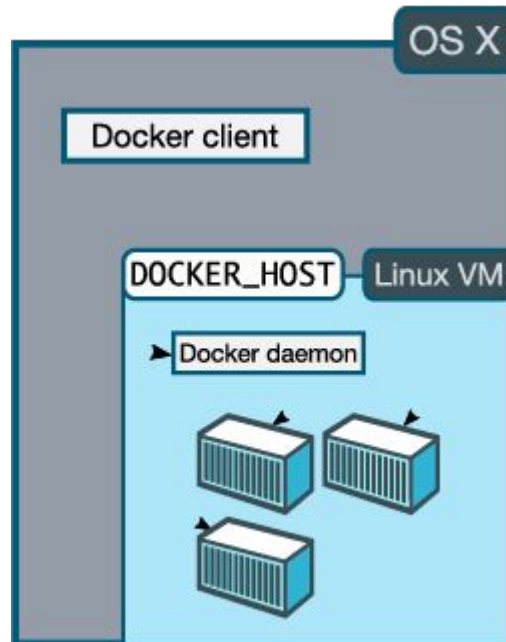
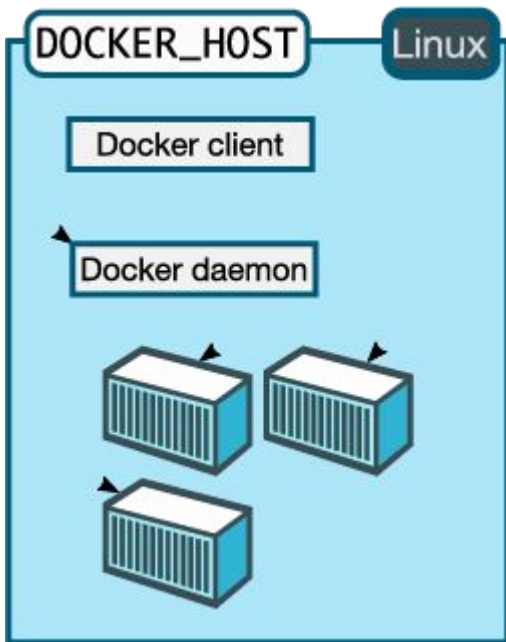


Minutes



Seconds





```
$ docker-machine
```



```
$ docker-machine
```

```
$ docker
```



```
FROM node:boron

# Create app directory
RUN mkdir -p /usr/src/app
WORKDIR /usr/src/app

# Install app dependencies
COPY package.json /usr/src/app/
RUN npm install

# Bundle app source
COPY . /usr/src/app

EXPOSE 8080
CMD [ "npm", "start" ]
```



Image → **Container**



Image → **Container**

Tag

ID





Tag

ID



Demo

<https://github.com/cassilup/express-cowsay>



```
$ docker-machine
```

```
$ docker
```



```
$ docker images # lists images
```

```
$ docker build # builds an image
```

```
$ docker run # runs image
```

```
$ docker ps # lists containers
```

```
$ docker stop # stops container
```

```
$ docker rm # removes container
```



```
$ docker-machine
```

```
$ docker
```

```
$ docker-compose
```



Why you should consider it



Why you should consider it

Development

Fast start

Automated config. Config as code.

Easy context switching

Lower resources usage



Why you should consider it

Development

Fast start

Automated config. Config as code.

Easy context switching

Lower resources usage

Deployment

Cattle as pets.

“Works on my machine” less of a problem

Same env as in production.*

* almost

Microservices (AWS Lambda, Quay.io, now.sh)

Continuous Delivery



<https://docs.docker.com/get-started>



Thank you! :)

Come say hi!

@cassilup



Resources

<http://www.docker.com>

<http://bit.ly/2qfGsto> – Docker intro

<https://docs.docker.com/engine/reference/builder>

<https://usersnap.com/blog/docker-for-web-developers>

<https://github.com/laradock/laradock> – Laradock (PHP)

<https://nodejs.org/en/docs/guides/nodejs-docker-webapp>

<http://jd1m.info/articles/2016/03/06/lessons-building-node-app-docker.html>

<https://semaphoreci.com/community/tutorials/dockerizing-a-php-application>

<https://stackoverflow.com/questions/16047306/how-is-docker-different-from-a-normal-virtual-machine>

